Structural econometrics - Replication

Saara Haikonen, Ella Mattinen, Matias Pousi

December 2023

Contents

This replication report covers the replication of the main estimation results of Lewbel et al. (2022). The report is structured the following way: First, we go through the main challenges of the replication. Then, we provide the results of our replication attempts. Finally, we estimate the models of Lewbel et al. (2022) with some alternative specifications which are not covered in the paper. These results are supplemented by the log files of the estimations in the Appendix.

1 Replication challenges

The replication files provided by the authors can be run without considerable problems. However, there are some caveats to this and the replication files are not very user-friendly in terms of estimating multiple columns or tables.

Firstly, in terms of errors in running the code, the code tries to use a data set which does not exist ("years1999-2004_sm.dta") in addition to referring to the actual data set "years1999-2004.dta". However, this problem is easily solved by removing the reference to the non-existing data set.

Secondly, each of us had different issues running the code. The scalar, J, is set to be 2 and then a local J is set to equal J (most likely due to locals not being able to handle numerical values properly, so they make it a scalar first and refer to J as text in the local). When running the "analysis wrapper.do" for two of us, it said "J not found". To fix this, we checked whether they were referring to J as a local later on and indeed they were not, so we put ‘’ around the J's to fix this. Also, when calling the "estim program.do", J was not found again, so we set scalar J=2 which fixed the issue.

The replication package also includes and generates files with very long names, namely some of the .ster-files. This led to zip-file extracting problems and more importantly, further replication issues. Files intended to be created could not be referred to in the code because their names exceeded the character limits when combined with any slightly longer folder names for the paths. This bad naming practice led to r(603) in Stata. After shortening the folder names (and changing locations in the folder), this problem disappeared.

For the ease of replication, the structural design of the code is not optimal either. You must manually modify the wrapper code to replicate the results of a few individual columns. Then, the code has to be run multiple times with these slight modifications to obtain all the results presented in the paper. Not all of these modifications are explicitly mentioned in the sparsely-commented readme-file. Hence, to replicate some of the columns, it was sometimes necessary to make educated guesses based on the instructions for other columns and the overall structure of the code. The code also does not create proper outcome tables, but rather only saves the estimates and requires further steps to replicate the exact tables used in the paper. These are not included in the code.

2 Main replication results

In this section we provide a summary of main replication results which we deem essential in understanding the core results of the paper. These include tables 1-8 of the main paper.

Table 1, the summary statistics for consumption data, is the only table of the tables presented in this replication report which was more or less automatically created. For other tables, the estimation results were manually typed into the tables because of the structure of the replication files. We added esttab commands to the code to ease this task instead of dealing with .ster-files.

Table 1: Summary statistics for consumption data

	Observations (N= 24,757)				Pairs (N= 128,640)			
	Mean	SD	Min	Max	Mean	SD	Min	Max
xi	.99	.59	.072	4.7	1	.59	.072	4.7
Qi1	.44	.21	0	2	.44	.21	0	2
Qi2	.44	.33	.0069	2.7	.44	.32	.0069	2.7
Qikbar1					.44	.15	.027	1.7
Qikbar2					.44	.24	.02	2.4
P1	1.1	.08	.94	1.3	1.1	.083	.94	1.3
P2	1.2	.11	.94	1.5	1.2	.12	.94	1.5
zi1	.38	.21	0	1.1	.38	.21	0	1.1
zi2	.39	.11	.17	.82	.4	.11	.17	.82
zi3	.84	.36	0	1	.84	.36	0	1
zi4	.15	.35	0	2.3	.16	.35	0	2.3
zi5	.14	.35	0	1	.13	.34	0	1
zi6	.46	.5	0	1	.47	.5	0	1
zi7	.26	.44	0	1	.26	.44	0	1

Table reports summary statistics for estimation sample.

Table 2 reports the replicated estimates of table 2 of Lewbel et al. (2022). The estimates and their standard errors match between the paper and replication for the estimates we were able to obtain like in most cases (with the exception of table 8). At least for our computers, the computational burden caused by district-level group estimations (over 10 times more observations than in other specifications) was too much and we were not able to replicate these estimates (e.g., table 2, columns (1) and (4) in the paper). To give some context for the computational issues, table 2 panel A estimates in table 2 were computed within approximately 10 minutes each. However after three hours, the district group-level estimations had not finalized the first iteration of the first step of the first GMM estimation (the RE approach). During these attempts, Stata notified several times that it was not responding. Hence, we deem that Stata or our computational capabilities do not seem a very optimal choice for GMM with such a large sample (clearly over 2 million within-group household-pairs). Thus, the tables exclude the columns for district-level estimations.

Table 2: Table 2 estimations Lewbel et al. (2022)

	Neighborhood	Neighborhood-Caste	Neighborhood	Neighborhood-Caste
	(2)	(3)	(5)	(6)
Panel A				
A				
	0.558***	0.606***	0.0876	0.266*
	(0.0362)	(0.0357)	(0.121)	(0.119)
Panel B				
A				
	0.568***	0.606***	0.156	0.266*
	(0.0345)	(0.0357)	(0.133)	(0.119)

Standard errors in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Notably, the readme-file is lacking in terms of instructions for the exact computation method for the difference between RE and FE model estimates, specifically the standard error of the difference because the difference between estimates is just a subtraction.

Specifically, instructions for estimating panel B of table 2 are missing. That is, it reports only the instructions to replicate panel A. However, based on the replication instructions of table 3 of

Lewbel et al. (2022), it seems that table 2 panel B can be replicated by setting groupComp_def to the Neighborhood-Caste -level. Our replication attempt estimates match those of the paper as seen in table 2.

Table 3: Table 3 estimations Lewbel et al. (2022)

	Neighborhood	Neighborhood-Caste	Neighborhood	Neighborhood-Caste
	(2)	(3)	(5)	(6)
Panel A				
A				
	0.0380*	0.0539***	0.559***	0.529***
	(0.0171)	(0.0156)	(0.0888)	(0.0900)
Panel B				
A				
	0.568***	0.606***	0.156	0.266*
	(0.0345)	(0.0357)	(0.133)	(0.119)

Standard errors in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Table 3 suffers of the same computational problems as table 2 for columns 1 and 4. That is, we were not able to estimate the district-level estimates. Otherwise, the replication proceeded well.

For tables 4-6, there were no replication issues.

Table 4: Table 4 Lewbel et al. (2022)

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Constant								
	0.606***	0.606***	0.600***	0.694***	0.266*	0.255*	0.0780	0.145
	(0.0357)	(0.0354)	(0.0565)	(0.0412)	(0.119)	(0.121)	(0.125)	(0.143)
Scheduled non-Hindu								
		0.168*				0.130		
		(0.0664)				(0.237)		
Scheduled Hindu								
		0.247**				-0.285		
		(0.0872)				(0.330)		
Nonscheduled non-Hindu								
		0.179**				-0.0747		
		(0.0547)				(0.153)		
Owns Land								
			0.0313				0.446**	
			(0.0540)				(0.136)	
High school or greater								
				-0.186**				0.454**
				(0.0570)				(0.163)

Standard errors in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Table 5: Table 5 estimations Lewbel et al. (2022)

	Own-Group	Upper Caste	Own-Group	Upper Caste
	(1)	(2)	(3)	(4)
A				
	0.802***	0.0523	0.445**	0.0106
	(0.0702)	(0.0388)	(0.138)	(0.302)

Standard errors in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Table 6: Table 6 Lewbel et al. (2022)

	(1)	(2)	(3)	(4)	(5)	(6)
A (group food on food)						
	0.411*	0.639***	0.606***	9.741***	2.228***	0.266*
	(0.171)	(0.0360)	(0.0357)	(2.066)	(0.382)	(0.119)
A (group nonfood on nonfood)						
	0.452**	0.572***	0.606***	5.400***	-0.911***	0.266*
	(0.171)	(0.0336)	(0.0357)	(1.577)	(0.276)	(0.119)
A (group food on own nonfood)						
	-0.397			-7.695***		
	(0.275)			(1.828)		
A (group nonfood on own food)						
	-0.0945			-6.383***		
	(0.102)			(1.860)		

Standard errors in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Replication of table 7 proceeded without notable problems like the other replications presented above. There are only a few worthy mentions. The required time for the estimation comparably lengthy which can be explained by the increased complexity of the model with the transition from a two to a three good model. Additionally, columns 2 and 4 of table 7 are an example of estimates for which the necessary changes for replication were not explicitly described in neither the readme nor the do-files. However, these problems were rather easily solved with paying attention on the paper and little experimentation.

Table 7: Table 7 Lewbel et al. (2022)

	(1)	(2)	(3)	(4)
A (group food on food)				
	0.848***	0.932***	2.393***	0.296**
	(0.0231)	(0.0145)	(0.426)	(0.0996)
A (group fuel on own fuel)				
	0.938***	0.932***	2.820**	0.296**
	(0.0184)	(0.0145)	(0.913)	(0.0996)
A (group other on own other)				
	0.740***	0.932***	-1.387***	0.296**
	(0.0232)	(0.0145)	(0.334)	(0.0996)

Standard errors in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

For the estimates in table 8, that is alternative classifications to luxury and / or visible goods, the convergence of the GMM estimation is notably slower than with e.g., food and non-food specifications. For example, the RE model with luxury / non-luxury good specification requires multiple iterations more than with the food / non-food specification.
For some reason, we are not exactly able to replicate all the estimates presented in the paper. Both point estimates and standard errors differ in columns (2), (3), (4) and (6) compared to the ones presented in the article. However in column (6), the difference seems to be rather small, 0.002 for the point estimate and 0.001 for the standard error. In column (1) our point estimate matches the one presented in the article but the standard error does not. Finally for column (5), both the point estimate and the standard error are the same as in the article. We are not quite sure what causes these differences as we acquire some of the correct estimates and this happening just by chance seems somewhat unlikely. What makes this even more puzzling is that columns (1) and (4), (2) and (5), and (3) and (6) are estimated with the same changes suggested by the readme-file. Hence, at least estimating column (5) correctly and column (2) incorrectly does not support us making a mistake in specifying the correct adjustments to the code between runs.

Table 8: Table 8 Lewbel et al. (2022)

	RE	RE	RE	FE	FE	FE
	(1)	(2)	(3)	(4)	(5)	(6)
	Lux	Visible	Vis. Lux	Lux	Visible	Vis. Lux
A						
	0.545***	0.390***	0.616***	0.613***	0.418***	0.654***
	(0.0840)	(0.0862)	(0.122)	(0.182)	(0.115)	(0.132)

Standard errors in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

3 Modification

The modification we chose to do was a combination of alternative specifications in the paper. Specifically, we allow the elements of A matrix to vary freely within the diagonal and we allow heterogeneity in peer effects based on the education level (high school or higher). That is, the peer effects of food and non-food goods are allowed to differ between each other and between those who have attained at least high school level education and those who have not.

However, the results of this estimation are not very convincing. The estimated peer effects with the fixed effects approach in column (2) are negative for non-food and a lot larger than 1 for food. Neither of these values are particularly believable as these suggest either no consumption equilibrium or negative peer effects. The high school or higher education estimates still seem to increase the peer effects as in the paper with only the education heterogeneity. However, we are not willing to make any causal, or even relationship-like, interpretation on these coefficients because of the non-plausible nature of the food and non-food peer effect estimates.

For the RE approach in column (1), the estimates seem to be more reasonable but they should suffer of the same too strong assumptions as mentioned in the paper. Additionally, now the education estimates are negative like in the RE education heterogeneity specification in column 4 in table 4 in the paper. Hence, these results seem to be rather robust in terms of scale when compared to the table 4 and table 6 estimates in the paper but we can't draw any conclusions on the actual effects due to the assumptions biasing the results. Namely, at least based on the paper, the estimates of the fixed effect approach yield more credibility and they suggest an increase in the peer effects due to higher education instead of a decrease.

We also repeated the same estimations as in columns (1) and (2) in columns (3) and (4) with higher minimum group size. Namely, we increased the minimum size of a group from 3 to 5. However, the results still stay mostly unchanged in terms of plausibility even though we technically try to improve the precision of the group mean estimator. This improvement, with the downside of decreasing the number of groups, is not sufficient for us to try to interpret the estimates in a causal manner.

	RE	FE	RE	FE
	(1)	(2)	(3)	(4)
A				
Food	0.705***	2.275**	0.732***	3.360***
	(0.0398)	(0.798)	(0.0432)	(0.710)
High school	-0.172***	0.252	-0.181*	1.597
	(0.0517)	(0.849)	(0.0921)	(0.847)
Non-food	0.650***	-1.488*	0.732***	-2.156***
	(0.0435)	(0.710)	(0.0512)	(0.633)
High school	-0.169*	1.244*	-0.276*	1.000
	(0.0769)	(0.584)	(0.117)	(0.640)

Columns (1) and (2) with minimum group size 3 as in the baseline estimation.
Columns (3) and (4) with minimum group size 5.
Standard errors in parentheses.
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Table 9: Differing peer effects for food and non-food with education level heterogeneity estimations

A Appendix: Main replication result log files

A.1 Table 1

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table1.smcl
  log type:  smcl
 opened on:   9 Dec 2023, 18:50:03

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
  C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 1

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
50. local group_def "dist"

51. local group_def "fsuXseg"

52. local group_def "fsuXsegXreligXsched"

53.
54. // comparision group
55. local groupComp_def ""

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.         syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
70.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)        expend_type   ->   (dropped)
  xij variables:
                          laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)        expend_type   ->   (dropped)
  xij variables:
                          laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations               840   ->   280
  Number of variables                    5   ->   6
  j variable (3 values)        expend_type   ->   (dropped)
  xij variables:
                          laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,637 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                    egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
```

```
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }
297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }
316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }
333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
```

```
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (82 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,576 real changes made, 82 to missing)

460 replace x2=x*x
  (73,576 real changes made, 82 to missing)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,576 real changes made)

478 replace x2=x2_temp
  (73,576 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (48,819 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)
```

```
488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_2eb0_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
    24,757
```

```
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,757 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
549 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
    128,640
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
  (file C:\Users\pousim2\Downloads\rep\replication/output/consumption_sumstats.tex not
     found)
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                          qui su `var'
    3.                          //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max
  > )' \\"
    4.                          file write sumstat "`sumstats_row_`var''" _n
    5.                 }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
  command BREAK is unrecognized
  r(199);
577 }
  r(199);
  r(199);

  end of do-file

  r(199);

578 log close
        name:  <unnamed>
         log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table1.smcl
    log type:  smcl
   closed on:   9 Dec 2023, 18:51:32
```

A.2 Table 2

A.2.1 Columns 2 and 5

A.2.1.1 Panel A

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table2_25.smcl
  log type:  smcl
 opened on:   7 Dec 2023, 16:10:31

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
  C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
50. *local group_def "dist"
51. local group_def "fsuXseg"

52. *local group_def "fsuXsegXreligXsched"
53.
54. // comparision group
55. local groupComp_def ""

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.          syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
70.          if "`rename'" == "" local rename = "`varlist'bar"
    3.
```

```
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)        expend_type   ->   (dropped)
  xij variables:
                          laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                   5   ->   5
  j variable (2 values)        expend_type   ->   (dropped)
  xij variables:
                          laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                  Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations                 840   ->   280
  Number of variables                      5   ->   6
  j variable (3 values)          expend_type   ->   (dropped)
  xij variables:
                                laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg
  (43,637 missing values generated)

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"       egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                  egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                    egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
```

```
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }
297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }
316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }
333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
```

```
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (82 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,576 real changes made, 82 to missing)

460 replace x2=x*x
  (73,576 real changes made, 82 to missing)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,576 real changes made)

478 replace x2=x2_temp
  (73,576 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (44,114 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)
```

```
488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_7b70_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
```

```
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (29,462 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
549 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var'" _n
    5.                 }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
```

```
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1
```

```
scalar estimate_FE=1

// estimation options
global w_initial "winit(unadjusted)"
if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"
global wmatrix "wmatrix(robust)"
global trace_level ""
global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"
global technique ""
global maxiter ""
global technique_RE ""
if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"
global derivatives "quickderivatives"
scalar init_FE_from_RE=1
global clust_var "state_district_round"
local clus_name "_clus_Dist"









// xxzx to keep names same
local wmatrix_desc ""
local sectorName "_urbOnly"
local dropZdesc "_dropPre"
local desc_struc_v0 ""
local inst_desc "_exactInst"
local inst_rootp "_noPPXdInst"
local weight_desc ""
local RE_sq_inst_desc ""
local cnstname ""
local actual_pqhat_desc ""
local expUpname "_updateExp"
```

```
661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //              ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T

694 local Tm1=`T'-1
```

```
695 local Tindiv : word count $zlist_indiv

696
697 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
706
707 if `J'==2 {
708          replace p1=P1
  (195,282 real changes made)
709          replace p2=P2
  (195,282 real changes made)
710          replace qi1=Qi1
  (195,262 real changes made)
711          replace qk1=Qk1
  (195,262 real changes made)
712          replace qikbar1=Qikbar1
  (195,282 real changes made)
713          replace qi2=Qi2
  (195,282 real changes made)
714          replace qk2=Qk2
  (195,282 real changes made)
715          replace qikbar2=Qikbar2
  (195,282 real changes made)
716
717          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721          forv j = 1(1)3 {
    2.                replace p`j'=P`j'
    3.                replace qi`j'=Qi`j'
    4.                replace qk`j'=Qk`j'
    5.                replace qikbar`j'=Qikbar`j'
    6.          }
722
723          foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                qui replace `var'=.
    3.          }
724 }

725
```

```
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }
```

```
751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
773         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j'"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.         local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.         if `jplus1'<=`J' {
    5.                 forval k=`jplus1'/`J' {
    6.                         local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                 }
    8.         }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                 if noMeasError==1                                local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.         }
```

```
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                              l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                    ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2
  > ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qikba
  > r1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d
  > 1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > }*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))        ))
```

```
836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) +
  >  ({A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C
  > 2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikb
  > ar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:
  > }) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C
  > 2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qi
  > kbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )
  > *{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > one}*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}
  > *ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi
  > 4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p
  > 1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         )
  > )

874
875 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi

882 g double `ximxk'=xi-xk
```

```
883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.          tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.          g double `xizi`t''=xi*zi`t'
    4.          g double `xkzk`t''=xk*zk`t'
    5.          g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.          g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.          g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.          g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.          g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.          forval j=1/`J' {
   11.                  tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                  g double `zi`t'p`j''= zi`t'*p`j'
   13.                  g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                  g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                  g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                  forv s=1/`T' {
   17.                          tempvar zi`t'zi`s'p`j'
   18.                          g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                  }
   20.          }
   21. }

891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }

903
904 local zrg ""

905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var''"
    6.          }
    7. }
```

```
906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var''"
    5.
911          forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var''"
   10.                  local xprg "`xprg' `xp`j'`var''"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""

925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }
```

```
929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.         forval k=`j'/`J' {
    3.                 tempvar p`j'p`k'
    4.                 g `p`j'p`k''=p`j'*p`k'
    5.                 local roundinsts "`roundinsts' `p`j'p`k''"
    6.         }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.         if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.         if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }

  Linear regression                               Number of obs     =    195,282
                                                  F(129, 4281)      =     183.43
                                                  Prob > F          =     0.0000
                                                  R-squared         =     0.2342
                                                  Root MSE          =     .12062
```

(Std. err. adjusted for **4,282** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	-6.714683	6.513648	-1.03	0.303	-19.48481	6.055442
__00009L	-3.919705	1.181145	-3.32	0.001	-6.235361	-1.604049
__00009O	-1.258104	3.47171	-0.36	0.717	-8.064455	5.548247
xi	.0987095	.0068752	14.36	0.000	.0852304	.1121885
__000002	-.0137634	.0021242	-6.48	0.000	-.0179279	-.0095988
__00000E	-.1040907	.079469	-1.31	0.190	-.2598911	.0517096
__00000P	-.049984	.045646	-1.10	0.274	-.1394738	.0395057
__000017	.0907086	.1479888	0.61	0.540	-.199426	.3808433
__00001I	.0211691	.093939	0.23	0.822	-.163	.2053382
__000020	.0000126	.0396298	0.00	1.000	-.0776823	.0777076
__00002B	-.0085439	.0216343	-0.39	0.693	-.0509584	.0338705
__00002T	.0772893	.0589771	1.31	0.190	-.0383364	.1929149
__000034	-.006948	.0399336	-0.17	0.862	-.0852386	.0713426
__00003M	.0368221	.0497676	0.74	0.459	-.0607481	.1343923
__00003X	-.0250394	.0333119	-0.75	0.452	-.0903481	.0402692
__00004F	-.0266583	.0400568	-0.67	0.506	-.1051903	.0518737
__00004Q	.0163697	.0222542	0.74	0.462	-.02726	.0599994
__000058	-.0278007	.0535634	-0.52	0.604	-.1328127	.0772112
__00005J	.0230217	.030364	0.76	0.448	-.0365075	.082551
__00005U	-.0101181	.33423	-0.03	0.976	-.6653821	.6451458
__00005V	.0045335	.0621372	0.07	0.942	-.1172875	.1263546
__00005W	-.040752	.1384158	-0.29	0.768	-.3121187	.2306147
__00005X	.1565259	.3622612	0.43	0.666	-.5536938	.8667456
__00005Y	-.0208584	.0858552	-0.24	0.808	-.1891791	.1474623
__00005Z	.0519392	.1429915	0.36	0.716	-.2283983	.3322767
__000060	.0517537	.0530793	0.98	0.330	-.0523091	.1558166
__000061	.021804	.0457293	0.48	0.634	-.0678491	.1114572
__000062	.0525029	.0296207	1.77	0.076	-.0055689	.1105748
__000063	-.0507446	.0395684	-1.28	0.200	-.1283191	.0268299
__000064	-.0428488	.0480362	-0.89	0.372	-.1370248	.0513271
__000065	.105548	.7094066	0.15	0.882	-1.285257	1.496352
__000066	.2236562	.1196076	1.87	0.062	-.0108366	.458149
__000067	-.2966657	.2845182	-1.04	0.297	-.8544689	.2611375
__000068	-.0993483	.7078088	-0.14	0.888	-1.48702	1.288324
__000069	-.047935	.1696049	-0.28	0.777	-.3804485	.2845785
__00006A	-.2744667	.2751774	-1.00	0.319	-.8139571	.2650237
__00006B	.0566733	.1033056	0.55	0.583	-.1458592	.2592058
__00006C	-.1958352	.0938335	-2.09	0.037	-.3797975	-.0118729
__00006D	.0900214	.0521045	1.73	0.084	-.0121304	.1921733
__00006E	-.087469	.084737	-1.03	0.302	-.2535974	.0786595
__00006F	.0696471	.10396	0.67	0.503	-.1341683	.2734625
__00006G	-.1439915	.1519908	-0.95	0.344	-.4419723	.1539892
__00006H	-.0078279	.02767	-0.28	0.777	-.0620754	.0464195
__00006I	.038554	.0609161	0.63	0.527	-.0808731	.1579812
__00006J	.1383306	.1732707	0.80	0.425	-.2013697	.4780309
__00006K	.0439324	.0413456	1.06	0.288	-.0371264	.1249912
__00006L	-.0236313	.0713791	-0.33	0.741	-.1635713	.1163088
__00006M	-.0373197	.0247383	-1.51	0.131	-.0858195	.0111801
__00006N	.0191101	.0209508	0.91	0.362	-.0219644	.0601845
__00006O	-.0168836	.0121926	-1.38	0.166	-.0407874	.0070203
__00006P	-.0069102	.0214167	-0.32	0.747	-.0488981	.0350778
__00006Q	.0101874	.0251396	0.41	0.685	-.0390991	.059474
__00006R	-.4193466	.4168028	-1.01	0.314	-1.236496	.397803
__00006S	-.0035883	.0439635	-0.08	0.935	-.0897796	.082603
__00006T	.1918489	.1597812	1.20	0.230	-.1214049	.5051028
__00006U	.1579041	.3651306	0.43	0.665	-.5579412	.8737494
__00006V	-.0933399	.0673983	-1.38	0.166	-.2254756	.0387957
__00006W	-.0073692	.1247353	-0.06	0.953	-.2519151	.2371767
__00006X	.0323711	.0475117	0.68	0.496	-.0607765	.1255187
__00006Y	.0496426	.0322125	1.54	0.123	-.0135107	.1127959
__00006Z	-.0494569	.0211465	-2.34	0.019	-.090915	-.0079989
__000070	-.0083785	.0363035	-0.23	0.817	-.0795521	.0627952
__000071	-.0211205	.0410597	-0.51	0.607	-.1016188	.0593778
__000072	-.349702	.2482541	-1.41	0.159	-.8364088	.1370047
__000073	-.0604974	.0382074	-1.58	0.113	-.1354038	.014409
__000074	.2444043	.0998065	2.45	0.014	.0487319	.4400767
__000075	.1016006	.2440994	0.42	0.677	-.3769609	.580162

__000076	-.0383117	.0545102	-0.70	0.482	-.1451799	.0685566
__000077	.1404905	.0934202	1.50	0.133	-.0426615	.3236425
__000078	.0269809	.0363289	0.74	0.458	-.0442426	.0982043
__000079	-.0035637	.0298703	-0.12	0.905	-.0621249	.0549975
__00007A	.0213402	.0190196	1.12	0.262	-.015948	.0586285
__00007B	.0122551	.0274024	0.45	0.655	-.0414678	.0659781
__00007C	-.0261736	.0322539	-0.81	0.417	-.089408	.0370608
__00007D	.1501923	.1810814	0.83	0.407	-.204821	.5052056
__00007E	.0186924	.0297673	0.63	0.530	-.0396668	.0770517
__00007F	-.0960229	.0740694	-1.30	0.195	-.2412372	.0491915
__00007G	-.036966	.1891706	-0.20	0.845	-.4078384	.3339064
__00007H	.0165474	.0447487	0.37	0.712	-.0711833	.104278
__00007I	-.0949843	.0756459	-1.26	0.209	-.2432894	.0533208
__00007J	-.0053479	.0275558	-0.19	0.846	-.0593716	.0486758
__00007K	-.0297525	.0236645	-1.26	0.209	-.0761471	.0166421
__00007L	.0272967	.0140789	1.94	0.053	-.0003052	.0548987
__00007M	-.0087321	.0234782	-0.37	0.710	-.0547614	.0372973
__00007N	.0462657	.0263354	1.76	0.079	-.0053654	.0978968
__00007O	-.0088212	.3326424	-0.03	0.979	-.6609727	.6433302
__00007P	.0922669	.0417672	2.21	0.027	.0103816	.1741522
__00007Q	-.1378633	.1267547	-1.09	0.277	-.3863681	.1106416
__00007R	.1011113	.2978586	0.34	0.734	-.4828459	.6850686
__00007S	.0466282	.05859	0.80	0.426	-.0682386	.1614949
__00007T	-.2858349	.1006521	-2.84	0.005	-.4831651	-.0885046
__00007U	.0126861	.0345988	0.37	0.714	-.0551454	.0805176
__00007V	-.083342	.029172	-2.86	0.004	-.1405343	-.0261497
__00007W	.0075605	.0178951	0.42	0.673	-.0275231	.0426441
__00007X	.0094192	.030578	0.31	0.758	-.0505295	.0693679
__00007Y	.055116	.0361508	1.52	0.127	-.0157584	.1259904
__000080	-22.65107	15.26445	-1.48	0.138	-52.57731	7.275173
__000082	-4.271119	7.640486	-0.56	0.576	-19.25043	10.70819
__000085	-.5001575	.3604299	-1.39	0.165	-1.206787	.2064719
__000087	.5029404	.2469202	2.04	0.042	.0188488	.987032
__00008A	5.392621	2.994827	1.80	0.072	-.4787922	11.26403
__00008C	-.4701594	1.436112	-0.33	0.743	-3.285684	2.345365
__00008F	26.76682	18.81978	1.42	0.155	-10.12971	63.66335
__00008H	7.984344	9.7927	0.82	0.415	-11.21442	27.18311
__00008K	-.4326991	.4636585	-0.93	0.351	-1.34171	.4763119
__00008M	-1.108652	.4331257	-2.56	0.011	-1.957802	-.2595007
__00008P	-1.327868	.833414	-1.59	0.111	-2.961791	.3060552
__00008R	-.1502675	.6168339	-0.24	0.808	-1.359582	1.059047
__00008U	.3049203	.3041771	1.00	0.316	-.2914245	.9012652
__00008W	.1260139	.2152904	0.59	0.558	-.2960669	.5480947
__00008Z	.7913615	.2158016	3.67	0.000	.3682785	1.214444
__000091	-.2611784	.1569008	-1.66	0.096	-.5687854	.0464286
__000094	.0581917	.173075	0.34	0.737	-.281125	.3975083
__000096	.0871662	.1323099	0.66	0.510	-.1722298	.3465623
__000099	.1644385	.2450035	0.67	0.502	-.3158953	.6447723
__00009B	-.2349664	.1839875	-1.28	0.202	-.5956773	.1257446
__00009E	-.0188946	.2852469	-0.07	0.947	-.5781263	.5403371
__00009G	-.5827322	.2126674	-2.74	0.006	-.9996705	-.1657938
xbarlag	28.1798	19.43797	1.45	0.147	-9.928701	66.28831
x2barlag	-.3078268	.3771324	-0.82	0.414	-1.047202	.431548
xbar2lag	-4.654213	3.805724	-1.22	0.221	-12.1154	2.806977
rootxbarlag	-36.45109	24.0315	-1.52	0.129	-83.56528	10.6631
z1barlag	1.698725	.6941054	2.45	0.014	.337919	3.059532
z2barlag	1.509663	1.082542	1.39	0.163	-.6126804	3.632007
z3barlag	-.5443645	.3903002	-1.39	0.163	-1.309555	.2208263
z4barlag	-.4424163	.2523297	-1.75	0.080	-.9371132	.0522806
z5barlag	-.2387804	.1754685	-1.36	0.174	-.5827897	.1052289
z6barlag	.0932053	.340593	0.27	0.784	-.5745336	.7609441
z7barlag	.5329691	.3701172	1.44	0.150	-.1926524	1.258591
_cons	12.86783	8.273149	1.56	0.120	-3.351834	29.08749

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	195,282	.4391322	.066687	.2247637	.8553479

Linear regression

Number of obs	**195,282**
F(129, 4281)	**129.20**
Prob > F	**0.0000**
R-squared	**0.3046**
Root MSE	**.19239**

(Std. err. adjusted for **4,282** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**5.439203**	**8.437017**	**0.64**	**0.519**	**-11.10172**	**21.98013**
__00009L	**-.560323**	**1.969256**	**-0.28**	**0.776**	**-4.421086**	**3.30044**
__00009O	**-1.151377**	**4.531356**	**-0.25**	**0.799**	**-10.03518**	**7.732428**
xi	**.1280307**	**.0130026**	**9.85**	**0.000**	**.102539**	**.1535225**
__000002	**-.0052441**	**.0046894**	**-1.12**	**0.264**	**-.0144378**	**.0039495**
__00000E	**-.1155163**	**.1132067**	**-1.02**	**0.308**	**-.3374601**	**.1064275**
__00000P	**-.1462888**	**.0680481**	**-2.15**	**0.032**	**-.2796984**	**-.0128792**
__000017	**.4345833**	**.2383273**	**1.82**	**0.068**	**-.0326618**	**.9018283**
__00001I	**.2394224**	**.1573674**	**1.52**	**0.128**	**-.0690994**	**.5479441**
__000020	**.0084234**	**.0589268**	**0.14**	**0.886**	**-.1071037**	**.1239506**
__00002B	**.0362406**	**.0349047**	**1.04**	**0.299**	**-.0321908**	**.1046719**
__00002T	**.2640651**	**.079803**	**3.31**	**0.001**	**.1076098**	**.4205204**
__000034	**.0015667**	**.0540939**	**0.03**	**0.977**	**-.1044854**	**.1076187**
__00003M	**-.0400876**	**.0732627**	**-0.55**	**0.584**	**-.1837204**	**.1035452**
__00003X	**.0055526**	**.046444**	**0.12**	**0.905**	**-.0855017**	**.096607**
__00004F	**-.0263637**	**.0546062**	**-0.48**	**0.629**	**-.1334201**	**.0806928**
__00004Q	**-.0003242**	**.0351143**	**-0.01**	**0.993**	**-.0691665**	**.0685181**
__000058	**-.0466603**	**.0826715**	**-0.56**	**0.573**	**-.2087392**	**.1154186**
__00005J	**.1017303**	**.0533565**	**1.91**	**0.057**	**-.0028761**	**.2063367**
__00005U	**-.3926164**	**.5235607**	**-0.75**	**0.453**	**-1.419067**	**.6338339**
__00005V	**.0769996**	**.0979041**	**0.79**	**0.432**	**-.1149433**	**.2689424**
__00005W	**-.0641166**	**.2185867**	**-0.29**	**0.769**	**-.4926598**	**.3644265**
__00005X	**.4054684**	**.5616211**	**0.72**	**0.470**	**-.6956001**	**1.506537**
__00005Y	**.0216321**	**.1284164**	**0.17**	**0.866**	**-.2301307**	**.2733949**
__00005Z	**.5232118**	**.1974632**	**2.65**	**0.008**	**.1360815**	**.910342**
__000060	**-.0380698**	**.0893003**	**-0.43**	**0.670**	**-.2131446**	**.1370049**
__000061	**.0346239**	**.054193**	**0.64**	**0.523**	**-.0716224**	**.1408702**
__000062	**.0163968**	**.041081**	**0.40**	**0.690**	**-.0641433**	**.0969368**
__000063	**-.0639632**	**.0529553**	**-1.21**	**0.227**	**-.167783**	**.0398567**
__000064	**-.0268231**	**.0644746**	**-0.42**	**0.677**	**-.1532268**	**.0995806**
__000065	**4.091551**	**1.069388**	**3.83**	**0.000**	**1.994996**	**6.188106**
__000066	**.4534598**	**.1952972**	**2.32**	**0.020**	**.070576**	**.8363436**
__000067	**-1.660848**	**.4419496**	**-3.76**	**0.000**	**-2.527298**	**-.7943975**
__000068	**-3.533975**	**1.143144**	**-3.09**	**0.002**	**-5.77513**	**-1.292819**
__000069	**-.2210461**	**.2683042**	**-0.82**	**0.410**	**-.7470613**	**.3049691**
__00006A	**-.2325463**	**.4402057**	**-0.53**	**0.597**	**-1.095578**	**.6304851**
__00006B	**.0792471**	**.1580984**	**0.50**	**0.616**	**-.2307077**	**.3892019**
__00006C	**-.6376054**	**.1131975**	**-5.63**	**0.000**	**-.8595313**	**-.4156796**
__00006D	**.1360327**	**.086029**	**1.58**	**0.114**	**-.0326287**	**.3046941**
__00006E	**.0219391**	**.1249639**	**0.18**	**0.861**	**-.2230548**	**.2669331**
__00006F	**.1766408**	**.156363**	**1.13**	**0.259**	**-.1299118**	**.4831933**
__00006G	**-.0302669**	**.2621416**	**-0.12**	**0.908**	**-.5442003**	**.4836664**
__00006H	**-.0002198**	**.0425971**	**-0.01**	**0.996**	**-.0837321**	**.0832925**
__00006I	**.0029321**	**.1035696**	**0.03**	**0.977**	**-.200118**	**.2059821**
__00006J	**-.0230365**	**.2685073**	**-0.09**	**0.932**	**-.5494501**	**.503377**
__00006K	**.0988179**	**.0627265**	**1.58**	**0.115**	**-.0241585**	**.2217942**
__00006L	**-.1366169**	**.1073481**	**-1.27**	**0.203**	**-.3470749**	**.0738411**
__00006M	**-.0204391**	**.0435557**	**-0.47**	**0.639**	**-.1058308**	**.0649526**
__00006N	**-.003578**	**.0270492**	**-0.13**	**0.895**	**-.0566085**	**.0494524**
__00006O	**.0061403**	**.0197607**	**0.31**	**0.756**	**-.032601**	**.0448815**
__00006P	**.0043417**	**.0295624**	**0.15**	**0.883**	**-.053616**	**.0622994**
__00006Q	**.0404013**	**.0346609**	**1.17**	**0.244**	**-.0275521**	**.1083547**
__00006R	**-.5571023**	**.4724726**	**-1.18**	**0.238**	**-1.483394**	**.3691889**
__00006S	**.0219314**	**.0569535**	**0.39**	**0.700**	**-.0897271**	**.1335898**
__00006T	**.2357088**	**.1664723**	**1.42**	**0.157**	**-.0906631**	**.5620808**
__00006U	**-.1020348**	**.4294443**	**-0.24**	**0.812**	**-.9439683**	**.7398986**
__00006V	**.1934944**	**.0924322**	**2.09**	**0.036**	**.0122794**	**.3747095**
__00006W	**.0635793**	**.1623983**	**0.39**	**0.695**	**-.2548055**	**.381964**
__00006X	**-.0073773**	**.0629564**	**-0.12**	**0.907**	**-.1308045**	**.1160498**
__00006Y	**.1171947**	**.0397766**	**2.95**	**0.003**	**.039212**	**.1951774**
__00006Z	**-.1065532**	**.0288086**	**-3.70**	**0.000**	**-.163033**	**-.0500733**

__000070	-.0794296	.0436765	-1.82	0.069	-.1650583	.0061991
__000071	.0243616	.0530909	0.46	0.646	-.0797241	.1284474
__000072	-.7255974	.3383489	-2.14	0.032	-1.388937	-.0622581
__000073	-.0791158	.055109	-1.44	0.151	-.1871579	.0289263
__000074	.3756923	.1421952	2.64	0.008	.096916	.6544686
__000075	.2829968	.3346673	0.85	0.398	-.3731246	.9391182
__000076	.1215848	.0769283	1.58	0.114	-.0292347	.2724042
__000077	.2607229	.1272463	2.05	0.041	.0112542	.5101916
__000078	-.0363359	.0490671	-0.74	0.459	-.1325329	.0598611
__000079	.0843746	.0378103	2.23	0.026	.0102468	.1585023
__00007A	.0484157	.0253211	1.91	0.056	-.0012268	.0980581
__00007B	.0715159	.0365842	1.95	0.051	-.0002081	.1432399
__00007C	.0916252	.043508	2.11	0.035	.006327	.1769235
__00007D	.1659837	.2472147	0.67	0.502	-.3186851	.6506526
__00007E	.0499496	.0407672	1.23	0.221	-.0299753	.1298745
__00007F	-.1361263	.1037625	-1.31	0.190	-.3395546	.067302
__00007G	.0263594	.2476484	0.11	0.915	-.4591598	.5118786
__00007H	-.0408588	.0596394	-0.69	0.493	-.157783	.0760653
__00007I	-.0767324	.1020413	-0.75	0.452	-.2767863	.1233215
__00007J	-.0279664	.0372818	-0.75	0.453	-.1010581	.0451253
__00007K	-.0673867	.0275482	-2.45	0.014	-.1213955	-.0133779
__00007L	.024324	.0209656	1.16	0.246	-.0167794	.0654275
__00007M	.0090948	.028044	0.32	0.746	-.045886	.0640756
__00007N	.0199372	.0328316	0.61	0.544	-.0444298	.0843043
__00007O	.192632	.4306766	0.45	0.655	-.6517173	1.036981
__00007P	.1927595	.0658687	2.93	0.003	.0636228	.3218963
__00007Q	-.3164722	.1714225	-1.85	0.065	-.6525491	.0196046
__00007R	.0388996	.4273645	0.09	0.927	-.7989563	.8767555
__00007S	-.1510172	.0909679	-1.66	0.097	-.3293615	.0273271
__00007T	-.4299568	.1549531	-2.77	0.006	-.7337453	-.1261684
__00007U	.0580428	.0557609	1.04	0.298	-.0512774	.1673631
__00007V	-.2099308	.0398959	-5.26	0.000	-.2881474	-.1317142
__00007W	-.0040343	.0290306	-0.14	0.889	-.0609492	.0528807
__00007X	.0116888	.044308	0.26	0.792	-.0751778	.0985554
__00007Y	.0155868	.0523245	0.30	0.766	-.0869963	.1181699
__000080	10.05086	19.55136	0.51	0.607	-28.27994	48.38166
__000082	-4.727773	10.07976	-0.47	0.639	-24.48933	15.03378
__000085	-.1321439	.5056997	-0.26	0.794	-1.123577	.8592895
__000087	.1006816	.375131	0.27	0.788	-.6347697	.8361328
__00008A	-1.640058	3.843595	-0.43	0.670	-9.175496	5.895381
__00008C	.7793676	1.951017	0.40	0.690	-3.045636	4.604372
__00008F	-12.49224	24.14608	-0.52	0.605	-59.83108	34.84659
__00008H	6.135525	12.84319	0.48	0.633	-19.04378	31.31483
__00008K	-.4949447	.665728	-0.74	0.457	-1.800117	.8102272
__00008M	-2.0956	.6227942	-3.36	0.001	-3.3166	-.8746008
__00008P	-1.487633	1.108948	-1.34	0.180	-3.661746	.6864798
__00008R	-.8570356	.9462338	-0.91	0.365	-2.712144	.998073
__00008U	-.4527762	.4364663	-1.04	0.300	-1.308476	.402924
__00008W	.107523	.3470776	0.31	0.757	-.572929	.7879749
__00008Z	.7350802	.2364273	3.11	0.002	.2715601	1.1986
__000091	-.1433496	.2035187	-0.70	0.481	-.5423516	.2556525
__000094	-.5229626	.2452563	-2.13	0.033	-1.003792	-.0421331
__000096	-.3912382	.1988635	-1.97	0.049	-.7811137	-.0013626
__000099	.8475826	.2940059	2.88	0.004	.2711786	1.423987
__00009B	.161125	.2645512	0.61	0.543	-.3575324	.6797824
__00009E	-.1245482	.3543252	-0.35	0.725	-.8192091	.5701128
__00009G	-.4130188	.2953828	-1.40	0.162	-.9921221	.1660845
xbarlag	-3.835202	23.05271	-0.17	0.868	-49.03046	41.36006
x2barlag	-.2402496	.557938	-0.43	0.667	-1.334097	.853598
xbar2lag	1.158026	4.580535	0.25	0.800	-7.822197	10.13825
rootxbarlag	4.160911	28.40881	0.15	0.884	-51.53508	59.8569
z1barlag	2.740105	.9747055	2.81	0.005	.8291772	4.651033
z2barlag	2.419492	1.56134	1.55	0.121	-.6415433	5.480526
z3barlag	.3131525	.5756933	0.54	0.586	-.8155047	1.44181
z4barlag	-.3707201	.2740526	-1.35	0.176	-.9080051	.166565
z5barlag	.9411127	.2485816	3.79	0.000	.4537639	1.428462
z6barlag	-1.05382	.4620898	-2.28	0.023	-1.959756	-.1478848
z7barlag	.4130635	.4979286	0.83	0.407	-.5631347	1.389262
_cons	-2.649469	9.73676	-0.27	0.786	-21.73857	16.43963

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A2	**195,282**	**.4385587**	**.1273027**	**.1088241**	**1.175086**

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"
957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9
960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
964 }
965
966 local r_RE      "`r_RE' `qhat_hats'"
967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }
978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
```

```
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS	Number of obs	=	195,282
				F(13, 195268)	=	85851.67
Model	8871.65252	13	682.434809	Prob > F	=	0.0000
Residual	1552.18508	195,268	.007948999	R-squared	=	0.8511
				Adj R-squared	=	0.8511
Total	10423.8376	195,281	.053378657	Root MSE	=	.08916

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	.0229268	.0008463	27.09	0.000	.0212681	.0245855
ybarx	-.0012198	.0008803	-1.39	0.166	-.0029452	.0005055
xi2	-.0395706	.000461	-85.85	0.000	-.040474	-.0386671
ybar	-.2104704	.0023427	-89.84	0.000	-.215062	-.2058788
xi	.4260222	.0013426	317.32	0.000	.4233908	.4286536
ybar_q1	.4578542	.0028207	162.32	0.000	.4523257	.4633827
pzi1	.2205423	.001061	207.87	0.000	.2184628	.2226217
pzi2	.0229202	.0017471	13.12	0.000	.0194959	.0263444
pzi3	-.0018793	.0005445	-3.45	0.001	-.0029464	-.0008122
pzi4	.0081199	.000558	14.55	0.000	.0070263	.0092135
pzi5	-.0134241	.0005633	-23.83	0.000	-.0145281	-.01232
pzi6	.0034407	.0004709	7.31	0.000	.0025177	.0043638
pzi7	-.0042157	.0006187	-6.81	0.000	-.0054283	-.003003
_cons	-.0301525	.0013735	-21.95	0.000	-.0328445	-.0274605

```
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = -_b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                  local coef = _b[pzi`j']
    3.                  local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
```

```
1025}

1026
1027if `J'==3 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00570997
  Iteration 1:  GMM criterion Q(b) =  .00078457
  Iteration 2:  GMM criterion Q(b) =  .00019385
  Iteration 3:  GMM criterion Q(b) =  .00014432
  Iteration 4:  GMM criterion Q(b) =  .00012914

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00811377
  Iteration 1:  GMM criterion Q(b) =  .00660025
  Iteration 2:  GMM criterion Q(b) =  .00647763

  GMM estimation

  Number of parameters =  20
  Number of moments    =  28
  Initial weight matrix: Unadjusted                 Number of obs   =    195,282
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 567 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
A11          |
         one |   .5580483   .0361521    15.44   0.000     .4871914    .6289051
-------------+----------------------------------------------------------------
C1           |
         zi1 |   .4634185   .0432554    10.71   0.000     .3786396    .5481975
         zi2 |  -.1606464   .0699637    -2.30   0.022    -.2977726   -.0235201
         zi3 |   .0000943   .0192217     0.00   0.996    -.0375796    .0377683
         zi4 |   .1458089   .0367076     3.97   0.000     .0738632    .2177545
         zi5 |   .0435282   .0233161     1.87   0.062    -.0021704    .0892269
         zi6 |   .0099479   .0177452     0.56   0.575     -.024832    .0447278
         zi7 |   .0002557   .0230576     0.01   0.991    -.0449363    .0454478
-------------+----------------------------------------------------------------
C2           |
         zi1 |   .2330917   .0696977     3.34   0.001     .0964866    .3696967
         zi2 |  -.3122623   .1131487    -2.76   0.006    -.5340297    -.090495
         zi3 |   .0017019   .0301408     0.06   0.955     -.057373    .0607768
         zi4 |   .2347928   .0534594     4.39   0.000     .1300142    .3395713
         zi5 |   .0902473   .0302828     2.98   0.003     .0308941    .1496005
```

```
         zi6 |   .0047865   .0262396     0.18   0.855    -.0466421    .0562151
         zi7 |   .0037342   .0383552     0.10   0.922    -.0714406    .0789089
-------------+----------------------------------------------------------------
         /d1 |  -.0845425   .0063545   -13.30   0.000    -.0969971   -.0720879
         /b1 |   .3956831   .0186523    21.21   0.000     .3591252    .4322409
      /AVA11 |   .5963418   .6637408     0.90   0.369    -.7045662     1.89725
      /AVA12 |  -.6032887    .595297    -1.01   0.311    -1.770049    .5634721
      /AVA22 |   .3086459   .5658364     0.55   0.585     -.800373    1.417665
------------------------------------------------------------------------------

Instruments for equation eq1_RE: __00009J __00009L __00009O xi __000002 __00000E
    __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M
    __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5
    __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
```

```
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Asame_fsuXseg_drop3_all_exactInst_noPPXdInst
      > _14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not found)
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXseg_drop3_all_exactIn
  > st_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  29462
  Number of records is  195282

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
                 e(N_hh) =  29462

1077unique group_round
  Number of unique values of group_round is  4282
  Number of records is  195282

1078estadd scalar N_grp = r(sum)

  added scalar:
                e(N_grp) =  4282

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
         e(avg_grp_size) =  6.8804297

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXseg_drop3_all_exactInst_noPP
  > XdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXseg_drop3_all_exactInst_noPPXdInst_
  > 14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
```

```
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*`p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
```

```
                                                    gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                 local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                         }
   30.                                 }
   31.                         }
   32.                 }
   33.         }
        if `Tindivp1'<=`T' {
                forval t=`Tindivp1'/`T' {
                        tempvar zi`t'xdiff
                        g double `zi`t'xdiff'=zi`t'*`ximxk'
                        local zg "`zg' zi`t'"
                        forval j=1/`J' {
                                tempvar zi`t'p`j'xdiff
                                g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
                                local pzg "`pzg' `zi`t'p`j''"
                                local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
                        }
                }
        }

        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
        // denoting zg as group-zs
        //  zg,pzgx,rprp,rprpx
        local rootprootp ""
        local rootprootpxdiff ""

        forval j=1/`J' {
                forval s=`j'/`J' {
                        tempvar xdiffrp`j'rp`s'
                        g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')

                     if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "

                     local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
                }
         }

        local pqhatxdiff ""
        forval j=1/`J' {
                forval s=1/`J' {
                        foreach name of global Alist {
                                tempvar   p`s'qikhat`j'xdiff`name'
                                g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'

                                local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
                        }
                }
         }

        // set instruments
        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
}
```

```
1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                tempvar ximxk`var'
    3.                g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                        local `++t'
    3.                        local p_nice = subinstr("`p'",":_cons","",.)
    4.                        local est = coefs[1,`t']
    5.                        if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                        if `j'>`i' {
   11.                                tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                        }
   15.                }
   16.         }
```

```
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >     `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1     `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
```

```
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =   .0008534
  Iteration 1:  GMM criterion Q(b) =  .00035072
  Iteration 2:  GMM criterion Q(b) =  .00026857

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01202369
  Iteration 1:  GMM criterion Q(b) =  .01107694
  Iteration 2:  GMM criterion Q(b) =   .0109312
  Iteration 3:  GMM criterion Q(b) =  .01091184

  GMM estimation

  Number of parameters =  17
  Number of moments    =  63
  Initial weight matrix: Unadjusted                 Number of obs   =    195,282
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 567 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
C11          |
       _cons |   .6283512   .0259823    24.18   0.000     .5774268    .6792755
-------------+----------------------------------------------------------------
C12          |
       _cons |  -.0306178   .0408795    -0.75   0.454    -.1107402    .0495045
-------------+----------------------------------------------------------------
C13          |
       _cons |  -.0143428   .0108988    -1.32   0.188    -.0357041    .0070185
-------------+----------------------------------------------------------------
C14          |
       _cons |   .0054945   .0116071     0.47   0.636    -.0172549     .028244
-------------+----------------------------------------------------------------
C15          |
       _cons |  -.0121923   .0181145    -0.67   0.501    -.0476962    .0233115
------------------------------------------------------------------------------
```

C16						
_cons	**-.014725**	**.011952**	**-1.23**	**0.218**	**-.0381505**	**.0087006**
C17						
_cons	**-.0089869**	**.0141285**	**-0.64**	**0.525**	**-.0366782**	**.0187044**
C21						
_cons	**.4684317**	**.0372166**	**12.59**	**0.000**	**.3954885**	**.5413749**
C22						
_cons	**-.0858763**	**.0591634**	**-1.45**	**0.147**	**-.2018344**	**.0300818**
C23						
_cons	**-.0207376**	**.0148123**	**-1.40**	**0.162**	**-.0497692**	**.008294**
C24						
_cons	**-.0035097**	**.0194081**	**-0.18**	**0.856**	**-.0415489**	**.0345296**
C25						
_cons	**.0100697**	**.0228478**	**0.44**	**0.659**	**-.0347112**	**.0548507**
C26						
_cons	**-.023591**	**.0158996**	**-1.48**	**0.138**	**-.0547536**	**.0075717**
C27						
_cons	**-.0044993**	**.0204382**	**-0.22**	**0.826**	**-.0445574**	**.0355589**
d1						
_cons	**-.0828039**	**.0062863**	**-13.17**	**0.000**	**-.0951249**	**-.070483**
b1						
_cons	**.441251**	**.016015**	**27.55**	**0.000**	**.4098621**	**.4726399**
A11						
one	**.0876088**	**.1207494**	**0.73**	**0.468**	**-.1490557**	**.3242732**

Instruments for equation **eq1_FE: __000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7 __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1 __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY __0000D5 _cons**

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
```

(file **diag_2good_norm_main_nopcross_state_Asame_fsuXseg_drop3_all_exactInst_noPPXdInst > _14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex** not found)
(output written to diag_2good_norm_main_nopcross_state_Asame_fsuXseg_drop3_all_exactIn > st_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex)

```
1242
1243        estat overid
```

Test of overidentifying restriction:

Hansen's J chi2(**46**) = **2130.89** (p = **0.0000**)

```
1244        di "$S_DATE $S_TIME"
```

7 Dec 2023 16:18:44

```
1245        unique hhidi
  Number of unique values of hhidi is  29462
  Number of records is  195282
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  29462
1248        unique group_round
  Number of unique values of group_round is  4282
  Number of records is  195282
1249        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4282
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  6.8804297
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXseg_drop3_all_exactInst_noPP
  > XdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXseg_drop3_all_exactInst_noPPXdInst_
  > 14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}

1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
```

```
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table2_25.smcl
   log type:  smcl
  closed on:   7 Dec 2023, 16:18:44
-------------------------------------------------------------------------------
```

A.2.1.2 Panel B

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table2_25b.smcl
  log type:  smcl
 opened on:   8 Dec 2023, 13:17:22

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
. local grp_type ""

. if $rand_grp==1 local grp_type "_Random"

.
. // make summary stats table
. local make_sumstats = 0

.
. // measure expenditure in per capita terms
. global meas_pc = 0

. local typeExp ""

. if $meas_pc==1 local typeExp "_ExpPC"

.
. // number of goods
. scalar J=2

. local J = J

.
. // group def options
. //local group_def "dist"
. local group_def "fsuXseg"

. //local group_def "fsuXsegXreligXsched"
.
. // comparision group
. local groupComp_def "fsuXsegXreligXsched"

. local desc_compGrp ""

. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

.
. // load data
. include "$CODE/prep demand system data.do"

. //clear all
. pause on

. set more off

. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

.
. scalar short_zlist=$short_zlist_val

.
. cap prog drop bysmeanw

. prog def bysmeanw
1.          syntax varname, weight(varname) by(varlist) [rename(name)]
2.
.         if "`rename'" == "" local rename = "`varlist'bar"
3.
```

```
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations            1,118    ->   280
  Number of variables                   5    ->   7
  j variable (4 values)       expend_type    ->   (dropped)
  xij variables:
                         laspeyres_state    ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  -----------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations              560    ->   280
  Number of variables                   5    ->   5
  j variable (2 values)       expend_type    ->   (dropped)
  xij variables:
                         laspeyres_state    ->   laspeyres_state1 laspeyres_state2
  -----------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations               840   ->   280
  Number of variables                    5   ->   6
  j variable (3 values)       expend_type   ->   (dropped)
  xij variables:
                         laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg
  (43,637 missing values generated)

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"       egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                  egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)
  (43,637 missing values generated)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)
  (43,637 missing values generated)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }
```

```
236
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
```

```
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }

297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }

316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }

333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
```

```
351
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.         drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.         }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.         g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (82 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,576 real changes made, 82 to missing)

460 replace x2=x*x
  (73,576 real changes made, 82 to missing)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,576 real changes made)

478 replace x2=x2_temp
  (73,576 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (44,114 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3
  (4,705 observations deleted)

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)
```

```
488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_4630_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
```

```
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,757 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
549 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var'" _n
    5.                 }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
```

```
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1
```

```
625 scalar estimate_FE=1

626
627 // estimation options
628 global w_initial "winit(unadjusted)"

629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

630 global wmatrix "wmatrix(robust)"

631 global trace_level ""

632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

633 global technique ""

634 global maxiter ""

635 global technique_RE ""

636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

637 global derivatives "quickderivatives"

638 scalar init_FE_from_RE=1

639 global clust_var "state_district_round"

640 local clus_name "_clus_Dist"

641
642
643
644
645
646
647
648
649 // xxzx to keep names same
650 local wmatrix_desc ""

651 local sectorName "_urbOnly"

652 local dropZdesc "_dropPre"

653 local desc_struc_v0 ""

654 local inst_desc "_exactInst"

655 local inst_rootp "_noPPXdInst"

656 local weight_desc ""

657 local RE_sq_inst_desc ""

658 local cnstname ""

659 local actual_pqhat_desc ""

660 local expUpname "_updateExp"
```

```
661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
  (0 observations deleted)
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //               ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T
```

```
694 local Tm1=`T'-1

695 local Tindiv : word count $zlist_indiv

696
697 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
706
707 if `J'==2 {
708          replace p1=P1
  (150,184 real changes made)
709          replace p2=P2
  (150,184 real changes made)
710          replace qi1=Qi1
  (150,173 real changes made)
711          replace qk1=Qk1
  (150,173 real changes made)
712          replace qikbar1=Qikbar1
  (150,184 real changes made)
713          replace qi2=Qi2
  (150,184 real changes made)
714          replace qk2=Qk2
  (150,184 real changes made)
715          replace qikbar2=Qikbar2
  (150,184 real changes made)
716
717          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                   qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721          forv j = 1(1)3 {
    2.                   replace p`j'=P`j'
    3.                   replace qi`j'=Qi`j'
    4.                   replace qk`j'=Qk`j'
    5.                   replace qikbar`j'=Qikbar`j'
    6.          }
722
723          foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                   qui replace `var'=.
    3.          }
724 }
```

```
725
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }
```

```
751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
773         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.         local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.         if `jplus1'<=`J' {
    5.                 forval k=`jplus1'/`J' {
    6.                         local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                 }
    8.         }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                 if noMeasError==1                                local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.         }
```

```
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                      l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                    ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2
  > ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qikba
  > r1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d
  > 1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > }*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))        ))
```

```
836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) +
  >  ({A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C
  > 2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikb
  > ar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:
  > }) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C
  > 2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qi
  > kbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )
  > *{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > one}*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}
  > *ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi
  > 4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p
  > 1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         )
  > )

874
875 /*******************************************************************************
  >  construct instruments
  > *******************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi

882 g double `ximxk'=xi-xk
```

```
883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.         tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.         g double `xizi`t''=xi*zi`t'
    4.         g double `xkzk`t''=xk*zk`t'
    5.         g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.         g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.         g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.         g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.         g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.         forval j=1/`J' {
   11.                 tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                 g double `zi`t'p`j''= zi`t'*p`j'
   13.                 g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                 g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                 g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                 forv s=1/`T' {
   17.                         tempvar zi`t'zi`s'p`j'
   18.                         g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                 }
   20.         }
   21. }

891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                 local rg "`rg' z`t'barlag"
    3. }

903
904 local zrg ""

905 forval t=1/`Tindiv' {
    2.         foreach var of varlist `rg' {
    3.                 tempvar zi`t'`var'
    4.                 g double `zi`t'`var''=zi`t'*`var'
    5.                 local zrg "`zrg' `zi`t'`var''"
    6.         }
    7. }
```

```
906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var''"
    5.
911          forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var''"
   10.                  local xprg "`xprg' `xp`j'`var''"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""

925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }
```

```
929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.         forval k=`j'/`J' {
    3.                 tempvar p`j'p`k'
    4.                 g `p`j'p`k''=p`j'*p`k'
    5.                 local roundinsts "`roundinsts' `p`j'p`k''"
    6.         }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.         if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.         if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }

  Linear regression                               Number of obs     =    150,184
                                                  F(128, 3940)      =          .
                                                  Prob > F          =          .
                                                  R-squared         =     0.2431
                                                  Root MSE          =     .11811
```

(Std. err. adjusted for **3,941** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	-8.796917	7.034192	-1.25	0.211	-22.58792	4.994082
__00009L	-3.764209	1.230855	-3.06	0.002	-6.177382	-1.351036
__00009O	1.078146	3.627116	0.30	0.766	-6.033055	8.189348
xi	.1000276	.0074252	13.47	0.000	.0854699	.1145853
__000002	-.0144237	.0022978	-6.28	0.000	-.0189286	-.0099187
__00000E	-.1355399	.0853439	-1.59	0.112	-.3028622	.0317825
__00000P	-.0350062	.0492259	-0.71	0.477	-.1315168	.0615044
__000017	.0463156	.1616702	0.29	0.775	-.2706495	.3632808
__00001I	.0103831	.1000978	0.10	0.917	-.1858653	.2066315
__000020	.0036667	.0442548	0.08	0.934	-.0830978	.0904312
__00002B	-.021251	.0225827	-0.94	0.347	-.0655259	.0230239
__00002T	.0932379	.0615338	1.52	0.130	-.0274032	.213879
__000034	-.0198576	.042829	-0.46	0.643	-.1038268	.0641115
__00003M	.0419562	.0527057	0.80	0.426	-.0613768	.1452892
__00003X	-.0403197	.035484	-1.14	0.256	-.1098885	.0292491
__00004F	-.0115768	.0443277	-0.26	0.794	-.0984842	.0753305
__00004Q	.0129628	.0235416	0.55	0.582	-.0331921	.0591177
__000058	-.0409803	.0580415	-0.71	0.480	-.1547744	.0728139
__00005J	.0203487	.0324366	0.63	0.530	-.0432454	.0839428
__00005U	-.0703763	.3808458	-0.18	0.853	-.8170497	.6762972
__00005V	-.0222761	.0644924	-0.35	0.730	-.1487177	.1041655
__00005W	.0031555	.1515545	0.02	0.983	-.2939771	.3002881
__00005X	.2312495	.4033821	0.57	0.566	-.5596078	1.022107
__00005Y	-.0429951	.0928682	-0.46	0.643	-.2250693	.1390791
__00005Z	.0556336	.1501553	0.37	0.711	-.2387558	.350023
__000060	.0326034	.0553248	0.59	0.556	-.0758646	.1410714
__000061	.0334185	.049067	0.68	0.496	-.0627806	.1296175
__000062	.0772437	.0312397	2.47	0.013	.0159962	.1384912
__000063	-.0376898	.0422079	-0.89	0.372	-.1204412	.0450617
__000064	-.0375722	.0518018	-0.73	0.468	-.1391331	.0639887
__000065	-.0933208	.8198473	-0.11	0.909	-1.700686	1.514044
__000066	.1816777	.1324913	1.37	0.170	-.0780802	.4414357
__000067	-.1779303	.3202878	-0.56	0.579	-.8058757	.450015
__000068	.1154195	.8004076	0.14	0.885	-1.453833	1.684672
__000069	-.1337839	.1849809	-0.72	0.470	-.4964512	.2288834
__00006A	-.1522514	.3010585	-0.51	0.613	-.7424965	.4379938
__00006B	.0083696	.1138287	0.07	0.941	-.2147991	.2315382
__00006C	-.1926544	.1023308	-1.88	0.060	-.3932807	.0079719
__00006D	.0668672	.0558923	1.20	0.232	-.0427134	.1764478
__00006E	-.0809838	.094059	-0.86	0.389	-.2653926	.103425
__00006F	.1173735	.118564	0.99	0.322	-.1150792	.3498261
__00006G	-.2256199	.1599718	-1.41	0.159	-.5392551	.0880153
__00006H	.0115792	.0293665	0.39	0.693	-.0459957	.0691541
__00006I	.0255117	.0636135	0.40	0.688	-.0992068	.1502301
__00006J	.2422553	.1878835	1.29	0.197	-.1261027	.6106133
__00006K	.0636527	.0470614	1.35	0.176	-.0286143	.1559198
__00006L	-.0546252	.0772868	-0.71	0.480	-.2061512	.0969007
__00006M	-.0549227	.0275047	-2.00	0.046	-.1088474	-.0009979
__00006N	.0291602	.0219285	1.33	0.184	-.013832	.0721523
__00006O	-.0237378	.0131444	-1.81	0.071	-.0495082	.0020326
__00006P	-.0155396	.0235679	-0.66	0.510	-.0617461	.0306669
__00006Q	.012124	.0277498	0.44	0.662	-.0422813	.0665292
__00006R	-.4652835	.4340837	-1.07	0.284	-1.316333	.3857663
__00006S	-.0237404	.0518065	-0.46	0.647	-.1253105	.0778297
__00006T	.228297	.1647202	1.39	0.166	-.0946478	.5512418
__00006U	.2281641	.3857761	0.59	0.554	-.5281755	.9845036
__00006V	-.1276221	.0713369	-1.79	0.074	-.2674829	.0122386
__00006W	.0124841	.1339121	0.09	0.926	-.2500596	.2750277
__00006X	.0086718	.0529041	0.16	0.870	-.0950502	.1123937
__00006Y	.0479063	.0341571	1.40	0.161	-.0190609	.1148735
__00006Z	-.0709526	.0228975	-3.10	0.002	-.1158447	-.0260606
__000070	-.013299	.0408222	-0.33	0.745	-.0933336	.0667355
__000071	-.0327481	.0461505	-0.71	0.478	-.1232292	.057733
__000072	-.3957121	.2731515	-1.45	0.148	-.9312437	.1398195
__000073	-.0692937	.0438799	-1.58	0.114	-.1553231	.0167358
__000074	.2667135	.1111481	2.40	0.016	.0488003	.4846268
__000075	.1596342	.2708665	0.59	0.556	-.3714174	.6906858

__000076	-.0289989	.0590864	-0.49	0.624	-.1448418	.0868439
__000077	.126095	.1023337	1.23	0.218	-.074537	.326727
__000078	.0042598	.0397476	0.11	0.915	-.073668	.0821876
__000079	.0057907	.0325545	0.18	0.859	-.0580346	.0696159
__00007A	.0110381	.0206927	0.53	0.594	-.0295313	.0516074
__00007B	.0329142	.0283928	1.16	0.246	-.0227517	.0885802
__00007C	-.0152101	.0351555	-0.43	0.665	-.0841349	.0537147
__00007D	.158262	.1892885	0.84	0.403	-.2128506	.5293745
__00007E	.0093531	.0316342	0.30	0.768	-.0526679	.0713741
__00007F	-.0695719	.073096	-0.95	0.341	-.2128815	.0737376
__00007G	-.0990247	.2079715	-0.48	0.634	-.5067667	.3087173
__00007H	.0102521	.0504914	0.20	0.839	-.0887397	.1092438
__00007I	-.0267836	.0826793	-0.32	0.746	-.1888819	.1353146
__00007J	-.0077047	.0307359	-0.25	0.802	-.0679644	.052555
__00007K	-.0275353	.0258652	-1.06	0.287	-.0782457	.023175
__00007L	.0188307	.0152704	1.23	0.218	-.0111079	.0487694
__00007M	.0027078	.0261276	0.10	0.917	-.048517	.0539326
__00007N	.0549995	.0302891	1.82	0.069	-.0043843	.1143832
__00007O	.0379662	.3320731	0.11	0.909	-.6130852	.6890176
__00007P	.0910566	.0447932	2.03	0.042	.0032366	.1788766
__00007Q	-.1434121	.1243705	-1.15	0.249	-.3872488	.1004245
__00007R	.0448395	.3092313	0.15	0.885	-.561429	.651108
__00007S	.0245799	.0653179	0.38	0.707	-.1034802	.15264
__00007T	-.2048683	.1098351	-1.87	0.062	-.4202073	.0104707
__00007U	.0161166	.0393365	0.41	0.682	-.0610052	.0932385
__00007V	-.0722283	.0319521	-2.26	0.024	-.1348726	-.009584
__00007W	.0058943	.0190409	0.31	0.757	-.0314367	.0432253
__00007X	.0247797	.0336215	0.74	0.461	-.0411374	.0906969
__00007Y	.0645344	.0405775	1.59	0.112	-.0150205	.1440894
__000080	-26.69175	16.53843	-1.61	0.107	-59.11644	5.732943
__000082	.0527788	8.007264	0.01	0.995	-15.64599	15.75155
__000085	-.4552559	.3828573	-1.19	0.234	-1.205873	.2953611
__000087	.4568422	.2638796	1.73	0.083	-.0605112	.9741957
__00008A	5.901538	3.231707	1.83	0.068	-.4344368	12.23751
__00008C	-1.07043	1.511183	-0.71	0.479	-4.033204	1.892344
__00008F	32.13742	20.35128	1.58	0.114	-7.762615	72.03746
__00008H	2.111943	10.25783	0.21	0.837	-17.99922	22.22311
__00008K	-.1443268	.4888127	-0.30	0.768	-1.102676	.8140228
__00008M	-1.082298	.4602793	-2.35	0.019	-1.984706	-.1798895
__00008P	-1.372353	.8857554	-1.55	0.121	-3.108936	.3642286
__00008R	-.6238773	.6429644	-0.97	0.332	-1.884452	.6366971
__00008U	.3790673	.3099321	1.22	0.221	-.2285751	.9867098
__00008W	.1032206	.2226208	0.46	0.643	-.3332423	.5396835
__00008Z	.7078229	.2290496	3.09	0.002	.258756	1.15689
__000091	-.2784454	.1597645	-1.74	0.081	-.5916743	.0347835
__000094	.1718188	.1831362	0.94	0.348	-.1872317	.5308694
__000096	.0595339	.1392413	0.43	0.669	-.213458	.3325257
__000099	.0697814	.2608128	0.27	0.789	-.4415594	.5811222
__00009B	-.2191422	.1956148	-1.12	0.263	-.6026579	.1643735
__00009E	-.0489392	.3039811	-0.16	0.872	-.6449143	.5470358
__00009G	-.5707256	.2348141	-2.43	0.015	-1.031094	-.110357
xbarlag	27.44746	20.51956	1.34	0.181	-12.78249	67.67741
x2barlag	-.2879211	.4247021	-0.68	0.498	-1.120578	.5447356
xbar2lag	-4.525775	4.028161	-1.12	0.261	-12.42325	3.371701
rootxbarlag	-35.31371	25.31408	-1.40	0.163	-84.94364	14.31622
z1barlag	1.411787	.7303894	1.93	0.053	-.0201901	2.843763
z2barlag	2.003292	1.164019	1.72	0.085	-.2788446	4.285428
z3barlag	-.5598997	.4087547	-1.37	0.171	-1.36129	.2414909
z4barlag	-.3413605	.2671899	-1.28	0.201	-.8652041	.1824831
z5barlag	-.3141638	.1873291	-1.68	0.094	-.681435	.0531073
z6barlag	.1541822	.3667657	0.42	0.674	-.5648863	.8732507
z7barlag	.5279346	.4013801	1.32	0.188	-.2589977	1.314867
_cons	12.18517	8.679136	1.40	0.160	-4.830853	29.20119

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	150,184	.4393224	.0669084	.2214191	.7909912

Linear regression

Number of obs	**150,184**
F(129, 3940)	**19.28**
Prob > F	**0.0000**
R-squared	**0.3160**
Root MSE	**.1858**

(Std. err. adjusted for **3,941** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**6.089608**	**8.620672**	**0.71**	**0.480**	**-10.81179**	**22.99101**
__00009L	**.464938**	**2.096608**	**0.22**	**0.825**	**-3.645601**	**4.575477**
__00009O	**-.338558**	**4.828432**	**-0.07**	**0.944**	**-9.805018**	**9.127902**
xi	**.1344222**	**.0129629**	**10.37**	**0.000**	**.1090077**	**.1598368**
__000002	**-.0076142**	**.0048075**	**-1.58**	**0.113**	**-.0170397**	**.0018112**
__00000E	**-.1644752**	**.1218907**	**-1.35**	**0.177**	**-.40345**	**.0744996**
__00000P	**-.1549225**	**.071221**	**-2.18**	**0.030**	**-.2945559**	**-.0152891**
__000017	**.3068592**	**.2621254**	**1.17**	**0.242**	**-.207055**	**.8207734**
__00001I	**.2437801**	**.1621552**	**1.50**	**0.133**	**-.0741358**	**.5616961**
__000020	**.0582368**	**.0645843**	**0.90**	**0.367**	**-.0683849**	**.1848586**
__00002B	**.0376458**	**.0361744**	**1.04**	**0.298**	**-.0332764**	**.1085681**
__00002T	**.2798033**	**.0852084**	**3.28**	**0.001**	**.1127466**	**.44686**
__000034	**-.0113675**	**.0564181**	**-0.20**	**0.840**	**-.121979**	**.099244**
__00003M	**-.043527**	**.0803174**	**-0.54**	**0.588**	**-.2009946**	**.1139405**
__00003X	**.0079431**	**.0491422**	**0.16**	**0.872**	**-.0884033**	**.1042896**
__00004F	**-.0191394**	**.0603826**	**-0.32**	**0.751**	**-.1375235**	**.0992448**
__00004Q	**-.010345**	**.0364253**	**-0.28**	**0.776**	**-.0817591**	**.0610692**
__000058	**-.0717195**	**.0891396**	**-0.80**	**0.421**	**-.2464836**	**.1030446**
__00005J	**.0854146**	**.0552764**	**1.55**	**0.122**	**-.0229584**	**.1937877**
__00005U	**-.688783**	**.5818106**	**-1.18**	**0.237**	**-1.829461**	**.4518953**
__00005V	**.0163691**	**.1051546**	**0.16**	**0.876**	**-.1897935**	**.2225316**
__00005W	**.0968475**	**.240401**	**0.40**	**0.687**	**-.3744746**	**.5681696**
__00005X	**.6631304**	**.6180525**	**1.07**	**0.283**	**-.5486026**	**1.874863**
__00005Y	**.0142267**	**.1374351**	**0.10**	**0.918**	**-.2552239**	**.2836773**
__00005Z	**.5697296**	**.2067679**	**2.76**	**0.006**	**.1643474**	**.9751119**
__000060	**-.083225**	**.0926459**	**-0.90**	**0.369**	**-.2648634**	**.0984135**
__000061	**.0476409**	**.0567774**	**0.84**	**0.401**	**-.063675**	**.1589567**
__000062	**.0157185**	**.0439877**	**0.36**	**0.721**	**-.0705223**	**.1019593**
__000063	**-.0057237**	**.0575246**	**-0.10**	**0.921**	**-.1185044**	**.107057**
__000064	**-.0034996**	**.0698407**	**-0.05**	**0.960**	**-.1404268**	**.1334277**
__000065	**3.716272**	**1.218455**	**3.05**	**0.002**	**1.32741**	**6.105133**
__000066	**.4143888**	**.218434**	**1.90**	**0.058**	**-.0138654**	**.8426431**
__000067	**-1.57466**	**.5041342**	**-3.12**	**0.002**	**-2.563048**	**-.5862713**
__000068	**-2.835514**	**1.254517**	**-2.26**	**0.024**	**-5.295077**	**-.3759511**
__000069	**-.4478759**	**.2878574**	**-1.56**	**0.120**	**-1.012239**	**.1164876**
__00006A	**-.165373**	**.4781971**	**-0.35**	**0.729**	**-1.10291**	**.772164**
__00006B	**-.0476723**	**.170343**	**-0.28**	**0.780**	**-.3816411**	**.2862965**
__00006C	**-.6318021**	**.1165253**	**-5.42**	**0.000**	**-.8602577**	**-.4033466**
__00006D	**.1274947**	**.0899064**	**1.42**	**0.156**	**-.0487726**	**.3037621**
__00006E	**-.052264**	**.137919**	**-0.38**	**0.705**	**-.3226634**	**.2181354**
__00006F	**.1271027**	**.1730808**	**0.73**	**0.463**	**-.2122336**	**.466439**
__00006G	**.0383991**	**.2676739**	**0.14**	**0.886**	**-.4863933**	**.5631915**
__00006H	**.0283894**	**.0444075**	**0.64**	**0.523**	**-.0586745**	**.1154532**
__00006I	**-.0534568**	**.1066442**	**-0.50**	**0.616**	**-.2625397**	**.1556262**
__00006J	**-.0883998**	**.2806149**	**-0.32**	**0.753**	**-.6385638**	**.4617643**
__00006K	**.1348518**	**.0690825**	**1.95**	**0.051**	**-.000589**	**.2702926**
__00006L	**-.1557193**	**.1105546**	**-1.41**	**0.159**	**-.3724689**	**.0610303**
__00006M	**-.0487534**	**.0473166**	**-1.03**	**0.303**	**-.1415208**	**.0440141**
__00006N	**.0018231**	**.0283442**	**0.06**	**0.949**	**-.0537477**	**.0573938**
__00006O	**-.0050388**	**.021137**	**-0.24**	**0.812**	**-.0464793**	**.0364017**
__00006P	**-.0285372**	**.0315215**	**-0.91**	**0.365**	**-.0903371**	**.0332628**
__00006Q	**.0191431**	**.0364472**	**0.53**	**0.599**	**-.052314**	**.0906002**
__00006R	**-.4141256**	**.5006414**	**-0.83**	**0.408**	**-1.395666**	**.5674149**
__00006S	**-.0129503**	**.0676917**	**-0.19**	**0.848**	**-.1456643**	**.1197637**
__00006T	**.2307477**	**.1807872**	**1.28**	**0.202**	**-.1236975**	**.5851929**
__00006U	**-.1792796**	**.4663031**	**-0.38**	**0.701**	**-1.093498**	**.7349386**
__00006V	**.1609471**	**.0979382**	**1.64**	**0.100**	**-.0310672**	**.3529615**
__00006W	**.0969977**	**.1739086**	**0.56**	**0.577**	**-.2439617**	**.4379571**
__00006X	**-.0205288**	**.0702537**	**-0.29**	**0.770**	**-.1582659**	**.1172082**
__00006Y	**.1086966**	**.0429847**	**2.53**	**0.011**	**.0244222**	**.1929711**
__00006Z	**-.1240148**	**.0306997**	**-4.04**	**0.000**	**-.1842036**	**-.063826**

__000070	-.076898	.0486952	-1.58	0.114	-.1723683	.0185722
__000071	.0200262	.06071	0.33	0.742	-.0989998	.1390523
__000072	-.7362104	.3724669	-1.98	0.048	-1.466456	-.0059643
__000073	-.0863865	.0644319	-1.34	0.180	-.2127095	.0399365
__000074	.3868157	.1597273	2.42	0.015	.0736597	.6999717
__000075	.2927027	.3753663	0.78	0.436	-.4432278	1.028633
__000076	.1348562	.0832216	1.62	0.105	-.0283052	.2980177
__000077	.266676	.1405454	1.90	0.058	-.0088725	.5422245
__000078	-.0579151	.0528354	-1.10	0.273	-.1615023	.0456722
__000079	.0970843	.0391333	2.48	0.013	.0203609	.1738077
__00007A	.0421752	.0270203	1.56	0.119	-.0107999	.0951502
__00007B	.0888308	.0386446	2.30	0.022	.0130656	.164596
__00007C	.1000081	.0488627	2.05	0.041	.0042096	.1958067
__00007D	.2487231	.266747	0.93	0.351	-.2742521	.7716983
__00007E	.0573584	.0453456	1.26	0.206	-.0315446	.1462614
__00007F	-.1546601	.1103663	-1.40	0.161	-.3710405	.0617204
__00007G	-.0694054	.2721041	-0.26	0.799	-.6028836	.4640728
__00007H	-.0530957	.066168	-0.80	0.422	-.1828224	.0766309
__00007I	-.0165483	.1094955	-0.15	0.880	-.2312214	.1981249
__00007J	-.0232984	.040649	-0.57	0.567	-.1029935	.0563967
__00007K	-.0570372	.0296343	-1.92	0.054	-.1151372	.0010627
__00007L	.010958	.0217214	0.50	0.614	-.0316283	.0535442
__00007M	.0296	.0303417	0.98	0.329	-.029887	.089087
__00007N	.0150273	.0368631	0.41	0.684	-.0572452	.0872999
__00007O	.2435464	.4377392	0.56	0.578	-.6146703	1.101763
__00007P	.2094575	.0708466	2.96	0.003	.0705579	.348357
__00007Q	-.3669741	.1760288	-2.08	0.037	-.7120901	-.021858
__00007R	.0480973	.4410605	0.11	0.913	-.8166311	.9128256
__00007S	-.1654594	.0988801	-1.67	0.094	-.3593204	.0284015
__00007T	-.3892751	.1658977	-2.35	0.019	-.7145286	-.0640216
__00007U	.0477318	.0608804	0.78	0.433	-.0716283	.1670918
__00007V	-.19278	.0424009	-4.55	0.000	-.2759098	-.1096502
__00007W	-.0107137	.0307982	-0.35	0.728	-.0710956	.0496683
__00007X	.0412066	.0475265	0.87	0.386	-.0519723	.1343855
__00007Y	.0324403	.0563942	0.58	0.565	-.0781244	.1430049
__000080	13.63042	19.69139	0.69	0.489	-24.97585	52.23669
__000082	-2.551455	10.79781	-0.24	0.813	-23.72128	18.61837
__000085	.1960833	.5327866	0.37	0.713	-.8484801	1.240647
__000087	.010633	.3867265	0.03	0.978	-.74757	.7688359
__00008A	-2.928065	3.820511	-0.77	0.443	-10.41843	4.562299
__00008C	.6684936	2.101185	0.32	0.750	-3.451019	4.788006
__00008F	-16.5049	24.43674	-0.68	0.499	-64.41475	31.40495
__00008H	3.053908	13.70234	0.22	0.824	-23.81043	29.91825
__00008K	.1295521	.6887722	0.19	0.851	-1.220831	1.479936
__00008M	-1.704772	.6459334	-2.64	0.008	-2.971167	-.4383764
__00008P	-1.635151	1.143278	-1.43	0.153	-3.876624	.6063226
__00008R	-1.289457	.9498686	-1.36	0.175	-3.151737	.5728233
__00008U	-.4250607	.4266047	-1.00	0.319	-1.261447	.4113261
__00008W	.0449181	.3684055	0.12	0.903	-.6773652	.7672015
__00008Z	.6391104	.2444237	2.61	0.009	.1599015	1.118319
__000091	-.0501266	.2046879	-0.24	0.807	-.4514307	.3511775
__000094	-.4826167	.2566175	-1.88	0.060	-.9857322	.0204989
__000096	-.3487109	.2036286	-1.71	0.087	-.7479383	.0505165
__000099	.784776	.3094548	2.54	0.011	.1780694	1.391483
__00009B	.0034317	.2682163	0.01	0.990	-.5224242	.5292875
__00009E	-.1222842	.375878	-0.33	0.745	-.8592179	.6146496
__00009G	-.5069032	.3157059	-1.61	0.108	-1.125866	.1120591
xbarlag	-9.60482	23.48093	-0.41	0.683	-55.64074	36.4311
x2barlag	-.4721261	.6202525	-0.76	0.447	-1.688172	.74392
xbar2lag	2.547392	4.642938	0.55	0.583	-6.555396	11.65018
rootxbarlag	11.2012	29.02425	0.39	0.700	-45.70276	68.10516
z1barlag	1.737129	1.012927	1.71	0.086	-.2487825	3.723039
z2barlag	2.967942	1.587918	1.87	0.062	-.1452773	6.081161
z3barlag	.4605951	.599841	0.77	0.443	-.7154329	1.636623
z4barlag	-.3892313	.2830116	-1.38	0.169	-.9440944	.1656318
z5barlag	.8752757	.2598649	3.37	0.001	.3657934	1.384758
z6barlag	-.8072529	.4812432	-1.68	0.094	-1.750762	.1362562
z7barlag	.5586904	.5309706	1.05	0.293	-.4823126	1.599693
_cons	-5.198733	9.986805	-0.52	0.603	-24.77853	14.38106

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A2	**150,184**	**.4349509**	**.1262332**	**.1125493**	**1.114139**

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"
957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9
960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
964 }
965
966 local r_RE      "`r_RE' `qhat_hats'"
967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }
978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
```

```
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS		
Model	6864.7597	13	528.058438	Number of obs	= 150,184
Residual	1164.17266	150,170	.007752365	F(13, 150170)	= 68115.79
				Prob > F	= 0.0000
				R-squared	= 0.8550
				Adj R-squared	= 0.8550
Total	8028.93236	150,183	.053460993	Root MSE	= .08805

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	.0239142	.0010556	22.65	0.000	.0218452	.0259833
ybarx	.0007801	.0010663	0.73	0.464	-.0013098	.00287
xi2	-.0398479	.0005318	-74.93	0.000	-.0408902	-.0388056
ybar	-.2250685	.002763	-81.46	0.000	-.2304839	-.219653
xi	.4263369	.0015386	277.09	0.000	.4233213	.4293525
ybar_q1	.4845681	.0032033	151.27	0.000	.4782897	.4908464
pzi1	.2192719	.0012031	182.25	0.000	.2169137	.22163
pzi2	.0216977	.0019671	11.03	0.000	.0178422	.0255533
pzi3	-.0026384	.000611	-4.32	0.000	-.0038359	-.0014409
pzi4	.0084738	.0006135	13.81	0.000	.0072714	.0096763
pzi5	-.0132719	.000639	-20.77	0.000	-.0145244	-.0120195
pzi6	.0031967	.000531	6.02	0.000	.0021558	.0042375
pzi7	-.0039162	.0006956	-5.63	0.000	-.0052796	-.0025528
_cons	-.029631	.0015825	-18.72	0.000	-.0327326	-.0265293

```
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = -_b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                  local coef = _b[pzi`j']
    3.                  local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
```

```
1025}

1026
1027if `J'==3 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =   .0059551
  Iteration 1:  GMM criterion Q(b) =  .00073409
  Iteration 2:  GMM criterion Q(b) =  .00023097
  Iteration 3:  GMM criterion Q(b) =  .00014086
  Iteration 4:  GMM criterion Q(b) =  .00012829

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00829904
  Iteration 1:  GMM criterion Q(b) =  .00680251
  Iteration 2:  GMM criterion Q(b) =   .0065602

  GMM estimation

  Number of parameters =  20
  Number of moments    =  28
  Initial weight matrix: Unadjusted                 Number of obs   =    150,184
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
A11          |
         one |   .5683408   .0344552    16.50   0.000     .5008098    .6358718
-------------+----------------------------------------------------------------
C1           |
         zi1 |   .4908137   .0447565    10.97   0.000     .4030925    .5785349
         zi2 |  -.1710747   .0743433    -2.30   0.021    -.3167848   -.0253646
         zi3 |  -.0065487   .0210316    -0.31   0.756    -.0477698    .0346724
         zi4 |   .1099215   .0336328     3.27   0.001     .0440025    .1758405
         zi5 |   .0448934   .0244776     1.83   0.067    -.0030818    .0928686
         zi6 |   .0155505   .0188225     0.83   0.409    -.0213408    .0524419
         zi7 |   .0094861   .0227733     0.42   0.677    -.0351487    .0541208
-------------+----------------------------------------------------------------
C2           |
         zi1 |   .2761768   .0709489     3.89   0.000     .1371194    .4152341
         zi2 |  -.3175068   .1184345    -2.68   0.007     -.549634   -.0853795
         zi3 |  -.0071198   .0325928    -0.22   0.827    -.0710005    .0567608
         zi4 |   .1771288   .0521204     3.40   0.001     .0749748    .2792829
         zi5 |   .0893823   .0314582     2.84   0.004     .0277253    .1510392
```

```
------------------------------------------------------------------------------
          zi6 |   .0133579    .027548     0.48   0.628    -.0406352    .0673511
          zi7 |   .0175412   .0369848     0.47   0.635    -.0549476      .09003
------------------------------------------------------------------------------
          /d1 |   -.084997   .0068963   -12.33   0.000    -.0985135   -.0714806
          /b1 |   .3946773   .0199469    19.79   0.000      .355582    .4337725
       /AVA11 |    .843183   .7183222     1.17   0.240    -.5647027    2.251069
       /AVA12 |  -.8516747   .6349841    -1.34   0.180    -2.096221    .3928713
       /AVA22 |   .5608702   .5903965     0.95   0.342    -.5962856    1.718026
------------------------------------------------------------------------------
Instruments for equation eq1_RE: __00009J __00009L __00009O xi __000002 __00000E
    __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M
    __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5
    __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
```

```
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_drop3_
      > all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex
      not found)
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXrelig
  > Xsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_
  > RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  150184

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  24757

1077unique group_round
  Number of unique values of group_round is  3941
  Number of records is  150184

1078estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  3941

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  6.2819081

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_
  > drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster
  >  not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_drop3_a
  > ll_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
```

```
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*`p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
```

```
                                                    gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                 local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                         }
   30.                                 }
   31.                         }
   32.                 }
   33.         }
        if `Tindivp1'<=`T' {
                forval t=`Tindivp1'/`T' {
                        tempvar zi`t'xdiff
                        g double `zi`t'xdiff'=zi`t'*`ximxk'
                        local zg "`zg' zi`t'"
                        forval j=1/`J' {
                                tempvar zi`t'p`j'xdiff
                                g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
                                local pzg "`pzg' `zi`t'p`j''"
                                local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
                        }
                }
        }

        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
        // denoting zg as group-zs
        //  zg,pzgx,rprp,rprpx
        local rootprootp ""
        local rootprootpxdiff ""

        forval j=1/`J' {
                forval s=`j'/`J' {
                        tempvar xdiffrp`j'rp`s'
                        g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')

                     if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "

                     local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
                }
         }

        local pqhatxdiff ""
        forval j=1/`J' {
                forval s=1/`J' {
                        foreach name of global Alist {
                                tempvar   p`s'qikhat`j'xdiff`name'
                                g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'

                                local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
                        }
                }
         }

        // set instruments
        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
}
```

```
1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                tempvar ximxk`var'
    3.                g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                        local `++t'
    3.                        local p_nice = subinstr("`p'",":_cons","",.)
    4.                        local est = coefs[1,`t']
    5.                        if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                        if `j'>`i' {
   11.                                tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                        }
   15.                }
   16.         }
```

```
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >     `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1     `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
```

```
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00073486
  Iteration 1:  GMM criterion Q(b) =   .0003449
  Iteration 2:  GMM criterion Q(b) =  .00026727

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01159607
  Iteration 1:  GMM criterion Q(b) =  .01084863
  Iteration 2:  GMM criterion Q(b) =  .01056366
  Iteration 3:  GMM criterion Q(b) =  .01054384

  GMM estimation

  Number of parameters =  17
  Number of moments    =  63
  Initial weight matrix: Unadjusted                 Number of obs   =    150,184
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
C11          |
       _cons |   .6346346   .0281423    22.55   0.000     .5794766    .6897926
-------------+----------------------------------------------------------------
C12          |
       _cons |  -.0581026   .0444603    -1.31   0.191    -.1452432    .0290379
-------------+----------------------------------------------------------------
C13          |
       _cons |  -.0107396   .0118905    -0.90   0.366    -.0340446    .0125654
-------------+----------------------------------------------------------------
C14          |
       _cons |   .0036452   .0124054     0.29   0.769    -.0206689    .0279594
-------------+----------------------------------------------------------------
C15          |
       _cons |  -.0141398   .0204313    -0.69   0.489    -.0541844    .0259048
------------------------------------------------------------------------------
```

C16						
_cons	**-.0212513**	**.0135876**	**-1.56**	**0.118**	**-.0478825**	**.0053799**
C17						
_cons	**-.0183233**	**.0154458**	**-1.19**	**0.236**	**-.0485964**	**.0119499**
C21						
_cons	**.4750328**	**.0395765**	**12.00**	**0.000**	**.3974642**	**.5526013**
C22						
_cons	**-.1230057**	**.0636367**	**-1.93**	**0.053**	**-.2477312**	**.0017199**
C23						
_cons	**-.0140738**	**.0161864**	**-0.87**	**0.385**	**-.0457985**	**.017651**
C24						
_cons	**-.0069473**	**.0208163**	**-0.33**	**0.739**	**-.0477465**	**.033852**
C25						
_cons	**.0068395**	**.025599**	**0.27**	**0.789**	**-.0433337**	**.0570127**
C26						
_cons	**-.031059**	**.0176734**	**-1.76**	**0.079**	**-.0656983**	**.0035802**
C27						
_cons	**-.0168854**	**.02156**	**-0.78**	**0.434**	**-.0591423**	**.0253715**
d1						
_cons	**-.0862064**	**.0063904**	**-13.49**	**0.000**	**-.0987313**	**-.0736815**
b1						
_cons	**.4405539**	**.018341**	**24.02**	**0.000**	**.4046062**	**.4765016**
A11						
one	**.1559822**	**.1327391**	**1.18**	**0.240**	**-.1041816**	**.4161461**

Instruments for equation **eq1_FE: __000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7 __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1 __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY __0000D5 _cons**

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
```

(file **diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_drop3_ > all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex** not found)

(output written to diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXrelig > Xsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_ > _FE.tex)

```
1242
1243        estat overid
```

Test of overidentifying restriction:

Hansen's J chi2(**46**) = **1583.52** (p = **0.0000**)

```
1244        di "$S_DATE $S_TIME"
   8 Dec 2023 13:24:42
1245        unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  150184
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  24757
1248        unique group_round
  Number of unique values of group_round is  3941
  Number of records is  150184
1249        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  3941
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  6.2819081
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_
  > drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster
  >  not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_drop3_a
  > ll_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}
```

```
1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table2_25b.smcl
   log type:  smcl
  closed on:   8 Dec 2023, 13:24:42
```

A.2.2 Columns 3 and 6

A.2.2.1 Panel A

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table2_36.smcl
  log type:  smcl
 opened on:   7 Dec 2023, 16:02:44

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
50. local group_def "dist"

51. local group_def "fsuXseg"

52. local group_def "fsuXsegXreligXsched"

53.
54. // comparision group
55. local groupComp_def ""

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.         syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
70.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations               840   ->   280
  Number of variables                    5   ->   6
  j variable (3 values)       expend_type   ->   (dropped)
  xij variables:
                          laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.             }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,637 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                    egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
```

```
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }
297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }
316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }
333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
```

```
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (82 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,576 real changes made, 82 to missing)

460 replace x2=x*x
  (73,576 real changes made, 82 to missing)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,576 real changes made)

478 replace x2=x2_temp
  (73,576 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (48,819 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)
```

```
488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_7b70_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
```

```
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,757 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
forval j=1/4 {
  2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
> ght_group_round-weighti-weightk)
  3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
> j')/(weight_group_round-weighti-weightk)
  4. }

// summary stats tables
if `make_sumstats'==1 {
        count
        local sumstats_n_obs_pair = r(N)

        cap file close sumstat
                file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
> place
                file write sumstat "\begin{table}[htbp]\centering" _n
                file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
                file write sumstat "\caption{Summary statistics for consumption data
> }\label{tab:sumstats}" _n
                file write sumstat "\begin{tabular}{lcccccccc}" _n
                file write sumstat "\toprule" _n
                file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
> mn{4}{c}{Pairs }\tabularnewline" _n
                file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
> _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
> wline" _n
                file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
> SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
                file write sumstat "\midrule" _n

                foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
  2.                         qui su `var'
  3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
> d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
> max)',.01)' "
                        local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
> gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
> )' \\"
  4.                         file write sumstat "`sumstats_row_`var'" _n
  5.                 }

                file write sumstat "\bottomrule" _n
                file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
>  reports summary statistics for estimation sample.}\\ " _n
                file write sumstat "\end{tabular}" _n
                file write sumstat "\end{table}" _n
                file close sumstat
                BREAK
}

// constant for estimation
g one=1

egen state_district_round=group(state district urban round)

egen state_district=group(state district)

```

```
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1
```

```
scalar estimate_FE=1

// estimation options
global w_initial "winit(unadjusted)"
if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"
global wmatrix "wmatrix(robust)"
global trace_level ""
global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"
global technique ""
global maxiter ""
global technique_RE ""
if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"
global derivatives "quickderivatives"
scalar init_FE_from_RE=1
global clust_var "state_district_round"
local clus_name "_clus_Dist"









// xxzx to keep names same
local wmatrix_desc ""
local sectorName "_urbOnly"
local dropZdesc "_dropPre"
local desc_struc_v0 ""
local inst_desc "_exactInst"
local inst_rootp "_noPPXdInst"
local weight_desc ""
local RE_sq_inst_desc ""
local cnstname ""
local actual_pqhat_desc ""
local expUpname "_updateExp"
```

```
661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //              ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T

694 local Tm1=`T'-1
```

```
695 local Tindiv : word count $zlist_indiv

696
697 /*****************************************************************************
  > Set weights
  > *****************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*****************************************************************************
  >  Update data for J=2, and J=3
  > *****************************************************************************/
706
707 if `J'==2 {
708          replace p1=P1
  (128,640 real changes made)
709          replace p2=P2
  (128,640 real changes made)
710          replace qi1=Qi1
  (128,632 real changes made)
711          replace qk1=Qk1
  (128,632 real changes made)
712          replace qikbar1=Qikbar1
  (128,640 real changes made)
713          replace qi2=Qi2
  (128,640 real changes made)
714          replace qk2=Qk2
  (128,640 real changes made)
715          replace qikbar2=Qikbar2
  (128,640 real changes made)
716
717          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721          forv j = 1(1)3 {
    2.                 replace p`j'=P`j'
    3.                 replace qi`j'=Qi`j'
    4.                 replace qk`j'=Qk`j'
    5.                 replace qikbar`j'=Qikbar`j'
    6.          }
722
723          foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
724 }

725
```

```
726
727 /*****************************************************************************
  > Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }
```

```
751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j'"
    3.          local pCi "`pCi' + `pCi`j'"
    4.          local pCk "`pCk' + `pCk`j'"
    5.          local pCg "`pCg' + `pCg`j'"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j'"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
773         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.          local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.          if `jplus1'<=`J' {
    5.                  forval k=`jplus1'/`J' {
    6.                          local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                  }
    8.          }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                  if noMeasError==1                               local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
```

```
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                      l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                    ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2
  > ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qikba
  > r1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d
  > 1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > }*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))        ))
```

```
836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) +
  >  ({A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C
  > 2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikb
  > ar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:
  > }) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C
  > 2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qi
  > kbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )
  > *{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > one}*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}
  > *ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi
  > 4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p
  > 1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         )
  > )

874
875 /*******************************************************************************
  >  construct instruments
  > *******************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi

882 g double `ximxk'=xi-xk
```

```
883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.          tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.          g double `xizi`t''=xi*zi`t'
    4.          g double `xkzk`t''=xk*zk`t'
    5.          g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.          g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.          g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.          g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.          g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.          forval j=1/`J' {
   11.                  tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                  g double `zi`t'p`j''= zi`t'*p`j'
   13.                  g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                  g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                  g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                  forv s=1/`T' {
   17.                          tempvar zi`t'zi`s'p`j'
   18.                          g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                  }
   20.          }
   21. }

891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }

903
904 local zrg ""

905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var''"
    6.          }
    7. }
```

```
906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var''"
    5.
911          forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var''"
   10.                  local xprg "`xprg' `xp`j'`var''"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""

925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }
```

```
929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.         forval k=`j'/`J' {
    3.                 tempvar p`j'p`k'
    4.                 g `p`j'p`k''=p`j'*p`k'
    5.                 local roundinsts "`roundinsts' `p`j'p`k''"
    6.         }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.         if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.         if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }

  Linear regression                               Number of obs     =    128,640
                                                  F(128, 4598)      =          .
                                                  Prob > F          =          .
                                                  R-squared         =     0.2238
                                                  Root MSE          =       .131
```

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	-12.79512	7.260347	-1.76	0.078	-27.02888	1.43865
__00009L	-3.637138	1.251887	-2.91	0.004	-6.091436	-1.182839
__00009O	1.004898	3.873148	0.26	0.795	-6.588331	8.598128
xi	.1090911	.0077876	14.01	0.000	.0938237	.1243585
__000002	-.0157421	.0025123	-6.27	0.000	-.0206674	-.0108169
__00000E	-.1156487	.0903327	-1.28	0.201	-.2927442	.0614468
__00000P	-.0139978	.0521173	-0.27	0.788	-.1161727	.0881771
__000017	.0509528	.1735472	0.29	0.769	-.289283	.3911887
__00001I	-.0239111	.1060934	-0.23	0.822	-.2319051	.184083
__000020	.0171582	.04793	0.36	0.720	-.0768076	.111124
__00002B	-.025461	.0245826	-1.04	0.300	-.0736547	.0227327
__00002T	.047008	.0663415	0.71	0.479	-.0830531	.1770692
__000034	.0018229	.0474534	0.04	0.969	-.0912086	.0948543
__00003M	.0130708	.0588349	0.22	0.824	-.1022739	.1284156
__00003X	.0098183	.0379708	0.26	0.796	-.0646228	.0842594
__00004F	-.025421	.0466786	-0.54	0.586	-.1169335	.0660916
__00004Q	.0074369	.0250442	0.30	0.767	-.0416617	.0565356
__000058	-.0217155	.0609657	-0.36	0.722	-.1412376	.0978066
__00005J	.0055081	.0340618	0.16	0.872	-.0612694	.0722856
__00005U	.0797255	.4007228	0.20	0.842	-.7058835	.8653344
__00005V	-.0346695	.0711757	-0.49	0.626	-.174208	.1048689
__00005W	-.0306542	.1635037	-0.19	0.851	-.3512	.2898916
__00005X	.1385994	.4167248	0.33	0.739	-.6783813	.9555801
__00005Y	-.0224787	.1028475	-0.22	0.827	-.2241091	.1791517
__00005Z	.0485407	.1674104	0.29	0.772	-.279664	.3767454
__000060	-.0305375	.0696472	-0.44	0.661	-.1670794	.1060045
__000061	.0243364	.0499235	0.49	0.626	-.0735376	.1222104
__000062	.0703686	.0341366	2.06	0.039	.0034444	.1372928
__000063	-.0376336	.047996	-0.78	0.433	-.1317289	.0564616
__000064	-.0701443	.057982	-1.21	0.226	-.1838168	.0435282
__000065	-.1665861	.877927	-0.19	0.850	-1.887744	1.554572
__000066	.2283964	.1442115	1.58	0.113	-.0543274	.5111203
__000067	-.3199538	.3445552	-0.93	0.353	-.9954474	.3555397
__000068	.5394784	.8576131	0.63	0.529	-1.141855	2.220812
__000069	-.1708187	.1878736	-0.91	0.363	-.5391411	.1975038
__00006A	-.424151	.3283207	-1.29	0.196	-1.067817	.2195151
__00006B	-.0463852	.1238777	-0.37	0.708	-.289245	.1964745
__00006C	-.2113415	.1076302	-1.96	0.050	-.4223483	-.0003347
__00006D	.0467054	.0589613	0.79	0.428	-.068887	.1622978
__00006E	-.1444075	.0994178	-1.45	0.146	-.339314	.0504991
__00006F	.030443	.1210695	0.25	0.801	-.2069113	.2677973
__00006G	-.1629455	.1849126	-0.88	0.378	-.5254629	.199572
__00006H	.0332459	.032772	1.01	0.310	-.031003	.0974948
__00006I	-.0177702	.0723866	-0.25	0.806	-.1596827	.1241422
__00006J	.1731219	.2093027	0.83	0.408	-.2372119	.5834557
__00006K	.0504318	.0524078	0.96	0.336	-.0523128	.1531763
__00006L	-.0507186	.0851838	-0.60	0.552	-.2177197	.1162825
__00006M	-.0292887	.0358816	-0.82	0.414	-.0996338	.0410563
__00006N	.0309542	.0234174	1.32	0.186	-.0149551	.0768636
__00006O	-.014908	.0145441	-1.03	0.305	-.0434214	.0136054
__00006P	-.0290572	.0253252	-1.15	0.251	-.0787067	.0205924
__00006Q	.0028381	.030826	0.09	0.927	-.0575957	.0632719
__00006R	-.6012457	.495913	-1.21	0.225	-1.573473	.3709819
__00006S	-.0369337	.0565668	-0.65	0.514	-.1478319	.0739644
__00006T	.2871989	.1903878	1.51	0.131	-.0860527	.6604504
__00006U	.3847497	.4323705	0.89	0.374	-.4629041	1.232403
__00006V	-.1362284	.0767153	-1.78	0.076	-.2866273	.0141705
__00006W	-.0523872	.1430883	-0.37	0.714	-.332909	.2281346
__00006X	.0125271	.0552197	0.23	0.821	-.0957301	.1207843
__00006Y	.0345059	.0349869	0.99	0.324	-.0340851	.103097
__00006Z	-.0725883	.0241739	-3.00	0.003	-.1199808	-.0251958
__000070	-.0109506	.0430715	-0.25	0.799	-.0953914	.0734902
__000071	-.0582328	.0494466	-1.18	0.239	-.1551719	.0387062
__000072	-.4534528	.305937	-1.48	0.138	-1.053236	.1463307
__000073	-.0906118	.0458174	-1.98	0.048	-.180436	-.0007877
__000074	.2961195	.1206212	2.45	0.014	.0596441	.5325949
__000075	.2853582	.2940211	0.97	0.332	-.2910644	.8617807

__000076	-.076137	.0648958	-1.17	0.241	-.2033639	.0510898
__000077	.0084989	.1095353	0.08	0.938	-.2062429	.2232407
__000078	-.0029301	.0439453	-0.07	0.947	-.0890839	.0832237
__000079	-.0005896	.0346299	-0.02	0.986	-.0684808	.0673017
__00007A	.0293777	.0211752	1.39	0.165	-.0121359	.0708912
__00007B	-.0056913	.0314004	-0.18	0.856	-.0672512	.0558686
__00007C	-.0796128	.0372845	-2.14	0.033	-.1527083	-.0065174
__00007D	.1277082	.2134185	0.60	0.550	-.2906944	.5461109
__00007E	.0131872	.0355408	0.37	0.711	-.0564899	.0828643
__00007F	-.0882595	.083501	-1.06	0.291	-.2519616	.0754427
__00007G	.006318	.2231374	0.03	0.977	-.4311384	.4437744
__00007H	-.0032218	.0521663	-0.06	0.951	-.1054929	.0990492
__00007I	-.0735209	.0881929	-0.83	0.405	-.2464212	.0993795
__00007J	-.0161166	.0327932	-0.49	0.623	-.0804071	.0481739
__00007K	-.0226722	.0278317	-0.81	0.415	-.0772358	.0318913
__00007L	.0113972	.0160408	0.71	0.477	-.0200505	.0428448
__00007M	-.0074927	.0281454	-0.27	0.790	-.0626712	.0476857
__00007N	.0512257	.0317663	1.61	0.107	-.0110516	.113503
__00007O	-.0216683	.3475372	-0.06	0.950	-.7030081	.6596714
__00007P	.0879331	.0479654	1.83	0.067	-.0061021	.1819684
__00007Q	-.123569	.1328924	-0.93	0.353	-.3841018	.1369638
__00007R	.1056987	.3234608	0.33	0.744	-.5284398	.7398371
__00007S	.0276477	.0679204	0.41	0.684	-.1055088	.1608043
__00007T	-.2030032	.1161087	-1.75	0.080	-.430632	.0246256
__00007U	.0181395	.0416299	0.44	0.663	-.063475	.0997541
__00007V	-.0771625	.0330746	-2.33	0.020	-.1420047	-.0123204
__00007W	-.0141401	.0196985	-0.72	0.473	-.0527587	.0244784
__00007X	.0035144	.0352185	0.10	0.921	-.0655307	.0725594
__00007Y	.0380758	.0423369	0.90	0.369	-.0449249	.1210764
__000080	-36.81121	17.02885	-2.16	0.031	-70.19593	-3.426501
__000082	.0760369	8.506323	0.01	0.993	-16.60044	16.75251
__000085	-.6294823	.4059554	-1.55	0.121	-1.42535	.1663852
__000087	.4964106	.2690436	1.85	0.065	-.0310441	1.023865
__00008A	8.027308	3.333334	2.41	0.016	1.492373	14.56224
__00008C	-1.201791	1.594737	-0.75	0.451	-4.32824	1.924659
__00008F	44.59295	20.99826	2.12	0.034	3.426269	85.75963
__00008H	2.168832	10.94407	0.20	0.843	-19.2868	23.62447
__00008K	-.4921474	.5029834	-0.98	0.328	-1.478236	.4939415
__00008M	-.8762901	.4582283	-1.91	0.056	-1.774638	.0220574
__00008P	-1.190171	.9196827	-1.29	0.196	-2.99319	.612849
__00008R	-1.132574	.642393	-1.76	0.078	-2.391973	.1268243
__00008U	.1009246	.3299369	0.31	0.760	-.5459101	.7477593
__00008W	.1590723	.2320867	0.69	0.493	-.295929	.6140737
__00008Z	.7462393	.2282715	3.27	0.001	.2987175	1.193761
__000091	-.4374243	.1583699	-2.76	0.006	-.7479053	-.1269433
__000094	.0122267	.1794758	0.07	0.946	-.339632	.3640855
__000096	.1116296	.1418125	0.79	0.431	-.1663908	.3896501
__000099	-.0594555	.2718099	-0.22	0.827	-.5923333	.4734223
__00009B	-.0418895	.1972203	-0.21	0.832	-.4285359	.3447569
__00009E	.0027748	.3171447	0.01	0.993	-.618981	.6245306
__00009G	-.4676324	.2253747	-2.07	0.038	-.9094749	-.0257898
xbarlag	38.11984	21.35657	1.78	0.074	-3.749302	79.98898
x2barlag	-.1896805	.4397159	-0.43	0.666	-1.051735	.6723737
xbar2lag	-6.519805	4.180346	-1.56	0.119	-14.71529	1.67568
rootxbarlag	-48.72697	26.43458	-1.84	0.065	-100.5514	3.097498
z1barlag	1.544271	.7682997	2.01	0.044	.0380348	3.050507
z2barlag	2.57067	1.188866	2.16	0.031	.2399211	4.901418
z3barlag	-.2976042	.4371373	-0.68	0.496	-1.154603	.5593949
z4barlag	-.1807586	.2681451	-0.67	0.500	-.7064517	.3449345
z5barlag	-.1795307	.1890717	-0.95	0.342	-.5502021	.1911407
z6barlag	.1391592	.3756895	0.37	0.711	-.5973725	.8756909
z7barlag	.430906	.4037833	1.07	0.286	-.3607032	1.222515
_cons	16.33104	9.106072	1.79	0.073	-1.521234	34.18331

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	128,640	.4424145	.0703047	.2011188	.8127233

Linear regression

Number of obs	= **128,640**
F(128, 4598)	= .
Prob > F	= .
R-squared	= **0.3053**
Root MSE	= **.20268**

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**4.564747**	**8.325434**	**0.55**	**0.584**	**-11.7571**	**20.88659**
__00009L	**.5937963**	**2.171845**	**0.27**	**0.785**	**-3.664063**	**4.851655**
__00009O	**-.8608651**	**4.900914**	**-0.18**	**0.861**	**-10.46901**	**8.74728**
xi	**.1450715**	**.0142385**	**10.19**	**0.000**	**.1171571**	**.1729859**
__000002	**-.0081835**	**.0053966**	**-1.52**	**0.129**	**-.0187634**	**.0023964**
__00000E	**-.1996901**	**.1318225**	**-1.51**	**0.130**	**-.4581254**	**.0587452**
__00000P	**-.1306909**	**.0773583**	**-1.69**	**0.091**	**-.2823502**	**.0209685**
__000017	**.3219988**	**.2747868**	**1.17**	**0.241**	**-.2167153**	**.8607128**
__00001I	**.2451433**	**.1716831**	**1.43**	**0.153**	**-.091438**	**.5817245**
__000020	**.0465067**	**.0713123**	**0.65**	**0.514**	**-.0932996**	**.186313**
__00002B	**.0499324**	**.0392375**	**1.27**	**0.203**	**-.026992**	**.1268568**
__00002T	**.3317416**	**.0880092**	**3.77**	**0.000**	**.1592013**	**.5042819**
__000034	**-.0424598**	**.0611807**	**-0.69**	**0.488**	**-.1624034**	**.0774838**
__00003M	**-.0824665**	**.0821755**	**-1.00**	**0.316**	**-.2435698**	**.0786368**
__00003X	**.0562813**	**.0513967**	**1.10**	**0.274**	**-.0444809**	**.1570436**
__00004F	**-.0214064**	**.0649749**	**-0.33**	**0.742**	**-.1487884**	**.1059755**
__00004Q	**-.0033885**	**.0382735**	**-0.09**	**0.929**	**-.0784229**	**.0716459**
__000058	**-.0616114**	**.095554**	**-0.64**	**0.519**	**-.248943**	**.1257203**
__00005J	**.0962271**	**.0576901**	**1.67**	**0.095**	**-.0168731**	**.2093274**
__00005U	**-1.072842**	**.6175735**	**-1.74**	**0.082**	**-2.283583**	**.1378984**
__00005V	**-.0686079**	**.112662**	**-0.61**	**0.543**	**-.2894794**	**.1522637**
__00005W	**.2956557**	**.2555724**	**1.16**	**0.247**	**-.2053889**	**.7967004**
__00005X	**1.107247**	**.6463172**	**1.71**	**0.087**	**-.1598453**	**2.374339**
__00005Y	**.006665**	**.1525531**	**0.04**	**0.965**	**-.2924124**	**.3057424**
__00005Z	**.4177337**	**.2272567**	**1.84**	**0.066**	**-.0277986**	**.863266**
__000060	**-.2003814**	**.1148141**	**-1.75**	**0.081**	**-.4254721**	**.0247093**
__000061	**.0627776**	**.0573305**	**1.10**	**0.274**	**-.0496177**	**.1751728**
__000062	**-.0079371**	**.0479**	**-0.17**	**0.868**	**-.101844**	**.0859698**
__000063	**.0234324**	**.0606999**	**0.39**	**0.699**	**-.0955686**	**.1424334**
__000064	**-.0476537**	**.0743598**	**-0.64**	**0.522**	**-.1934346**	**.0981272**
__000065	**4.601567**	**1.273473**	**3.61**	**0.000**	**2.104948**	**7.098186**
__000066	**.5353632**	**.231593**	**2.31**	**0.021**	**.0813297**	**.9893968**
__000067	**-2.104046**	**.5308536**	**-3.96**	**0.000**	**-3.144774**	**-1.063318**
__000068	**-3.209865**	**1.309425**	**-2.45**	**0.014**	**-5.776967**	**-.6427636**
__000069	**-.5733963**	**.2998063**	**-1.91**	**0.056**	**-1.16116**	**.0143679**
__00006A	**-.3073944**	**.5099257**	**-0.60**	**0.547**	**-1.307093**	**.6923047**
__00006B	**-.0013787**	**.1805182**	**-0.01**	**0.994**	**-.3552811**	**.3525236**
__00006C	**-.6917272**	**.1201482**	**-5.76**	**0.000**	**-.9272752**	**-.4561791**
__00006D	**.0978464**	**.0930479**	**1.05**	**0.293**	**-.0845721**	**.280265**
__00006E	**-.1126402**	**.1459926**	**-0.77**	**0.440**	**-.3988558**	**.1735754**
__00006F	**.0293601**	**.1813666**	**0.16**	**0.871**	**-.3262055**	**.3849256**
__00006G	**.228589**	**.3106349**	**0.74**	**0.462**	**-.3804045**	**.8375825**
__00006H	**.0378065**	**.0490674**	**0.77**	**0.441**	**-.0583892**	**.1340023**
__00006I	**-.1229966**	**.1220438**	**-1.01**	**0.314**	**-.362261**	**.1162679**
__00006J	**-.2612815**	**.3194111**	**-0.82**	**0.413**	**-.8874806**	**.3649176**
__00006K	**.042626**	**.077334**	**0.55**	**0.582**	**-.1089857**	**.1942377**
__00006L	**-.0857245**	**.1255653**	**-0.68**	**0.495**	**-.3318928**	**.1604438**
__00006M	**.0143417**	**.0615696**	**0.23**	**0.816**	**-.1063642**	**.1350476**
__00006N	**.0142187**	**.0300003**	**0.47**	**0.636**	**-.0445962**	**.0730337**
__00006O	**.0108951**	**.0238843**	**0.46**	**0.648**	**-.0359295**	**.0577198**
__00006P	**-.051378**	**.0341179**	**-1.51**	**0.132**	**-.1182654**	**.0155094**
__00006Q	**.0061354**	**.0401718**	**0.15**	**0.879**	**-.0726205**	**.0848914**
__00006R	**-.3954866**	**.5294751**	**-0.75**	**0.455**	**-1.433512**	**.6425387**
__00006S	**.0027485**	**.070072**	**0.04**	**0.969**	**-.1346264**	**.1401233**
__00006T	**.2422518**	**.1891001**	**1.28**	**0.200**	**-.1284752**	**.6129788**
__00006U	**-.2749271**	**.4753902**	**-0.58**	**0.563**	**-1.20692**	**.657066**
__00006V	**.1610918**	**.0976758**	**1.65**	**0.099**	**-.0303997**	**.3525833**
__00006W	**.1256103**	**.177919**	**0.71**	**0.480**	**-.2231964**	**.474417**
__00006X	**.0319531**	**.0688723**	**0.46**	**0.643**	**-.1030697**	**.1669758**
__00006Y	**.0856173**	**.0433777**	**1.97**	**0.048**	**.0005762**	**.1706584**
__00006Z	**-.1346033**	**.0315098**	**-4.27**	**0.000**	**-.1963778**	**-.0728289**

__000070	-.0896637	.0497818	-1.80	0.072	-.18726	.0079325
__000071	-.0242239	.0620782	-0.39	0.696	-.145927	.0974792
__000072	-.817985	.3639092	-2.25	0.025	-1.531422	-.1045482
__000073	-.1058926	.0632791	-1.67	0.094	-.22995	.0181648
__000074	.4129247	.154788	2.67	0.008	.109466	.7163835
__000075	.4604809	.3667051	1.26	0.209	-.258437	1.179399
__000076	.1273306	.0859626	1.48	0.139	-.0411973	.2958585
__000077	.1296378	.1391035	0.93	0.351	-.1430718	.4023475
__000078	-.0674194	.0576484	-1.17	0.242	-.1804379	.0455991
__000079	.0872997	.0407137	2.14	0.032	.0074813	.167118
__00007A	.0612698	.02713	2.26	0.024	.008082	.1144575
__00007B	.0328333	.0413449	0.79	0.427	-.0482225	.1138891
__00007C	.0255229	.0482849	0.53	0.597	-.0691387	.1201845
__00007D	.354494	.2770601	1.28	0.201	-.1886768	.8976647
__00007E	.0890569	.0489815	1.82	0.069	-.0069704	.1850841
__00007F	-.2285898	.1151561	-1.99	0.047	-.4543511	-.0028285
__00007G	-.1664105	.2810873	-0.59	0.554	-.7174767	.3846556
__00007H	-.0569692	.0661062	-0.86	0.389	-.186569	.0726307
__00007I	-.0611722	.1122723	-0.54	0.586	-.2812799	.1589354
__00007J	.0315184	.0411415	0.77	0.444	-.0491387	.1121755
__00007K	-.0752832	.0302422	-2.49	0.013	-.1345724	-.015994
__00007L	.0125053	.0227207	0.55	0.582	-.0320382	.0570487
__00007M	.0286478	.0317049	0.90	0.366	-.0335091	.0908047
__00007N	.0174569	.0380542	0.46	0.646	-.0571476	.0920614
__00007O	.2851267	.4558075	0.63	0.532	-.6084747	1.178728
__00007P	.2207786	.0723679	3.05	0.002	.0789028	.3626543
__00007Q	-.3915989	.1823773	-2.15	0.032	-.7491459	-.0340519
__00007R	-.012365	.4576998	-0.03	0.978	-.9096763	.8849463
__00007S	-.1570488	.1017663	-1.54	0.123	-.3565595	.042462
__00007T	-.4053457	.1710711	-2.37	0.018	-.7407271	-.0699642
__00007U	.0972112	.0620585	1.57	0.117	-.0244534	.2188757
__00007V	-.2339363	.0420127	-5.57	0.000	-.3163014	-.1515711
__00007W	-.0218837	.0324529	-0.67	0.500	-.085507	.0417396
__00007X	.0212272	.0497507	0.43	0.670	-.076308	.1187624
__00007Y	.005633	.0584831	0.10	0.923	-.109022	.1202879
__000080	8.445766	18.96277	0.45	0.656	-28.73037	45.6219
__000082	-2.538594	11.18002	-0.23	0.820	-24.45681	19.37962
__000085	-.0754758	.5249525	-0.14	0.886	-1.104635	.9536832
__000087	.2107624	.3791253	0.56	0.578	-.5325052	.9540299
__00008A	-1.497372	3.684007	-0.41	0.684	-8.719793	5.72505
__00008C	.2927278	2.177493	0.13	0.893	-3.976205	4.56166
__00008F	-10.63528	23.58522	-0.45	0.652	-56.87363	35.60308
__00008H	3.123759	14.11996	0.22	0.825	-24.55815	30.80567
__00008K	-.1941854	.6776997	-0.29	0.774	-1.522802	1.134431
__00008M	-1.39456	.6534375	-2.13	0.033	-2.675611	-.1135085
__00008P	-1.343119	1.111617	-1.21	0.227	-3.522422	.8361837
__00008R	-1.271986	.9151353	-1.39	0.165	-3.066091	.5221185
__00008U	-.7730286	.4412188	-1.75	0.080	-1.638029	.091972
__00008W	.2278244	.3407778	0.67	0.504	-.4402638	.8959125
__00008Z	.601064	.2397331	2.51	0.012	.1310721	1.071056
__000091	-.0640346	.2041853	-0.31	0.754	-.4643358	.3362665
__000094	-.7268351	.2604519	-2.79	0.005	-1.237446	-.2162243
__000096	-.228038	.2082336	-1.10	0.274	-.6362758	.1801998
__000099	.7647842	.3113365	2.46	0.014	.1544152	1.375153
__00009B	.1977191	.261654	0.76	0.450	-.3152483	.7106865
__00009E	-.0660116	.3747056	-0.18	0.860	-.8006145	.6685913
__00009G	-.243422	.302943	-0.80	0.422	-.8373358	.3504917
xbarlag	-4.769469	22.185	-0.21	0.830	-48.26272	38.72378
x2barlag	-.4538931	.5918067	-0.77	0.443	-1.614118	.7063322
xbar2lag	1.725566	4.421582	0.39	0.696	-6.942858	10.39399
rootxbarlag	5.399595	27.44805	0.20	0.844	-48.41176	59.21095
z1barlag	1.81121	1.002554	1.81	0.071	-.1542766	3.776696
z2barlag	2.690187	1.511498	1.78	0.075	-.2730747	5.653448
z3barlag	.552162	.563691	0.98	0.327	-.5529429	1.657267
z4barlag	-.2955395	.2702011	-1.09	0.274	-.8252634	.2341844
z5barlag	1.024167	.2637745	3.88	0.000	.5070427	1.541292
z6barlag	-.9630149	.4686881	-2.05	0.040	-1.881869	-.0441613
z7barlag	.2824898	.5031912	0.56	0.575	-.7040064	1.268986
_cons	-3.250004	9.429693	-0.34	0.730	-21.73673	15.23672

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A2	**128,640**	**.4438109**	**.13429**	**.0976942**	**1.111178**

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"
957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9
960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
964 }
965
966 local r_RE      "`r_RE' `qhat_hats'"
967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }
978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
```

```
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS	Number of obs	=	128,640
				F(13, 128626)	=	57321.16
Model	5933.80789	13	456.446761	Prob > F	=	0.0000
Residual	1024.24524	128,626	.007962972	R-squared	=	0.8528
				Adj R-squared	=	0.8528
Total	6958.05313	128,639	.054089764	Root MSE	=	.08924

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	.0180296	.0009529	18.92	0.000	.0161619	.0198974
ybarx	.0030935	.0010502	2.95	0.003	.0010352	.0051518
xi2	-.0393058	.0005754	-68.31	0.000	-.0404336	-.0381779
ybar	-.1946598	.0026658	-73.02	0.000	-.1998848	-.1894348
xi	.4196294	.0016617	252.52	0.000	.4163724	.4228864
ybar_q1	.4287804	.0032091	133.61	0.000	.4224907	.4350702
pzi1	.22573	.001319	171.13	0.000	.2231448	.2283153
pzi2	.0193004	.0021507	8.97	0.000	.0150851	.0235157
pzi3	-.00296	.0006651	-4.45	0.000	-.0042635	-.0016564
pzi4	.0106379	.0006603	16.11	0.000	.0093438	.011932
pzi5	-.0145877	.0007109	-20.52	0.000	-.0159811	-.0131944
pzi6	.0026414	.000587	4.50	0.000	.0014908	.003792
pzi7	-.0039312	.0007606	-5.17	0.000	-.005422	-.0024403
_cons	-.0249549	.0016336	-15.28	0.000	-.0281566	-.0217531

```
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = -_b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                  local coef = _b[pzi`j']
    3.                  local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
```

```
1025}

1026
1027if `J'==3 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00498017
  Iteration 1:  GMM criterion Q(b) =  .00061572
  Iteration 2:  GMM criterion Q(b) =  .00018123
  Iteration 3:  GMM criterion Q(b) =  .00012528

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00675202
  Iteration 1:  GMM criterion Q(b) =  .00512747
  Iteration 2:  GMM criterion Q(b) =  .00508207

  GMM estimation

  Number of parameters =  20
  Number of moments    =  28
  Initial weight matrix: Unadjusted                 Number of obs   =    128,640
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
A11          |
         one |   .6063488   .0357253    16.97   0.000     .5363286     .676369
-------------+----------------------------------------------------------------
C1           |
         zi1 |   .4928823   .0463509    10.63   0.000     .4020362    .5837284
         zi2 |  -.1965658   .0766228    -2.57   0.010    -.3467437    -.046388
         zi3 |   .0011543   .0222672     0.05   0.959    -.0424887    .0447973
         zi4 |   .1027277   .0314652     3.26   0.001     .0410571    .1643984
         zi5 |     .05721   .0235431     2.43   0.015     .0110664    .1033536
         zi6 |   .0080166   .0186304     0.43   0.667    -.0284983    .0445315
         zi7 |   .0001728   .0230086     0.01   0.994    -.0449232    .0452689
-------------+----------------------------------------------------------------
C2           |
         zi1 |   .2864128   .0725217     3.95   0.000     .1442729    .4285527
         zi2 |  -.3656803   .1220157    -3.00   0.003    -.6048266    -.126534
         zi3 |   .0050626   .0344776     0.15   0.883    -.0625123    .0726375
         zi4 |   .1673259   .0487888     3.43   0.001     .0717016    .2629502
         zi5 |   .1038069   .0293474     3.54   0.000     .0462871    .1613267
         zi6 |   .0000514   .0269239     0.00   0.998    -.0527184    .0528212
```

```
------------------------------------------------------------------------------
          zi7 |  -.0008804    .037622    -0.02   0.981    -.0746182    .0728573
------------------------------------------------------------------------------
          /d1 |  -.0908835   .0076594   -11.87   0.000    -.1058958   -.0758713
          /b1 |   .3974894   .0218415    18.20   0.000     .3546808    .4402979
       /AVA11 |   .5914223   .6227972     0.95   0.342    -.6292378    1.812082
       /AVA12 |   -.595739   .5485919    -1.09   0.278    -1.670959    .4794815
       /AVA22 |   .2854211   .5181407     0.55   0.582    -.7301159    1.300958
------------------------------------------------------------------------------
Instruments for equation eq1_RE: __00009J __00009L __00009O xi __000002 __00000E
     __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M
     __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5
     __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
```

```
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  24757

1077unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640

1078estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4599

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  5.3831268

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
```

```
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*`p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
1119                                                       gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                  local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                          }
   30.                                  }
   31.                          }
   32.                  }
   33.          }
```

```
1120        if `Tindivp1'<=`T' {
1121                forval t=`Tindivp1'/`T' {
    2.                          tempvar zi`t'xdiff
    3.                          g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                          local zg "`zg' zi`t'"
    5.                          forval j=1/`J' {
    6.                                  tempvar zi`t'p`j'xdiff
    7.                                  g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                  local pzg "`pzg' `zi`t'p`j''"
    9.                                  local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                          }
   11.                 }
1122        }
1123
1124        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1125        // denoting zg as group-zs
1126        //  zg,pzgx,rprp,rprpx
1127        local rootprootp ""
1128        local rootprootpxdiff ""
1129
1130        forval j=1/`J' {
    2.                  forval s=`j'/`J' {
    3.                          tempvar xdiffrp`j'rp`s'
    4.                          g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1131                            if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1132                            local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                  }
    8.          }
1133
1134        local pqhatxdiff ""
1135        forval j=1/`J' {
    2.                  forval s=1/`J' {
    3.                          foreach name of global Alist {
    4.                                  tempvar   p`s'qikhat`j'xdiff`name'
    5.                                  g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
1136                                    local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
1137
1138        // set instruments
1139        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1140}

1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
```

```
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                      local `++t'
    3.                      local p_nice = subinstr("`p'",":_cons","",.)
    4.                      local est = coefs[1,`t']
    5.                      if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.              }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                 g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                 local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                 g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                 local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                 }
   16.         }
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
```

```
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1    `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
```

```
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00070619
  Iteration 1:  GMM criterion Q(b) =  .00034284
  Iteration 2:  GMM criterion Q(b) =  .00025979

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01121013
  Iteration 1:  GMM criterion Q(b) =  .01091197
  Iteration 2:  GMM criterion Q(b) =  .01017896
  Iteration 3:  GMM criterion Q(b) =  .01015145

  GMM estimation

  Number of parameters =  17
  Number of moments    =  63
  Initial weight matrix: Unadjusted                 Number of obs   =    128,640
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11						
_cons	.6316835	.0285784	22.10	0.000	.5756707	.6876962
C12						
_cons	-.0602877	.0444385	-1.36	0.175	-.1473854	.0268101
C13						
_cons	-.0125609	.0120696	-1.04	0.298	-.0362168	.0110951
C14						
_cons	.003487	.0117769	0.30	0.767	-.0195953	.0265692
C15						
_cons	-.0116801	.020477	-0.57	0.568	-.0518143	.028454
C16						
_cons	-.0225087	.0135312	-1.66	0.096	-.0490293	.0040118
C17						
_cons	-.0176941	.015536	-1.14	0.255	-.0481441	.0127559
C21						
_cons	.4719309	.0398987	11.83	0.000	.393731	.5501309
C22						

	_cons	-.1299939	.0629262	-2.07	0.039	-.2533269	-.0066609
C23							
	_cons	-.0165189	.0162531	-1.02	0.309	-.0483744	.0153366
C24							
	_cons	-.0070961	.0199483	-0.36	0.722	-.046194	.0320019
C25							
	_cons	.0100266	.0255116	0.39	0.694	-.0399752	.0600285
C26							
	_cons	-.0332436	.0175596	-1.89	0.058	-.0676597	.0011725
C27							
	_cons	-.0175159	.0218638	-0.80	0.423	-.0603682	.0253364
d1							
	_cons	-.0925405	.0065272	-14.18	0.000	-.1053335	-.0797475
b1							
	_cons	.4307305	.0189708	22.70	0.000	.3935483	.4679126
A11							
	one	.265553	.1190223	2.23	0.026	.0322736	.4988324

Instruments for equation **eq1_FE**: **__000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7 __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1 __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY __0000D5 _cons**

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
```
(output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex)
```
1242
1243        estat overid
```

Test of overidentifying restriction:

Hansen's J chi2(**46**) = **1305.88** (p = **0.0000**)
```
1244        di "$S_DATE $S_TIME"
```
7 Dec 2023 16:07:58
```
1245        unique hhidi
```
Number of unique values of hhidi is **24757**
Number of records is **128640**
```
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)
```

added scalar:
e(N_hh) = **24757**
```
1248        unique group_round
```
Number of unique values of group_round is **4599**
Number of records is **128640**
```
1249        estadd scalar N_grp = r(sum)
```

added scalar:
e(N_grp) = **4599**

```
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  5.3831268
1251        estimates save "$filename_FE", replace
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}

1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table2_36.smcl
   log type:  smcl
  closed on:   7 Dec 2023, 16:07:58
```

A.2.2.2 Panel B

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table2_36b.smcl
  log type:  smcl
 opened on:   8 Dec 2023, 13:05:05

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
  C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
50. //local group_def "dist"
51. //local group_def "fsuXseg"
52. local group_def "fsuXsegXreligXsched"

53.
54. // comparision group
55. local groupComp_def "fsuXsegXreligXsched"

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.          syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
70.          if "`rename'" == "" local rename = "`varlist'bar"
    3.
```

```
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations            1,118    ->   280
  Number of variables                   5    ->   7
  j variable (4 values)       expend_type    ->   (dropped)
  xij variables:
                         laspeyres_state    ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  -----------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations              560    ->   280
  Number of variables                   5    ->   5
  j variable (2 values)       expend_type    ->   (dropped)
  xij variables:
                         laspeyres_state    ->   laspeyres_state1 laspeyres_state2
  -----------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                  Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations                 840   ->   280
  Number of variables                      5   ->   6
  j variable (3 values)          expend_type   ->   (dropped)
  xij variables:
                                laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.             }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,637 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)
  (43,637 missing values generated)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)
  (43,637 missing values generated)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }
```

```
236
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
```

```
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }

297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }

316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }

333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
```

```
351
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.         drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.         }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.         g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (82 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,576 real changes made, 82 to missing)

460 replace x2=x*x
  (73,576 real changes made, 82 to missing)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,576 real changes made)

478 replace x2=x2_temp
  (73,576 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (48,819 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3
  (0 observations deleted)

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)
```

```
488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_4630_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
```

```
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,757 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
549 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var''" _n
    5.                 }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
```

```
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1
```

```
625 scalar estimate_FE=1

626
627 // estimation options
628 global w_initial "winit(unadjusted)"

629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

630 global wmatrix "wmatrix(robust)"

631 global trace_level ""

632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

633 global technique ""

634 global maxiter ""

635 global technique_RE ""

636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

637 global derivatives "quickderivatives"

638 scalar init_FE_from_RE=1

639 global clust_var "state_district_round"

640 local clus_name "_clus_Dist"

641
642
643
644
645
646
647
648
649 // xxzx to keep names same
650 local wmatrix_desc ""

651 local sectorName "_urbOnly"

652 local dropZdesc "_dropPre"

653 local desc_struc_v0 ""

654 local inst_desc "_exactInst"

655 local inst_rootp "_noPPXdInst"

656 local weight_desc ""

657 local RE_sq_inst_desc ""

658 local cnstname ""

659 local actual_pqhat_desc ""

660 local expUpname "_updateExp"
```

```
661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
  (0 observations deleted)
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //               ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T
```

```
694 local Tm1=`T'-1

695 local Tindiv : word count $zlist_indiv

696
697 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
706
707 if `J'==2 {
708          replace p1=P1
  (128,640 real changes made)
709          replace p2=P2
  (128,640 real changes made)
710          replace qi1=Qi1
  (128,632 real changes made)
711          replace qk1=Qk1
  (128,632 real changes made)
712          replace qikbar1=Qikbar1
  (128,640 real changes made)
713          replace qi2=Qi2
  (128,640 real changes made)
714          replace qk2=Qk2
  (128,640 real changes made)
715          replace qikbar2=Qikbar2
  (128,640 real changes made)
716
717          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721          forv j = 1(1)3 {
    2.                 replace p`j'=P`j'
    3.                 replace qi`j'=Qi`j'
    4.                 replace qk`j'=Qk`j'
    5.                 replace qikbar`j'=Qikbar`j'
    6.          }
722
723          foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
724 }
```

```
725
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }
```

```
751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
773         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j'"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                    "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.         local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.         if `jplus1'<=`J' {
    5.                 forval k=`jplus1'/`J' {
    6.                         local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                 }
    8.         }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                 if noMeasError==1                                local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.         }
```

```
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                    l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                    ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2
  > ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qikba
  > r1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d
  > 1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > }*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))        ))
```

```
836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) +
  >  ({A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C
  > 2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikb
  > ar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:
  > }) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C
  > 2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qi
  > kbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )
  > *{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > one}*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}
  > *ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi
  > 4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p
  > 1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         )
  > )

874
875 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi

882 g double `ximxk'=xi-xk
```

```
883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.          tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.          g double `xizi`t''=xi*zi`t'
    4.          g double `xkzk`t''=xk*zk`t'
    5.          g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.          g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.          g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.          g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.          g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.          forval j=1/`J' {
   11.                  tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                  g double `zi`t'p`j''= zi`t'*p`j'
   13.                  g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                  g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                  g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                  forv s=1/`T' {
   17.                          tempvar zi`t'zi`s'p`j'
   18.                          g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                  }
   20.          }
   21. }

891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }

903
904 local zrg ""

905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var''"
    6.          }
    7. }
```

```
906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var''"
    5.
911          forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var''"
   10.                  local xprg "`xprg' `xp`j'`var''"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""

925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }
```

```
929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.         forval k=`j'/`J' {
    3.                 tempvar p`j'p`k'
    4.                 g `p`j'p`k''=p`j'*p`k'
    5.                 local roundinsts "`roundinsts' `p`j'p`k''"
    6.         }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.         if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.         if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }

  Linear regression                               Number of obs     =    128,640
                                                  F(127, 4598)      =          .
                                                  Prob > F          =          .
                                                  R-squared         =     0.2238
                                                  Root MSE          =       .131
```

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	-12.79512	7.260346	-1.76	0.078	-27.02888	1.438647
__00009L	-3.637138	1.251887	-2.91	0.004	-6.091436	-1.182839
__00009O	1.004898	3.873148	0.26	0.795	-6.588332	8.598128
xi	.1090911	.0077876	14.01	0.000	.0938237	.1243585
__000002	-.0157421	.0025123	-6.27	0.000	-.0206674	-.0108169
__00000E	-.1156487	.0903327	-1.28	0.201	-.2927442	.0614468
__00000P	-.0139978	.0521173	-0.27	0.788	-.1161727	.0881771
__000017	.0509528	.1735472	0.29	0.769	-.289283	.3911886
__00001I	-.0239111	.1060934	-0.23	0.822	-.2319051	.184083
__000020	.0171582	.04793	0.36	0.720	-.0768076	.111124
__00002B	-.025461	.0245826	-1.04	0.300	-.0736547	.0227327
__00002T	.047008	.0663415	0.71	0.479	-.0830531	.1770692
__000034	.0018229	.0474534	0.04	0.969	-.0912086	.0948543
__00003M	.0130708	.0588349	0.22	0.824	-.1022739	.1284156
__00003X	.0098183	.0379708	0.26	0.796	-.0646228	.0842594
__00004F	-.025421	.0466786	-0.54	0.586	-.1169335	.0660916
__00004Q	.0074369	.0250442	0.30	0.767	-.0416617	.0565356
__000058	-.0217155	.0609657	-0.36	0.722	-.1412376	.0978066
__00005J	.0055081	.0340618	0.16	0.872	-.0612694	.0722856
__00005U	.0797255	.4007228	0.20	0.842	-.7058835	.8653344
__00005V	-.0346695	.0711757	-0.49	0.626	-.174208	.1048689
__00005W	-.0306542	.1635038	-0.19	0.851	-.3512	.2898916
__00005X	.1385994	.4167248	0.33	0.739	-.6783813	.9555801
__00005Y	-.0224787	.1028475	-0.22	0.827	-.2241091	.1791517
__00005Z	.0485407	.1674104	0.29	0.772	-.279664	.3767454
__000060	-.0305375	.0696472	-0.44	0.661	-.1670794	.1060045
__000061	.0243364	.0499235	0.49	0.626	-.0735376	.1222104
__000062	.0703686	.0341366	2.06	0.039	.0034444	.1372928
__000063	-.0376336	.047996	-0.78	0.433	-.1317289	.0564616
__000064	-.0701443	.057982	-1.21	0.226	-.1838168	.0435282
__000065	-.1665861	.877927	-0.19	0.850	-1.887744	1.554572
__000066	.2283964	.1442115	1.58	0.113	-.0543274	.5111203
__000067	-.3199538	.3445552	-0.93	0.353	-.9954474	.3555397
__000068	.5394784	.8576131	0.63	0.529	-1.141855	2.220812
__000069	-.1708187	.1878736	-0.91	0.363	-.5391411	.1975038
__00006A	-.424151	.3283207	-1.29	0.196	-1.067817	.2195151
__00006B	-.0463852	.1238777	-0.37	0.708	-.289245	.1964745
__00006C	-.2113415	.1076302	-1.96	0.050	-.4223483	-.0003347
__00006D	.0467054	.0589613	0.79	0.428	-.068887	.1622978
__00006E	-.1444075	.0994178	-1.45	0.146	-.339314	.0504991
__00006F	.030443	.1210695	0.25	0.801	-.2069113	.2677973
__00006G	-.1629455	.1849126	-0.88	0.378	-.5254629	.199572
__00006H	.0332459	.032772	1.01	0.310	-.031003	.0974948
__00006I	-.0177702	.0723866	-0.25	0.806	-.1596827	.1241422
__00006J	.1731219	.2093027	0.83	0.408	-.2372119	.5834557
__00006K	.0504318	.0524078	0.96	0.336	-.0523128	.1531763
__00006L	-.0507186	.0851838	-0.60	0.552	-.2177197	.1162825
__00006M	-.0292887	.0358816	-0.82	0.414	-.0996338	.0410563
__00006N	.0309542	.0234174	1.32	0.186	-.0149551	.0768636
__00006O	-.014908	.0145441	-1.03	0.305	-.0434214	.0136054
__00006P	-.0290572	.0253252	-1.15	0.251	-.0787067	.0205924
__00006Q	.0028381	.030826	0.09	0.927	-.0575957	.0632719
__00006R	-.6012457	.495913	-1.21	0.225	-1.573473	.3709819
__00006S	-.0369337	.0565668	-0.65	0.514	-.1478319	.0739644
__00006T	.2871989	.1903878	1.51	0.131	-.0860527	.6604504
__00006U	.3847497	.4323705	0.89	0.374	-.4629041	1.232403
__00006V	-.1362284	.0767153	-1.78	0.076	-.2866273	.0141705
__00006W	-.0523872	.1430883	-0.37	0.714	-.332909	.2281346
__00006X	.0125271	.0552197	0.23	0.821	-.0957301	.1207843
__00006Y	.0345059	.0349869	0.99	0.324	-.0340851	.103097
__00006Z	-.0725883	.0241739	-3.00	0.003	-.1199808	-.0251958
__000070	-.0109506	.0430715	-0.25	0.799	-.0953914	.0734902
__000071	-.0582328	.0494466	-1.18	0.239	-.1551719	.0387062
__000072	-.4534528	.305937	-1.48	0.138	-1.053236	.1463307
__000073	-.0906118	.0458174	-1.98	0.048	-.180436	-.0007877
__000074	.2961195	.1206212	2.45	0.014	.0596441	.5325949
__000075	.2853582	.2940211	0.97	0.332	-.2910644	.8617807

__000076	-.076137	.0648958	-1.17	0.241	-.2033639	.0510898
__000077	.0084989	.1095353	0.08	0.938	-.2062429	.2232407
__000078	-.0029301	.0439453	-0.07	0.947	-.0890839	.0832237
__000079	-.0005896	.0346299	-0.02	0.986	-.0684808	.0673017
__00007A	.0293777	.0211752	1.39	0.165	-.0121359	.0708912
__00007B	-.0056913	.0314004	-0.18	0.856	-.0672512	.0558686
__00007C	-.0796128	.0372845	-2.14	0.033	-.1527083	-.0065174
__00007D	.1277082	.2134185	0.60	0.550	-.2906944	.5461109
__00007E	.0131872	.0355408	0.37	0.711	-.0564899	.0828643
__00007F	-.0882595	.083501	-1.06	0.291	-.2519616	.0754427
__00007G	.006318	.2231374	0.03	0.977	-.4311384	.4437744
__00007H	-.0032218	.0521663	-0.06	0.951	-.1054929	.0990492
__00007I	-.0735209	.0881929	-0.83	0.405	-.2464212	.0993795
__00007J	-.0161166	.0327932	-0.49	0.623	-.0804071	.0481739
__00007K	-.0226722	.0278317	-0.81	0.415	-.0772358	.0318913
__00007L	.0113972	.0160408	0.71	0.477	-.0200505	.0428448
__00007M	-.0074927	.0281454	-0.27	0.790	-.0626712	.0476857
__00007N	.0512257	.0317663	1.61	0.107	-.0110516	.113503
__00007O	-.0216683	.3475372	-0.06	0.950	-.7030081	.6596714
__00007P	.0879331	.0479654	1.83	0.067	-.0061021	.1819684
__00007Q	-.123569	.1328924	-0.93	0.353	-.3841018	.1369638
__00007R	.1056987	.3234608	0.33	0.744	-.5284398	.7398371
__00007S	.0276477	.0679204	0.41	0.684	-.1055088	.1608043
__00007T	-.2030032	.1161087	-1.75	0.080	-.430632	.0246256
__00007U	.0181395	.0416299	0.44	0.663	-.063475	.0997541
__00007V	-.0771625	.0330746	-2.33	0.020	-.1420047	-.0123204
__00007W	-.0141401	.0196985	-0.72	0.473	-.0527587	.0244784
__00007X	.0035144	.0352185	0.10	0.921	-.0655307	.0725594
__00007Y	.0380758	.0423369	0.90	0.369	-.0449249	.1210764
__000080	-36.81121	17.02884	-2.16	0.031	-70.19592	-3.426507
__000082	.0760369	8.506324	0.01	0.993	-16.60044	16.75252
__000085	-.6294823	.4059554	-1.55	0.121	-1.42535	.1663852
__000087	.4964106	.2690436	1.85	0.065	-.0310441	1.023865
__00008A	8.027308	3.333333	2.41	0.016	1.492375	14.56224
__00008C	-1.201791	1.594737	-0.75	0.451	-4.328241	1.924659
__00008F	44.59295	20.99826	2.12	0.034	3.426278	85.75962
__00008H	2.168832	10.94407	0.20	0.843	-19.28681	23.62447
__00008K	-.4921474	.5029834	-0.98	0.328	-1.478236	.4939415
__00008M	-.8762901	.4582283	-1.91	0.056	-1.774638	.0220574
__00008P	-1.190171	.9196827	-1.29	0.196	-2.99319	.612849
__00008R	-1.132574	.642393	-1.76	0.078	-2.391973	.1268243
__00008U	.1009246	.3299369	0.31	0.760	-.5459101	.7477593
__00008W	.1590723	.2320867	0.69	0.493	-.295929	.6140737
__00008Z	.7462393	.2282715	3.27	0.001	.2987175	1.193761
__000091	-.4374243	.1583699	-2.76	0.006	-.7479053	-.1269433
__000094	.0122267	.1794758	0.07	0.946	-.339632	.3640855
__000096	.1116296	.1418125	0.79	0.431	-.1663908	.3896501
__000099	-.0594555	.2718099	-0.22	0.827	-.5923333	.4734223
__00009B	-.0418895	.1972203	-0.21	0.832	-.4285359	.3447569
__00009E	.0027748	.3171447	0.01	0.993	-.618981	.6245306
__00009G	-.4676324	.2253747	-2.07	0.038	-.9094749	-.0257898
xbarlag	38.11984	21.35657	1.78	0.074	-3.749283	79.98896
x2barlag	-.1896805	.4397159	-0.43	0.666	-1.051735	.6723737
xbar2lag	-6.519805	4.180344	-1.56	0.119	-14.71529	1.675677
rootxbarlag	-48.72697	26.43457	-1.84	0.065	-100.5514	3.097473
z1barlag	1.544271	.7682997	2.01	0.044	.0380348	3.050507
z2barlag	2.57067	1.188866	2.16	0.031	.2399211	4.901418
z3barlag	-.2976042	.4371373	-0.68	0.496	-1.154603	.5593949
z4barlag	-.1807586	.2681451	-0.67	0.500	-.7064517	.3449345
z5barlag	-.1795307	.1890717	-0.95	0.342	-.5502021	.1911407
z6barlag	.1391592	.3756895	0.37	0.711	-.5973725	.8756909
z7barlag	.430906	.4037833	1.07	0.286	-.3607032	1.222515
_cons	16.33104	9.106068	1.79	0.073	-1.521225	34.1833

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	128,640	.4424145	.0703047	.2011188	.8127233

Linear regression

Number of obs	= **128,640**
F(128, 4598)	= .
Prob > F	= .
R-squared	= **0.3053**
Root MSE	= **.20268**

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**4.564747**	**8.325443**	**0.55**	**0.584**	**-11.75712**	**20.88661**
__00009L	**.5937963**	**2.171845**	**0.27**	**0.785**	**-3.664063**	**4.851655**
__00009O	**-.8608651**	**4.900916**	**-0.18**	**0.861**	**-10.46901**	**8.747282**
xi	**.1450715**	**.0142385**	**10.19**	**0.000**	**.1171571**	**.1729859**
__000002	**-.0081835**	**.0053966**	**-1.52**	**0.129**	**-.0187634**	**.0023964**
__00000E	**-.1996901**	**.1318225**	**-1.51**	**0.130**	**-.4581254**	**.0587452**
__00000P	**-.1306909**	**.0773583**	**-1.69**	**0.091**	**-.2823502**	**.0209685**
__000017	**.3219988**	**.2747868**	**1.17**	**0.241**	**-.2167153**	**.8607128**
__00001I	**.2451433**	**.1716831**	**1.43**	**0.153**	**-.091438**	**.5817245**
__000020	**.0465067**	**.0713123**	**0.65**	**0.514**	**-.0932996**	**.186313**
__00002B	**.0499324**	**.0392375**	**1.27**	**0.203**	**-.026992**	**.1268568**
__00002T	**.3317416**	**.0880092**	**3.77**	**0.000**	**.1592013**	**.5042819**
__000034	**-.0424598**	**.0611807**	**-0.69**	**0.488**	**-.1624034**	**.0774838**
__00003M	**-.0824665**	**.0821755**	**-1.00**	**0.316**	**-.2435698**	**.0786368**
__00003X	**.0562813**	**.0513967**	**1.10**	**0.274**	**-.0444809**	**.1570436**
__00004F	**-.0214064**	**.0649749**	**-0.33**	**0.742**	**-.1487884**	**.1059755**
__00004Q	**-.0033885**	**.0382735**	**-0.09**	**0.929**	**-.0784229**	**.0716459**
__000058	**-.0616114**	**.095554**	**-0.64**	**0.519**	**-.248943**	**.1257203**
__00005J	**.0962271**	**.0576901**	**1.67**	**0.095**	**-.0168731**	**.2093274**
__00005U	**-1.072842**	**.6175735**	**-1.74**	**0.082**	**-2.283583**	**.1378984**
__00005V	**-.0686079**	**.112662**	**-0.61**	**0.543**	**-.2894794**	**.1522637**
__00005W	**.2956557**	**.2555724**	**1.16**	**0.247**	**-.2053889**	**.7967004**
__00005X	**1.107247**	**.6463172**	**1.71**	**0.087**	**-.1598453**	**2.374339**
__00005Y	**.006665**	**.1525531**	**0.04**	**0.965**	**-.2924124**	**.3057424**
__00005Z	**.4177337**	**.2272567**	**1.84**	**0.066**	**-.0277986**	**.863266**
__000060	**-.2003814**	**.1148141**	**-1.75**	**0.081**	**-.4254721**	**.0247093**
__000061	**.0627776**	**.0573305**	**1.10**	**0.274**	**-.0496177**	**.1751728**
__000062	**-.0079371**	**.0479**	**-0.17**	**0.868**	**-.101844**	**.0859698**
__000063	**.0234324**	**.0606999**	**0.39**	**0.699**	**-.0955686**	**.1424334**
__000064	**-.0476537**	**.0743598**	**-0.64**	**0.522**	**-.1934346**	**.0981272**
__000065	**4.601567**	**1.273473**	**3.61**	**0.000**	**2.104948**	**7.098186**
__000066	**.5353632**	**.231593**	**2.31**	**0.021**	**.0813297**	**.9893968**
__000067	**-2.104046**	**.5308536**	**-3.96**	**0.000**	**-3.144774**	**-1.063318**
__000068	**-3.209865**	**1.309425**	**-2.45**	**0.014**	**-5.776967**	**-.6427636**
__000069	**-.5733963**	**.2998063**	**-1.91**	**0.056**	**-1.161161**	**.0143679**
__00006A	**-.3073944**	**.5099257**	**-0.60**	**0.547**	**-1.307093**	**.6923047**
__00006B	**-.0013787**	**.1805182**	**-0.01**	**0.994**	**-.3552811**	**.3525236**
__00006C	**-.6917272**	**.1201482**	**-5.76**	**0.000**	**-.9272752**	**-.4561791**
__00006D	**.0978464**	**.0930479**	**1.05**	**0.293**	**-.0845721**	**.280265**
__00006E	**-.1126402**	**.1459926**	**-0.77**	**0.440**	**-.3988558**	**.1735754**
__00006F	**.0293601**	**.1813666**	**0.16**	**0.871**	**-.3262055**	**.3849256**
__00006G	**.228589**	**.3106349**	**0.74**	**0.462**	**-.3804045**	**.8375825**
__00006H	**.0378065**	**.0490674**	**0.77**	**0.441**	**-.0583892**	**.1340023**
__00006I	**-.1229966**	**.1220438**	**-1.01**	**0.314**	**-.362261**	**.1162679**
__00006J	**-.2612815**	**.3194111**	**-0.82**	**0.413**	**-.8874806**	**.3649176**
__00006K	**.042626**	**.077334**	**0.55**	**0.582**	**-.1089857**	**.1942377**
__00006L	**-.0857245**	**.1255653**	**-0.68**	**0.495**	**-.3318928**	**.1604438**
__00006M	**.0143417**	**.0615696**	**0.23**	**0.816**	**-.1063642**	**.1350476**
__00006N	**.0142187**	**.0300003**	**0.47**	**0.636**	**-.0445962**	**.0730337**
__00006O	**.0108951**	**.0238843**	**0.46**	**0.648**	**-.0359295**	**.0577198**
__00006P	**-.051378**	**.0341179**	**-1.51**	**0.132**	**-.1182654**	**.0155094**
__00006Q	**.0061354**	**.0401718**	**0.15**	**0.879**	**-.0726205**	**.0848914**
__00006R	**-.3954866**	**.5294751**	**-0.75**	**0.455**	**-1.433512**	**.6425387**
__00006S	**.0027485**	**.070072**	**0.04**	**0.969**	**-.1346264**	**.1401233**
__00006T	**.2422518**	**.1891001**	**1.28**	**0.200**	**-.1284752**	**.6129788**
__00006U	**-.2749271**	**.4753902**	**-0.58**	**0.563**	**-1.20692**	**.657066**
__00006V	**.1610918**	**.0976758**	**1.65**	**0.099**	**-.0303997**	**.3525833**
__00006W	**.1256103**	**.177919**	**0.71**	**0.480**	**-.2231964**	**.474417**
__00006X	**.0319531**	**.0688723**	**0.46**	**0.643**	**-.1030697**	**.1669758**
__00006Y	**.0856173**	**.0433777**	**1.97**	**0.048**	**.0005762**	**.1706584**
__00006Z	**-.1346033**	**.0315098**	**-4.27**	**0.000**	**-.1963778**	**-.0728289**

__000070	-.0896637	.0497818	-1.80	0.072	-.18726	.0079325
__000071	-.0242239	.0620782	-0.39	0.696	-.145927	.0974792
__000072	-.817985	.3639092	-2.25	0.025	-1.531422	-.1045482
__000073	-.1058926	.0632791	-1.67	0.094	-.2299501	.0181648
__000074	.4129247	.154788	2.67	0.008	.109466	.7163835
__000075	.4604809	.3667051	1.26	0.209	-.258437	1.179399
__000076	.1273306	.0859626	1.48	0.139	-.0411973	.2958585
__000077	.1296378	.1391035	0.93	0.351	-.1430718	.4023475
__000078	-.0674194	.0576484	-1.17	0.242	-.1804379	.0455991
__000079	.0872997	.0407137	2.14	0.032	.0074813	.167118
__00007A	.0612698	.02713	2.26	0.024	.008082	.1144575
__00007B	.0328333	.0413449	0.79	0.427	-.0482225	.1138891
__00007C	.0255229	.0482849	0.53	0.597	-.0691387	.1201845
__00007D	.354494	.2770601	1.28	0.201	-.1886768	.8976647
__00007E	.0890569	.0489815	1.82	0.069	-.0069704	.1850841
__00007F	-.2285898	.1151561	-1.99	0.047	-.4543511	-.0028285
__00007G	-.1664105	.2810873	-0.59	0.554	-.7174767	.3846556
__00007H	-.0569692	.0661062	-0.86	0.389	-.186569	.0726307
__00007I	-.0611722	.1122723	-0.54	0.586	-.2812799	.1589354
__00007J	.0315184	.0411415	0.77	0.444	-.0491387	.1121755
__00007K	-.0752832	.0302422	-2.49	0.013	-.1345724	-.015994
__00007L	.0125053	.0227207	0.55	0.582	-.0320382	.0570487
__00007M	.0286478	.0317049	0.90	0.366	-.0335091	.0908047
__00007N	.0174569	.0380542	0.46	0.646	-.0571476	.0920614
__00007O	.2851267	.4558075	0.63	0.532	-.6084747	1.178728
__00007P	.2207786	.0723679	3.05	0.002	.0789028	.3626543
__00007Q	-.3915989	.1823773	-2.15	0.032	-.7491459	-.0340519
__00007R	-.012365	.4576998	-0.03	0.978	-.9096763	.8849463
__00007S	-.1570488	.1017663	-1.54	0.123	-.3565595	.042462
__00007T	-.4053457	.1710711	-2.37	0.018	-.7407271	-.0699642
__00007U	.0972112	.0620585	1.57	0.117	-.0244534	.2188757
__00007V	-.2339363	.0420127	-5.57	0.000	-.3163014	-.1515711
__00007W	-.0218837	.0324529	-0.67	0.500	-.085507	.0417396
__00007X	.0212272	.0497507	0.43	0.670	-.076308	.1187624
__00007Y	.005633	.0584831	0.10	0.923	-.109022	.1202879
__000080	8.445766	18.96279	0.45	0.656	-28.73041	45.62194
__000082	-2.538594	11.18003	-0.23	0.820	-24.45681	19.37962
__000085	-.0754758	.5249525	-0.14	0.886	-1.104635	.9536832
__000087	.2107624	.3791253	0.56	0.578	-.5325052	.9540299
__00008A	-1.497372	3.68401	-0.41	0.684	-8.7198	5.725057
__00008C	.2927278	2.177494	0.13	0.893	-3.976205	4.561661
__00008F	-10.63528	23.58524	-0.45	0.652	-56.87368	35.60312
__00008H	3.123759	14.11997	0.22	0.825	-24.55816	30.80567
__00008K	-.1941854	.6776997	-0.29	0.774	-1.522802	1.134431
__00008M	-1.39456	.6534375	-2.13	0.033	-2.675611	-.1135085
__00008P	-1.343119	1.111617	-1.21	0.227	-3.522422	.8361837
__00008R	-1.271986	.9151353	-1.39	0.165	-3.066091	.5221185
__00008U	-.7730286	.4412188	-1.75	0.080	-1.638029	.091972
__00008W	.2278244	.3407778	0.67	0.504	-.4402638	.8959125
__00008Z	.601064	.2397331	2.51	0.012	.1310721	1.071056
__000091	-.0640346	.2041853	-0.31	0.754	-.4643358	.3362665
__000094	-.7268351	.2604519	-2.79	0.005	-1.237446	-.2162243
__000096	-.228038	.2082336	-1.10	0.274	-.6362758	.1801998
__000099	.7647842	.3113365	2.46	0.014	.1544152	1.375153
__00009B	.1977191	.261654	0.76	0.450	-.3152483	.7106865
__00009E	-.0660116	.3747056	-0.18	0.860	-.8006145	.6685913
__00009G	-.243422	.302943	-0.80	0.422	-.8373358	.3504917
xbarlag	-4.769469	22.18503	-0.21	0.830	-48.26277	38.72383
x2barlag	-.4538931	.5918067	-0.77	0.443	-1.614118	.7063322
xbar2lag	1.725566	4.421587	0.39	0.696	-6.942867	10.394
rootxbarlag	5.399595	27.44809	0.20	0.844	-48.41184	59.21103
z1barlag	1.81121	1.002554	1.81	0.071	-.1542767	3.776696
z2barlag	2.690187	1.511498	1.78	0.075	-.2730747	5.653448
z3barlag	.552162	.563691	0.98	0.327	-.5529429	1.657267
z4barlag	-.2955395	.2702011	-1.09	0.274	-.8252634	.2341844
z5barlag	1.024167	.2637745	3.88	0.000	.5070427	1.541292
z6barlag	-.9630149	.4686881	-2.05	0.040	-1.881869	-.0441613
z7barlag	.2824898	.5031912	0.56	0.575	-.7040064	1.268986
_cons	-3.250004	9.429703	-0.34	0.730	-21.73675	15.23674

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A2	**128,640**	**.4438109**	**.13429**	**.0976942**	**1.111178**

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"
957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9
960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
964 }
965
966 local r_RE      "`r_RE' `qhat_hats'"
967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }
978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
```

```
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS	Number of obs	=	128,640
				F(13, 128626)	=	57321.16
Model	5933.80789	13	456.446761	Prob > F	=	0.0000
Residual	1024.24524	128,626	.007962972	R-squared	=	0.8528
				Adj R-squared	=	0.8528
Total	6958.05313	128,639	.054089764	Root MSE	=	.08924

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	.0180296	.0009529	18.92	0.000	.0161619	.0198974
ybarx	.0030935	.0010502	2.95	0.003	.0010352	.0051518
xi2	-.0393058	.0005754	-68.31	0.000	-.0404336	-.0381779
ybar	-.1946598	.0026658	-73.02	0.000	-.1998848	-.1894348
xi	.4196294	.0016617	252.52	0.000	.4163724	.4228864
ybar_q1	.4287804	.0032091	133.61	0.000	.4224907	.4350702
pzi1	.22573	.001319	171.13	0.000	.2231448	.2283153
pzi2	.0193004	.0021507	8.97	0.000	.0150851	.0235157
pzi3	-.00296	.0006651	-4.45	0.000	-.0042635	-.0016564
pzi4	.0106379	.0006603	16.11	0.000	.0093438	.011932
pzi5	-.0145877	.0007109	-20.52	0.000	-.0159811	-.0131944
pzi6	.0026414	.000587	4.50	0.000	.0014908	.003792
pzi7	-.0039312	.0007606	-5.17	0.000	-.005422	-.0024403
_cons	-.0249549	.0016336	-15.28	0.000	-.0281566	-.0217531

```
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = -_b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                  local coef = _b[pzi`j']
    3.                  local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
```

```
1025}

1026
1027if `J'==3 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00498017
  Iteration 1:  GMM criterion Q(b) =  .00061572
  Iteration 2:  GMM criterion Q(b) =  .00018123
  Iteration 3:  GMM criterion Q(b) =  .00012528

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00675202
  Iteration 1:  GMM criterion Q(b) =  .00512747
  Iteration 2:  GMM criterion Q(b) =  .00508207

  GMM estimation

  Number of parameters =  20
  Number of moments    =  28
  Initial weight matrix: Unadjusted                 Number of obs   =    128,640
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
A11          |
         one |   .6063488   .0357253    16.97   0.000     .5363286     .676369
-------------+----------------------------------------------------------------
C1           |
         zi1 |   .4928823   .0463509    10.63   0.000     .4020362    .5837284
         zi2 |  -.1965658   .0766228    -2.57   0.010    -.3467437    -.046388
         zi3 |   .0011543   .0222672     0.05   0.959    -.0424887    .0447973
         zi4 |   .1027277   .0314652     3.26   0.001     .0410571    .1643984
         zi5 |     .05721   .0235431     2.43   0.015     .0110664    .1033536
         zi6 |   .0080166   .0186304     0.43   0.667    -.0284983    .0445315
         zi7 |   .0001728   .0230086     0.01   0.994    -.0449232    .0452689
-------------+----------------------------------------------------------------
C2           |
         zi1 |   .2864128   .0725217     3.95   0.000     .1442729    .4285527
         zi2 |  -.3656803   .1220157    -3.00   0.003    -.6048266   -.1265339
         zi3 |   .0050626   .0344776     0.15   0.883    -.0625123    .0726375
         zi4 |   .1673259   .0487888     3.43   0.001     .0717016    .2629502
         zi5 |   .1038069   .0293474     3.54   0.000     .0462871    .1613267
         zi6 |   .0000514   .0269239     0.00   0.998    -.0527184    .0528212
```

```
          zi7 |  -.0008804    .037622    -0.02   0.981    -.0746182    .0728573
--------------+----------------------------------------------------------------
          /d1 |  -.0908835   .0076594   -11.87   0.000    -.1058958   -.0758713
          /b1 |   .3974894   .0218415    18.20   0.000     .3546808    .4402979
       /AVA11 |   .5914222   .6227972     0.95   0.342    -.6292379    1.812082
       /AVA12 |   -.595739   .5485921    -1.09   0.278     -1.67096    .4794818
       /AVA22 |   .2854211    .518141     0.55   0.582    -.7301166    1.300959
------------------------------------------------------------------------------

Instruments for equation eq1_RE: __00009J __00009L __00009O xi __000002 __00000E
     __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M
     __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5
     __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
```

```
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compfsuXsegXreligX
      > sched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dro
      > pPre_RE.tex not found)
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compf
  > suXsegXreligXsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_D
  > ist_dropPre_RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  24757

1077unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640

1078estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4599

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  5.3831268

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compfsuXsegX
  > religXsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dro
  > pPre_RE.ster not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compfsuXsegXreligXs
  > ched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE
  > .ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
```

```
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
```

```
                                                    gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                 local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                         }
   30.                                 }
   31.                         }
   32.                 }
   33.         }
        if `Tindivp1'<=`T' {
                forval t=`Tindivp1'/`T' {
                        tempvar zi`t'xdiff
                        g double `zi`t'xdiff'=zi`t'*`ximxk'
                        local zg "`zg' zi`t'"
                        forval j=1/`J' {
                                tempvar zi`t'p`j'xdiff
                                g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
                                local pzg "`pzg' `zi`t'p`j''"
                                local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
                        }
                }
        }

        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
        // denoting zg as group-zs
        //  zg,pzgx,rprp,rprpx
        local rootprootp ""
        local rootprootpxdiff ""

        forval j=1/`J' {
                forval s=`j'/`J' {
                        tempvar xdiffrp`j'rp`s'
                        g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')

                     if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "

                     local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
                }
         }

        local pqhatxdiff ""
        forval j=1/`J' {
                forval s=1/`J' {
                        foreach name of global Alist {
                                tempvar   p`s'qikhat`j'xdiff`name'
                                g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'

                                local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
                        }
                }
         }

        // set instruments
        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
}
```

```
1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                          local `++t'
    3.                          local p_nice = subinstr("`p'",":_cons","",.)
    4.                          local est = coefs[1,`t']
    5.                          if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                  }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                 }
   16.         }
```

```
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >     `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1     `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
```

```
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00070619
  Iteration 1:  GMM criterion Q(b) =  .00034284
  Iteration 2:  GMM criterion Q(b) =  .00025979

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01121013
  Iteration 1:  GMM criterion Q(b) =  .01091197
  Iteration 2:  GMM criterion Q(b) =  .01017896
  Iteration 3:  GMM criterion Q(b) =  .01015145

  GMM estimation

  Number of parameters =  17
  Number of moments    =  63
  Initial weight matrix: Unadjusted                 Number of obs   =    128,640
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
C11          |
       _cons |   .6316835   .0285784    22.10   0.000     .5756707    .6876962
-------------+----------------------------------------------------------------
C12          |
       _cons |  -.0602877   .0444385    -1.36   0.175    -.1473854    .0268101
-------------+----------------------------------------------------------------
C13          |
       _cons |  -.0125609   .0120696    -1.04   0.298    -.0362168    .0110951
-------------+----------------------------------------------------------------
C14          |
       _cons |    .003487   .0117769     0.30   0.767    -.0195953    .0265692
-------------+----------------------------------------------------------------
C15          |
       _cons |  -.0116801    .020477    -0.57   0.568    -.0518143     .028454
------------------------------------------------------------------------------
```

C16						
_cons	**-.0225087**	**.0135312**	**-1.66**	**0.096**	**-.0490293**	**.0040118**
C17						
_cons	**-.0176941**	**.015536**	**-1.14**	**0.255**	**-.0481441**	**.0127559**
C21						
_cons	**.4719309**	**.0398987**	**11.83**	**0.000**	**.393731**	**.5501309**
C22						
_cons	**-.1299939**	**.0629262**	**-2.07**	**0.039**	**-.2533269**	**-.0066609**
C23						
_cons	**-.0165189**	**.0162531**	**-1.02**	**0.309**	**-.0483744**	**.0153366**
C24						
_cons	**-.0070961**	**.0199483**	**-0.36**	**0.722**	**-.046194**	**.0320019**
C25						
_cons	**.0100266**	**.0255116**	**0.39**	**0.694**	**-.0399752**	**.0600285**
C26						
_cons	**-.0332436**	**.0175596**	**-1.89**	**0.058**	**-.0676597**	**.0011725**
C27						
_cons	**-.0175159**	**.0218638**	**-0.80**	**0.423**	**-.0603682**	**.0253364**
d1						
_cons	**-.0925405**	**.0065272**	**-14.18**	**0.000**	**-.1053335**	**-.0797475**
b1						
_cons	**.4307305**	**.0189708**	**22.70**	**0.000**	**.3935483**	**.4679126**
A11						
one	**.265553**	**.1190223**	**2.23**	**0.026**	**.0322736**	**.4988324**

Instruments for equation **eq1_FE: __000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7 __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1 __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY __0000D5 _cons**

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
```

(file **diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compfsuXsegXreligX > sched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dro > pPre_FE.tex** not found)
(output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compf > suXsegXreligXsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_D > ist_dropPre_FE.tex)

```
1242
1243        estat overid
```

Test of overidentifying restriction:

Hansen's J chi2(**46**) = **1305.88** (p = **0.0000**)

```
1244        di "$S_DATE $S_TIME"
   8 Dec 2023 13:12:45
1245        unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  24757
1248        unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640
1249        estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  4599
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  5.3831268
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compfsuXsegX
  > religXsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dro
  > pPre_FE.ster not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compfsuXsegXreligXs
  > ched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE
  > .ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}
```

```
1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table2_36b.smcl
   log type:  smcl
  closed on:   8 Dec 2023, 13:12:45
```

A.3 Table 3

A.3.1 Columns 2 and 5

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\rep_table3_25.sm
> cl
  log type:  smcl
 opened on:   7 Dec 2023, 20:20:46

1 .
2 . // wrapper for analysis
3 .
4 . *global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"
5 . *global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"
6 . global ROOT "C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication"

7 .
8 . global CODE "$ROOT/code"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
  >                                                    change these according to what c
  > olumn we are doing (uncomment below)
50. *local group_def "dist"
51. local group_def "fsuXseg"

52. *local group_def "fsuXsegXreligXsched"
53.
54. // comparision group
55. local groupComp_def "fsuXsegXreligXsched"

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.         syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
70.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\mdelma\OneDrive -
      TUNI.fi\Tiedostot\replication/data/laspeyres_state_reshape.dta saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file C:\Users\mdelma\OneDrive -
      TUNI.fi\Tiedostot\replication/data/laspeyres_state_2cats_reshape.dta saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                               Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations              840   ->   280
  Number of variables                   5   ->   6
  j variable (3 values)       expend_type   ->   (dropped)
  xij variables:
                            laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------
93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file C:\Users\mdelma\OneDrive -
      TUNI.fi\Tiedostot\replication/data/laspeyres_state_3cats_reshape.dta saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg
  (43,637 missing values generated)

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"       egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                  egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)
  (43,637 missing values generated)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)
  (43,637 missing values generated)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }
```

```
236
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
```

```
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }

297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }

316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }

333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
```

```
351
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.         drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.         }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.         g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (82 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,576 real changes made, 82 to missing)

460 replace x2=x*x
  (73,576 real changes made, 82 to missing)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,576 real changes made)

478 replace x2=x2_temp
  (73,576 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (44,114 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3
  (4,705 observations deleted)

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)
```

```
488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\mdelma\AppData\Local\Temp\ST_2c78_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
```

```
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,757 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
549 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var'" _n
    5.                 }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
```

```
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=1

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1
```

```
625 scalar estimate_FE=1

626
627 // estimation options
628 global w_initial "winit(unadjusted)"

629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

630 global wmatrix "wmatrix(robust)"

631 global trace_level ""

632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

633 global technique ""

634 global maxiter ""

635 global technique_RE ""

636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

637 global derivatives "quickderivatives"

638 scalar init_FE_from_RE=1

639 global clust_var "state_district_round"

640 local clus_name "_clus_Dist"

641
642
643
644
645
646
647
648
649 // xxzx to keep names same
650 local wmatrix_desc ""

651 local sectorName "_urbOnly"

652 local dropZdesc "_dropPre"

653 local desc_struc_v0 ""

654 local inst_desc "_exactInst"

655 local inst_rootp "_noPPXdInst"

656 local weight_desc ""

657 local RE_sq_inst_desc ""

658 local cnstname ""

659 local actual_pqhat_desc ""

660 local expUpname "_updateExp"
```

```
661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
  (0 observations deleted)
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //               ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T
```

```
694 local Tm1=`T'-1

695 local Tindiv : word count $zlist_indiv

696
697 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
706
707 if `J'==2 {
708          replace p1=P1
  (150,184 real changes made)
709          replace p2=P2
  (150,184 real changes made)
710          replace qi1=Qi1
  (150,173 real changes made)
711          replace qk1=Qk1
  (150,173 real changes made)
712          replace qikbar1=Qikbar1
  (150,184 real changes made)
713          replace qi2=Qi2
  (150,184 real changes made)
714          replace qk2=Qk2
  (150,184 real changes made)
715          replace qikbar2=Qikbar2
  (150,184 real changes made)
716
717          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721          forv j = 1(1)3 {
    2.                 replace p`j'=P`j'
    3.                 replace qi`j'=Qi`j'
    4.                 replace qk`j'=Qk`j'
    5.                 replace qikbar`j'=Qikbar`j'
    6.          }
722
723          foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
724 }
```

```
725
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }
```

```
751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
773         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j'"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.         local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.         if `jplus1'<=`J' {
    5.                 forval k=`jplus1'/`J' {
    6.                         local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                 }
    8.         }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                 if noMeasError==1                                local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.         }
```

```
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                      l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                    )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2) )*( ({A11:}*p1*qbar1) +
  > ({A11:}*p2*qbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2) )
  >    + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:}) ) -
  >  ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}) )
  >  - ( 0 ) )-( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2) ) )*{b1} + ({A11:}*p1*qbar1)  + (
  > p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d1}/exp(
  >  {b1}*ln(p1/p2) + ln(p2) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > }*p1*qbar1) + ({A11:}*p2*qbar2) )  + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1
  > /p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1
  > *{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13
  > }*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))           ))
```

```
836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
                local RE_cz_2 "`RE_cz_2')^2"
        }

        local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}     `RE
> _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
> ' +{v0})))))"

        local xi_hat "({b}*p1*xi `FE_czi')"
        local xk_hat "({b}*p1*xk `FE_czk')"

        local eqs_FE
>                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
> - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
}

di "`eqs_RE'"
  (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qbar1) + ({A11:}*p2*qbar2) )*( ({A11:}*p1*qbar1)
>  + ({A11:}*p2*qbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C
> 2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2)
>  )    + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
> ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
> ) ) - ( 0 ) )-( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2) ) )*{b1} + ({A11:}*p1*qbar1)
> + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d1}/e
> xp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

di "`eqs_FE'"
  (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
>  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
> C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
> ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
> 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
> k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
> b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
> 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
> 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
> )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
> zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
>  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
> one}*p1*qbar1) + ({A11:}*p2*qbar2) )  + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
> (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
>  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
> C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))            )
> )

/*******************************************************************************
>  construct instruments
> *******************************************************************************/

//make squares and differences of x
tempvar x2i x2k ximxk x2imx2k xiP

g double `xiP'=xi*local_cpi

g double `x2i'=xi*xi

g double `x2k'=xi*xi

g double `ximxk'=xi-xk
```

```
883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.          tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.          g double `xizi`t''=xi*zi`t'
    4.          g double `xkzk`t''=xk*zk`t'
    5.          g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.          g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.          g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.          g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.          g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.          forval j=1/`J' {
   11.                  tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                  g double `zi`t'p`j''= zi`t'*p`j'
   13.                  g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                  g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                  g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                  forv s=1/`T' {
   17.                          tempvar zi`t'zi`s'p`j'
   18.                          g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                  }
   20.          }
   21. }

891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }

903
904 local zrg ""

905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var''"
    6.          }
    7. }
```

```
906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var''"
    5.
911          forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var''"
   10.                  local xprg "`xprg' `xp`j'`var''"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""

925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }
```

```
929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.         forval k=`j'/`J' {
    3.                 tempvar p`j'p`k'
    4.                 g `p`j'p`k''=p`j'*p`k'
    5.                 local roundinsts "`roundinsts' `p`j'p`k''"
    6.         }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.         if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.         if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }

  Linear regression                               Number of obs     =    150,184
                                                  F(129, 3940)      =     104.09
                                                  Prob > F          =     0.0000
                                                  R-squared         =     0.7989
                                                  Root MSE          =     .05753
```

(Std. err. adjusted for **3,941** clusters in **group_round**)

qbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**1.472261**	**.6918438**	**2.13**	**0.033**	**.1158558**	**2.828667**
__00009L	**-2.293606**	**.7230971**	**-3.17**	**0.002**	**-3.711286**	**-.8759264**
__00009O	**1.517118**	**.4854116**	**3.13**	**0.002**	**.5654359**	**2.468799**
xi	**.0266459**	**.0037155**	**7.17**	**0.000**	**.0193614**	**.0339305**
__000002	**-.0096958**	**.0012445**	**-7.79**	**0.000**	**-.0121358**	**-.0072559**
__00000E	**-.0896064**	**.037461**	**-2.39**	**0.017**	**-.1630511**	**-.0161617**
__00000P	**.030442**	**.025656**	**1.19**	**0.235**	**-.0198583**	**.0807422**
__000017	**.1760478**	**.0761769**	**2.31**	**0.021**	**.0266978**	**.3253977**
__00001I	**.0113994**	**.0519124**	**0.22**	**0.826**	**-.0903782**	**.1131771**
__000020	**-.0020968**	**.0219674**	**-0.10**	**0.924**	**-.0451654**	**.0409718**
__00002B	**-.0255406**	**.013045**	**-1.96**	**0.050**	**-.0511163**	**.0000351**
__00002T	**-.0571805**	**.0306253**	**-1.87**	**0.062**	**-.1172234**	**.0028624**
__000034	**-.0150745**	**.0256636**	**-0.59**	**0.557**	**-.0653896**	**.0352406**
__00003M	**.0502063**	**.0269018**	**1.87**	**0.062**	**-.0025365**	**.1029491**
__00003X	**-.0385089**	**.0214543**	**-1.79**	**0.073**	**-.0805714**	**.0035537**
__00004F	**.0321422**	**.019581**	**1.64**	**0.101**	**-.0062477**	**.070532**
__00004Q	**.024961**	**.0126072**	**1.98**	**0.048**	**.0002437**	**.0496783**
__000058	**.0637696**	**.0285256**	**2.24**	**0.025**	**.0078432**	**.1196959**
__00005J	**.0141898**	**.0169472**	**0.84**	**0.402**	**-.0190364**	**.047416**
__00005U	**.0655076**	**.0108513**	**6.04**	**0.000**	**.0442329**	**.0867823**
__00005V	**-.0313905**	**.0283405**	**-1.11**	**0.268**	**-.0869539**	**.0241729**
__00005W	**.0349377**	**.0384549**	**0.91**	**0.364**	**-.0404558**	**.1103312**
__00005X	**.0720165**	**.0778043**	**0.93**	**0.355**	**-.0805239**	**.224557**
__00005Y	**-.0726669**	**.0481613**	**-1.51**	**0.131**	**-.1670905**	**.0217566**
__00005Z	**-.0942904**	**.0715166**	**-1.32**	**0.187**	**-.2345035**	**.0459226**
__000060	**.0490123**	**.0269513**	**1.82**	**0.069**	**-.0038275**	**.101852**
__000061	**.010546**	**.0233546**	**0.45**	**0.652**	**-.0352422**	**.0563342**
__000062	**.0431798**	**.0162154**	**2.66**	**0.008**	**.0113885**	**.0749712**
__000063	**-.0138105**	**.0233626**	**-0.59**	**0.554**	**-.0596144**	**.0319934**
__000064	**-.0297774**	**.028281**	**-1.05**	**0.292**	**-.0852241**	**.0256693**
__000065	**-.0726562**	**.0181915**	**-3.99**	**0.000**	**-.1083219**	**-.0369905**
__000066	**.0438101**	**.0537893**	**0.81**	**0.415**	**-.0616473**	**.1492675**
__000067	**.0187583**	**.073931**	**0.25**	**0.800**	**-.1261884**	**.163705**
__000068	**-.3664581**	**.160804**	**-2.28**	**0.023**	**-.6817249**	**-.0511912**
__000069	**.213219**	**.0884642**	**2.41**	**0.016**	**.039779**	**.386659**
__00006A	**-.0449871**	**.1416955**	**-0.32**	**0.751**	**-.3227906**	**.2328163**
__00006B	**.0649789**	**.0539415**	**1.20**	**0.228**	**-.0407771**	**.1707348**
__00006C	**.0825627**	**.0554274**	**1.49**	**0.136**	**-.0261064**	**.1912317**
__00006D	**.02163**	**.029695**	**0.73**	**0.466**	**-.036589**	**.079849**
__00006E	**-.0523173**	**.0413956**	**-1.26**	**0.206**	**-.133476**	**.0288414**
__00006F	**.0713193**	**.0557106**	**1.28**	**0.201**	**-.037905**	**.1805436**
__00006G	**.0042771**	**.0045094**	**0.95**	**0.343**	**-.0045639**	**.0131181**
__00006H	**.0033586**	**.012944**	**0.26**	**0.795**	**-.022019**	**.0287363**
__00006I	**-.011659**	**.0198946**	**-0.59**	**0.558**	**-.0506637**	**.0273458**
__00006J	**.0199948**	**.0401403**	**0.50**	**0.618**	**-.0587029**	**.0986924**
__00006K	**-.0010501**	**.0239789**	**-0.04**	**0.965**	**-.0480623**	**.045962**
__00006L	**.0427175**	**.039465**	**1.08**	**0.279**	**-.0346562**	**.1200913**
__00006M	**-.0013286**	**.0147066**	**-0.09**	**0.928**	**-.0301618**	**.0275046**
__00006N	**.0144598**	**.0111774**	**1.29**	**0.196**	**-.0074542**	**.0363737**
__00006O	**-.0125683**	**.0075553**	**-1.66**	**0.096**	**-.0273809**	**.0022444**
__00006P	**-.0013814**	**.0117907**	**-0.12**	**0.907**	**-.0244978**	**.021735**
__00006Q	**.0058201**	**.014886**	**0.39**	**0.696**	**-.0233649**	**.0350052**
__00006R	**.0282911**	**.0085884**	**3.29**	**0.001**	**.0114529**	**.0451292**
__00006S	**-.0046756**	**.0252911**	**-0.18**	**0.853**	**-.0542604**	**.0449093**
__00006T	**.0029807**	**.0384967**	**0.08**	**0.938**	**-.0724947**	**.078456**
__00006U	**.0955095**	**.0688468**	**1.39**	**0.165**	**-.0394693**	**.2304883**
__00006V	**-.1376087**	**.0383964**	**-3.58**	**0.000**	**-.2128873**	**-.0623301**
__00006W	**.0375429**	**.0748489**	**0.50**	**0.616**	**-.1092034**	**.1842891**
__00006X	**.0092751**	**.0262504**	**0.35**	**0.724**	**-.0421906**	**.0607408**
__00006Y	**-.0144463**	**.0185238**	**-0.78**	**0.436**	**-.0507635**	**.0218708**
__00006Z	**-.0041786**	**.0129912**	**-0.32**	**0.748**	**-.0296487**	**.0212915**
__000070	**.0171026**	**.0190858**	**0.90**	**0.370**	**-.0203164**	**.0545216**
__000071	**-.0324389**	**.0270362**	**-1.20**	**0.230**	**-.0854452**	**.0205674**
__000072	**.0138758**	**.0081667**	**1.70**	**0.089**	**-.0021355**	**.0298871**
__000073	**-.0086359**	**.0221328**	**-0.39**	**0.696**	**-.0520287**	**.0347569**
__000074	**.0227287**	**.0298388**	**0.76**	**0.446**	**-.0357723**	**.0812297**
__000075	**.0056539**	**.0528147**	**0.11**	**0.915**	**-.0978929**	**.1092007**

__000076	-.0690788	.0315304	-2.19	0.029	-.1308964	-.0072613
__000077	-.0239226	.0489419	-0.49	0.625	-.1198764	.0720312
__000078	.0230691	.0194447	1.19	0.236	-.0150536	.0611918
__000079	-.0352916	.0212294	-1.66	0.097	-.0769133	.0063302
__00007A	.0002382	.0106598	0.02	0.982	-.020661	.0211374
__00007B	-.0162609	.0152881	-1.06	0.288	-.0462343	.0137124
__00007C	-.0385959	.0192282	-2.01	0.045	-.0762941	-.0008977
__00007D	-.0269314	.0053024	-5.08	0.000	-.037327	-.0165357
__00007E	-.0224253	.012535	-1.79	0.074	-.0470011	.0021504
__00007F	.0360008	.0183725	1.96	0.050	-.0000197	.0720212
__00007G	-.0490725	.0350147	-1.40	0.161	-.1177212	.0195761
__00007H	.0408719	.0234146	1.75	0.081	-.0050339	.0867777
__00007I	-.042028	.0367111	-1.14	0.252	-.1140027	.0299466
__00007J	-.0088674	.0139958	-0.63	0.526	-.0363071	.0185722
__00007K	.0010534	.0131447	0.08	0.936	-.0247177	.0268245
__00007L	.0122346	.0077478	1.58	0.114	-.0029555	.0274246
__00007M	-.0155798	.0115227	-1.35	0.176	-.0381708	.0070112
__00007N	.02653	.013688	1.94	0.053	-.0003063	.0533662
__00007O	-.0535043	.00701	-7.63	0.000	-.0672478	-.0397608
__00007P	-.0119392	.0199126	-0.60	0.549	-.0509792	.0271008
__00007Q	.0353063	.0273944	1.29	0.198	-.0184021	.0890148
__00007R	-.055516	.0567498	-0.98	0.328	-.1667777	.0557456
__00007S	.0863973	.0327288	2.64	0.008	.0222303	.1505644
__00007T	-.093108	.0532905	-1.75	0.081	-.1975877	.0113716
__00007U	-.0152682	.0198612	-0.77	0.442	-.0542074	.023671
__00007V	.0143846	.0164621	0.87	0.382	-.0178905	.0466597
__00007W	.0080573	.0104612	0.77	0.441	-.0124526	.0285671
__00007X	-.0153566	.0152502	-1.01	0.314	-.0452556	.0145425
__00007Y	.0132969	.0199013	0.67	0.504	-.0257209	.0523148
__000080	-.0409156	.0612033	-0.67	0.504	-.1609086	.0790775
__000082	-.1534962	.035607	-4.31	0.000	-.2233061	-.0836863
__000085	-.2028893	.1979479	-1.02	0.305	-.5909792	.1852006
__000087	.2538647	.1291869	1.97	0.049	.0005852	.5071441
__00008A	.7032561	.3269819	2.15	0.032	.0621864	1.344326
__00008C	-.2731025	.1976551	-1.38	0.167	-.6606184	.1144134
__00008F	-.9914366	.7395236	-1.34	0.180	-2.441322	.4584485
__00008H	-.1893579	.4099057	-0.46	0.644	-.9930052	.6142893
__00008K	-.3708142	.2607129	-1.42	0.155	-.8819592	.1403307
__00008M	.1525818	.2352379	0.65	0.517	-.3086177	.6137812
__00008P	.2990701	.4602059	0.65	0.516	-.603194	1.201334
__00008R	-.417898	.346351	-1.21	0.228	-1.096942	.2611461
__00008U	.3427594	.1731561	1.98	0.048	.0032754	.6822433
__00008W	.1310861	.1295726	1.01	0.312	-.1229496	.3851218
__00008Z	.2272457	.1234746	1.84	0.066	-.0148343	.4693258
__000091	-.1952287	.0997203	-1.96	0.050	-.390737	.0002796
__000094	.2683066	.1005018	2.67	0.008	.0712662	.4653469
__000096	-.0408552	.0778518	-0.52	0.600	-.1934887	.1117783
__000099	-.1816952	.1265735	-1.44	0.151	-.4298509	.0664606
__00009B	-.1632564	.1051443	-1.55	0.121	-.3693988	.042886
__00009E	-.0842415	.1642115	-0.51	0.608	-.4061891	.237706
__00009G	-.2036313	.1235552	-1.65	0.099	-.4458694	.0386069
xbar	.5774685	.0685478	8.42	0.000	.443076	.711861
x2barlag	-.1711193	.2146963	-0.80	0.425	-.5920457	.2498071
xbar2lag	-.3386727	.4029565	-0.84	0.401	-1.128696	.4513502
rootxbarlag	1.388775	.954018	1.46	0.146	-.4816411	3.25919
z1barlag	.2410824	.3921089	0.61	0.539	-.5276731	1.009838
z2barlag	.0868094	.6185475	0.14	0.888	-1.125894	1.299513
z3barlag	-.5872006	.2311831	-2.54	0.011	-1.04045	-.1339509
z4barlag	-.0176472	.1396663	-0.13	0.899	-.2914723	.2561778
z5barlag	-.27089	.1036452	-2.61	0.009	-.4740932	-.0676867
z6barlag	.3844184	.1845639	2.08	0.037	.0225687	.7462681
z7barlag	.2666148	.21424	1.24	0.213	-.1534168	.6866465
_cons	-.6744719	.7444171	-0.91	0.365	-2.133951	.7850072

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	150,184	.4400029	.1146201	.1343808	1.091543

Linear regression

Number of obs	**150,184**
F(129, 3940)	**46.92**
Prob > F	**0.0000**
R-squared	**0.6432**
Root MSE	**.02006**

(Std. err. adjusted for **3,941** clusters in **group_round**)

qbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**1.268115**	**.2573269**	**4.93**	**0.000**	**.7636091**	**1.772622**
__00009L	**-2.254662**	**.2567492**	**-8.78**	**0.000**	**-2.758036**	**-1.751288**
__00009O	**.7106114**	**.1829407**	**3.88**	**0.000**	**.351944**	**1.069279**
xi	**.0005362**	**.0017254**	**0.31**	**0.756**	**-.0028466**	**.003919**
__000002	**-.0002947**	**.0006094**	**-0.48**	**0.629**	**-.0014895**	**.0009001**
__00000E	**-.0022198**	**.0119312**	**-0.19**	**0.852**	**-.0256118**	**.0211722**
__00000P	**-.0102697**	**.0086816**	**-1.18**	**0.237**	**-.0272907**	**.0067512**
__000017	**.0256941**	**.0275542**	**0.93**	**0.351**	**-.0283277**	**.0797158**
__00001I	**.008931**	**.0179816**	**0.50**	**0.619**	**-.0263231**	**.044185**
__000020	**.0166961**	**.0071963**	**2.32**	**0.020**	**.0025874**	**.0308048**
__00002B	**-.0005393**	**.0044702**	**-0.12**	**0.904**	**-.0093034**	**.0082249**
__00002T	**-.021952**	**.0112602**	**-1.95**	**0.051**	**-.0440283**	**.0001244**
__000034	**.004035**	**.0077707**	**0.52**	**0.604**	**-.0112**	**.0192701**
__00003M	**.0039042**	**.0092637**	**0.42**	**0.673**	**-.0142578**	**.0220662**
__00003X	**.0009482**	**.0061172**	**0.16**	**0.877**	**-.011045**	**.0129415**
__00004F	**-.0112635**	**.0070488**	**-1.60**	**0.110**	**-.0250831**	**.002556**
__00004Q	**.0055266**	**.0045512**	**1.21**	**0.225**	**-.0033963**	**.0144495**
__000058	**-.0097235**	**.0102956**	**-0.94**	**0.345**	**-.0299088**	**.0104617**
__00005J	**.0098522**	**.0061504**	**1.60**	**0.109**	**-.0022061**	**.0219105**
__00005U	**-.0060878**	**.0037607**	**-1.62**	**0.106**	**-.0134609**	**.0012852**
__00005V	**-.0020102**	**.0118539**	**-0.17**	**0.865**	**-.0252505**	**.0212301**
__00005W	**-.0012622**	**.0155708**	**-0.08**	**0.935**	**-.0317899**	**.0292655**
__00005X	**.0321501**	**.0279015**	**1.15**	**0.249**	**-.0225526**	**.0868527**
__00005Y	**-.0310631**	**.0179671**	**-1.73**	**0.084**	**-.0662888**	**.0041626**
__00005Z	**.0293287**	**.0345026**	**0.85**	**0.395**	**-.038316**	**.0969734**
__000060	**.0072138**	**.0112275**	**0.64**	**0.521**	**-.0147985**	**.0292262**
__000061	**-.0012647**	**.0065028**	**-0.19**	**0.846**	**-.0140138**	**.0114844**
__000062	**.0087747**	**.0063237**	**1.39**	**0.165**	**-.0036233**	**.0211726**
__000063	**-.0120661**	**.0078314**	**-1.54**	**0.123**	**-.0274201**	**.0032879**
__000064	**-.021517**	**.0088396**	**-2.43**	**0.015**	**-.0388476**	**-.0041864**
__000065	**.0194192**	**.006141**	**3.16**	**0.002**	**.0073793**	**.031459**
__000066	**.0169204**	**.0206533**	**0.82**	**0.413**	**-.0235718**	**.0574125**
__000067	**-.0247215**	**.0279265**	**-0.89**	**0.376**	**-.0794733**	**.0300302**
__000068	**-.0017119**	**.0540535**	**-0.03**	**0.975**	**-.1076874**	**.1042635**
__000069	**-.0243774**	**.0317847**	**-0.77**	**0.443**	**-.0866934**	**.0379386**
__00006A	**-.0922716**	**.057248**	**-1.61**	**0.107**	**-.2045101**	**.0199669**
__00006B	**-.0065158**	**.0212218**	**-0.31**	**0.759**	**-.0481225**	**.0350908**
__00006C	**.0150637**	**.0178663**	**0.84**	**0.399**	**-.0199644**	**.0500918**
__00006D	**-.0155799**	**.0115671**	**-1.35**	**0.178**	**-.038258**	**.0070982**
__00006E	**-.0064152**	**.0145951**	**-0.44**	**0.660**	**-.0350298**	**.0221995**
__00006F	**.0251992**	**.019714**	**1.28**	**0.201**	**-.0134515**	**.0638499**
__00006G	**.000083**	**.0016681**	**0.05**	**0.960**	**-.0031874**	**.0033534**
__00006H	**.0019697**	**.0051755**	**0.38**	**0.704**	**-.0081772**	**.0121166**
__00006I	**.0012555**	**.0075773**	**0.17**	**0.868**	**-.0136003**	**.0161113**
__00006J	**-.0181925**	**.0123892**	**-1.47**	**0.142**	**-.0424824**	**.0060974**
__00006K	**.0141037**	**.0086285**	**1.63**	**0.102**	**-.0028131**	**.0310204**
__00006L	**-.0165611**	**.013979**	**-1.18**	**0.236**	**-.0439679**	**.0108456**
__00006M	**-.0039106**	**.0055438**	**-0.71**	**0.481**	**-.0147795**	**.0069583**
__00006N	**-.0035746**	**.0040345**	**-0.89**	**0.376**	**-.0114845**	**.0043353**
__00006O	**-.0030225**	**.0031577**	**-0.96**	**0.339**	**-.0092133**	**.0031683**
__00006P	**.0014991**	**.0036668**	**0.41**	**0.683**	**-.0056899**	**.0086881**
__00006Q	**.0010408**	**.0045476**	**0.23**	**0.819**	**-.007875**	**.0099566**
__00006R	**-.0054859**	**.0022181**	**-2.47**	**0.013**	**-.0098346**	**-.0011372**
__00006S	**-.0142932**	**.0092451**	**-1.55**	**0.122**	**-.0324189**	**.0038324**
__00006T	**.0043673**	**.0121405**	**0.36**	**0.719**	**-.019435**	**.0281696**
__00006U	**.0356445**	**.0243498**	**1.46**	**0.143**	**-.012095**	**.083384**
__00006V	**-.0049048**	**.0106518**	**-0.46**	**0.645**	**-.0257883**	**.0159788**
__00006W	**.0184877**	**.0264712**	**0.70**	**0.485**	**-.0334109**	**.0703862**
__00006X	**-.0087396**	**.0094821**	**-0.92**	**0.357**	**-.02733**	**.0098508**
__00006Y	**.0101674**	**.0058967**	**1.72**	**0.085**	**-.0013935**	**.0217284**
__00006Z	**.0078657**	**.0051821**	**1.52**	**0.129**	**-.0022943**	**.0180256**

__000070	-.0019016	.0060652	-0.31	0.754	-.0137929	.0099897
__000071	.0021768	.0077581	0.28	0.779	-.0130336	.0173871
__000072	.0003483	.0022497	0.15	0.877	-.0040624	.0047589
__000073	-.0153273	.0071587	-2.14	0.032	-.0293624	-.0012923
__000074	.0224614	.0094911	2.37	0.018	.0038535	.0410693
__000075	-.028314	.0172271	-1.64	0.100	-.0620888	.0054609
__000076	.001668	.0105511	0.16	0.874	-.0190181	.0223541
__000077	.0202338	.0180463	1.12	0.262	-.0151472	.0556147
__000078	.0059866	.0068863	0.87	0.385	-.0075145	.0194878
__000079	.0051142	.0059095	0.87	0.387	-.0064717	.0167001
__00007A	-.0053255	.004041	-1.32	0.188	-.0132482	.0025972
__00007B	.0017733	.0047202	0.38	0.707	-.0074811	.0110276
__00007C	.0102242	.006168	1.66	0.097	-.0018685	.022317
__00007D	-.0025402	.0018027	-1.41	0.159	-.0060744	.0009941
__00007E	-.0016479	.0046359	-0.36	0.722	-.0107369	.0074411
__00007F	-.0012453	.0065054	-0.19	0.848	-.0139995	.011509
__00007G	.004754	.0125404	0.38	0.705	-.0198322	.0293402
__00007H	-.0108534	.0088091	-1.23	0.218	-.0281243	.0064175
__00007I	.0080317	.0144294	0.56	0.578	-.0202582	.0363215
__00007J	.0065156	.0052235	1.25	0.212	-.0037254	.0167565
__00007K	-.0030172	.004294	-0.70	0.482	-.0114359	.0054015
__00007L	-.0016641	.0037618	-0.44	0.658	-.0090394	.0057112
__00007M	.0049575	.0038576	1.29	0.199	-.0026056	.0125207
__00007N	.0046486	.0049103	0.95	0.344	-.0049783	.0142756
__00007O	.00121	.0024732	0.49	0.625	-.003639	.0060589
__00007P	.0093174	.0076235	1.22	0.222	-.005629	.0242638
__00007Q	-.014358	.0091437	-1.57	0.116	-.0322848	.0035689
__00007R	-.0062047	.0268126	-0.23	0.817	-.0587725	.0463631
__00007S	-.0187593	.0124429	-1.51	0.132	-.0431545	.0056358
__00007T	.0107971	.0238842	0.45	0.651	-.0360295	.0576237
__00007U	.0124962	.0083694	1.49	0.135	-.0039125	.0289049
__00007V	-.0054457	.004492	-1.21	0.225	-.0142526	.0033612
__00007W	.0002044	.0047643	0.04	0.966	-.0091364	.0095452
__00007X	.0006726	.0052633	0.13	0.898	-.0096464	.0109916
__00007Y	.0011474	.0066848	0.17	0.864	-.0119586	.0142534
__000080	.0506739	.0193062	2.62	0.009	.0128227	.088525
__000082	-.0380091	.0155484	-2.44	0.015	-.0684928	-.0075254
__000085	.0310938	.0683459	0.45	0.649	-.1029029	.1650904
__000087	-.0742174	.0462761	-1.60	0.109	-.1649447	.0165099
__00008A	-.1025461	.1156783	-0.89	0.375	-.3293411	.124249
__00008C	-.0155359	.0771627	-0.20	0.840	-.1668186	.1357467
__00008F	.1827979	.2826161	0.65	0.518	-.3712897	.7368855
__00008H	.5162288	.2020291	2.56	0.011	.1201373	.9123203
__00008K	.6783722	.0886852	7.65	0.000	.504499	.8522454
__00008M	-.2393485	.0951935	-2.51	0.012	-.4259816	-.0527153
__00008P	-.7984727	.1582341	-5.05	0.000	-1.108701	-.4882443
__00008R	.1286617	.1590164	0.81	0.419	-.1831005	.440424
__00008U	-.1623582	.0617927	-2.63	0.009	-.2835069	-.0412096
__00008W	-.0528083	.0556521	-0.95	0.343	-.161918	.0563014
__00008Z	.0261445	.0432467	0.60	0.546	-.0586434	.1109325
__000091	.0465468	.0292669	1.59	0.112	-.0108329	.1039265
__000094	.0023976	.0366211	0.07	0.948	-.0694004	.0741956
__000096	.085122	.0338844	2.51	0.012	.0186894	.1515546
__000099	-.0012575	.0430353	-0.03	0.977	-.085631	.083116
__00009B	.0256931	.0384438	0.67	0.504	-.0496785	.1010646
__00009E	-.0044042	.0525207	-0.08	0.933	-.1073744	.098566
__00009G	.0157774	.0457536	0.34	0.730	-.0739257	.1054804
xbar	.0546318	.0237303	2.30	0.021	.0081069	.1011567
x2barlag	.0544669	.0707712	0.77	0.442	-.0842848	.1932186
xbar2lag	.1267469	.1357989	0.93	0.351	-.1394959	.3929897
rootxbarlag	-.7869211	.3264446	-2.41	0.016	-1.426937	-.1469049
z1barlag	-.3981229	.136814	-2.91	0.004	-.6663559	-.12989
z2barlag	.6742386	.2241703	3.01	0.003	.234738	1.113739
z3barlag	.2184938	.0872519	2.50	0.012	.0474307	.3895569
z4barlag	-.083326	.0463003	-1.80	0.072	-.1741007	.0074488
z5barlag	-.0896707	.0394488	-2.27	0.023	-.1670128	-.0123286
z6barlag	-.0172463	.0636288	-0.27	0.786	-.1419947	.1075021
z7barlag	-.0145517	.0720601	-0.20	0.840	-.1558303	.1267269
_cons	.3466295	.2474142	1.40	0.161	-.1384424	.8317014

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A2	**150,184**	**.0897089**	**.0269184**	**.0241319**	**.2270752**

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"
957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9
960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
964 }
965
966 local r_RE      "`r_RE' `qhat_hats'"
967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }
978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
```

```
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS	Number of obs	=	150,184
				F(13, 150170)	=	68115.79
Model	6864.7597	13	528.058438	Prob > F	=	0.0000
Residual	1164.17266	150,170	.007752365	R-squared	=	0.8550
				Adj R-squared	=	0.8550
Total	8028.93236	150,183	.053460993	Root MSE	=	.08805

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	.0239142	.0010556	22.65	0.000	.0218452	.0259833
ybarx	.0007801	.0010663	0.73	0.464	-.0013098	.00287
xi2	-.0398479	.0005318	-74.93	0.000	-.0408902	-.0388056
ybar	-.2250685	.002763	-81.46	0.000	-.2304839	-.219653
xi	.4263369	.0015386	277.09	0.000	.4233213	.4293525
ybar_q1	.4845681	.0032033	151.27	0.000	.4782897	.4908464
pzi1	.2192719	.0012031	182.25	0.000	.2169137	.22163
pzi2	.0216977	.0019671	11.03	0.000	.0178422	.0255533
pzi3	-.0026384	.000611	-4.32	0.000	-.0038359	-.0014409
pzi4	.0084738	.0006135	13.81	0.000	.0072714	.0096763
pzi5	-.0132719	.000639	-20.77	0.000	-.0145244	-.0120195
pzi6	.0031967	.000531	6.02	0.000	.0021558	.0042375
pzi7	-.0039162	.0006956	-5.63	0.000	-.0052796	-.0025528
_cons	-.029631	.0015825	-18.72	0.000	-.0327326	-.0265293

```
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = -_b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                  local coef = _b[pzi`j']
    3.                  local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
```

```
1025}

1026
1027if `J'==3 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .07339596
  Iteration 1:  GMM criterion Q(b) =  .00198058
  Iteration 2:  GMM criterion Q(b) =   .0002836
  Iteration 3:  GMM criterion Q(b) =  .00014193

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01137726
  Iteration 1:  GMM criterion Q(b) =  .00933308
  Iteration 2:  GMM criterion Q(b) =  .00892579

  GMM estimation

  Number of parameters =  20
  Number of moments    =  28
  Initial weight matrix: Unadjusted                 Number of obs   =    150,184
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
A11          |
         one |   .0380387   .0171205     2.22   0.026     .0044831    .0715943
-------------+----------------------------------------------------------------
C1           |
         zi1 |   .4501073   .0561473     8.02   0.000     .3400606    .5601539
         zi2 |  -.2438472   .0870934    -2.80   0.005    -.4145471   -.0731473
         zi3 |  -.0316469   .0221652    -1.43   0.153    -.0750899     .011796
         zi4 |   .1410297   .0431481     3.27   0.001     .0564609    .2255984
         zi5 |   .0516662   .0335526     1.54   0.124    -.0140956    .1174281
         zi6 |   .0436955   .0238725     1.83   0.067    -.0030937    .0904847
         zi7 |    .042254   .0270718     1.56   0.119    -.0108058    .0953139
-------------+----------------------------------------------------------------
C2           |
         zi1 |   .1634808   .0961175     1.70   0.089     -.024906    .3518677
         zi2 |  -.3998531   .1430218    -2.80   0.005    -.6801707   -.1195355
         zi3 |  -.0521307   .0351929    -1.48   0.139    -.1211075    .0168461
         zi4 |   .2221515   .0698896     3.18   0.001     .0851703    .3591327
         zi5 |   .1080079   .0479615     2.25   0.024     .0140051    .2020106
         zi6 |   .0720837   .0376144     1.92   0.055    -.0016392    .1458066
```

```
------------------------------------------------------------------------------
          zi7 |   .0962101   .0450929     2.13   0.033     .0078295    .1845906
------------------------------------------------------------------------------
          /d1 |  -.0667852   .0054587   -12.23   0.000    -.0774841   -.0560862
          /b1 |   .4531422   .0181015    25.03   0.000     .4176639    .4886206
       /AVA11 |    4.45648   1.647805     2.70   0.007     1.226842    7.686119
       /AVA12 |  -3.687284   1.504303    -2.45   0.014    -6.635665   -.7389037
       /AVA22 |   2.935842   1.360636     2.16   0.031     .2690435    5.60264
------------------------------------------------------------------------------
Instructments for equation eq1_RE: __00009J __00009L __00009O xi __000002 __00000E
     __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M
     __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5
     __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
```

```
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_drop3_
      > all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dro
      > pPre_RE.tex not found)
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXrelig
  > Xsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_D
  > ist_dropPre_RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  150184

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  24757

1077unique group_round
  Number of unique values of group_round is  3941
  Number of records is  150184

1078estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  3941

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  6.2819081

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_
  > drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dro
  > pPre_RE.ster not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_drop3_a
  > ll_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dropPre_RE
  > .ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
```

```
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
```

```
1119                                                    gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                     local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                          }
   30.                                    }
   31.                           }
   32.                   }
   33.           }
1120          if `Tindivp1'<=`T' {
1121                  forval t=`Tindivp1'/`T' {
    2.                           tempvar zi`t'xdiff
    3.                           g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                           local zg "`zg' zi`t'"
    5.                           forval j=1/`J' {
    6.                                   tempvar zi`t'p`j'xdiff
    7.                                   g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                   local pzg "`pzg' `zi`t'p`j''"
    9.                                   local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                           }
   11.                   }
1122          }
1123
1124          // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1125          // denoting zg as group-zs
1126          //  zg,pzgx,rprp,rprpx
1127          local rootprootp ""
1128          local rootprootpxdiff ""
1129
1130          forval j=1/`J' {
    2.                   forval s=`j'/`J' {
    3.                           tempvar xdiffrp`j'rp`s'
    4.                           g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1131                           if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1132                           local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                   }
    8.           }
1133
1134          local pqhatxdiff ""
1135          forval j=1/`J' {
    2.                   forval s=1/`J' {
    3.                           foreach name of global Alist {
    4.                                   tempvar   p`s'qikhat`j'xdiff`name'
    5.                                   g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
1136                                    local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                           }
    8.                   }
    9.           }
1137
1138          // set instruments
1139          local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1140}
```

```
1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                tempvar ximxk`var'
    3.                g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                        local `++t'
    3.                        local p_nice = subinstr("`p'",":_cons","",.)
    4.                        local est = coefs[1,`t']
    5.                        if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                        if `j'>`i' {
   11.                                tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                        }
   15.                }
   16.         }
```

```
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >     `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1     `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
```

```
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00096614
  Iteration 1:  GMM criterion Q(b) =   .0005016
  Iteration 2:  GMM criterion Q(b) =  .00024085

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01246014
  Iteration 1:  GMM criterion Q(b) =   .0111269
  Iteration 2:  GMM criterion Q(b) =  .01100962
  Iteration 3:  GMM criterion Q(b) =  .01099593

  GMM estimation

  Number of parameters =  17
  Number of moments    =  63
  Initial weight matrix: Unadjusted                 Number of obs   =    150,184
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
C11          |
       _cons |   .6018655   .0259454    23.20   0.000     .5510135    .6527175
-------------+----------------------------------------------------------------
C12          |
       _cons |  -.0568268    .045108    -1.26   0.208    -.1452368    .0315832
-------------+----------------------------------------------------------------
C13          |
       _cons |  -.0063337   .0112319    -0.56   0.573    -.0283478    .0156803
-------------+----------------------------------------------------------------
C14          |
       _cons |   .0041332   .0120867     0.34   0.732    -.0195562    .0278226
-------------+----------------------------------------------------------------
C15          |
       _cons |  -.0205276   .0193979    -1.06   0.290    -.0585468    .0174916
------------------------------------------------------------------------------
```

C16						
_cons	**-.0197834**	**.0138242**	**-1.43**	**0.152**	**-.0468783**	**.0073115**
C17						
_cons	**-.0171346**	**.0156252**	**-1.10**	**0.273**	**-.0477594**	**.0134902**
C21						
_cons	**.4331446**	**.0366339**	**11.82**	**0.000**	**.3613435**	**.5049457**
C22						
_cons	**-.1123753**	**.0650498**	**-1.73**	**0.084**	**-.2398706**	**.01512**
C23						
_cons	**-.0083976**	**.0154441**	**-0.54**	**0.587**	**-.0386675**	**.0218724**
C24						
_cons	**-.0056939**	**.0203723**	**-0.28**	**0.780**	**-.0456229**	**.0342351**
C25						
_cons	**-.0009164**	**.0247008**	**-0.04**	**0.970**	**-.0493291**	**.0474962**
C26						
_cons	**-.0296676**	**.0180979**	**-1.64**	**0.101**	**-.0651389**	**.0058037**
C27						
_cons	**-.0147515**	**.0217432**	**-0.68**	**0.497**	**-.0573675**	**.0278644**
d1						
_cons	**-.0911591**	**.0038943**	**-23.41**	**0.000**	**-.0987919**	**-.0835264**
b1						
_cons	**.4195372**	**.0139553**	**30.06**	**0.000**	**.3921853**	**.446889**
A11						
one	**.5593808**	**.0887801**	**6.30**	**0.000**	**.385375**	**.7333866**

Instruments for equation **eq1_FE: __000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7 __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1 __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY __0000D5 _cons**

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
```

(file **diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dropPre_FE.tex** not found)
(output written to diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dropPre_FE.tex)

```
1242
1243        estat overid
```

Test of overidentifying restriction:

Hansen's J chi2(**46**) = **1651.41** (p = **0.0000**)

```
1244        di "$S_DATE $S_TIME"
   7 Dec 2023 20:26:16
1245        unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  150184
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  24757
1248        unique group_round
  Number of unique values of group_round is  3941
  Number of records is  150184
1249        estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  3941
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  6.2819081
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_
  > drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dro
  > pPre_FE.ster not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXseg_compfsuXsegXreligXsched_drop3_a
  > ll_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dropPre_FE
  > .ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}
```

```
1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\rep_table3_25.sm
  > cl
    log type:  smcl
   closed on:   7 Dec 2023, 20:26:16
  ──────────────────────────────────────────────────────────────────────────────────────
```

A.3.2 Columns 3 and 6

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\rep_table3_36.sm
> cl
  log type:  smcl
 opened on:   7 Dec 2023, 20:32:11

1 .
2 . // wrapper for analysis
3 .
4 . *global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"
5 . *global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"
6 . global ROOT "C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication"

7 .
8 . global CODE "$ROOT/code"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
  >                                                    change these according to what c
  > olumn we are doing (uncomment below)
50. *local group_def "dist"
51. local group_def "fsuXseg"

52. local group_def "fsuXsegXreligXsched"

53.
54. // comparision group
55. local groupComp_def "fsuXsegXreligXsched"

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.          syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
70.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\mdelma\OneDrive -
      TUNI.fi\Tiedostot\replication/data/laspeyres_state_reshape.dta saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file C:\Users\mdelma\OneDrive -
      TUNI.fi\Tiedostot\replication/data/laspeyres_state_2cats_reshape.dta saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations               840   ->   280
  Number of variables                    5   ->   6
  j variable (3 values)       expend_type   ->   (dropped)
  xij variables:
                         laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------
93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file C:\Users\mdelma\OneDrive -
      TUNI.fi\Tiedostot\replication/data/laspeyres_state_3cats_reshape.dta saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean           .6966798
                       Largest        Std. dev.       1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance       1.079168
95%     2.451768       26.1213        Skewness       5.702664
99%      4.56912       29.15594       Kurtosis       76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,637 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)
  (43,637 missing values generated)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)
  (43,637 missing values generated)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }
```

```
236
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
```

```
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }

297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }

316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }

333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
```

```
351
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.         drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.         }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.         g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (82 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,576 real changes made, 82 to missing)

460 replace x2=x*x
  (73,576 real changes made, 82 to missing)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,576 real changes made)

478 replace x2=x2_temp
  (73,576 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (48,819 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3
  (0 observations deleted)

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)
```

```
488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\mdelma\AppData\Local\Temp\ST_2c78_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
```

```
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,757 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
forval j=1/4 {
2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
> ght_group_round-weighti-weightk)
3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
> j')/(weight_group_round-weighti-weightk)
4. }

// summary stats tables
if `make_sumstats'==1 {
        count
        local sumstats_n_obs_pair = r(N)

        cap file close sumstat
                file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
> place
                file write sumstat "\begin{table}[htbp]\centering" _n
                file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
                file write sumstat "\caption{Summary statistics for consumption data
> }\label{tab:sumstats}" _n
                file write sumstat "\begin{tabular}{lcccccccc}" _n
                file write sumstat "\toprule" _n
                file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
> mn{4}{c}{Pairs }\tabularnewline" _n
                file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
> _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
> wline" _n
                file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
> SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
                file write sumstat "\midrule" _n

                foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
2.                         qui su `var'
3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
> d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
> max)',.01)' "
                        local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
> gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
> )' \\"
4.                         file write sumstat "`sumstats_row_`var'" _n
5.                 }

                file write sumstat "\bottomrule" _n
                file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
>  reports summary statistics for estimation sample.}\\ " _n
                file write sumstat "\end{tabular}" _n
                file write sumstat "\end{table}" _n
                file close sumstat
                BREAK
}

// constant for estimation
g one=1

egen state_district_round=group(state district urban round)

egen state_district=group(state district)

```

```
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=1

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1
```

```
625 scalar estimate_FE=1

626
627 // estimation options
628 global w_initial "winit(unadjusted)"

629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

630 global wmatrix "wmatrix(robust)"

631 global trace_level ""

632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

633 global technique ""

634 global maxiter ""

635 global technique_RE ""

636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

637 global derivatives "quickderivatives"

638 scalar init_FE_from_RE=1

639 global clust_var "state_district_round"

640 local clus_name "_clus_Dist"

641
642
643
644
645
646
647
648
649 // xxzx to keep names same
650 local wmatrix_desc ""

651 local sectorName "_urbOnly"

652 local dropZdesc "_dropPre"

653 local desc_struc_v0 ""

654 local inst_desc "_exactInst"

655 local inst_rootp "_noPPXdInst"

656 local weight_desc ""

657 local RE_sq_inst_desc ""

658 local cnstname ""

659 local actual_pqhat_desc ""

660 local expUpname "_updateExp"
```

```
661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
  (0 observations deleted)
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //               ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T
```

```
694 local Tm1=`T'-1

695 local Tindiv : word count $zlist_indiv

696
697 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
706
707 if `J'==2 {
708          replace p1=P1
  (128,640 real changes made)
709          replace p2=P2
  (128,640 real changes made)
710          replace qi1=Qi1
  (128,632 real changes made)
711          replace qk1=Qk1
  (128,632 real changes made)
712          replace qikbar1=Qikbar1
  (128,640 real changes made)
713          replace qi2=Qi2
  (128,640 real changes made)
714          replace qk2=Qk2
  (128,640 real changes made)
715          replace qikbar2=Qikbar2
  (128,640 real changes made)
716
717          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721          forv j = 1(1)3 {
    2.                 replace p`j'=P`j'
    3.                 replace qi`j'=Qi`j'
    4.                 replace qk`j'=Qk`j'
    5.                 replace qikbar`j'=Qikbar`j'
    6.          }
722
723          foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
724 }
```

```
725
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }
```

```
751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j'"
    3.          local pCi "`pCi' + `pCi`j'"
    4.          local pCk "`pCk' + `pCk`j'"
    5.          local pCg "`pCg' + `pCg`j'"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j'"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
773         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.         local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.         if `jplus1'<=`J' {
    5.                 forval k=`jplus1'/`J' {
    6.                         local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                 }
    8.         }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                 if noMeasError==1                                local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.         }
```

```
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                      l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                     )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2) )*( ({A11:}*p1*qbar1) +
  > ({A11:}*p2*qbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2) )
  >    + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:}) ) -
  >  ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}) )
  >  - ( 0 ) )-( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2) ) )*{b1} + ({A11:}*p1*qbar1)  + (
  > p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d1}/exp(
  >  {b1}*ln(p1/p2) + ln(p2) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > }*p1*qbar1) + ({A11:}*p2*qbar2) )  + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1
  > /p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1
  > *{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13
  > }*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))           ))
```

```
836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qbar1) + ({A11:}*p2*qbar2) )*( ({A11:}*p1*qbar1)
  >  + ({A11:}*p2*qbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C
  > 2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2)
  >  )    + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2) ) )*{b1} + ({A11:}*p1*qbar1)
  > + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d1}/e
  > xp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > one}*p1*qbar1) + ({A11:}*p2*qbar2) )  + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))            )
  > )

874
875 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi

882 g double `ximxk'=xi-xk
```

```
883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.          tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.          g double `xizi`t''=xi*zi`t'
    4.          g double `xkzk`t''=xk*zk`t'
    5.          g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.          g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.          g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.          g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.          g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.          forval j=1/`J' {
   11.                  tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                  g double `zi`t'p`j''= zi`t'*p`j'
   13.                  g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                  g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                  g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                  forv s=1/`T' {
   17.                          tempvar zi`t'zi`s'p`j'
   18.                          g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                  }
   20.          }
   21. }

891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }

903
904 local zrg ""

905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var''"
    6.          }
    7. }
```

```
906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var''"
    5.
911          forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var''"
   10.                  local xprg "`xprg' `xp`j'`var''"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""

925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }
```

```
929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.         forval k=`j'/`J' {
    3.                 tempvar p`j'p`k'
    4.                 g `p`j'p`k''=p`j'*p`k'
    5.                 local roundinsts "`roundinsts' `p`j'p`k''"
    6.         }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.         if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.         if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }

  Linear regression                               Number of obs     =    128,640
                                                  F(129, 4598)      =     118.07
                                                  Prob > F          =     0.0000
                                                  R-squared         =     0.8006
                                                  Root MSE          =      .0607
```

(Std. err. adjusted for **4,599** clusters in **group_round**)

qbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	1.377733	.7233567	1.90	0.057	-.0403931	2.79586
__00009L	-2.097563	.7597341	-2.76	0.006	-3.587006	-.608119
__00009O	1.553148	.4981071	3.12	0.002	.5766194	2.529677
xi	.0286665	.0040719	7.04	0.000	.0206837	.0366493
__000002	-.0109584	.0013579	-8.07	0.000	-.0136205	-.0082962
__00000E	-.0747394	.0403336	-1.85	0.064	-.1538127	.0043339
__00000P	.0279856	.0257329	1.09	0.277	-.0224632	.0784345
__000017	.2021934	.0813308	2.49	0.013	.0427459	.3616409
__00001I	-.001796	.0551311	-0.03	0.974	-.1098794	.1062874
__000020	.0080278	.0233157	0.34	0.731	-.0376822	.0537377
__00002B	-.0316132	.0134583	-2.35	0.019	-.057998	-.0052284
__00002T	-.0962501	.0321952	-2.99	0.003	-.1593682	-.0331321
__000034	.0091249	.0284852	0.32	0.749	-.0467197	.0649696
__00003M	.0425839	.0292451	1.46	0.145	-.0147507	.0999184
__00003X	-.0238057	.022708	-1.05	0.295	-.0683243	.0207129
__00004F	.0319966	.0201442	1.59	0.112	-.0074957	.0714889
__00004Q	.0205072	.0133226	1.54	0.124	-.0056116	.046626
__000058	.0833191	.0293717	2.84	0.005	.0257364	.1409018
__00005J	-.0015768	.0183506	-0.09	0.932	-.0375527	.0343991
__00005U	.063911	.0096163	6.65	0.000	.0450585	.0827635
__00005V	-.0125186	.0316794	-0.40	0.693	-.0746255	.0495883
__00005W	.0204862	.0422975	0.48	0.628	-.0624372	.1034096
__00005X	.0358449	.0820548	0.44	0.662	-.1250218	.1967116
__00005Y	-.0556555	.0515832	-1.08	0.281	-.1567833	.0454724
__00005Z	-.0171623	.0743769	-0.23	0.818	-.1629768	.1286522
__000060	.0474738	.027642	1.72	0.086	-.0067178	.1016654
__000061	-.0025432	.0247905	-0.10	0.918	-.0511446	.0460582
__000062	.037586	.0167438	2.24	0.025	.0047601	.0704118
__000063	-.027161	.0249255	-1.09	0.276	-.0760271	.021705
__000064	-.0380997	.0294676	-1.29	0.196	-.0958702	.0196709
__000065	-.0818605	.0180611	-4.53	0.000	-.1172689	-.0464522
__000066	.0398229	.0595446	0.67	0.504	-.0769131	.1565589
__000067	.0049155	.0810126	0.06	0.952	-.1539081	.1637392
__000068	-.279866	.1669769	-1.68	0.094	-.607221	.0474889
__000069	.2324353	.0917789	2.53	0.011	.0525045	.412366
__00006A	-.1605312	.15037	-1.07	0.286	-.4553287	.1342662
__00006B	.0448277	.0582075	0.77	0.441	-.069287	.1589423
__00006C	.0939268	.0587942	1.60	0.110	-.021338	.2091917
__00006D	.0236966	.0303581	0.78	0.435	-.0358199	.0832131
__00006E	-.0833222	.0441991	-1.89	0.059	-.1699737	.0033293
__00006F	.0544052	.0567827	0.96	0.338	-.0569161	.1657265
__00006G	.0042694	.0045688	0.93	0.350	-.0046876	.0132263
__00006H	.0112019	.0141595	0.79	0.429	-.0165575	.0389613
__00006I	-.0210649	.0219323	-0.96	0.337	-.0640627	.0219328
__00006J	.0182496	.0422972	0.43	0.666	-.0646732	.1011724
__00006K	.0198813	.0244116	0.81	0.415	-.0279771	.0677397
__00006L	.0285153	.0426834	0.67	0.504	-.0551647	.1121952
__00006M	-.0062334	.0138748	-0.45	0.653	-.0334346	.0209678
__00006N	.0092966	.0116915	0.80	0.427	-.0136244	.0322175
__00006O	-.0123172	.0077693	-1.59	0.113	-.0275488	.0029144
__00006P	-.0010748	.0130467	-0.08	0.934	-.0266526	.024503
__00006Q	.0054379	.0165759	0.33	0.743	-.0270587	.0379346
__00006R	.0316194	.0088273	3.58	0.000	.0143136	.0489252
__00006S	-.013146	.0271739	-0.48	0.629	-.0664199	.0401279
__00006T	.0041755	.0418359	0.10	0.921	-.077843	.0861939
__00006U	.1349361	.0714839	1.89	0.059	-.0052066	.2750788
__00006V	-.1399381	.0432548	-3.24	0.001	-.2247382	-.055138
__00006W	.0121051	.0825303	0.15	0.883	-.1496939	.1739042
__00006X	-.0058284	.0255576	-0.23	0.820	-.0559335	.0442767
__00006Y	-.0133605	.0198228	-0.67	0.500	-.0522227	.0255017
__00006Z	-.0015224	.0138109	-0.11	0.912	-.0285984	.0255536
__000070	.0235671	.0204858	1.15	0.250	-.0165948	.063729
__000071	-.027152	.0294347	-0.92	0.356	-.0848581	.030554
__000072	.0258673	.0074644	3.47	0.001	.0112334	.0405012
__000073	-.0120603	.0214469	-0.56	0.574	-.0541066	.029986
__000074	.0274528	.0296584	0.93	0.355	-.0306919	.0855975
__000075	.0003439	.0577372	0.01	0.995	-.1128487	.1135365

__000076	-.1021578	.0349086	-2.93	0.003	-.1705954	-.0337202
__000077	-.0408773	.0518526	-0.79	0.431	-.1425333	.0607787
__000078	.0303513	.0211844	1.43	0.152	-.0111803	.071883
__000079	-.0387628	.0220278	-1.76	0.079	-.0819478	.0044223
__00007A	.0016651	.010922	0.15	0.879	-.0197473	.0230776
__00007B	-.0164244	.0168392	-0.98	0.329	-.0494373	.0165885
__00007C	-.0468427	.0197903	-2.37	0.018	-.0856411	-.0080443
__00007D	-.0277055	.00534	-5.19	0.000	-.0381745	-.0172366
__00007E	-.0313793	.0141406	-2.22	0.027	-.0591017	-.0036568
__00007F	.0353335	.0200647	1.76	0.078	-.0040029	.0746699
__00007G	.0008952	.0360421	0.02	0.980	-.0697647	.071555
__00007H	.0369877	.0251104	1.47	0.141	-.0122408	.0862161
__00007I	-.0580765	.0381457	-1.52	0.128	-.1328605	.0167074
__00007J	-.035405	.0144783	-2.45	0.015	-.0637894	-.0070207
__00007K	.0124492	.0134002	0.93	0.353	-.0138216	.03872
__00007L	.0109754	.0080986	1.36	0.175	-.0049017	.0268524
__00007M	-.0228577	.0122021	-1.87	0.061	-.0467797	.0010643
__00007N	.0264033	.0142805	1.85	0.065	-.0015933	.0543999
__00007O	-.0533577	.0072023	-7.41	0.000	-.0674777	-.0392378
__00007P	-.0181736	.0212414	-0.86	0.392	-.0598169	.0234697
__00007Q	.0420639	.0292822	1.44	0.151	-.0153433	.0994711
__00007R	-.0449207	.0559626	-0.80	0.422	-.1546342	.0647928
__00007S	.0909393	.0357267	2.55	0.011	.0208978	.1609808
__00007T	-.0879365	.0552126	-1.59	0.111	-.1961796	.0203067
__00007U	-.0313519	.0205183	-1.53	0.127	-.0715777	.0088739
__00007V	.0269638	.0170275	1.58	0.113	-.0064182	.0603459
__00007W	.0046729	.0108156	0.43	0.666	-.0165309	.0258767
__00007X	-.022104	.0163234	-1.35	0.176	-.0541058	.0098977
__00007Y	.0089604	.0208969	0.43	0.668	-.0320075	.0499283
__000080	-.0384196	.0564503	-0.68	0.496	-.1490893	.07225
__000082	-.1299515	.0358288	-3.63	0.000	-.2001931	-.0597098
__000085	-.2064757	.2036348	-1.01	0.311	-.6056977	.1927462
__000087	.224298	.1378739	1.63	0.104	-.0460011	.4945971
__00008A	.7026172	.3373749	2.08	0.037	.0412005	1.364034
__00008C	-.229666	.2100972	-1.09	0.274	-.6415573	.1822254
__00008F	-.9132789	.7708282	-1.18	0.236	-2.424472	.5979144
__00008H	-.2374495	.3982853	-0.60	0.551	-1.01828	.5433807
__00008K	-.500225	.269964	-1.85	0.064	-1.029484	.0290341
__00008M	.2260132	.2530274	0.89	0.372	-.2700421	.7220684
__00008P	.3532354	.4755864	0.74	0.458	-.5791423	1.285613
__00008R	-.6946224	.3556451	-1.95	0.051	-1.391858	.0026127
__00008U	.311103	.1701294	1.83	0.068	-.0224322	.6446383
__00008W	.1062077	.132066	0.80	0.421	-.152705	.3651205
__00008Z	.2332741	.1222152	1.91	0.056	-.0063263	.4728745
__000091	-.2699383	.1038885	-2.60	0.009	-.4736096	-.066267
__000094	.2580975	.0992624	2.60	0.009	.0634957	.4526994
__000096	-.0496404	.0787286	-0.63	0.528	-.2039864	.1047055
__000099	-.2734954	.131055	-2.09	0.037	-.5304262	-.0165646
__00009B	-.1200271	.1078233	-1.11	0.266	-.3314125	.0913583
__00009E	-.1140745	.1665379	-0.68	0.493	-.4405687	.2124198
__00009G	-.2109157	.1238315	-1.70	0.089	-.4536849	.0318534
xbar	.5534121	.0662994	8.35	0.000	.4234334	.6833908
x2barlag	-.1443595	.2196268	-0.66	0.511	-.5749334	.2862144
xbar2lag	-.3667885	.4162412	-0.88	0.378	-1.182821	.449244
rootxbarlag	1.313706	1.013191	1.30	0.195	-.6726354	3.300047
z1barlag	.2670312	.4210897	0.63	0.526	-.5585069	1.092569
z2barlag	.4181552	.6421627	0.65	0.515	-.840792	1.677102
z3barlag	-.4913618	.2276912	-2.16	0.031	-.9377459	-.0449778
z4barlag	.0596818	.1380641	0.43	0.666	-.21099	.3303537
z5barlag	-.2463329	.1010338	-2.44	0.015	-.4444076	-.0482582
z6barlag	.4537685	.1856684	2.44	0.015	.0897692	.8177678
z7barlag	.3220941	.2106281	1.53	0.126	-.0908381	.7350264
_cons	-.860962	.7935572	-1.08	0.278	-2.416715	.6947911

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	128,640	.4432915	.1215803	.0980408	1.14881

Linear regression

Number of obs	**128,640**
F(129, 4598)	**49.44**
Prob > F	**0.0000**
R-squared	**0.6347**
Root MSE	**.02135**

(Std. err. adjusted for **4,599** clusters in **group_round**)

qbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**1.360707**	**.2691102**	**5.06**	**0.000**	**.8331222**	**1.888292**
__00009L	**-2.362566**	**.266613**	**-8.86**	**0.000**	**-2.885255**	**-1.839877**
__00009O	**.7058641**	**.1914231**	**3.69**	**0.000**	**.330583**	**1.081145**
xi	**.0009618**	**.0020043**	**0.48**	**0.631**	**-.0029676**	**.0048911**
__000002	**-.0003966**	**.0006949**	**-0.57**	**0.568**	**-.0017588**	**.0009657**
__00000E	**.0112668**	**.0130327**	**0.86**	**0.387**	**-.0142836**	**.0368172**
__00000P	**-.0115041**	**.0092264**	**-1.25**	**0.213**	**-.0295923**	**.006584**
__000017	**.0143324**	**.0295681**	**0.48**	**0.628**	**-.0436353**	**.0723**
__00001I	**.0107806**	**.0194519**	**0.55**	**0.579**	**-.0273545**	**.0489157**
__000020	**.0147858**	**.0075532**	**1.96**	**0.050**	**-.0000221**	**.0295938**
__00002B	**.0002663**	**.0047367**	**0.06**	**0.955**	**-.0090199**	**.0095526**
__00002T	**-.0252263**	**.0123689**	**-2.04**	**0.041**	**-.0494752**	**-.0009774**
__000034	**.0045848**	**.0084408**	**0.54**	**0.587**	**-.0119632**	**.0211327**
__00003M	**-.0008958**	**.010177**	**-0.09**	**0.930**	**-.0208476**	**.019056**
__00003X	**.0035043**	**.0065906**	**0.53**	**0.595**	**-.0094165**	**.0164251**
__00004F	**-.0141667**	**.0076803**	**-1.84**	**0.065**	**-.0292239**	**.0008905**
__00004Q	**.0072009**	**.00511**	**1.41**	**0.159**	**-.0028171**	**.017219**
__000058	**-.0119357**	**.011006**	**-1.08**	**0.278**	**-.0335127**	**.0096413**
__00005J	**.0137431**	**.0067375**	**2.04**	**0.041**	**.0005343**	**.0269518**
__00005U	**-.0053057**	**.0038382**	**-1.38**	**0.167**	**-.0128303**	**.002219**
__00005V	**.000759**	**.0136825**	**0.06**	**0.956**	**-.0260653**	**.0275833**
__00005W	**-.00014**	**.0176981**	**-0.01**	**0.994**	**-.0348367**	**.0345567**
__00005X	**.0095413**	**.0299207**	**0.32**	**0.750**	**-.0491177**	**.0682003**
__00005Y	**-.0214169**	**.0197547**	**-1.08**	**0.278**	**-.0601456**	**.0173117**
__00005Z	**.0490611**	**.0368565**	**1.33**	**0.183**	**-.0231953**	**.1213174**
__000060	**.0017558**	**.012201**	**0.14**	**0.886**	**-.022164**	**.0256757**
__000061	**-.0056987**	**.0070051**	**-0.81**	**0.416**	**-.019432**	**.0080346**
__000062	**.0042473**	**.0066303**	**0.64**	**0.522**	**-.0087512**	**.0172459**
__000063	**-.0144707**	**.0084872**	**-1.71**	**0.088**	**-.0311097**	**.0021683**
__000064	**-.0222872**	**.0092796**	**-2.40**	**0.016**	**-.0404796**	**-.0040947**
__000065	**.0196756**	**.0065186**	**3.02**	**0.003**	**.006896**	**.0324551**
__000066	**.023523**	**.022514**	**1.04**	**0.296**	**-.0206153**	**.0676612**
__000067	**-.0441622**	**.0299993**	**-1.47**	**0.141**	**-.1029752**	**.0146508**
__000068	**.0476144**	**.0581371**	**0.82**	**0.413**	**-.0663621**	**.161591**
__000069	**-.0468602**	**.0341963**	**-1.37**	**0.171**	**-.1139014**	**.020181**
__00006A	**-.1018251**	**.0623046**	**-1.63**	**0.102**	**-.2239721**	**.0203219**
__00006B	**-.0172525**	**.0237717**	**-0.73**	**0.468**	**-.0638566**	**.0293515**
__00006C	**.0111881**	**.0196918**	**0.57**	**0.570**	**-.0274173**	**.0497934**
__00006D	**-.0180913**	**.0122443**	**-1.48**	**0.140**	**-.0420959**	**.0059134**
__00006E	**-.0103657**	**.0160071**	**-0.65**	**0.517**	**-.0417472**	**.0210158**
__00006F	**.0179496**	**.0212349**	**0.85**	**0.398**	**-.0236811**	**.0595803**
__00006G	**.0011893**	**.0017663**	**0.67**	**0.501**	**-.0022735**	**.0046521**
__00006H	**.0039413**	**.0057115**	**0.69**	**0.490**	**-.0072559**	**.0151386**
__00006I	**-.0008365**	**.0082366**	**-0.10**	**0.919**	**-.0169843**	**.0153113**
__00006J	**-.0204385**	**.0134255**	**-1.52**	**0.128**	**-.0467588**	**.0058819**
__00006K	**.0121784**	**.0093513**	**1.30**	**0.193**	**-.0061547**	**.0305116**
__00006L	**-.0102922**	**.0150136**	**-0.69**	**0.493**	**-.039726**	**.0191417**
__00006M	**-.0030276**	**.0059514**	**-0.51**	**0.611**	**-.0146951**	**.0086399**
__00006N	**-.0025608**	**.0042377**	**-0.60**	**0.546**	**-.0108688**	**.0057472**
__00006O	**-.0026843**	**.0032851**	**-0.82**	**0.414**	**-.0091246**	**.0037561**
__00006P	**.0012805**	**.0038367**	**0.33**	**0.739**	**-.0062413**	**.0088024**
__00006Q	**.0000288**	**.0050225**	**0.01**	**0.995**	**-.0098177**	**.0098753**
__00006R	**-.0053853**	**.002394**	**-2.25**	**0.025**	**-.0100787**	**-.0006919**
__00006S	**-.0128745**	**.0100673**	**-1.28**	**0.201**	**-.0326113**	**.0068622**
__00006T	**.0018107**	**.0132149**	**0.14**	**0.891**	**-.0240968**	**.0277182**
__00006U	**.0393519**	**.0265713**	**1.48**	**0.139**	**-.0127406**	**.0914444**
__00006V	**-.0020913**	**.0115391**	**-0.18**	**0.856**	**-.0247134**	**.0205309**
__00006W	**.0013938**	**.0285665**	**0.05**	**0.961**	**-.0546102**	**.0573977**
__00006X	**-.0056812**	**.0104914**	**-0.54**	**0.588**	**-.0262494**	**.014887**
__00006Y	**.0123697**	**.0062343**	**1.98**	**0.047**	**.0001474**	**.024592**
__00006Z	**.0101166**	**.005572**	**1.82**	**0.069**	**-.0008071**	**.0210404**

__000070	.0007076	.0065555	0.11	0.914	-.0121442	.0135595
__000071	.0039815	.0083727	0.48	0.634	-.012433	.020396
__000072	.0001899	.0022809	0.08	0.934	-.0042818	.0046615
__000073	-.0175826	.0077637	-2.26	0.024	-.0328032	-.002362
__000074	.0256209	.0100977	2.54	0.011	.0058245	.0454172
__000075	-.0235372	.0185901	-1.27	0.206	-.0599828	.0129084
__000076	-.0000533	.0121365	-0.00	0.996	-.0238467	.02374
__000077	.0264157	.0196641	1.34	0.179	-.0121353	.0649668
__000078	.0031299	.007631	0.41	0.682	-.0118305	.0180903
__000079	.0054913	.0063349	0.87	0.386	-.006928	.0179107
__00007A	-.0053633	.004275	-1.25	0.210	-.0137444	.0030177
__00007B	-.0005283	.0053796	-0.10	0.922	-.0110749	.0100183
__00007C	.0073089	.0068724	1.06	0.288	-.0061644	.0207821
__00007D	-.0050338	.0018443	-2.73	0.006	-.0086495	-.0014181
__00007E	-.0029036	.0052665	-0.55	0.581	-.0132285	.0074212
__00007F	-.0014639	.0072855	-0.20	0.841	-.015747	.0128192
__00007G	.0124304	.0135379	0.92	0.359	-.0141104	.0389712
__00007H	-.0119058	.0099379	-1.20	0.231	-.0313889	.0075773
__00007I	.0049898	.0157878	0.32	0.752	-.0259618	.0359415
__00007J	.0069318	.005803	1.19	0.232	-.0044449	.0183085
__00007K	-.0056026	.004831	-1.16	0.246	-.0150737	.0038685
__00007L	-.0007704	.0040615	-0.19	0.850	-.0087329	.0071921
__00007M	.0032851	.0042047	0.78	0.435	-.0049582	.0115284
__00007N	.0026542	.00559	0.47	0.635	-.0083048	.0136132
__00007O	-.0005014	.0027517	-0.18	0.855	-.0058961	.0048933
__00007P	.007879	.0082074	0.96	0.337	-.0082115	.0239695
__00007Q	-.0137104	.0100206	-1.37	0.171	-.0333556	.0059349
__00007R	.0022406	.0285859	0.08	0.938	-.0538014	.0582827
__00007S	-.0273539	.0136528	-2.00	0.045	-.0541199	-.0005878
__00007T	.003742	.0258974	0.14	0.885	-.0470293	.0545134
__00007U	.0164178	.0090471	1.81	0.070	-.0013188	.0341544
__00007V	-.0067466	.0048204	-1.40	0.162	-.0161969	.0027037
__00007W	.0008315	.0051279	0.16	0.871	-.0092216	.0108847
__00007X	-.0056354	.0057144	-0.99	0.324	-.0168384	.0055676
__00007Y	-.0078539	.0071697	-1.10	0.273	-.02191	.0062022
__000080	.0563403	.0210907	2.67	0.008	.0149925	.0976881
__000082	-.0315283	.0161788	-1.95	0.051	-.0632465	.0001898
__000085	.0110789	.0744382	0.15	0.882	-.1348558	.1570136
__000087	-.043608	.0504579	-0.86	0.387	-.1425297	.0553137
__00008A	-.0766457	.1240739	-0.62	0.537	-.3198901	.1665987
__00008C	-.0740808	.0851554	-0.87	0.384	-.2410264	.0928647
__00008F	.1602937	.292201	0.55	0.583	-.4125606	.7331479
__00008H	.5706521	.2135358	2.67	0.008	.1520194	.9892848
__00008K	.6555445	.0943125	6.95	0.000	.4706467	.8404423
__00008M	-.2177851	.102889	-2.12	0.034	-.419497	-.0160732
__00008P	-.8046061	.1669684	-4.82	0.000	-1.131944	-.4772679
__00008R	.1038157	.1675129	0.62	0.535	-.2245901	.4322215
__00008U	-.162378	.0665528	-2.44	0.015	-.2928534	-.0319026
__00008W	-.0461807	.059143	-0.78	0.435	-.1621294	.069768
__00008Z	.0148629	.0452332	0.33	0.742	-.073816	.1035417
__000091	.0627298	.0301977	2.08	0.038	.0035277	.1219318
__000094	-.0096521	.0382038	-0.25	0.801	-.08455	.0652457
__000096	.1001981	.0356779	2.81	0.005	.0302522	.1701439
__000099	-.0035588	.0458554	-0.08	0.938	-.0934573	.0863398
__00009B	.0267518	.0406022	0.66	0.510	-.0528481	.1063517
__00009E	-.0027479	.0561281	-0.05	0.961	-.112786	.1072901
__00009G	.0268485	.0489753	0.55	0.584	-.0691666	.1228636
xbar	.0394624	.024255	1.63	0.104	-.0080889	.0870138
x2barlag	.035303	.0755558	0.47	0.640	-.1128226	.1834286
xbar2lag	.1798176	.1416568	1.27	0.204	-.0978977	.457533
rootxbarlag	-.8498505	.3313012	-2.57	0.010	-1.49936	-.2003411
z1barlag	-.3897896	.1476607	-2.64	0.008	-.6792755	-.1003037
z2barlag	.7075497	.2355803	3.00	0.003	.2456991	1.1694
z3barlag	.2166604	.0940368	2.30	0.021	.0323031	.4010176
z4barlag	-.087425	.0483046	-1.81	0.070	-.1821251	.0072752
z5barlag	-.0915324	.0414099	-2.21	0.027	-.1727157	-.0103492
z6barlag	-.0116257	.0674729	-0.17	0.863	-.1439049	.1206535
z7barlag	-.0214518	.075838	-0.28	0.777	-.1701307	.127227
_cons	.3787886	.2515386	1.51	0.132	-.1143478	.871925

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A2	**128,640**	**.0912606**	**.0281262**	**.0216847**	**.2422223**

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"
957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9
960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
964 }
965
966 local r_RE      "`r_RE' `qhat_hats'"
967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }
978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
```

```
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS	Number of obs	=	128,640
				F(13, 128626)	=	57321.16
Model	5933.80789	13	456.446761	Prob > F	=	0.0000
Residual	1024.24524	128,626	.007962972	R-squared	=	0.8528
				Adj R-squared	=	0.8528
Total	6958.05313	128,639	.054089764	Root MSE	=	.08924

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	.0180296	.0009529	18.92	0.000	.0161619	.0198974
ybarx	.0030935	.0010502	2.95	0.003	.0010352	.0051518
xi2	-.0393058	.0005754	-68.31	0.000	-.0404336	-.0381779
ybar	-.1946598	.0026658	-73.02	0.000	-.1998848	-.1894348
xi	.4196294	.0016617	252.52	0.000	.4163724	.4228864
ybar_q1	.4287804	.0032091	133.61	0.000	.4224907	.4350702
pzi1	.22573	.001319	171.13	0.000	.2231448	.2283153
pzi2	.0193004	.0021507	8.97	0.000	.0150851	.0235157
pzi3	-.00296	.0006651	-4.45	0.000	-.0042635	-.0016564
pzi4	.0106379	.0006603	16.11	0.000	.0093438	.011932
pzi5	-.0145877	.0007109	-20.52	0.000	-.0159811	-.0131944
pzi6	.0026414	.000587	4.50	0.000	.0014908	.003792
pzi7	-.0039312	.0007606	-5.17	0.000	-.005422	-.0024403
_cons	-.0249549	.0016336	-15.28	0.000	-.0281566	-.0217531

```
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = -_b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                local coef = _b[pzi`j']
    3.                local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
```

```
1025}

1026
1027if `J'==3 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .05507561
  Iteration 1:  GMM criterion Q(b) =  .00136896
  Iteration 2:  GMM criterion Q(b) =  .00023032
  Iteration 3:  GMM criterion Q(b) =  .00012766

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01010302
  Iteration 1:  GMM criterion Q(b) =  .00806673
  Iteration 2:  GMM criterion Q(b) =  .00788191

  GMM estimation

  Number of parameters =  20
  Number of moments    =  28
  Initial weight matrix: Unadjusted                 Number of obs   =    128,640
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
A11          |
         one |   .0538881   .0155691     3.46   0.001     .0233733    .0844029
-------------+----------------------------------------------------------------
C1           |
         zi1 |   .4355214   .0556558     7.83   0.000      .326438    .5446048
         zi2 |  -.2436268   .0858061    -2.84   0.005    -.4118037   -.0754498
         zi3 |  -.0278793   .0219094    -1.27   0.203    -.0708209    .0150623
         zi4 |   .1367652   .0425061     3.22   0.001     .0534549    .2200756
         zi5 |    .055434   .0330749     1.68   0.094    -.0093916    .1202596
         zi6 |   .0463404   .0235613     1.97   0.049      .000161    .0925198
         zi7 |   .0517905   .0271854     1.91   0.057     -.001492     .105073
-------------+----------------------------------------------------------------
C2           |
         zi1 |   .1423762   .0972178     1.46   0.143    -.0481671    .3329195
         zi2 |  -.4053659    .142629    -2.84   0.004    -.6849136   -.1258182
         zi3 |  -.0464188   .0353405    -1.31   0.189    -.1156849    .0228474
         zi4 |   .2194292    .069774     3.14   0.002     .0826747    .3561837
         zi5 |   .1151847   .0477909     2.41   0.016     .0215163    .2088531
         zi6 |   .0766288   .0376327     2.04   0.042      .00287     .1503877
```

```
------------------------------------------------------------------------------
          zi7 |   .1136157   .0458213     2.48   0.013     .0238076    .2034237
------------------------------------------------------------------------------
          /d1 |  -.0677776     .00551   -12.30   0.000     -.078577   -.0569783
          /b1 |   .4507909   .0185755    24.27   0.000     .4143835    .4871983
       /AVA11 |   4.462369   1.613224     2.77   0.006     1.300507     7.62423
       /AVA12 |  -3.719511    1.47569    -2.52   0.012     -6.61181    -.827212
       /AVA22 |   2.964857   1.335161     2.22   0.026       .34799    5.581723
------------------------------------------------------------------------------

Instructments for equation eq1_RE: __00009J __00009L __00009O xi __000002 __00000E
     __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M
     __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5
     __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons

1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
```

```
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compfsuXsegXreligX
      > sched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_c
      > lus_Dist_dropPre_RE.tex not found)
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compf
  > suXsegXreligXsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_ur
  > bOnly_clus_Dist_dropPre_RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  24757

1077unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640

1078estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4599

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  5.3831268

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compfsuXsegX
  > religXsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_c
  > lus_Dist_dropPre_RE.ster not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compfsuXsegXreligXs
  > ched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dis
  > t_dropPre_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
```

```
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
```

```
                                                    gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
                                                       local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
                                            }
                                    }
                            }
                    }
            }
        if `Tindivp1'<=`T' {
                forval t=`Tindivp1'/`T' {
                            tempvar zi`t'xdiff
                            g double `zi`t'xdiff'=zi`t'*`ximxk'
                            local zg "`zg' zi`t'"
                            forval j=1/`J' {
                                    tempvar zi`t'p`j'xdiff
                                    g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
                                    local pzg "`pzg' `zi`t'p`j''"
                                    local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
                            }
                    }
        }

        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
        // denoting zg as group-zs
        //  zg,pzgx,rprp,rprpx
        local rootprootp ""
        local rootprootpxdiff ""

        forval j=1/`J' {
                    forval s=`j'/`J' {
                            tempvar xdiffrp`j'rp`s'
                            g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')

                         if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "

                         local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
                    }
            }

        local pqhatxdiff ""
        forval j=1/`J' {
                    forval s=1/`J' {
                            foreach name of global Alist {
                                    tempvar   p`s'qikhat`j'xdiff`name'
                                    g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'

                                 local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
                            }
                    }
            }

        // set instruments
        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
}
```

```
1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                tempvar ximxk`var'
    3.                g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                        local `++t'
    3.                        local p_nice = subinstr("`p'",":_cons","",.)
    4.                        local est = coefs[1,`t']
    5.                        if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                        if `j'>`i' {
   11.                                tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                        }
   15.                }
   16.         }
```

```
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >     `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1     `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
```

```
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00098958
  Iteration 1:  GMM criterion Q(b) =  .00047174
  Iteration 2:  GMM criterion Q(b) =  .00022374

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01155527
  Iteration 1:  GMM criterion Q(b) =  .01024522
  Iteration 2:  GMM criterion Q(b) =  .01012454
  Iteration 3:  GMM criterion Q(b) =  .01011089

  GMM estimation

  Number of parameters =  17
  Number of moments    =  63
  Initial weight matrix: Unadjusted                 Number of obs   =    128,640
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
C11          |
       _cons |   .5990252   .0265977    22.52   0.000     .5468946    .6511558
-------------+----------------------------------------------------------------
C12          |
       _cons |  -.0662563   .0450475    -1.47   0.141    -.1545478    .0220351
-------------+----------------------------------------------------------------
C13          |
       _cons |  -.0117946   .0110115    -1.07   0.284    -.0333767    .0097876
-------------+----------------------------------------------------------------
C14          |
       _cons |   .0041523    .011996     0.35   0.729    -.0193595    .0276641
-------------+----------------------------------------------------------------
C15          |
       _cons |  -.0204903   .0192751    -1.06   0.288    -.0582689    .0172883
------------------------------------------------------------------------------
```

C16						
_cons	**-.0235301**	**.0134696**	**-1.75**	**0.081**	**-.0499301**	**.0028699**
C17						
_cons	**-.0210753**	**.0154346**	**-1.37**	**0.172**	**-.0513265**	**.0091758**
C21						
_cons	**.4318972**	**.0373995**	**11.55**	**0.000**	**.3585954**	**.5051989**
C22						
_cons	**-.1272414**	**.0647853**	**-1.96**	**0.050**	**-.2542183**	**-.0002645**
C23						
_cons	**-.0157598**	**.0150973**	**-1.04**	**0.297**	**-.0453499**	**.0138303**
C24						
_cons	**-.0057307**	**.0204521**	**-0.28**	**0.779**	**-.0458161**	**.0343547**
C25						
_cons	**-.0025809**	**.0247127**	**-0.10**	**0.917**	**-.0510169**	**.0458552**
C26						
_cons	**-.0353004**	**.0178605**	**-1.98**	**0.048**	**-.0703064**	**-.0002944**
C27						
_cons	**-.0211871**	**.0220233**	**-0.96**	**0.336**	**-.064352**	**.0219777**
d1						
_cons	**-.0920375**	**.0042705**	**-21.55**	**0.000**	**-.1004075**	**-.0836675**
b1						
_cons	**.4278921**	**.0135198**	**31.65**	**0.000**	**.4013938**	**.4543904**
A11						
one	**.5293325**	**.0900367**	**5.88**	**0.000**	**.3528638**	**.7058012**

Instruments for equation **eq1_FE: __000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7 __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1 __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY __0000D5 _cons**

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
```

(file **diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compfsuXsegXreligX > sched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_c > lus_Dist_dropPre_FE.tex** not found)
(output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compf > suXsegXreligXsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_ur > bOnly_clus_Dist_dropPre_FE.tex)

```
1242
1243        estat overid
```

Test of overidentifying restriction:

Hansen's J chi2(**46**) = **1300.67** (p = **0.0000**)

```
1244        di "$S_DATE $S_TIME"
   7 Dec 2023 20:34:57
1245        unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  24757
1248        unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640
1249        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4599
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  5.3831268
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compfsuXsegX
  > religXsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_c
  > lus_Dist_dropPre_FE.ster not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_compfsuXsegXreligXs
  > ched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dis
  > t_dropPre_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}
```

```
1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\rep_table3_36.sm
  > cl
    log type:  smcl
   closed on:   7 Dec 2023, 20:34:57
  ──────────────────────────────────────────────────────────────────────────────────────
```

A.4 Table 4

A.4.1 Columns 2 and 6

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table4_26.smcl
  log type:  smcl
 opened on:   7 Dec 2023, 20:16:51

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=2

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
50. *local group_def "dist"
51. *local group_def "fsuXseg"
52. local group_def "fsuXsegXreligXsched"

53.
54. // comparision group
55. local groupComp_def ""

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.          syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
70.          if "`rename'" == "" local rename = "`varlist'bar"
    3.
```

```
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations            1,118    ->   280
  Number of variables                   5    ->   7
  j variable (4 values)       expend_type    ->   (dropped)
  xij variables:
                         laspeyres_state    ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560    ->   280
  Number of variables                   5    ->   5
  j variable (2 values)       expend_type    ->   (dropped)
  xij variables:
                         laspeyres_state    ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                  Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations                 840   ->   280
  Number of variables                      5   ->   6
  j variable (3 values)          expend_type   ->   (dropped)
  xij variables:
                                laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,637 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
224         g land=exp(demog23)
225         replace z4=0 if land<=0.005
226         replace z4=ln(land+1) if land>0.005
227         g z5=demog24
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
```

```
242         g z3=married
243         g z4=.
  (86,380 missing values generated)
244         g land=exp(demog23)
  (12,776 missing values generated)
245         replace z4=0 if land<=0.005
  (29,502 real changes made)
246         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
247         g z5=demog24
  (29 missing values generated)
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }
297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }
316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }
333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
```

```
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.         drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                 replace `var' = `var'/hhsize
    3.         }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.         g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
```

```
382         g pq4=inec_norm
383 }

384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }
```

```
422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }

424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (82 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,576 real changes made, 82 to missing)
```

```
460 replace x2=x*x
  (73,576 real changes made, 82 to missing)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)

464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,576 real changes made)
```

```
478 replace x2=x2_temp
  (73,576 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (48,819 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)

488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi
```

```
507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_41e4_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk
```

```
539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,757 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
549 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var''" _n
    5.                 }
570
571                 file write sumstat "\bottomrule" _n
```

```
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
```

```
613 //global Alist "one"
614 global Alist "one schednhi schedhi nschednhi"

615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1

625 scalar estimate_FE=1

626
627 // estimation options
628 global w_initial "winit(unadjusted)"

629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

630 global wmatrix "wmatrix(robust)"

631 global trace_level ""

632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

633 global technique ""

634 global maxiter ""

635 global technique_RE ""

636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

637 global derivatives "quickderivatives"

638 scalar init_FE_from_RE=1

639 global clust_var "state_district_round"

640 local clus_name "_clus_Dist"

641
642
643
644
645
646
647
648
649 // xxzx to keep names same
650 local wmatrix_desc ""

651 local sectorName "_urbOnly"
```

```
652 local dropZdesc "_dropPre"

653 local desc_struc_v0 ""

654 local inst_desc "_exactInst"

655 local inst_rootp "_noPPXdInst"

656 local weight_desc ""

657 local RE_sq_inst_desc ""

658 local cnstname ""

659 local actual_pqhat_desc ""

660 local expUpname "_updateExp"

661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //              ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*****************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *****************************************************************************/
```

```
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T

694 local Tm1=`T'-1

695 local Tindiv : word count $zlist_indiv

696
697 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
706
707 if `J'==2 {
708          replace p1=P1
  (128,640 real changes made)
709          replace p2=P2
  (128,640 real changes made)
710          replace qi1=Qi1
  (128,632 real changes made)
711          replace qk1=Qk1
  (128,632 real changes made)
712          replace qikbar1=Qikbar1
  (128,640 real changes made)
713          replace qi2=Qi2
  (128,640 real changes made)
714          replace qk2=Qk2
  (128,640 real changes made)
715          replace qikbar2=Qikbar2
  (128,640 real changes made)
716
717          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                   qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721          forv j = 1(1)3 {
    2.                   replace p`j'=P`j'
    3.                   replace qi`j'=Qi`j'
    4.                   replace qk`j'=Qk`j'
    5.                   replace qikbar`j'=Qikbar`j'
    6.          }
```

```
722
723         foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                 qui replace `var'=.
    3.         }
724 }

725
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.         global zilist "$zilist zi`t'"
    3.         global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.         local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.         local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.         local pCi`j' "( 0 "
    3.         local pCk`j' "( 0 "
    4.         local pCg`j' "( 0 "
    5.         forval t=1/`Tindiv' {
    6.                 local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                 local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.         }
    9.         forval t=`Tindivp1'/`T' {
   10.                 local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.         }
   12.         local pCi`j' "`pCi`j'' )"
   13.         local pCk`j' "`pCk`j'' )"
   14.         local pCg`j' "`pCg`j'' )"
   15. }
```

```
751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
773         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.          local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.          if `jplus1'<=`J' {
    5.                  forval k=`jplus1'/`J' {
    6.                          local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                  }
    8.          }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                  if noMeasError==1                               local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
```

```
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                              l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7 zi8 zi9 zi10}
  > ) + (p2*{C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7 zi8 zi9 zi10}) ) - ( 0 ) )*( ({A11:}*p1*qik
  > bar1) + ({A11:}*p2*qikbar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi -
  > ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi
  > - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2)
  > ))*{b1} + ({A11:}*p1*qikbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA1
  > 2}*p2 + p2*{AVA22}*p2 )*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 + p1*{C18}*zi8 + p1*{C19}
  > *zi9 + p1*{C110}*zi10 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi
  > 4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 + p2*{C28}*zi8 + p2*{C29}*zi9 + p2*{C
  > 210}*zi10 ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*z
  > k4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 + p1*{C18}*zk8 + p1*{C19}*zk9 + p1*{
  > C110}*zk10 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25
  > }*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 + p2*{C28}*zk8 + p2*{C29}*zk9 + p2*{C210}*zk10 )
  >  ))^2) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi
  > 2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 + p1*{C
  > 18}*zi8 + p1*{C19}*zi9 + p1*{C110}*zi10 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}
  > *zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 + p2*{C28}*zi8 + p2
  > *{C29}*zi9 + p2*{C210}*zi10 ) )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*
  > zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 + p1*{C18}*zk8 + p1*
  > {C19}*zk9 + p1*{C110}*zk10 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 + p2*{C2
  > 4}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 + p2*{C28}*zk8 + p2*{C29}*zk9 +
  > p2*{C210}*zk10 ) )))*({b1} - 2*(( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( (
  > 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{
  > C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7
  > + p1*{C18}*zi8 + p1*{C19}*zi9 + p1*{C110}*zi10 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p
  > 1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 + p1*{C18}*z
  > k8 + p1*{C19}*zk9 + p1*{C110}*zk10 ))         ))
```

```
836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one schednhi schedhi nschednhi}*p1*qk1) + ({A11:}*p2*qk2
  > ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3
  >  zi4 zi5 zi6 zi7 zi8 zi9 zi10}) + (p2*{C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7 zi8 zi9 zi10}
  > ) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) )   + (xi - ( (p1*{C1:}) +
  >  (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )) * {d1}/exp( {b1
  > }*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )-( ({A11:}*p1*q
  > ikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qikbar1) + (p1*{C1:}) + ( 0 ) - (
  > 0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d1}/exp( {b1}*ln(p1/p2) + ln(
  > p2) ) ) ) )

873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 + p1*{C18}*zi8 + p1*{C19}
  > *zi9 + p1*{C110}*zi10 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi
  > 4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 + p2*{C28}*zi8 + p2*{C29}*zi9 + p2*{C
  > 210}*zi10 ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*z
  > k4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 + p1*{C18}*zk8 + p1*{C19}*zk9 + p1*{
  > C110}*zk10 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25
  > }*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 + p2*{C28}*zk8 + p2*{C29}*zk9 + p2*{C210}*zk10 )
  >  ))^2) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi
  > 2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 + p1*{C
  > 18}*zi8 + p1*{C19}*zi9 + p1*{C110}*zi10 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}
  > *zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 + p2*{C28}*zi8 + p2
  > *{C29}*zi9 + p2*{C210}*zi10 ) )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*
  > zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 + p1*{C18}*zk8 + p1*
  > {C19}*zk9 + p1*{C110}*zk10 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 + p2*{C2
  > 4}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 + p2*{C28}*zk8 + p2*{C29}*zk9 +
  > p2*{C210}*zk10 ) )))*({b1} - 2*(( ({A11:one schednhi schedhi nschednhi}*p1*qikbar1)
  > + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1/p2) + ln(p
  > 2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{C15}*zi5
  > + p1*{C16}*zi6 + p1*{C17}*zi7 + p1*{C18}*zi8 + p1*{C19}*zi9 + p1*{C110}*zi10 ) - ( p
  > 1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*z
  > k6 + p1*{C17}*zk7 + p1*{C18}*zk8 + p1*{C19}*zk9 + p1*{C110}*zk10 ))         ))

874
875 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi
```

```
880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi

882 g double `ximxk'=xi-xk

883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.         tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.         g double `xizi`t''=xi*zi`t'
    4.         g double `xkzk`t''=xk*zk`t'
    5.         g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.         g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.         g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.         g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.         g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.         forval j=1/`J' {
   11.                 tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                 g double `zi`t'p`j''= zi`t'*p`j'
   13.                 g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                 g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                 g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                 forv s=1/`T' {
   17.                         tempvar zi`t'zi`s'p`j'
   18.                         g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                 }
   20.         }
   21. }

891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                 local rg "`rg' z`t'barlag"
    3. }
```

```
903
904 local zrg ""

905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var'"
    6.          }
    7. }

906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var'"
    5.
911          forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var'"
   10.                  local xprg "`xprg' `xp`j'`var'"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
```

```
923 local zinsts ""

924 local xzinsts ""

925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }

929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.          forval k=`j'/`J' {
    3.                  tempvar p`j'p`k'
    4.                  g `p`j'p`k''=p`j'*p`k'
    5.                  local roundinsts "`roundinsts' `p`j'p`k''"
    6.          }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 25

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""
```

```
949 local pqhat_sq ""

950 forval j=1/`J' {
    2.         if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.         if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }
```

Linear regression

Number of obs	**128,640**
F(206, 4598)	.
Prob > F	.
R-squared	**0.2662**
Root MSE	**.12742**

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__0000FS	**-9.815662**	**7.742679**	**-1.27**	**0.205**	**-24.99503**	**5.363705**
__0000FU	**-4.79649**	**1.429275**	**-3.36**	**0.001**	**-7.598555**	**-1.994425**
__0000FX	**2.529539**	**4.130309**	**0.61**	**0.540**	**-5.56785**	**10.62693**
xi	**.0947758**	**.0075449**	**12.56**	**0.000**	**.0799842**	**.1095674**
__000002	**-.0130556**	**.0024379**	**-5.36**	**0.000**	**-.0178351**	**-.0082762**
__00000E	**-.1592736**	**.0877181**	**-1.82**	**0.069**	**-.3312433**	**.012696**
__00000S	**-.0246076**	**.049596**	**-0.50**	**0.620**	**-.1218396**	**.0726244**
__00001D	**.1719511**	**.1673053**	**1.03**	**0.304**	**-.1560477**	**.4999499**
__00001R	**.0264758**	**.1073189**	**0.25**	**0.805**	**-.1839207**	**.2368723**
__00002C	**.0174048**	**.0438072**	**0.40**	**0.691**	**-.0684783**	**.1032878**
__00002Q	**-.0479958**	**.0235142**	**-2.04**	**0.041**	**-.0940949**	**-.0018967**
__00003B	**.0188792**	**.0607705**	**0.31**	**0.756**	**-.1002601**	**.1380185**
__00003P	**-.005076**	**.0449696**	**-0.11**	**0.910**	**-.093238**	**.083086**
__00004A	**-.0476031**	**.0586106**	**-0.81**	**0.417**	**-.162508**	**.0673017**
__00004O	**.0020287**	**.0377763**	**0.05**	**0.957**	**-.072031**	**.0760885**
__000059	**-.0099762**	**.0418305**	**-0.24**	**0.812**	**-.091984**	**.0720317**
__00005N	**.0139468**	**.0246468**	**0.57**	**0.572**	**-.0343727**	**.0622663**
__000068	**-.0036266**	**.0580204**	**-0.06**	**0.950**	**-.1173744**	**.1101212**
__00006M	**.0085709**	**.0330713**	**0.26**	**0.796**	**-.0562646**	**.0734064**
__000077	**.3765503**	**.2547154**	**1.48**	**0.139**	**-.1228141**	**.8759147**
__00007L	**.2388798**	**.1960687**	**1.22**	**0.223**	**-.145509**	**.6232686**
__000086	**-.239469**	**.1132155**	**-2.12**	**0.034**	**-.4614258**	**-.0175123**
__00008K	**.028478**	**.0755265**	**0.38**	**0.706**	**-.1195903**	**.1765463**
__000095	**-.2360092**	**.0890015**	**-2.65**	**0.008**	**-.4104948**	**-.0615236**
__00009J	**.0424865**	**.0530434**	**0.80**	**0.423**	**-.061504**	**.146477**
__00009X	**.2070713**	**.3986368**	**0.52**	**0.603**	**-.5744482**	**.9885908**
__00009Y	**.0075413**	**.0700982**	**0.11**	**0.914**	**-.1298849**	**.1449675**
__00009Z	**-.1060011**	**.163268**	**-0.65**	**0.516**	**-.4260848**	**.2140826**
__0000A0	**-.0903784**	**.412816**	**-0.22**	**0.827**	**-.8996959**	**.7189391**
__0000A1	**-.0487361**	**.0967323**	**-0.50**	**0.614**	**-.2383779**	**.1409057**
__0000A2	**.3324296**	**.1636032**	**2.03**	**0.042**	**.0116888**	**.6531703**
__0000A3	**.0016657**	**.060493**	**0.03**	**0.978**	**-.1169297**	**.1202612**
__0000A4	**-.0379789**	**.0508057**	**-0.75**	**0.455**	**-.1375825**	**.0616246**
__0000A5	**.0011914**	**.0323205**	**0.04**	**0.971**	**-.0621723**	**.0645551**
__0000A6	**.0156821**	**.0453941**	**0.35**	**0.730**	**-.0733122**	**.1046764**

0000A7	-.0226468	.0575546	-0.39	0.694	-.1354814	.0901879
0000A8	.1113188	.0547286	2.03	0.042	.0040244	.2186132
0000A9	.0491149	.049525	0.99	0.321	-.0479778	.1462077
0000AA	.0348594	.0472341	0.74	0.461	-.057742	.1274608
0000AB	.2343404	.8414477	0.28	0.781	-1.415301	1.883982
0000AC	.2227567	.1425105	1.56	0.118	-.0566323	.5021456
0000AD	-.3854275	.3322578	-1.16	0.246	-1.036812	.2659573
0000AE	.0540862	.8184095	0.07	0.947	-1.550389	1.658562
0000AF	-.009387	.1803576	-0.05	0.958	-.3629745	.3442005
0000AG	-.6819498	.3268569	-2.09	0.037	-1.322746	-.0411534
0000AH	-.1362065	.1215638	-1.12	0.263	-.3745299	.1021168
0000AI	-.127765	.0924011	-1.38	0.167	-.3089155	.0533856
0000AJ	.1118528	.0618714	1.81	0.071	-.0094448	.2331504
0000AK	-.1425448	.0910531	-1.57	0.118	-.3210527	.0359631
0000AL	.0121579	.1126909	0.11	0.914	-.2087704	.2330862
0000AM	-.0151194	.1075441	-0.14	0.888	-.2259575	.1957188
0000AN	-.0042337	.0907533	-0.05	0.963	-.1821538	.1736864
0000AO	.0974131	.0872823	1.12	0.264	-.0737021	.2685283
0000AP	-.1191066	.1744672	-0.68	0.495	-.4611461	.222933
0000AQ	.0427475	.0319568	1.34	0.181	-.0199032	.1053982
0000AR	-.0424915	.0696272	-0.61	0.542	-.1789943	.0940113
0000AS	.1427259	.1964586	0.73	0.468	-.2424274	.5278791
0000AT	.0575296	.0485933	1.18	0.237	-.0377365	.1527957
0000AU	-.0848209	.0841077	-1.01	0.313	-.2497125	.0800706
0000AV	-.0200974	.0303908	-0.66	0.508	-.079678	.0394831
0000AW	.0336474	.021447	1.57	0.117	-.008399	.0756939
0000AX	-.0190188	.014883	-1.28	0.201	-.0481967	.0101591
0000AY	-.0099923	.0232144	-0.43	0.667	-.0555037	.0355192
0000AZ	.012565	.0305265	0.41	0.681	-.0472815	.0724115
0000B0	.0157148	.0267752	0.59	0.557	-.0367774	.0682071
0000B1	.027874	.0252232	1.11	0.269	-.0215756	.0773235
0000B2	.0208986	.0235847	0.89	0.376	-.0253388	.067136
0000B3	-.6053945	.4537148	-1.33	0.182	-1.494893	.2841043
0000B4	-.052948	.05259	-1.01	0.314	-.1560497	.0501538
0000B5	.3080303	.1812214	1.70	0.089	-.0472506	.6633112
0000B6	.382417	.3843788	0.99	0.320	-.37115	1.135984
0000B7	-.1523559	.0684856	-2.22	0.026	-.2866205	-.0180913
0000B8	.1115289	.1263865	0.88	0.378	-.1362493	.3593071
0000B9	.0293483	.0476462	0.62	0.538	-.064061	.1227577
0000BA	.0258941	.0300138	0.86	0.388	-.0329473	.0847355
0000BB	-.0627522	.0229008	-2.74	0.006	-.1076487	-.0178557
0000BC	-.0051258	.0348356	-0.15	0.883	-.0734202	.0631687
0000BD	-.0569208	.0424165	-1.34	0.180	-.1400775	.0262359
0000BE	-.0422312	.0380568	-1.11	0.267	-.1168408	.0323783
0000BF	.0282685	.040486	0.70	0.485	-.0511035	.1076405
0000BG	-.0539543	.0295129	-1.83	0.068	-.1118137	.0039052
0000BH	-.4776043	.2952365	-1.62	0.106	-1.05641	.101201
0000BI	-.0900144	.0451231	-1.99	0.046	-.1784773	-.0015515
0000BJ	.2700027	.1176345	2.30	0.022	.0393827	.5006227
0000BK	.4164107	.2813485	1.48	0.139	-.1351675	.9679889
0000BL	-.0808319	.061634	-1.31	0.190	-.2016642	.0400004
0000BM	.0253749	.1065362	0.24	0.812	-.1834872	.2342371
0000BN	.0211695	.0435972	0.49	0.627	-.064302	.106641
0000BO	-.0054623	.0311852	-0.18	0.861	-.0666002	.0556756
0000BP	.0238631	.0211413	1.13	0.259	-.0175839	.0653101
0000BQ	-.0342565	.0292588	-1.17	0.242	-.0916179	.0231048
0000BR	-.0928259	.0367006	-2.53	0.011	-.1647766	-.0208752
0000BS	.0327091	.0363703	0.90	0.369	-.0385941	.1040123
0000BT	-.0233676	.0304374	-0.77	0.443	-.0830395	.0363042
0000BU	-.0072375	.0284994	-0.25	0.800	-.0631101	.0486351
0000BV	.2078043	.2081852	1.00	0.318	-.2003387	.6159473
0000BW	.011381	.0347299	0.33	0.743	-.0567062	.0794682
0000BX	-.1105319	.0828417	-1.33	0.182	-.2729414	.0518776
0000BY	-.0268365	.2065763	-0.13	0.897	-.4318252	.3781522
0000BZ	.0045472	.0480595	0.09	0.925	-.0896725	.0987669
0000C0	-.112718	.0879309	-1.28	0.200	-.2851047	.0596687
0000C1	-.0170737	.0316955	-0.54	0.590	-.0792121	.0450647
0000C2	.0143872	.0242282	0.59	0.553	-.0331116	.061886
0000C3	.031857	.0154654	2.06	0.039	.0015375	.0621765
0000C4	-.0489343	.0248951	-1.97	0.049	-.0977406	-.000128
0000C5	-.0121797	.0293075	-0.42	0.678	-.0696364	.045277
0000C6	.0120496	.0290778	0.41	0.679	-.0449569	.0690562

__0000C7	-.0373079	.0253608	-1.47	0.141	-.0870272	.0124113
__0000C8	-.0026065	.0243575	-0.11	0.915	-.0503588	.0451459
__0000C9	.1999402	.3445774	0.58	0.562	-.4755969	.8754774
__0000CA	.1063852	.0476751	2.23	0.026	.0129192	.1998512
__0000CB	-.2019958	.1335705	-1.51	0.131	-.463858	.0598664
__0000CC	-.100789	.3126485	-0.32	0.747	-.7137301	.5121521
__0000CD	.0663495	.0632771	1.05	0.294	-.057704	.190403
__0000CE	-.2075081	.1174528	-1.77	0.077	-.437772	.0227557
__0000CF	.0217692	.0407216	0.53	0.593	-.0580648	.1016031
__0000CG	-.041722	.0283962	-1.47	0.142	-.0973921	.0139481
__0000CH	.0116164	.0199483	0.58	0.560	-.0274919	.0507246
__0000CI	-.0261593	.0313924	-0.83	0.405	-.0877034	.0353848
__0000CJ	-.0145325	.0399786	-0.36	0.716	-.0929098	.0638448
__0000CK	.0394405	.0372947	1.06	0.290	-.033675	.1125561
__0000CL	-.0117677	.0336055	-0.35	0.726	-.0776506	.0541151
__0000CM	.0296811	.0317134	0.94	0.349	-.0324924	.0918545
__0000CN	-.5549581	.8717007	-0.64	0.524	-2.26391	1.153994
__0000CO	-.0214202	.1735397	-0.12	0.902	-.3616414	.318801
__0000CP	.3142663	.3370595	0.93	0.351	-.346532	.9750646
__0000CQ	-.4605157	1.147586	-0.40	0.688	-2.710334	1.789303
__0000CR	.4861692	.2861445	1.70	0.089	-.0748114	1.04715
__0000CS	-.5576526	.6391533	-0.87	0.383	-1.8107	.6953946
__0000CT	-.0829785	.1424897	-0.58	0.560	-.3623267	.1963697
__0000CU	-.1320916	.0999419	-1.32	0.186	-.3280257	.0638424
__0000CV	-.0542094	.0741554	-0.73	0.465	-.1995896	.0911707
__0000CW	.1943826	.1482621	1.31	0.190	-.0962824	.4850475
__0000CX	.4120155	.1470941	2.80	0.005	.1236404	.7003906
__0000CY	-.1034482	.2114535	-0.49	0.625	-.5179986	.3111021
__0000CZ	.148254	.2517103	0.59	0.556	-.3452191	.6417271
__0000D0	-.1514543	.2241994	-0.68	0.499	-.5909927	.2880842
__0000D1	-.3803467	.5883421	-0.65	0.518	-1.53378	.7730862
__0000D2	-.0545911	.0876915	-0.62	0.534	-.2265085	.1173263
__0000D3	.0932009	.2130896	0.44	0.662	-.324557	.5109587
__0000D4	.6734483	.577258	1.17	0.243	-.4582545	1.805151
__0000D5	-.2509714	.1273719	-1.97	0.049	-.5006815	-.0012612
__0000D6	-.0156201	.2211072	-0.07	0.944	-.4490962	.4178561
__0000D7	.0218329	.0951083	0.23	0.818	-.1646249	.2082908
__0000D8	.0611618	.0712477	0.86	0.391	-.0785178	.2008415
__0000D9	.0560435	.0473596	1.18	0.237	-.036804	.148891
__0000DA	-.0824898	.0611981	-1.35	0.178	-.2024674	.0374878
__0000DB	-.04531	.0695739	-0.65	0.515	-.1817081	.0910882
__0000DC	.2096913	.1188496	1.76	0.078	-.0233108	.4426935
__0000DD	.114451	.0702873	1.63	0.104	-.0233458	.2522479
__0000DE	.1259804	.0696845	1.81	0.071	-.0106347	.2625955
__0000DF	-1.280804	.3535299	-3.62	0.000	-1.973892	-.5877155
__0000DG	-.1071987	.0676821	-1.58	0.113	-.2398882	.0254908
__0000DH	.4309744	.1371666	3.14	0.002	.162062	.6998867
__0000DI	1.357818	.3968501	3.42	0.001	.5798014	2.135835
__0000DJ	-.0445689	.1006424	-0.44	0.658	-.2418763	.1527385
__0000DK	-.0596885	.1686091	-0.35	0.723	-.3902432	.2708662
__0000DL	-.0439942	.0705169	-0.62	0.533	-.1822412	.0942528
__0000DM	-.1087913	.052566	-2.07	0.039	-.2118459	-.0057368
__0000DN	-.0140877	.0345535	-0.41	0.684	-.081829	.0536537
__0000DO	-.0362474	.0464089	-0.78	0.435	-.1272311	.0547364
__0000DP	.0392977	.053324	0.74	0.461	-.065243	.1438384
__0000DQ	-.0989194	.0729357	-1.36	0.175	-.2419083	.0440696
__0000DR	-.0477089	.0520902	-0.92	0.360	-.1498306	.0544128
__0000DS	.0093987	.0421009	0.22	0.823	-.0731393	.0919367
__0000DU	-30.36277	18.00903	-1.69	0.092	-65.66911	4.943574
__0000DW	5.028086	9.092214	0.55	0.580	-12.79702	22.85319
__0000DZ	-.0666001	.4088355	-0.16	0.871	-.868114	.7349138
__0000E1	.094879	.2785552	0.34	0.733	-.4512229	.640981
__0000E4	6.001129	3.535024	1.70	0.090	-.9292156	12.93147
__0000E6	-2.114007	1.732759	-1.22	0.223	-5.511047	1.283033
__0000E9	37.1058	22.29101	1.66	0.096	-6.595274	80.80687
__0000EB	-1.664794	11.64802	-0.14	0.886	-24.50051	21.17092
__0000EE	-.5918336	.5154358	-1.15	0.251	-1.602335	.4186679
__0000EG	-1.123771	.4706561	-2.39	0.017	-2.046483	-.2010588
__0000EJ	-1.450199	1.038182	-1.40	0.163	-3.485533	.5851358
__0000EL	-.9598537	.658161	-1.46	0.145	-2.250165	.3304578
__0000EO	.2206097	.357377	0.62	0.537	-.4800208	.9212402
__0000EQ	-.0194876	.2486138	-0.08	0.938	-.50689	.4679149

__0000ET	.763183	.3132455	2.44	0.015	.1490715	1.377295
__0000EV	-.1018437	.1731637	-0.59	0.556	-.4413278	.2376403
__0000EY	-.1169112	.2176042	-0.54	0.591	-.54352	.3096976
__0000F0	.4587998	.1962774	2.34	0.019	.074002	.8435977
__0000F3	-.0944695	.2678689	-0.35	0.724	-.6196211	.4306821
__0000F5	-.1238351	.197252	-0.63	0.530	-.5105437	.2628736
__0000F8	-.329857	.3362878	-0.98	0.327	-.9891426	.3294286
__0000FA	-.6594106	.2476904	-2.66	0.008	-1.145003	-.1738186
__0000FD	.1750725	.4627553	0.38	0.705	-.7321501	1.082295
__0000FF	-.646568	.3001428	-2.15	0.031	-1.234992	-.0581441
__0000FI	.1434763	.3386188	0.42	0.672	-.5203792	.8073317
__0000FK	-.413402	.2054632	-2.01	0.044	-.8162085	-.0105956
__0000FN	.390942	.3007359	1.30	0.194	-.1986447	.9805286
__0000FP	-.5862811	.1892664	-3.10	0.002	-.957334	-.2152282
xbarlag	24.09844	22.30035	1.08	0.280	-19.62096	67.81783
x2barlag	-.2078749	.4465753	-0.47	0.642	-1.083377	.667627
xbar2lag	-3.211018	4.450365	-0.72	0.471	-11.93587	5.513834
rootxbarlag	-34.47103	27.46501	-1.26	0.210	-88.31564	19.37357
z1barlag	1.876152	.7925939	2.37	0.018	.3222877	3.430017
z2barlag	2.772264	1.264143	2.19	0.028	.2939376	5.250591
z3barlag	-.1468984	.4807398	-0.31	0.760	-1.089379	.7955823
z4barlag	-.539387	.3206459	-1.68	0.093	-1.168007	.0892329
z5barlag	-.4695038	.3298849	-1.42	0.155	-1.116237	.177229
z6barlag	.33752	.3635442	0.93	0.353	-.3752011	1.050241
z7barlag	1.039115	.4129952	2.52	0.012	.2294462	1.848784
z8barlag	.5325707	.6235403	0.85	0.393	-.6898677	1.755009
z9barlag	.3377235	.4055929	0.83	0.405	-.4574333	1.13288
z10barlag	.152191	.3420614	0.44	0.656	-.5184135	.8227955
_cons	11.9035	9.410864	1.26	0.206	-6.546316	30.35331

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G2	128,640	.4424145	.0766732	.115227	.8069276

Linear regression

Number of obs	=	128,640
F(205, 4598)	=	.
Prob > F	=	.
R-squared	=	0.3423
Root MSE	=	.19727

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__0000FS	3.389337	9.850192	0.34	0.731	-15.92177	22.70044
__0000FU	-.6395598	2.382098	-0.27	0.788	-5.309615	4.030496
__0000FX	-2.274178	5.508235	-0.41	0.680	-13.07296	8.524608
xi	.1223223	.014062	8.70	0.000	.094754	.1498905
__000002	-.0036564	.0053265	-0.69	0.492	-.0140989	.0067862
__00000E	-.2690711	.1236573	-2.18	0.030	-.5114987	-.0266435
__00000S	-.102828	.0757181	-1.36	0.175	-.2512718	.0456158
__00001D	.5740899	.2698126	2.13	0.033	.0451276	1.103052
__00001R	.3129436	.1847335	1.69	0.090	-.0492228	.67511
__00002C	.0734819	.0682419	1.08	0.282	-.060305	.2072688
__00002Q	.023902	.0387965	0.62	0.538	-.0521578	.0999618
__00003B	.1993333	.0835512	2.39	0.017	.0355329	.3631337
__00003P	-.0318031	.0595886	-0.53	0.594	-.1486254	.0850191
__00004A	-.0732387	.0811652	-0.90	0.367	-.2323614	.085884
__00004O	.0736722	.0508641	1.45	0.148	-.0260458	.1733902
__000059	.0146672	.063278	0.23	0.817	-.109388	.1387225
__00005N	-.0115724	.0385629	-0.30	0.764	-.0871742	.0640294
__000068	.0315816	.0976949	0.32	0.747	-.1599473	.2231105
__00006M	.1015343	.060459	1.68	0.093	-.0169944	.220063
__000077	1.323568	.3287563	4.03	0.000	.6790482	1.968089
__00007L	.0107608	.2566924	0.04	0.967	-.4924795	.5140011
__000086	.2108388	.137597	1.53	0.126	-.0589174	.480595
__00008K	.0912556	.0910781	1.00	0.316	-.0873011	.2698124
__000095	.0637875	.1069487	0.60	0.551	-.1458832	.2734583
__00009J	.0368363	.0626566	0.59	0.557	-.0860008	.1596734

__00009X	-1.016167	.6322174	-1.61	0.108	-2.255617	.2232825
__00009Y	-.0362802	.1109002	-0.33	0.744	-.2536979	.1811375
__00009Z	.2292892	.2583514	0.89	0.375	-.2772036	.7357821
__0000A0	1.102554	.6451399	1.71	0.088	-.1622298	2.367338
__0000A1	.0115443	.1368425	0.08	0.933	-.2567327	.2798212
__0000A2	.4721217	.2217937	2.13	0.033	.0372996	.9069437
__0000A3	-.1340506	.0898227	-1.49	0.136	-.3101462	.0420449
__0000A4	.020143	.0569637	0.35	0.724	-.0915332	.1318192
__0000A5	.0092869	.0472177	0.20	0.844	-.0832826	.1018563
__0000A6	.0162827	.059835	0.27	0.786	-.1010225	.133588
__0000A7	-.0885135	.0742429	-1.19	0.233	-.2340652	.0570383
__0000A8	.0302017	.0728061	0.41	0.678	-.1125332	.1729366
__0000A9	-.0793198	.0623564	-1.27	0.203	-.2015683	.0429286
__0000AA	-.0167782	.0606079	-0.28	0.782	-.1355989	.1020424
__0000AB	4.873457	1.216033	4.01	0.000	2.489449	7.257466
__0000AC	.4220158	.2324345	1.82	0.069	-.0336674	.877699
__0000AD	-1.946324	.5067881	-3.84	0.000	-2.939872	-.9527763
__0000AE	-3.623081	1.260619	-2.87	0.004	-6.094499	-1.151663
__0000AF	-.285124	.2890858	-0.99	0.324	-.851871	.281623
__0000AG	-.9526336	.5109147	-1.86	0.062	-1.954272	.0490044
__0000AH	-.215154	.1786418	-1.20	0.229	-.5653777	.1350697
__0000AI	-.4769981	.1144011	-4.17	0.000	-.7012792	-.2527171
__0000AJ	.2557313	.1059151	2.41	0.016	.0480868	.4633758
__0000AK	-.0441098	.1438532	-0.31	0.759	-.3261312	.2379115
__0000AL	.0321836	.178632	0.18	0.857	-.318021	.3823881
__0000AM	-.3361265	.1640413	-2.05	0.041	-.6577263	-.0145267
__0000AN	-.0835223	.1328373	-0.63	0.530	-.3439472	.1769025
__0000AO	.0340001	.1306215	0.26	0.795	-.2220808	.2900809
__0000AP	.1831102	.2962894	0.62	0.537	-.3977593	.7639796
__0000AQ	.0311011	.0482996	0.64	0.520	-.0635892	.1257914
__0000AR	-.092335	.1163613	-0.79	0.428	-.3204591	.135789
__0000AS	-.2121	.3030966	-0.70	0.484	-.8063149	.3821148
__0000AT	.1088699	.0705614	1.54	0.123	-.0294642	.2472041
__0000AU	-.1329533	.1212153	-1.10	0.273	-.3705933	.1046868
__0000AV	-.0289066	.0491882	-0.59	0.557	-.1253391	.0675259
__0000AW	.0472432	.0297425	1.59	0.112	-.0110664	.1055528
__0000AX	.016512	.0251307	0.66	0.511	-.0327563	.0657802
__0000AY	-.0220415	.0346293	-0.64	0.524	-.0899315	.0458485
__0000AZ	.0282373	.0402626	0.70	0.483	-.0506967	.1071713
__0000B0	-.0549972	.0371796	-1.48	0.139	-.1278871	.0178927
__0000B1	-.0166694	.0332107	-0.50	0.616	-.0817782	.0484395
__0000B2	-.0130263	.0324615	-0.40	0.688	-.0766664	.0506138
__0000B3	-.5041776	.5014633	-1.01	0.315	-1.487286	.4789312
__0000B4	-.0076969	.0677564	-0.11	0.910	-.1405319	.1251381
__0000B5	.2728147	.1792813	1.52	0.128	-.0786628	.6242922
__0000B6	-.1528628	.4480043	-0.34	0.733	-1.031166	.7254407
__0000B7	.1036079	.0931305	1.11	0.266	-.0789726	.2861884
__0000B8	.427444	.1771878	2.41	0.016	.0800708	.7748171
__0000B9	.0692293	.0651999	1.06	0.288	-.0585937	.1970524
__0000BA	.0526889	.0418324	1.26	0.208	-.0293226	.1347004
__0000BB	-.1283754	.0336026	-3.82	0.000	-.1942525	-.0624982
__0000BC	-.0701709	.0455533	-1.54	0.124	-.1594773	.0191354
__0000BD	-.0441296	.0609368	-0.72	0.469	-.1635951	.0753358
__0000BE	-.0020201	.0499686	-0.04	0.968	-.0999825	.0959424
__0000BF	-.0258658	.0483771	-0.53	0.593	-.1207081	.0689764
__0000BG	-.0501879	.0410114	-1.22	0.221	-.1305899	.030214
__0000BH	-.7076759	.3638237	-1.95	0.052	-1.420945	.0055933
__0000BI	-.061445	.0609383	-1.01	0.313	-.1809134	.0580234
__0000BJ	.3306494	.1529407	2.16	0.031	.0308123	.6304865
__0000BK	.3870006	.3564941	1.09	0.278	-.3118989	1.0859
__0000BL	.0903474	.0845185	1.07	0.285	-.0753494	.2560442
__0000BM	.1025866	.1401329	0.73	0.464	-.1721413	.3773144
__0000BN	-.0504902	.0552614	-0.91	0.361	-.1588291	.0578487
__0000BO	.087156	.0388645	2.24	0.025	.0109629	.1633492
__0000BP	.0460906	.0301049	1.53	0.126	-.0129294	.1051106
__0000BQ	-.0023542	.0428961	-0.05	0.956	-.0864512	.0817428
__0000BR	-.0005148	.0484378	-0.01	0.992	-.0954761	.0944466
__0000BS	.0008544	.0472844	0.02	0.986	-.0918457	.0935545
__0000BT	.0074836	.0396525	0.19	0.850	-.0702544	.0852215
__0000BU	.0065224	.0373257	0.17	0.861	-.0666538	.0796987
__0000BV	.5045778	.2876166	1.75	0.079	-.0592887	1.068444
__0000BW	.0994028	.0499541	1.99	0.047	.0014688	.1973368

__0000BX	-.2902786	.118127	-2.46	0.014	-.5218643	-.0586929
__0000BY	-.2995776	.2768337	-1.08	0.279	-.8423044	.2431493
__0000BZ	-.0242703	.0669053	-0.36	0.717	-.1554368	.1068961
__0000C0	-.0953875	.1187083	-0.80	0.422	-.3281127	.1373378
__0000C1	.0051926	.041072	0.13	0.899	-.0753284	.0857135
__0000C2	-.0415102	.0293789	-1.41	0.158	-.0991069	.0160866
__0000C3	.0327929	.0241839	1.36	0.175	-.0146192	.080205
__0000C4	.04613	.0307032	1.50	0.133	-.0140629	.106323
__0000C5	.0139943	.0394572	0.35	0.723	-.0633609	.0913494
__0000C6	-.0164661	.0359425	-0.46	0.647	-.0869307	.0539985
__0000C7	-.0142613	.0307019	-0.46	0.642	-.0744518	.0459292
__0000C8	-.0000871	.0309956	-0.00	0.998	-.0608533	.0606791
__0000C9	.6231241	.4813508	1.29	0.196	-.3205544	1.566803
__0000CA	.2046177	.076076	2.69	0.007	.0554723	.3537631
__0000CB	-.4436407	.1914594	-2.32	0.021	-.818993	-.0682884
__0000CC	-.3936307	.4626235	-0.85	0.395	-1.300595	.5133335
__0000CD	-.076854	.1008142	-0.76	0.446	-.2744982	.1207902
__0000CE	-.5249856	.1776894	-2.95	0.003	-.8733421	-.1766292
__0000CF	.0437648	.0601414	0.73	0.467	-.0741413	.1616708
__0000CG	-.155813	.0396389	-3.93	0.000	-.2335243	-.0781016
__0000CH	.0474844	.0352354	1.35	0.178	-.0215938	.1165626
__0000CI	.0728783	.0489919	1.49	0.137	-.0231693	.1689259
__0000CJ	.0159489	.0608988	0.26	0.793	-.103442	.1353397
__0000CK	-.0796506	.0567101	-1.40	0.160	-.1908297	.0315285
__0000CL	-.0160531	.0489481	-0.33	0.743	-.1120148	.0799086
__0000CM	.0232686	.0500484	0.46	0.642	-.0748503	.1213875
__0000CN	1.255815	1.22514	1.03	0.305	-1.146048	3.657678
__0000CO	-.3156563	.2051711	-1.54	0.124	-.7178901	.0865776
__0000CP	.3667818	.4362254	0.84	0.400	-.4884294	1.221993
__0000CQ	-2.343202	1.585685	-1.48	0.140	-5.451906	.7655013
__0000CR	-.2372921	.3477211	-0.68	0.495	-.9189925	.4444082
__0000CS	.0773117	.7457818	0.10	0.917	-1.384779	1.539402
__0000CT	-.2745776	.1856597	-1.48	0.139	-.6385597	.0894045
__0000CU	.0079775	.116042	0.07	0.945	-.2195205	.2354755
__0000CV	-.1157573	.0923909	-1.25	0.210	-.2968879	.0653732
__0000CW	-.0822292	.166522	-0.49	0.621	-.4086923	.2442339
__0000CX	.1249143	.1731466	0.72	0.471	-.2145362	.4643647
__0000CY	-.1422547	.2609648	-0.55	0.586	-.6538709	.3693615
__0000CZ	.1474008	.2954136	0.50	0.618	-.4317517	.7265532
__0000D0	.0151331	.2772853	0.05	0.956	-.5284791	.5587454
__0000D1	.8660032	.7207836	1.20	0.230	-.5470787	2.279085
__0000D2	.0942748	.1037639	0.91	0.364	-.1091522	.2977018
__0000D3	-.2499386	.2661535	-0.94	0.348	-.7717273	.27185
__0000D4	-1.102697	.70114	-1.57	0.116	-2.477269	.2718735
__0000D5	-.5258652	.1572843	-3.34	0.001	-.8342179	-.2175124
__0000D6	.5411497	.2661827	2.03	0.042	.0193038	1.062996
__0000D7	.0602148	.1118489	0.54	0.590	-.1590628	.2794924
__0000D8	.0027675	.077382	0.04	0.971	-.1489384	.1544734
__0000D9	.0256167	.05556	0.46	0.645	-.0833074	.1345409
__0000DA	-.0501007	.0645436	-0.78	0.438	-.1766372	.0764358
__0000DB	-.0434587	.0760166	-0.57	0.568	-.1924876	.1055703
__0000DC	.2047761	.1614725	1.27	0.205	-.1117874	.5213396
__0000DD	.120927	.0852512	1.42	0.156	-.0462063	.2880602
__0000DE	.079012	.0829232	0.95	0.341	-.0835574	.2415813
__0000DF	-.0687147	.4556613	-0.15	0.880	-.9620297	.8246003
__0000DG	-.0288465	.0756407	-0.38	0.703	-.1771386	.1194456
__0000DH	.0726756	.1778484	0.41	0.683	-.2759927	.4213439
__0000DI	.1698645	.4680779	0.36	0.717	-.7477929	1.087522
__0000DJ	-.3096267	.1260253	-2.46	0.014	-.5566968	-.0625566
__0000DK	.0399255	.1971344	0.20	0.840	-.3465526	.4264035
__0000DL	.1061665	.0750176	1.42	0.157	-.0409041	.2532371
__0000DM	-.0877391	.0586374	-1.50	0.135	-.2026965	.0272182
__0000DN	.0121142	.0431749	0.28	0.779	-.0725293	.0967577
__0000DO	-.116237	.052043	-2.23	0.026	-.2182664	-.0142077
__0000DP	-.0459981	.0639551	-0.72	0.472	-.1713807	.0793845
__0000DQ	-.0418487	.0973231	-0.43	0.667	-.2326487	.1489513
__0000DR	-.0120883	.0584683	-0.21	0.836	-.1267142	.1025377
__0000DS	-.0366322	.0500407	-0.73	0.464	-.1347359	.0614715
__0000DU	9.152845	22.73975	0.40	0.687	-35.42797	53.73367
__0000DW	-2.73138	12.45205	-0.22	0.826	-27.14337	21.68061
__0000DZ	.5413764	.5649458	0.96	0.338	-.5661886	1.648941
__0000E1	.2460351	.3872673	0.64	0.525	-.5131948	1.005265

__0000E4	-2.808637	4.506962	-0.62	0.533	-11.64445	6.027173
__0000E6	-.5430099	2.413357	-0.23	0.822	-5.274348	4.188328
__0000E9	-10.10867	28.02977	-0.36	0.718	-65.06047	44.84314
__0000EB	6.393056	15.74463	0.41	0.685	-24.47397	37.26009
__0000EE	.2954145	.7223615	0.41	0.683	-1.120761	1.71159
__0000EG	-1.44771	.6499883	-2.23	0.026	-2.721999	-.1734208
__0000EJ	-1.078427	1.314223	-0.82	0.412	-3.654935	1.498082
__0000EL	-1.639731	.949335	-1.73	0.084	-3.500883	.2214212
__0000EO	-.2077712	.4396889	-0.47	0.637	-1.069773	.6542301
__0000EQ	.2816809	.3411195	0.83	0.409	-.387077	.9504389
__0000ET	-.2727196	.3887682	-0.70	0.483	-1.034892	.4894527
__0000EV	.2122087	.2271126	0.93	0.350	-.2330411	.6574585
__0000EY	-.8164115	.307942	-2.65	0.008	-1.420126	-.2126974
__0000F0	-.1098108	.2809745	-0.39	0.696	-.6606556	.4410341
__0000F3	.4289897	.3332119	1.29	0.198	-.2242657	1.082245
__0000F5	.2445698	.2862323	0.85	0.393	-.3165828	.8057225
__0000F8	-.647714	.4428608	-1.46	0.144	-1.515934	.2205056
__0000FA	-.1046339	.349424	-0.30	0.765	-.7896728	.5804049
__0000FD	-.0117316	.5899835	-0.02	0.984	-1.168382	1.144919
__0000FF	.2466887	.4301439	0.57	0.566	-.5965999	1.089977
__0000FI	-.0248526	.3656804	-0.07	0.946	-.7417617	.6920565
__0000FK	-.5954683	.2774041	-2.15	0.032	-1.139314	-.0516231
__0000FN	.8080644	.3498996	2.31	0.021	.1220932	1.494036
__0000FP	-.3751199	.2751371	-1.36	0.173	-.9145207	.1642809
xbarlag	-8.096306	27.16222	-0.30	0.766	-61.34729	45.15468
x2barlag	-1.082772	.6269191	-1.73	0.084	-2.311834	.1462904
xbar2lag	4.478617	5.583237	0.80	0.423	-6.467209	15.42444
rootxbarlag	4.488399	32.97986	0.14	0.892	-60.16796	69.14475
z1barlag	1.40378	1.045271	1.34	0.179	-.6454532	3.453013
z2barlag	3.073449	1.653122	1.86	0.063	-.1674643	6.314362
z3barlag	-.0153116	.5844448	-0.03	0.979	-1.161104	1.130481
z4barlag	.2274196	.4022212	0.57	0.572	-.5611271	1.015966
z5barlag	.8547018	.4470831	1.91	0.056	-.0217957	1.731199
z6barlag	-.6274022	.5064445	-1.24	0.215	-1.620277	.3654722
z7barlag	.7883646	.5799309	1.36	0.174	-.3485785	1.925308
z8barlag	.0771658	.844388	0.09	0.927	-1.57824	1.732572
z9barlag	.8780704	.4758521	1.85	0.065	-.0548282	1.810969
z10barlag	-.4189417	.4362161	-0.96	0.337	-1.274135	.4362511
_cons	-.4308714	11.17074	-0.04	0.969	-22.33089	21.46915

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000GB	128,640	.4438109	.1421897	-.0131757	1.133586

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"

957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
__0000G5 __0000G8 __0000GE __0000GH __0000G6 __0000G9 __0000GF __0000GI

960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
  2.                 foreach var2 of global Alist {
  3.                         g `var1'_`var2'=`var1'*`var2'
  4.                         sum `var1' `var2' `var1'_`var2'
  5.                         local qhat_hats "`qhat_hats' `var1'_`var2'"
  6.                 }
  7.         }
```

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G5	128,640	.4805346	.0994099	.1145993	.9998097
one	128,640	1	0	1	1
__0000G5_one	128,640	.4805346	.0994099	.1145993	.9998097

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G5	128,640	.4805346	.0994099	.1145993	.9998097
schednhi	128,640	.0818719	.2741705	0	1
__0000G5_s~i	128,640	.0519162	.1773907	0	.9998097

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G5	128,640	.4805346	.0994099	.1145993	.9998097
schedhi	128,640	.0961443	.2947901	0	1
__0000G5~dhi	128,640	.0358377	.1123062	0	.6451631

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G5	128,640	.4805346	.0994099	.1145993	.9998097
nschednhi	128,640	.1535448	.3605132	0	1
__0000G5_n~i	128,640	.0775048	.1851197	0	.8314313

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G8	128,640	.5308714	.1178934	.1140262	1.089443
one	128,640	1	0	1	1
__0000G8_one	128,640	.5308714	.1178934	.1140262	1.089443

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G8	128,640	.5308714	.1178934	.1140262	1.089443
schednhi	128,640	.0818719	.2741705	0	1
__0000G8_s~i	128,640	.0542249	.1845708	0	.9958048

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G8	128,640	.5308714	.1178934	.1140262	1.089443
schedhi	128,640	.0961443	.2947901	0	1
__0000G8~dhi	128,640	.0398963	.1263003	0	.7901694

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G8	128,640	.5308714	.1178934	.1140262	1.089443
nschednhi	128,640	.1535448	.3605132	0	1
__0000G8_n~i	128,640	.0865605	.2071044	0	1.060837

Variable	Obs	Mean	Std. dev.	Min	Max
__0000GE	128,640	.4859174	.1747112	-.0127293	1.32599
one	128,640	1	0	1	1
__0000GE_one	128,640	.4859174	.1747112	-.0127293	1.32599

Variable	Obs	Mean	Std. dev.	Min	Max
__0000GE	128,640	.4859174	.1747112	-.0127293	1.32599
schednhi	128,640	.0818719	.2741705	0	1
__0000GE_s~i	128,640	.054963	.1914501	0	1.268115

Variable	Obs	Mean	Std. dev.	Min	Max
__0000GE	128,640	.4859174	.1747112	-.0127293	1.32599
schedhi	128,640	.0961443	.2947901	0	1
__0000GE~dhi	128,640	.0288252	.09309	0	.8587021

Variable	Obs	Mean	Std. dev.	Min	Max
__0000GE	128,640	.4859174	.1747112	-.0127293	1.32599
nschednhi	128,640	.1535448	.3605132	0	1
__0000GE_n~i	128,640	.0663605	.1670422	-.0127293	1.018219

Variable	Obs	Mean	Std. dev.	Min	Max
__0000GH	128,640	.5317179	.1886221	-.0157666	1.66005
one	128,640	1	0	1	1
__0000GH_one	128,640	.5317179	.1886221	-.0157666	1.66005

Variable	Obs	Mean	Std. dev.	Min	Max
__0000GH	128,640	.5317179	.1886221	-.0157666	1.66005
schednhi	128,640	.0818719	.2741705	0	1
__0000GH_s~i	128,640	.0571488	.1978471	0	1.263035

Variable	Obs	Mean	Std. dev.	Min	Max
__0000GH	128,640	.5317179	.1886221	-.0157666	1.66005
schedhi	128,640	.0961443	.2947901	0	1
__0000GH~dhi	128,640	.0319017	.1035573	0	1.025252

Variable	Obs	Mean	Std. dev.	Min	Max
__0000GH	128,640	.5317179	.1886221	-.0157666	1.66005
nschednhi	128,640	.1535448	.3605132	0	1
__0000GH_n~i	128,640	.0731742	.1828561	-.0157666	1.299163

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G6	128,640	.5168546	.3955048	.0222147	3.671393
one	128,640	1	0	1	1
__0000G6_one	128,640	.5168546	.3955048	.0222147	3.671393

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G6	128,640	.5168546	.3955048	.0222147	3.671393
schednhi	128,640	.0818719	.2741705	0	1
__0000G6_s~i	128,640	.0719502	.2847952	0	3.655755

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G6	128,640	.5168546	.3955048	.0222147	3.671393
schedhi	128,640	.0961443	.2947901	0	1
__0000G6~dhi	128,640	.0270828	.1065892	0	2.447787

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G6	128,640	.5168546	.3955048	.0222147	3.671393
nschednhi	128,640	.1535448	.3605132	0	1
__0000G6_n~i	128,640	.0814167	.2412095	0	3.671393

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G9	128,640	.5703316	.441691	.0221036	4.684393
one	128,640	1	0	1	1
__0000G9_one	128,640	.5703316	.441691	.0221036	4.684392

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G9	128,640	.5703316	.441691	.0221036	4.684393
schednhi	128,640	.0818719	.2741705	0	1
__0000G9_s~i	128,640	.0747731	.2939247	0	3.641111

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G9	128,640	.5703316	.441691	.0221036	4.684393
schedhi	128,640	.0961443	.2947901	0	1
__0000G9~dhi	128,640	.0303338	.1216035	0	3.123173

Variable	Obs	Mean	Std. dev.	Min	Max
__0000G9	128,640	.5703316	.441691	.0221036	4.684393
nschednhi	128,640	.1535448	.3605132	0	1
__0000G9_n~i	128,640	.0905217	.2695731	0	4.684392

Variable	Obs	Mean	Std. dev.	Min	Max
__0000GF	128,640	.5538199	.5212261	-.0031559	6.017912
one	128,640	1	0	1	1
__0000GF_one	128,640	.5538199	.5212261	-.0031559	6.017912

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000GF |    128,640    .5538199    .5212261  -.0031559   6.017912
     schednhi |    128,640    .0818719    .2741705          0          1
__0000GF_s~i |    128,640    .0782696    .3237378          0   5.029466

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000GF |    128,640    .5538199    .5212261  -.0031559   6.017912
      schedhi |    128,640    .0961443    .2947901          0          1
__0000GF~dhi |    128,640    .0228499    .0995676          0   3.225104

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000GF |    128,640    .5538199    .5212261  -.0031559   6.017912
    nschednhi |    128,640    .1535448    .3605132          0          1
__0000GF_n~i |    128,640    .0742251    .2457567  -.0031559   4.403024

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000GI |    128,640    .6069913    .5810036  -.0039089   7.534021
           one |    128,640           1           0          1          1
__0000GI_one |    128,640    .6069913    .5810036  -.0039089   7.534021

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000GI |    128,640    .6069913    .5810036  -.0039089   7.534021
     schednhi |    128,640    .0818719    .2741705          0          1
__0000GI_s~i |    128,640    .0810708    .3326981          0    5.00932

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000GI |    128,640    .6069913    .5810036  -.0039089   7.534021
      schedhi |    128,640    .0961443    .2947901          0          1
__0000GI~dhi |    128,640    .0254645    .1128858          0   4.114964

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000GI |    128,640    .6069913    .5810036  -.0039089   7.534021
    nschednhi |    128,640    .1535448    .3605132          0          1
__0000GI_n~i |    128,640    .0816984    .2720997  -.0039089   5.617892
964 }

965
966 local r_RE      "`r_RE' `qhat_hats'"

967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                 local r_RE "`r_RE' zi`i'"
    3.         }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
```

```
977 }

978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                 g pzi`j' = p1*zi`j'
    3.          }
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS	Number of obs	=	128,640
				F(16, 128623)	=	46631.28
Model	5934.91193	16	370.931996	Prob > F	=	0.0000
Residual	1023.1412	128,623	.007954574	R-squared	=	0.8530
				Adj R-squared	=	0.8529
Total	6958.05313	128,639	.054089764	Root MSE	=	.08919

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	.0184165	.0009592	19.20	0.000	.0165364	.0202966
ybarx	.0028598	.0010506	2.72	0.006	.0008005	.004919
xi2	-.0393257	.0005751	-68.37	0.000	-.0404529	-.0381984
ybar	-.1939297	.0027181	-71.35	0.000	-.1992571	-.1886024
xi	.4199094	.001663	252.50	0.000	.4166499	.4231689
ybar_q1	.4204275	.0033072	127.12	0.000	.4139454	.4269096
pzi1	.2239723	.0013308	168.30	0.000	.221364	.2265806
pzi2	.0183818	.0021733	8.46	0.000	.0141221	.0226414
pzi3	-.0030657	.0006648	-4.61	0.000	-.0043687	-.0017626
pzi4	.0097026	.0006721	14.44	0.000	.0083853	.0110198
pzi5	-.0146946	.0007121	-20.64	0.000	-.0160903	-.013299
pzi6	.0019168	.0006005	3.19	0.001	.0007398	.0030938
pzi7	-.004409	.0007736	-5.70	0.000	-.0059253	-.0028927
pzi8	.0097843	.0010764	9.09	0.000	.0076744	.0118941
pzi9	.0062135	.0009824	6.32	0.000	.004288	.008139
pzi10	.0018325	.000844	2.17	0.030	.0001782	.0034868
_cons	-.0235577	.001692	-13.92	0.000	-.0268739	-.0202414

```
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
```

```
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = -_b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                local coef = _b[pzi`j']
    3.                local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1025}

1026
1027if `J'==3 & simple_model==0
  >                             local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                             local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00504739
  Iteration 1:  GMM criterion Q(b) =  .00068149
  Iteration 2:  GMM criterion Q(b) =  .00026544
  Iteration 3:  GMM criterion Q(b) =   .0001769

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00778784
  Iteration 1:  GMM criterion Q(b) =  .00609319
  Iteration 2:  GMM criterion Q(b) =  .00606826

  GMM estimation
```

Number of parameters = **29**
Number of moments = **58**
Initial weight matrix: **Unadjusted** Number of obs = **128,640**
GMM weight matrix: **Robust**

(Std. err. adjusted for **564** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
A11						
one	.6058117	.0353814	17.12	0.000	.5364654	.6751581
schednhi	.1678525	.066441	2.53	0.012	.0376305	.2980745
schedhi	.2474445	.0871718	2.84	0.005	.0765909	.418298
nschednhi	.1793084	.054727	3.28	0.001	.0720453	.2865714
C1						
zi1	.472338	.0478942	9.86	0.000	.378467	.566209
zi2	-.1808125	.0760098	-2.38	0.017	-.3297889	-.031836
zi3	-.0035677	.0219407	-0.16	0.871	-.0465706	.0394353
zi4	.0814969	.0278483	2.93	0.003	.0269152	.1360787
zi5	.0513747	.0244921	2.10	0.036	.0033711	.0993784
zi6	.0174868	.0187851	0.93	0.352	-.0193313	.0543049
zi7	-.0029191	.0241081	-0.12	0.904	-.0501701	.0443318
zi8	-.0421232	.0397586	-1.06	0.289	-.1200486	.0358022
zi9	.0317341	.028182	1.13	0.260	-.0235016	.0869698
zi10	.0465227	.0240261	1.94	0.053	-.0005676	.093613
C2						
zi1	.2526695	.0774881	3.26	0.001	.1007956	.4045433
zi2	-.3513105	.1245497	-2.82	0.005	-.5954235	-.1071976
zi3	-.0017155	.035132	-0.05	0.961	-.0705729	.067142
zi4	.1426795	.047254	3.02	0.003	.0500634	.2352956
zi5	.1015866	.032742	3.10	0.002	.0374135	.1657597
zi6	.0152437	.0281899	0.54	0.589	-.0400075	.0704949
zi7	-.0116606	.0408582	-0.29	0.775	-.0917412	.06842
zi8	-.0871661	.0424502	-2.05	0.040	-.170367	-.0039653
zi9	.0477992	.0247768	1.93	0.054	-.0007625	.0963609
zi10	-.0022855	.0229268	-0.10	0.921	-.0472212	.0426501
/d1	-.0738388	.0057603	-12.82	0.000	-.0851287	-.0625488
/b1	.3738563	.0180674	20.69	0.000	.3384449	.4092678
/AVA11	.5939613	.7348825	0.81	0.419	-.8463819	2.034305
/AVA12	-.8284247	.6559083	-1.26	0.207	-2.113981	.4571318
/AVA22	.2866836	.6300031	0.46	0.649	-.9480998	1.521467

Instruments for equation **eq1_RE**: **__0000FS __0000FU __0000FX xi __000002 __00000E __00000S __00001D __00001R __00002C __00002Q __00003B __00003P __00004A __00004O __000059 __00005N __000068 __00006M __000077 __00007L __000086 __00008K __000095 __00009J __0000G5_one __0000G5_schednhi __0000G5_schedhi __0000G5_nschednhi __0000G8_one __0000G8_schednhi __0000G8_schedhi __0000G8_nschednhi __0000GE_one __0000GE_schednhi __0000GE_schedhi __0000GE_nschednhi __0000GH_one __0000GH_schednhi __0000GH_schedhi __0000GH_nschednhi __0000G6_one __0000G6_schednhi __0000G6_schedhi __0000G6_nschednhi __0000G9_one __0000G9_schednhi __0000G9_schedhi __0000G9_nschednhi __0000GF_one __0000GF_schednhi __0000GF_schedhi __0000GF_nschednhi __0000GI_one __0000GI_schednhi __0000GI_schedhi __0000GI_nschednhi _cons**

```
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
> _b[/a]+1.19*_b[/b])
```

```
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactIns
      > t_noPPXdInst_14_zlist2_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not found)
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist2_updateExp_urbOnly_clus_Dist_dropPre_RE.tex)

1070
1071
```

```
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  24757

1077unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640

1078estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  4599

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  5.3831268

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist2_updateExp_urbOnly_clus_Dist_dropPre_RE.ster not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist2_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
```

```
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*`p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
1119                                                 gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                  local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                          }
   30.                                  }
   31.                          }
   32.                  }
   33.          }
```

```
1120        if `Tindivp1'<=`T' {
1121                forval t=`Tindivp1'/`T' {
    2.                          tempvar zi`t'xdiff
    3.                          g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                          local zg "`zg' zi`t'"
    5.                          forval j=1/`J' {
    6.                                  tempvar zi`t'p`j'xdiff
    7.                                  g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                  local pzg "`pzg' `zi`t'p`j''"
    9.                                  local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                          }
   11.                 }
1122        }
1123
1124        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1125        // denoting zg as group-zs
1126        //  zg,pzgx,rprp,rprpx
1127        local rootprootp ""
1128        local rootprootpxdiff ""
1129
1130        forval j=1/`J' {
    2.                  forval s=`j'/`J' {
    3.                          tempvar xdiffrp`j'rp`s'
    4.                          g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1131                            if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1132                            local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                  }
    8.          }
1133
1134        local pqhatxdiff ""
1135        forval j=1/`J' {
    2.                  forval s=1/`J' {
    3.                          foreach name of global Alist {
    4.                                  tempvar   p`s'qikhat`j'xdiff`name'
    5.                                  g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
1136                                    local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
1137
1138        // set instruments
1139        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1140}

1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
```

```
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                          local `++t'
    3.                          local p_nice = subinstr("`p'",":_cons","",.)
    4.                          local est = coefs[1,`t']
    5.                          if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                  }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                 }
   16.         }
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
```

```
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1    `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
```

```
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique
  note: instruments __00007A __00007O __000089 __00008N __000098 __00009M __0000M1
    __0000M2 __0000M3 __0000M4 __0000M6 __0000M7 __0000M8 __0000M9 __0000ME
    __0000MF __0000MG __0000MH __0000MJ __0000MK __0000ML __0000MM __0000MS
    __0000MT __0000MU __0000MV __0000MX __0000MY __0000MZ __0000N0 __0000N5
    __0000N6 __0000N7 __0000N8 __0000NA __0000NB __0000NC __0000ND __0000NJ
    __0000NK __0000NL __0000NM __0000NO __0000NP __0000NQ __0000NR __0000NW
    __0000NX __0000NY __0000NZ __0000O1 __0000O2 __0000O3 __0000O4 omitted because
    of collinearity.

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00101813
  Iteration 1:  GMM criterion Q(b) =  .00048882
  Iteration 2:  GMM criterion Q(b) =  .00042247

  Step 2
  Iteration 0:  GMM criterion Q(b) =   .0169248
  Iteration 1:  GMM criterion Q(b) =  .01586015
  Iteration 2:  GMM criterion Q(b) =  .01576403
  Iteration 3:  GMM criterion Q(b) =  .01573856

  GMM estimation

  Number of parameters =  26
  Number of moments    = 165
  Initial weight matrix: Unadjusted                 Number of obs   =    128,640
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11						
_cons	.7006295	.0352048	19.90	0.000	.6316293	.7696296
C12						
_cons	-.0530542	.0518312	-1.02	0.306	-.1546415	.0485332
C13						
_cons	-.0243017	.0142222	-1.71	0.088	-.0521767	.0035733
C14						
_cons	-.0045796	.0132502	-0.35	0.730	-.0305495	.0213903
C15						
_cons	.0170841	.027673	0.62	0.537	-.037154	.0713222
C16						
_cons	-.0438755	.0192235	-2.28	0.022	-.0815529	-.0061981

C17						
_cons	-.0518984	.0225214	-2.30	0.021	-.0960396	-.0077572
C18						
_cons	1.594011	.7453152	2.14	0.032	.1332197	3.054802
C19						
_cons	1.002749	.3430838	2.92	0.003	.3303167	1.67518
C110						
_cons	.7611957	.1983591	3.84	0.000	.3724191	1.149972
C21						
_cons	.5742177	.0498946	11.51	0.000	.476426	.6720093
C22						
_cons	-.1145026	.0739106	-1.55	0.121	-.2593647	.0303595
C23						
_cons	-.0320429	.01937	-1.65	0.098	-.0700074	.0059216
C24						
_cons	-.0231467	.0228096	-1.01	0.310	-.0678526	.0215593
C25						
_cons	.0443209	.0337598	1.31	0.189	-.0218471	.1104888
C26						
_cons	-.0595294	.0248823	-2.39	0.017	-.1082977	-.010761
C27						
_cons	-.0693107	.0321432	-2.16	0.031	-.1323102	-.0063111
C28						
_cons	-1.766084	.5712014	-3.09	0.002	-2.885618	-.6465499
C29						
_cons	-.9931676	.3430393	-2.90	0.004	-1.665512	-.3208229
C210						
_cons	-.907638	.1828267	-4.96	0.000	-1.265972	-.5493042
d1						
_cons	-.0826538	.0057526	-14.37	0.000	-.0939286	-.071379
b1						
_cons	.4550525	.0366205	12.43	0.000	.3832777	.5268274
A11						
one	.2554541	.1210604	2.11	0.035	.0181801	.492728
schednhi	.1298939	.2372344	0.55	0.584	-.3350769	.5948647
schedhi	-.2853451	.3300544	-0.86	0.387	-.9322398	.3615496
nschednhi	-.0746894	.1531849	-0.49	0.626	-.3749262	.2255474

Instruments for equation **eq1_FE**: __000004 __00000H __00000V __00001G __00001U
__00002F __00002T __00003E __00003S __00004D __00004R __00005C __00005Q
__00006B __00006P o.__00007A o.__00007O o.__000089 o.__00008N o.__000098
o.__00009M __000005 __0000O8 __0000O9 __0000OA __0000OB __0000OC __0000OD
__0000OE __0000OF __0000OG __0000OH __0000OI __0000OJ __0000OK __0000OL
__0000OM __0000ON __0000GS __0000GT __0000GU __0000GV __0000GX __0000GY
__0000GZ __0000H0 __0000H5 __0000H6 __0000H7 __0000H8 __0000HA __0000HB
__0000HC __0000HD __0000HJ __0000HK __0000HL __0000HM __0000HO __0000HP
__0000HQ __0000HR __0000HW __0000HX __0000HY __0000HZ __0000I1 __0000I2
__0000I3 __0000I4 __0000IA __0000IB __0000IC __0000ID __0000IF __0000IG
__0000IH __0000II __0000IN __0000IO __0000IP __0000IQ __0000IS __0000IT
__0000IU __0000IV __0000J1 __0000J2 __0000J3 __0000J4 __0000J6 __0000J7
__0000J8 __0000J9 __0000JE __0000JF __0000JG __0000JH __0000JJ __0000JK
__0000JL __0000JM __0000JS __0000JT __0000JU __0000JV __0000JX __0000JY
__0000JZ __0000K0 __0000K5 __0000K6 __0000K7 __0000K8 __0000KA __0000KB
__0000KC __0000KD __0000KJ __0000KK __0000KL __0000KM __0000KO __0000KP
__0000KQ __0000KR __0000KW __0000KX __0000KY __0000KZ __0000L1 __0000L2

```
      __0000L3 __0000L4 __0000LA __0000LB __0000LC __0000LD __0000LF __0000LG
      __0000LH __0000LI __0000LN __0000LO __0000LP __0000LQ __0000LS __0000LT
      __0000LU __0000LV o.__0000M1 o.__0000M2 o.__0000M3 o.__0000M4 o.__0000M6
      o.__0000M7 o.__0000M8 o.__0000M9 o.__0000ME o.__0000MF o.__0000MG o.__0000MH
      o.__0000MJ o.__0000MK o.__0000ML o.__0000MM o.__0000MS o.__0000MT o.__0000MU
      o.__0000MV o.__0000MX o.__0000MY o.__0000MZ o.__0000N0 o.__0000N5 o.__0000N6
      o.__0000N7 o.__0000N8 o.__0000NA o.__0000NB o.__0000NC o.__0000ND o.__0000NJ
      o.__0000NK o.__0000NL o.__0000NM o.__0000NO o.__0000NP o.__0000NQ o.__0000NR
      o.__0000NW o.__0000NX o.__0000NY o.__0000NZ o.__0000O1 o.__0000O2 o.__0000O3
      o.__0000O4 __0000GP __0000H2 __0000HG __0000HT __0000I7 __0000IK __0000IY
      __0000JB __0000JP __0000K2 __0000KG __0000KT __0000L7 __0000LK __0000LY
      __0000MB __0000MP __0000N2 __0000NG __0000NT _cons
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactIns
      > t_noPPXdInst_14_zlist2_updateExp_urbOnly_clus_Dist_dropPre_FE.tex not found)
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist2_updateExp_urbOnly_clus_Dist_dropPre_FE.tex)
1242
1243        estat overid

    Test of overidentifying restriction:

    Hansen's J chi2(139) = 2024.61 (p = 0.0000)
1244        di "$S_DATE $S_TIME"
   7 Dec 2023 20:32:24
1245        unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  24757
1248        unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640
1249        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4599
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  5.3831268
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist2_updateExp_urbOnly_clus_Dist_dropPre_FE.ster not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist2_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
```

```
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}

1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table4_26.smcl
   log type:  smcl
  closed on:   7 Dec 2023, 20:32:24
```

A.4.2 Columns 3 and 7

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table4_37.smcl
  log type:  smcl
 opened on:   7 Dec 2023, 20:45:23

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
  C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=3

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
50. *local group_def "dist"
51. *local group_def "fsuXseg"
52. *local group_def "fsuXsegXreligXsched"
53. local group_def "fsuXsegXreligXschedXownsland"

54.
55. // comparision group
56. local groupComp_def ""

57. local desc_compGrp ""

58. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

59.
60. // load data
61. include "$CODE/prep demand system data.do"

62. //clear all
63. pause on

64. set more off

65. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

66.
67. scalar short_zlist=$short_zlist_val

68.
69. cap prog drop bysmeanw

70. prog def bysmeanw
    1.         syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
71.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
72.         tempvar totweight totsum varXweight
    4.
73.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
74. end

75.
76. // load prices
77. use "$DATA/laspeyres_state_core.dta", clear

78. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

79. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)        expend_type   ->   (dropped)
  xij variables:
                         laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

80. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

81. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

82.
83. use "$DATA/laspeyres_state_core_2cats.dta", clear

84. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

85. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

86. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

87. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                   5   ->   5
  j variable (2 values)        expend_type   ->   (dropped)
  xij variables:
                         laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

88. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
89. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

90.
91. if $use_norm==1 {
92.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
93.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations               840   ->   280
  Number of variables                    5   ->   6
  j variable (3 values)       expend_type   ->   (dropped)
  xij variables:
                         laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

94.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
95.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
96. }

97.
98. // read in consumption data
99. use "$DATA/years1999_2004.dta", clear

100 *use "$DATA/years1999_2004_sm.dta", clear
101 g hhid = _n

102
103 // update expenditure if normal goods
104 if $use_norm==1 {
105         egen update_exp = rowtotal(food fuel clothing other_nondur)
106         replace expenditure = update_exp
  (197,188 real changes made)
107 }

108
109 // possible groups
110 egen group_district=group(state district)

111 egen group_round_district=group(state district round)

112 egen group_round_districtUrb=group(state district round urban)

113
114 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
115 drop _merge

116 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
117 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

118 drop _merge

119
120 //make weights comparable across rounds
121 bys round: egen meanweight=mean(weight)

122 bys round: replace weight=weight/meanweight
  (234590 real changes made)

123 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
124
125 // clean hh chars
126 replace religion=1 if religion==.
  (20 real changes made)

127 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
128 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

129 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
130 g scheduled = socgroup==1 | socgroup==2

131 g hindu = religion==1

132 g muslim = religion==2

133 egen hinduXscheduled = group(religion hindu)

134 g schednh = scheduled==1 & hindu==0

135 g schedh  = scheduled==1 & hindu==1

136 g nschedh = scheduled==0 & hindu==1

137 g nschednh = scheduled==0 & hindu==0

138 g nonHindu=religion>1

139
140 // simplify education categories
141 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

142
143 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
144 local cutoff_group_size=10

145 scalar cutoff_expenditure_percentile=99

146 scalar scale_factor=1

147
```

```
148 // generate variables
149 g low_educ = z9==0 & z10==0

150
151 // drop top and bottom percentiles of expenditure in each round/state
152 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

153 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

154 drop expenditure_cutoff

155 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

156 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

157
158 // keep urban houselds smaller than 13 and with head older than 20
159 keep if urban==1
  (141,042 observations deleted)

160 drop if hhsize>12
  (656 observations deleted)

161 gen married=marstat==2

162 drop if age<20
  (1,182 observations deleted)

163 g hhsizem1=hhsize-1

164
165 // winsorize weights
166 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean           .6966798
                       Largest        Std. dev.       1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance       1.079168
95%     2.451768       26.1213        Skewness       5.702664
99%      4.56912       29.15594       Kurtosis       76.01125

167 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

168 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

169
170 g lux=ilux+vlux
```

```
171 g necc=inec+vnec

172
173 // education variables
174 drop if zone==.
  (799 observations deleted)

175 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
176 g educmed=educsimp==2

177 g educhigh=educsimp==3

178
179 // scale age (helps numerical performance)
180 drop if age==.
  (1 observation deleted)

181 replace age=age/40
  (86,380 real changes made)

182
183 // rename z's so that they can be used in each file
184 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

185
186 // landowner dummy
187 g owns_land = landowned>.005 & landowned<.

188
189 egen group_inst=group(state district urban)

190
191 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

192 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

193 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)

194 if "`group_def'"=="fsuXsegXreligXschedXownsland"       egen group = group(geogroup_
  > seg religion scheduled owns_land)
  (43,637 missing values generated)

195 if "`group_def'"=="fsuXsegXreligXschedXehigh"                  egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
196
197 if $rand_grp==1 {
198         sort round
199         sort round state district
200         g shuffle = _n
201         g rand = runiform()
202         sort round rand
203         sort round state district rand
204         g group_new = group[shuffle]
205         replace group = group_new
206         drop group_new shuffle rand
207 }

208
209 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

210 egen district_round=group(state district round urban)

211
212 // make comparision group, so can restrict to sample in other regs
213 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

214 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

215 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)

216 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

217
218 if short_zlist==1 {
219         global T=7
220
221         g z1=hhsizem1/10
222         g z2=age/3
223         g z3=married
224         g z4=.
225         g land=exp(demog23)
226         replace z4=0 if land<=0.005
227         replace z4=ln(land+1) if land>0.005
228         g z5=demog24
229         g z6=educmed
230         g z7=educhigh
231         order land, after(z7)
232
233         global zlist "z1 z2 z3 z4 z5 z6 z7"
234         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
235         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
236 }

237
238 if short_zlist==2 {
239         global T=10
240
241         g z1=hhsizem1/10
242         g z2=age/3
```

```
243         g z3=married
244         g z4=.
245         g land=exp(demog23)
246         replace z4=0 if land<=0.005
247         replace z4=ln(land+1) if land>0.005
248         g z5=demog24
249         g z6=educmed
250         g z7=educhigh
251         g z8 = scheduled==1 & hindu==0
252         g z9 = scheduled==0 & hindu==0
253         g z10 = scheduled==0 & hindu==1
254
255         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
257         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
258 }

259
260 if short_zlist==3 {
261         global T=8
262
263         g z1=hhsizem1/10
264         g z2=age/3
265         g z3=married
266         g z4=.
  (86,380 missing values generated)
267         g land=exp(demog23)
  (12,776 missing values generated)
268         replace z4=0 if land<=0.005
  (29,502 real changes made)
269         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
270         g z5=demog24
  (29 missing values generated)
271         g z6=educmed
272         g z7=educhigh
273         g z8 = owns_land
274
275         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
276         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
277         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
278 }

279
280 if short_zlist==4 {
281         global T=7
282
283         g z1=hhsizem1/10
284         g z2=age/3
285         g z3=married
286         g z4=.
287         g land=exp(demog23)
288         replace z4=0 if land<=0.005
289         replace z4=ln(land+1) if land>0.005
290         g z5=demog24
291         g z6=educmed
292         g z7=educhigh
293
294         global zlist "z1 z2 z3 z4 z5 z6 z7"
295         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
296         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
297 }
298
299 if short_zlist==5 {
300         global T=6
301
302         g z1=age/3
303         g z2=married
304         g z3=.
305         g land=exp(demog23)
306         replace z3=0 if land<=0.005
307         replace z3=ln(land+1) if land>0.005
308         g z4=demog24
309         g z5=educmed
310         g z6=educhigh
311         order land, after(z6)
312
313         global zlist "z1 z2 z3 z4 z5 z6"
314         global zlist_indiv "z1 z2 z3 z4 z5 z6"
315         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
316 }
317
318 if short_zlist==6 {
319         global T=5
320
321         g z1=hhsizem1/10
322         g z2=p1
323         g z3=p2
324         g z4=p3
325         g z5=p4
326         g land=exp(demog23)
327
328         global zlist "z1 z2 z3 z4 z5"
329         global zlist_indiv "z1"
330         global xzlist_indiv "xz1"
331
332         drop if land==. | demog24==.
333 }
334
335 if short_zlist==7 {
336         global T=11
337
338         g z1=hhsizem1/10
339         g z2=age/3
340         g z3=married
341         g z4=.
342         g land=exp(demog23)
343         replace z4=0 if land<=0.005
344         replace z4=ln(land+1) if land>0.005
345         g z5=demog24
346         g z6=educmed
347         g z7=educhigh
348         g z8=p1
349         g z9=p2
350         g z10=p3
351         g z11=p4
352
```

```
353         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
354         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
355         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
356 }

357
358
359
360 // drop missings and calculate size of grp
361 forv j = 1(1)$T {
    2.         drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

362 bys group_round: egen size_group_round=count(group_round)

363
364 // size of comparision group
365 if "`groupComp_def'"!="" {
366         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
367 }

368
369 // turn things int oper capita terms if ption flagged
370 if $meas_pc==1 {
371         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.         }
372 }

373
374 // normalize expenditure by overall mean
375 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

376 scalar expenditure_mean=r(mean)

377 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.         g `var'_norm=`var'/expenditure_mean
    3. }

378
379 if $use_norm==0 {
380         g pq1=vlux_norm
381         g pq2=ilux_norm
382         g pq3=vnec_norm
383         g pq4=inec_norm
384 }
```

```
385 else {
386         g pq1=food_norm
387         g pq2=fuel_norm
388         g pq3=clothing_norm
389         g pq4=other_nondur_norm
390 }

391 g x=expenditure_norm

392 g x2=x*x

393
394 // make q averages
395 qui bysort group_round: egen weight_group_round=sum(weight)

396
397 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

398
399 // 2 category consumption
400 g PQ1=pq1+pq2

401 g PQ2=pq3+pq4

402 if $use_vis==1 {
403         drop PQ1 PQ2
404         g PQ1=pq1+pq3
405         g PQ2=pq2+pq4
406 }

407 if $use_vislux==1 {
408         drop PQ1 PQ2
409         g PQ1=pq1
410         g PQ2=pq2+pq3+pq4
411 }

412 if $use_norm==1 {
413         drop PQ1 PQ2
414         g PQ1=pq1
415         g PQ2=pq2+pq3+pq4
416         if `J'==3 {
417                 drop PQ1 PQ2
418                 g PQ1=pq1
419                 g PQ2=pq2
420                 g PQ3=pq3+pq4
421         }
422 }

423
424 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
425 if `J'==3 {
426         qui g Q`J'=PQ`J'/P`J'
427         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
428 }

429
430 // only scheduled groups (with some of other group)
431 bys geogroup_seg religion: egen share_sched = mean(scheduled)

432 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

433
434 // update group epend wiht other group
435 if $only_scheduled==2 {
436         preserve
437                 keep if scheduled==0
438                 keep geogroup_seg religion Qbar* qbar*
439                 duplicates drop
440                 tempfile update_exp
441                 save `update_exp'
442         restore
443         keep if scheduled==1 & share_sched>0 & share_sched<1
444         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
445 }

446 drop share_sched

447
448 // laysperes_cpi varies by district and round
449 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

450
451 // make instruments
452 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
453 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
454 // measure all the x-stuff in reals, using aggregate laysperes index.
455 g weight_temp=weight

456 global instlist ""

457 g xreal=x/local_cpi
  (82 missing values generated)

458 g x_temp=x

459 g x2_temp=x2

460 replace x=xreal
  (73,576 real changes made, 82 to missing)

461 replace x2=x*x
  (73,576 real changes made, 82 to missing)

462
463 // make z x interactions
464 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
465
466 levelsof round, local(roundlist)
  59 60 61 62

467 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

468
469 g xbar2lag=xbarlag^2
  (4 missing values generated)

470 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

471
472 bysmeanw x, weight(weight) by(group) rename(xbar)

473 bysmeanw x2, weight(weight) by(group) rename(x2bar)

474 g xbar2=xbar^2

475 g rootxbar = sqrt(xbar)

476 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

477
478 replace x=x_temp
  (73,576 real changes made)

479 replace x2=x2_temp
  (73,576 real changes made)

480
481 // add x terms to inst lsit
482 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

483
484 // only groups of size 3 or more
485 drop if size_group_round<3
  (51,880 observations deleted)

486 if "`groupComp_def'"!="" drop if size_groupComp_round<3

487 drop size_group_round

488 bys group_round: egen size_group_round=count(group_round)
```

```
489
490 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
491 if `J'==3 local extrakeep "Q3 P3 Qbar3"

492 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

493 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

494
495 g obs_numi=_n

496
497 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

498 rename x xi

499
500 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

501 rename weight weighti

502 rename land landi

503 rename owns_land owns_landi

504 rename schednh schednhi

505 rename schedh schedhi

506 rename nschedh nschedhi

507 rename nschednh nschednhi

508 g urbani = urban

509 rename hhid hhidi

510 rename scheduled scheduledi

511 rename hhsize hhsizei

512
513 tempfile obs_i

514 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_41e4_000001.tmp saved as .dta format

515
516 // calculate sumstats at hh level
517 if `make_sumstats'==1 {
518         count
```

```
519         local sumstats_n_obs = r(N)
520
521         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
522                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
523         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
524         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
525 }

526
527 rename obs_numi obs_numk

528
529 forval j=1/4 {
    2.          rename qi`j' qk`j'
    3.          capture rename Qi`j' Qk`j'
    4. }

530 rename xi xk

531
532 forval t=1/$T {
    2.          rename zi`t' zk`t'
    3. }

533 rename weighti weightk

534 rename landi landk

535 rename owns_landi owns_landk

536 rename urbani urbank

537 rename schednhi schednhik

538 rename schedhi schedhk

539 rename nschedhi nschedhk

540 rename nschednhi nschednhk

541 rename scheduledi scheduledk

542 rename hhsizei hhsizek

543 drop geogroup

544
545 // combine into pairs within groups
546 joinby group_round using "`obs_i'"

547 drop if obs_numi==obs_numk
  (21,696 observations deleted)

548
549 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
550 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

551
552 // summary stats tables
553 if `make_sumstats'==1 {
554         count
555         local sumstats_n_obs_pair = r(N)
556
557         cap file close sumstat
558                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
559                 file write sumstat "\begin{table}[htbp]\centering" _n
560                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
561                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
562                 file write sumstat "\begin{tabular}{lcccccccc}" _n
563                 file write sumstat "\toprule" _n
564                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
565                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
566                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
567                 file write sumstat "\midrule" _n
568
569                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
570                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var'" _n
    5.                 }
571
572                 file write sumstat "\bottomrule" _n
573                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
574                 file write sumstat "\end{tabular}" _n
575                 file write sumstat "\end{table}" _n
576                 file close sumstat
577                 BREAK
578 }

579
580 // constant for estimation
581 g one=1

582 egen state_district_round=group(state district urban round)

583 egen state_district=group(state district)

584
```

```
585 // keep only thick round
586 keep if round==61
  (0 observations deleted)

587
588
589
590
591
592
593 // options
594 local output_number="14"

595
596 // dimension of A matrix
597 scalar Afull=0

598 local descA "full"

599 scalar Adiag=1

600 if Adiag==1 local descA "diag"

601 scalar same_spillover=1

602 if same_spillover==1 local descA "same"

603
604 // minimum group size (must be 3 or more)
605 local min_group_size = 3

606 scalar min_group_size=`min_group_size'

607
608 // flag for generic model
609 scalar simple_model=0

610 local simp_name ""

611 if simple_model==1 local simp_name "_simple"

612
613 // interactions with peer effects
614 //global Alist "one"
615 //global Alist "one schednhi schedhi nschednhi"
616 global Alist "one   zi8"

617 //global Alist "one   zi7"
618
619 // turn on or off measurement error correction
620 scalar noMeasError=0

621 local measError_desc ""

622 if noMeasError==1 local measError_desc "_NoMeasError"

623
624 // estimate RE and or FE
625 scalar estimate_RE=1
```

```
626 scalar estimate_FE=1
627
628 // estimation options
629 global w_initial "winit(unadjusted)"
630 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"
631 global wmatrix "wmatrix(robust)"
632 global trace_level ""
633 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"
634 global technique ""
635 global maxiter ""
636 global technique_RE ""
637 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"
638 global derivatives "quickderivatives"
639 scalar init_FE_from_RE=1
640 global clust_var "state_district_round"
641 local clus_name "_clus_Dist"
642
643
644
645
646
647
648
649
650 // xxzx to keep names same
651 local wmatrix_desc ""
652 local sectorName "_urbOnly"
653 local dropZdesc "_dropPre"
654 local desc_struc_v0 ""
655 local inst_desc "_exactInst"
656 local inst_rootp "_noPPXdInst"
657 local weight_desc ""
658 local RE_sq_inst_desc ""
659 local cnstname ""
660 local actual_pqhat_desc ""
661 local expUpname "_updateExp"
```

```
662
663
664 if $rand_grp==0 {
665         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
666         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
667
668         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
669
670         drop if size_group_round<min_group_size
  (0 observations deleted)
671         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
672
673         //do "$CODE/estim program.do"
674         do "$CODE/estim program.do"

675 // This code estimates a QES demand system
676 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
677 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
678 // i indexes observations, k indexes other observations in the group
679 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
680 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
681 //              ensure that all zt are z-scored or otherwise have similar scales
682 // qikbarj are leave-two-out group averages
683 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
684 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
685 // note that qi and qk are actually quantities, and not spending
686 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
687
688 /*****************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *****************************************************************************/
689
690 set seed 339487731

691
692 local J=J

693 local Jm1=`J'-1

694 local T=$T

695 local Tm1=`T'-1
```

```
696 local Tindiv : word count $zlist_indiv

697
698 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
699
700 tempvar clust_weight num_group_round

701 bys group_round: egen `num_group_round'=count(group_round)

702 g double `clust_weight'=size_group_round/`num_group_round'

703 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

704 global GMM_weight "[aweight=`clust_weight']"

705
706 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
707
708 if `J'==2 {
709          replace p1=P1
  (100,756 real changes made)
710          replace p2=P2
  (100,756 real changes made)
711          replace qi1=Qi1
  (100,752 real changes made)
712          replace qk1=Qk1
  (100,752 real changes made)
713          replace qikbar1=Qikbar1
  (100,756 real changes made)
714          replace qi2=Qi2
  (100,756 real changes made)
715          replace qk2=Qk2
  (100,756 real changes made)
716          replace qikbar2=Qikbar2
  (100,756 real changes made)
717
718          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
719 }

720
721 if `J'==3 {
722          forv j = 1(1)3 {
    2.                 replace p`j'=P`j'
    3.                 replace qi`j'=Qi`j'
    4.                 replace qk`j'=Qk`j'
    5.                 replace qikbar`j'=Qikbar`j'
    6.          }
723
724          foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
725 }

726
```

```
727
728 /*****************************************************************************
  > Make moment equations
  > *****************************************************************************/
729
730 global zilist ""

731 global zklist ""

732 forval t=1/`Tm1' {
    2.         global zilist "$zilist zi`t'"
    3.         global zklist "$zklist zk`t'"
    4. }

733 global zilistTm1 "$zilist"

734 global zklistTm1 "$zklist"

735 if `T'>0 global zilist "$zilist zi`T'"

736 if `T'>0 global zklist "$zklist zk`T'"

737
738 // create b_p and pC
739 // remember b_p doesn't vary within group
740 local b_p "exp( 0"

741 forval j=1/`Jm1' {
    2.         local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

742 local b_p "`b_p' + ln(p`J') )"

743 if simple_model==1      local b_p "1"

744 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

745
746 // pCj is the sum of these two, for use in the RE model
747 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
748 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
749 forval j=1/`J' {
    2.         local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

750 local Tindivp1=`Tindiv'+1

751 forval j=1/`J' {
    2.         local pCi`j' "( 0 "
    3.         local pCk`j' "( 0 "
    4.         local pCg`j' "( 0 "
    5.         forval t=1/`Tindiv' {
    6.                 local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                 local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.         }
    9.         forval t=`Tindivp1'/`T' {
   10.                 local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.         }
   12.         local pCi`j' "`pCi`j'' )"
   13.         local pCk`j' "`pCk`j'' )"
   14.         local pCg`j' "`pCg`j'' )"
   15. }
```

```
752
753 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
754 local pC "( 0"

755 local pCi "( 0"

756 local pCk "( 0"

757 local pCg "( 0"

758 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j'"
    3.          local pCi "`pCi' + `pCi`j'"
    4.          local pCk "`pCk' + `pCk`j'"
    5.          local pCg "`pCg' + `pCg`j'"
    6. }

759 local pC "`pC' )"

760 local pCi "`pCi' )"

761 local pCk "`pCk' )"

762 local pCg "`pCg' )"

763
764 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
765 local pC_2nd "( 0"

766 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j'"
    4. }

767 local pC_2nd "`pC_2nd' )"

768
769 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
770 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
771 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
772 local rpDrp "( 0"

773 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
774         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

775
776 local rpDrp "`rpDrp' )"

777 di "`rpDrp'"
  ( 0 )

778
779 if simple_model==1      {
780         local pC "0"
781         local pC1 "0"
782         local pCi "0"
783         local pCk "0"
784         local pC_2nd "0"
785         local rpDrp "0"
786         local Drp1 "0"
787 }

788
789 // create Aq
790 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
791 // k means use qk as the group quantity (for use in RE only)
792 local qtypelist "ikbar k bar"

793 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
794                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
795                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
796         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
797                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
798         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
799
800 // Aikbar is A'qbar-minus-ik; Ak is A'qk
801 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
802 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
803 local xi_hat            "(xi - `pCi')"

804 local xk_hat            "(xk - `pCk')"

805 local x_hat                    "(xi - `pC' - `rpDrp' )"

806 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

807
808 // construct structural vo
809 * let AVA be A'VA, and ensure its diagonals are positive.
810 local v0 "(0 "

811 forval j=1/`J' {
    2.          local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
812         local jplus1=`j'+1
    4.          if `jplus1'<=`J' {
    5.                  forval k=`jplus1'/`J' {
    6.                          local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                  }
    8.          }
    9. }

813
814 local v0 "`v0' )"

815
816 // make all equations, then put pieces together
817
818 * RE equations; first equation has first call to C, other equations use {Cj: } form
819 * note "-vj:", this is because we subtract the structural v0 term from E[q]
820 local j=1

821
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

822 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

823 if `J'>2 {
824         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                  if noMeasError==1                               local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
```

```
825 }

826
827 *       FE equations
828 forval j=1/`Jm1' {
    2.                                                                              l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

829
830 // clean up equations, and combine
831 local eqs_RE ""

832 local eqs_FE ""

833 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

834
835 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7 zi8}) + (p2*{
  > C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7 zi8}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2
  > *qikbar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2
  > *{C2:}) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (
  > p2*{C2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*
  > p1*qikbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}
  > *p2 )*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) )

836 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 + p1*{C18}*zi8 ) + ( p2*{
  > C21}*zi1 + p2*{C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6
  > + p2*{C27}*zi7 + p2*{C28}*zi8 ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C
  > 13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 + p1*{C18}*zk8 )
  >  + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{
  > C26}*zk6 + p2*{C27}*zk7 + p2*{C28}*zk8 ) ))^2) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) )
  >  - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{C15}*z
  > i5 + p1*{C16}*zi6 + p1*{C17}*zi7 + p1*{C18}*zi8 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 +
  > p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 + p2*{C28}*
  > zi8 ) )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*
  > {C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 + p1*{C18}*zk8 ) + ( p2*{C21}*zk1 + p2*{C22}
  > *zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 + p2
  > *{C28}*zk8 ) )))*({b1} - 2*(( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 )
  > + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}
  > *zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 + p1
  > *{C18}*zi8 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15
  > }*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 + p1*{C18}*zk8 ))         ))
```

```
837
838 * replace the first occurrence of Ajj with the Alist
839 if Afull==0 {
840         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
841 }

842 if Afull==1 {
843         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
844 }

845
846 if simple_model==1 {
847         // construct z sums and interactions
848         if `T'>0 {
849
850                 local RE_cz_2            "+ {d}*(0"
851                 local RE_acz     ""
852                 local RE_acz1     ""
853                 local RE_acz2     ""
854                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
855                 local RE_czcz    ""
856                 local FE_czi     ""
857                 local FE_czk     ""
858                 local FE_czg     ""
859                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
860                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
861                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
862                 local RE_cz_2 "`RE_cz_2')^2"
863         }
864
865         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
866
867         local xi_hat "({b}*p1*xi `FE_czi')"
868         local xk_hat "({b}*p1*xk `FE_czk')"
869
870         local eqs_FE
  >                             "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
871 }

872
873 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one   zi8}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikb
  > ar1) + ({A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7 zi8}
  > ) + (p2*{C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7 zi8}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + (
  > {A11:}*p2*qikbar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1
  > :}) + (p2*{C2:}) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{
  > C1:}) + (p2*{C2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} +
  >  ({A11:}*p1*qikbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p
  > 2*{AVA22}*p2 )*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) )

874 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 + p1*{C18}*zi8 ) + ( p2*{
  > C21}*zi1 + p2*{C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6
  > + p2*{C27}*zi7 + p2*{C28}*zi8 ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C
  > 13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 + p1*{C18}*zk8 )
  >  + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{
  > C26}*zk6 + p2*{C27}*zk7 + p2*{C28}*zk8 ) ))^2) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) )
  >  - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{C15}*z
  > i5 + p1*{C16}*zi6 + p1*{C17}*zi7 + p1*{C18}*zi8 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 +
  > p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 + p2*{C28}*
  > zi8 ) )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*
  > {C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 + p1*{C18}*zk8 ) + ( p2*{C21}*zk1 + p2*{C22}
  > *zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 + p2
  > *{C28}*zk8 ) )))*({b1} - 2*(( ({A11:one   zi8}*p1*qikbar1) + ({A11:}*p2*qikbar2) ) +
  >  ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 +
  >  p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}
  > *zi7 + p1*{C18}*zi8 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4
  > + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 + p1*{C18}*zk8 ))         ))

875
876 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
877
878 //make squares and differences of x
879 tempvar x2i x2k ximxk x2imx2k xiP

880 g double `xiP'=xi*local_cpi

881 g double `x2i'=xi*xi

882 g double `x2k'=xi*xi
```

```
883 g double `ximxk'=xi-xk

884 g double `x2imx2k'=xi*xi-xk*xk

885
886 g xbarlag_x2barlag = xbarlag*x2barlag

887 g xbarlag2  = xbarlag^2

888 g x2barlag2 = x2barlag^2

889
890 //make interactions with z and zp
891 forval t=1/`T' {
    2.         tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.         g double `xizi`t''=xi*zi`t'
    4.         g double `xkzk`t''=xk*zk`t'
    5.         g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.         g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.         g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.         g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.         g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.         forval j=1/`J' {
   11.                 tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                 g double `zi`t'p`j''= zi`t'*p`j'
   13.                 g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                 g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                 g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                 forv s=1/`T' {
   17.                         tempvar zi`t'zi`s'p`j'
   18.                         g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                 }
   20.         }
   21. }

892
893 // make group-level instruments, and the instrument lists rg, xrg
894 if noMeasError==0       local rg "xbarlag"

895 if noMeasError==1       local rg "xbar"

896
897 local rg "`rg' x2barlag"

898 local rg "`rg' xbar2lag"

899 local rg "`rg' rootxbarlag"

900
901 local Tindivm1=`Tindiv'-1

902 // lag instruments
903 forval t=1/`Tindiv' {
    2.                 local rg "`rg' z`t'barlag"
    3. }

904
905 local zrg ""
```

```
906 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var'"
    6.          }
    7. }

907
908 local xrg ""

909 local prg ""

910 local xprg ""

911 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var'"
    5.
912         forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var'"
   10.                  local xprg "`xprg' `xp`j'`var'"
   11.          }
   12. }

913
914 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
915 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
916 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
917 local budget "xi `x2i'"

918
919 local rootprootp ""

920 local xrootprootp ""

921 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

922
923 // demographics multiplied by prices and budget
924 local zinsts ""

925 local xzinsts ""
```

```
926 local xPzinsts ""

927 local pzinsts ""

928 local xpzinsts ""

929 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }

930
931 // price insts
932 local roundinsts " "

933 forval j=1/`J' {
    2.          forval k=`j'/`J' {
    3.                  tempvar p`j'p`k'
    4.                  g `p`j'p`k''=p`j'*p`k'
    5.                  local roundinsts "`roundinsts' `p`j'p`k''"
    6.          }
    7. }

934
935 // instruments for all equations are ri:  x,x2,p,z,zx,
936 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
937 local qhat_insts ""

938 local qhat_insts "`qhat_insts' `zrg'"

939 local qhat_insts "`qhat_insts' `prg'"

940 local qhat_insts "`qhat_insts' `rg' "

941
942 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

943
944 local count: word count `r_RE'

945 di "total instruments: `count'"
  total instruments: 21

946
947 local qhat_hats ""

948 local pqhat ""

949 local xpqhat ""

950 local pqhat_sq ""

951 forval j=1/`J' {
    2.          if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.          if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
```

```
952         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
953         // use all prices times all qikhats as instruments, interacted with xi
954         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
955                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }
```

Linear regression

Number of obs	= **100,756**
F(152, 4451)	= .
Prob > F	= .
R-squared	= **0.2591**
Root MSE	= **.13302**

(Std. err. adjusted for **4,452** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__0000BG	**-6.136836**	**8.067614**	**-0.76**	**0.447**	**-21.95337**	**9.679697**
__0000BI	**-1.855359**	**1.320743**	**-1.40**	**0.160**	**-4.444672**	**.7339546**
__0000BL	**-.0073369**	**3.81666**	**-0.00**	**0.998**	**-7.489889**	**7.475215**
__xi	**.1067058**	**.0084043**	**12.70**	**0.000**	**.0902292**	**.1231824**
__000002	**-.0140433**	**.0027596**	**-5.09**	**0.000**	**-.0194536**	**-.0086331**
__00000E	**-.1665548**	**.0963372**	**-1.73**	**0.084**	**-.3554237**	**.022314**
__00000Q	**.0084275**	**.0544122**	**0.15**	**0.877**	**-.0982474**	**.1151025**
__000019	**-.0113268**	**.1900727**	**-0.06**	**0.952**	**-.3839638**	**.3613101**
__00001L	**-.0250594**	**.1137593**	**-0.22**	**0.826**	**-.2480843**	**.1979654**
__000024	**-.0144562**	**.0521368**	**-0.28**	**0.782**	**-.1166703**	**.0877578**
__00002G	**.0003534**	**.0267725**	**0.01**	**0.989**	**-.0521339**	**.0528408**
__00002Z	**.0145157**	**.074461**	**0.19**	**0.845**	**-.1314649**	**.1604962**
__00003B	**-.0068283**	**.0510321**	**-0.13**	**0.894**	**-.1068767**	**.09322**
__00003U	**.0298845**	**.0659481**	**0.45**	**0.650**	**-.0994067**	**.1591756**
__000046	**-.0082441**	**.0388828**	**-0.21**	**0.832**	**-.0844736**	**.0679854**
__00004P	**-.084293**	**.051086**	**-1.65**	**0.099**	**-.1844469**	**.0158609**
__000051	**.0269479**	**.0270206**	**1.00**	**0.319**	**-.026026**	**.0799217**
__00005K	**-.1056481**	**.0657503**	**-1.61**	**0.108**	**-.2345515**	**.0232552**
__00005W	**.012243**	**.0362378**	**0.34**	**0.735**	**-.058801**	**.0832871**
__00006F	**.098297**	**.0830023**	**1.18**	**0.236**	**-.0644289**	**.2610229**
__00006R	**.0233058**	**.0554646**	**0.42**	**0.674**	**-.0854323**	**.132044**
__000073	**-.0497733**	**.449834**	**-0.11**	**0.912**	**-.9316715**	**.832125**
__000074	**-.0468552**	**.0744917**	**-0.63**	**0.529**	**-.1928961**	**.0991856**
__000075	**.0171729**	**.1788025**	**0.10**	**0.923**	**-.333369**	**.3677147**
__000076	**.2559959**	**.462904**	**0.55**	**0.580**	**-.6515261**	**1.163518**
__000077	**-.0485249**	**.1075903**	**-0.45**	**0.652**	**-.2594554**	**.1624056**
__000078	**.1210318**	**.1938611**	**0.62**	**0.532**	**-.2590324**	**.501096**
__000079	**-.0604199**	**.0735811**	**-0.82**	**0.412**	**-.2046755**	**.0838357**
__00007A	**-.0078395**	**.0602914**	**-0.13**	**0.897**	**-.1260406**	**.1103617**
__00007B	**.0780856**	**.036466**	**2.14**	**0.032**	**.0065941**	**.1495771**
__00007C	**-.0530125**	**.0526194**	**-1.01**	**0.314**	**-.1561726**	**.0501477**
__00007D	**.0026351**	**.0645606**	**0.04**	**0.967**	**-.1239358**	**.1292059**
__00007E	**.0080066**	**.0319663**	**0.25**	**0.802**	**-.0546634**	**.0706765**
__00007F	**-.1151403**	**.9523473**	**-0.12**	**0.904**	**-1.982214**	**1.751934**
__00007G	**.1999496**	**.1492668**	**1.34**	**0.180**	**-.0926875**	**.4925866**
__00007H	**-.3023201**	**.3650372**	**-0.83**	**0.408**	**-1.017975**	**.4133343**
__00007I	**.5330902**	**.9484764**	**0.56**	**0.574**	**-1.326395**	**2.392575**
__00007J	**-.177548**	**.2050007**	**-0.87**	**0.386**	**-.5794512**	**.2243552**
__00007K	**-.1636846**	**.3634899**	**-0.45**	**0.653**	**-.8763056**	**.5489363**
__00007L	**-.0354076**	**.1367334**	**-0.26**	**0.796**	**-.3034731**	**.2326578**
__00007M	**-.0176611**	**.124926**	**-0.14**	**0.888**	**-.2625782**	**.227256**
__00007N	**.1023024**	**.0614961**	**1.66**	**0.096**	**-.0182605**	**.2228654**
__00007O	**-.1757269**	**.1150542**	**-1.53**	**0.127**	**-.4012903**	**.0498366**

00007P	.0664773	.1378695	0.48	0.630	-.2038156	.3367701
00007Q	-.1424833	.0613225	-2.32	0.020	-.262706	-.0222606
00007R	-.1939417	.2083463	-0.93	0.352	-.602404	.2145206
00007S	.0089375	.0351046	0.25	0.799	-.059885	.07776
00007T	.0206445	.0806048	0.26	0.798	-.1373809	.1786699
00007U	.2086079	.2330222	0.90	0.371	-.2482313	.6654472
00007V	.0098551	.0543687	0.18	0.856	-.0967346	.1164449
00007W	-.0280922	.1006405	-0.28	0.780	-.2253976	.1692133
00007X	-.0343985	.0388406	-0.89	0.376	-.1105454	.0417483
00007Y	.0216789	.0301999	0.72	0.473	-.037528	.0808857
00007Z	-.0021541	.0155333	-0.14	0.890	-.0326072	.0282989
000080	-.0319446	.0275408	-1.16	0.246	-.0859382	.0220491
000081	.0037119	.0353507	0.11	0.916	-.0655931	.0730168
000082	.0028309	.0160949	0.18	0.860	-.0287232	.0343849
000083	-.9990617	.5260869	-1.90	0.058	-2.030454	.0323301
000084	.0092169	.0646708	0.14	0.887	-.11757	.1360038
000085	.3679074	.1988993	1.85	0.064	-.0220341	.7578489
000086	.635894	.4741752	1.34	0.180	-.293725	1.565513
000087	-.1267095	.0858206	-1.48	0.140	-.2949605	.0415415
000088	.0131861	.165459	0.08	0.936	-.3111957	.337568
000089	.0004237	.0646385	0.01	0.995	-.1262998	.1271472
00008A	.0398661	.0444912	0.90	0.370	-.0473587	.127091
00008B	-.0685959	.0273277	-2.51	0.012	-.1221718	-.01502
00008C	.0180747	.0490452	0.37	0.712	-.0780783	.1142277
00008D	-.0442749	.0552496	-0.80	0.423	-.1525915	.0640417
00008E	.0103481	.0264762	0.39	0.696	-.0415585	.0622547
00008F	-.4036782	.3206962	-1.26	0.208	-1.032402	.2250458
00008G	-.0432754	.0489724	-0.88	0.377	-.1392857	.052735
00008H	.2279436	.1296595	1.76	0.079	-.0262534	.4821406
00008I	.1905252	.3111026	0.61	0.540	-.4193905	.8004409
00008J	.0490754	.0681175	0.72	0.471	-.0844688	.1826196
00008K	-.0821497	.1273337	-0.65	0.519	-.331787	.1674876
00008L	.0153721	.045813	0.34	0.737	-.0744441	.1051882
00008M	-.0256656	.0393047	-0.65	0.514	-.1027222	.0513911
00008N	.0553072	.0224483	2.46	0.014	.0112973	.0993171
00008O	-.0106073	.0338608	-0.31	0.754	-.0769914	.0557767
00008P	-.0740494	.0385052	-1.92	0.055	-.1495387	.0014398
00008Q	.0446726	.0218808	2.04	0.041	.0017754	.0875698
00008R	-.0531045	.2366306	-0.22	0.822	-.5170181	.410809
00008S	-.0222451	.0380838	-0.58	0.559	-.0969083	.0524181
00008T	.0022475	.0925684	0.02	0.981	-.1792325	.1837275
00008U	.2156297	.245697	0.88	0.380	-.2660585	.6973178
00008V	-.0635279	.0549629	-1.16	0.248	-.1712824	.0442266
00008W	-.0180826	.0984224	-0.18	0.854	-.2110396	.1748743
00008X	-.0193708	.0350733	-0.55	0.581	-.088132	.0493904
00008Y	.018025	.0344702	0.52	0.601	-.0495536	.0856037
00008Z	.0171213	.0169995	1.01	0.314	-.0162062	.0504489
000090	-.038565	.0305486	-1.26	0.207	-.0984553	.0213254
000091	.0216117	.0341881	0.63	0.527	-.045414	.0886373
000092	-.0234076	.0164251	-1.43	0.154	-.055609	.0087939
000093	-.2217099	.3573987	-0.62	0.535	-.9223889	.4789692
000094	.0485284	.0499986	0.97	0.332	-.0494938	.1465505
000095	-.0308868	.1368235	-0.23	0.821	-.2991288	.2373553
000096	.3759638	.3350545	1.12	0.262	-.2809096	1.032837
000097	.0010442	.0709785	0.01	0.988	-.1381089	.1401972
000098	-.101638	.1263513	-0.80	0.421	-.3493493	.1460733
000099	-.0139127	.0433864	-0.32	0.748	-.0989716	.0711461
00009A	-.0211637	.039961	-0.53	0.596	-.0995071	.0571797
00009B	-.008247	.0206408	-0.40	0.690	-.0487132	.0322193
00009C	-.0187937	.0374697	-0.50	0.616	-.092253	.0546655
00009D	.026675	.0448928	0.59	0.552	-.0613373	.1146873
00009E	-.0418668	.020554	-2.04	0.042	-.0821629	-.0015706
00009F	.9412584	.5683065	1.66	0.098	-.172905	2.055422
00009G	.0601653	.0598564	1.01	0.315	-.057183	.1775137
00009H	-.4162995	.1989321	-2.09	0.036	-.8063052	-.0262938
00009I	-.6089851	.5353762	-1.14	0.255	-1.658589	.4406183
00009J	.0277464	.0895201	0.31	0.757	-.1477574	.2032502
00009K	-.35787	.1670087	-2.14	0.032	-.6852901	-.0304499
00009L	.079479	.0744874	1.07	0.286	-.0665534	.2255113
00009M	-.0953507	.0597349	-1.60	0.111	-.2124609	.0217594
00009N	.0353377	.0290582	1.22	0.224	-.0216307	.0923062
00009O	-.0748672	.0488778	-1.53	0.126	-.170692	.0209575

__00009P	-.0539812	.0560711	-0.96	0.336	-.1639084	.055946
__00009Q	.1031755	.0283907	3.63	0.000	.0475156	.1588354
__00009S	-18.92658	19.16979	-0.99	0.324	-56.5089	18.65574
__00009U	1.522652	8.5012	0.18	0.858	-15.14393	18.18923
__00009X	-.3198875	.4174922	-0.77	0.444	-1.13838	.4986047
__00009Z	.52503	.3005386	1.75	0.081	-.0641751	1.114235
__0000A2	4.183952	3.772161	1.11	0.267	-3.211359	11.57926
__0000A4	-1.676875	1.665902	-1.01	0.314	-4.942871	1.589121
__0000A7	22.34515	23.50896	0.95	0.342	-23.7441	68.43439
__0000A9	1.342112	10.85273	0.12	0.902	-19.93463	22.61886
__0000AC	.2454641	.5226476	0.47	0.639	-.779185	1.270113
__0000AE	-.9014749	.4800358	-1.88	0.060	-1.842584	.0396339
__0000AH	-1.340179	1.014877	-1.32	0.187	-3.329841	.6494839
__0000AJ	-1.326274	.7158595	-1.85	0.064	-2.729714	.0771667
__0000AM	-.1441877	.3541825	-0.41	0.684	-.8385614	.5501861
__0000AO	.2887853	.265653	1.09	0.277	-.2320266	.8095971
__0000AR	.8651448	.2528776	3.42	0.001	.369379	1.360911
__0000AT	-.4304691	.2009747	-2.14	0.032	-.8244794	-.0364587
__0000AW	-.0490779	.1888419	-0.26	0.795	-.4193018	.321146
__0000AY	-.07464	.1536498	-0.49	0.627	-.37587	.22659
__0000B1	.2551805	.3042637	0.84	0.402	-.3413277	.8516887
__0000B3	.0196418	.2199793	0.09	0.929	-.411627	.4509106
__0000B6	.2152593	.337461	0.64	0.524	-.446332	.8768505
__0000B8	-.6546302	.2474436	-2.65	0.008	-1.139743	-.1695177
__0000BB	.0991809	.1733516	0.57	0.567	-.2406745	.4390362
__0000BD	-.0487279	.121826	-0.40	0.689	-.2875673	.1901116
xbarlag	15.80143	23.04703	0.69	0.493	-29.38219	60.98506
x2barlag	-.5528668	.5278777	-1.05	0.295	-1.587769	.4820358
xbar2lag	-1.457282	4.648527	-0.31	0.754	-10.5707	7.656141
rootxbarlag	-22.52228	28.09869	-0.80	0.423	-77.60968	32.56511
z1barlag	.8751936	.8228524	1.06	0.288	-.7380061	2.488393
z2barlag	2.850403	1.30922	2.18	0.030	.2836802	5.417125
z3barlag	-.2166715	.510584	-0.42	0.671	-1.21767	.7843269
z4barlag	-.3498766	.2992619	-1.17	0.242	-.9365788	.2368255
z5barlag	.0228806	.2125022	0.11	0.914	-.3937294	.4394906
z6barlag	-.1837059	.427467	-0.43	0.667	-1.021754	.6543419
z7barlag	.4362906	.4383178	1.00	0.320	-.4230301	1.295611
z8barlag	-.0531232	.2204126	-0.24	0.810	-.4852414	.3789951
_cons	8.014652	9.497281	0.84	0.399	-10.60474	26.63404

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000BQ	100,756	.4481204	.0786038	.1870056	.9076644

Linear regression

Number of obs	=	100,756
F(153, 4451)	=	26.90
Prob > F	=	0.0000
R-squared	=	0.3380
Root MSE	=	.20521

(Std. err. adjusted for **4,452** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__0000BG	5.951484	10.06462	0.59	0.554	-13.78017	25.68314
__0000BI	-.668439	2.255213	-0.30	0.767	-5.089777	3.752899
__0000BL	1.290508	5.117684	0.25	0.801	-8.742696	11.32371
xi	.1477566	.0156952	9.41	0.000	.1169862	.178527
__000002	-.006841	.0061101	-1.12	0.263	-.0188198	.0051377
__00000E	-.1722349	.1375278	-1.25	0.211	-.4418578	.097388
__00000Q	-.0731327	.079271	-0.92	0.356	-.2285433	.0822778
__000019	.008807	.2888398	0.03	0.976	-.5574625	.5750766
__00001L	.1659763	.1733402	0.96	0.338	-.1738567	.5058093
__000024	.0139235	.0766284	0.18	0.856	-.1363063	.1641533
__00002G	.0446813	.0402402	1.11	0.267	-.0342096	.1235722
__00002Z	.3361824	.1082401	3.11	0.002	.1239779	.5483869
__00003B	-.0512038	.0670863	-0.76	0.445	-.1827263	.0803187
__00003U	-.0771703	.0849077	-0.91	0.363	-.2436317	.0892911
__000046	.0358288	.0500018	0.72	0.474	-.0621996	.1338571

__00004P	-.1314429	.067829	-1.94	0.053	-.2644215	.0015358
__000051	.0021734	.0375178	0.06	0.954	-.0713801	.075727
__00005K	-.2649898	.0989887	-2.68	0.007	-.4590568	-.0709228
__00005W	.0906325	.0613411	1.48	0.140	-.0296266	.2108916
__00006F	.343618	.1261149	2.72	0.006	.0963702	.5908658
__00006R	.0289219	.0902707	0.32	0.749	-.1480537	.2058974
__000073	-.9334369	.678209	-1.38	0.169	-2.263064	.3961899
__000074	-.0309087	.1019455	-0.30	0.762	-.2307726	.1689552
__000075	.2226404	.2631187	0.85	0.398	-.2932029	.7384838
__000076	.9620468	.6852331	1.40	0.160	-.3813507	2.305444
__000077	-.0308738	.1567928	-0.20	0.844	-.3382657	.2765181
__000078	.1693812	.2413129	0.70	0.483	-.3037121	.6424744
__000079	-.1694104	.1232166	-1.37	0.169	-.4109763	.0721555
__00007A	-.0363648	.0711282	-0.51	0.609	-.1758114	.1030817
__00007B	-.0215987	.0501256	-0.43	0.667	-.1198697	.0766724
__00007C	-.0017668	.0652616	-0.03	0.978	-.1297119	.1261783
__00007D	-.0333165	.0825179	-0.40	0.686	-.1950926	.1284597
__00007E	.0666934	.0459309	1.45	0.147	-.023354	.1567408
__00007F	4.510829	1.392276	3.24	0.001	1.781276	7.240381
__00007G	.3620735	.2285089	1.58	0.113	-.0859176	.8100646
__00007H	-1.925942	.5664682	-3.40	0.001	-3.036502	-.815383
__00007I	-2.924657	1.404968	-2.08	0.037	-5.679093	-.1702208
__00007J	-.5202377	.3155474	-1.65	0.099	-1.138867	.0983921
__00007K	.7150412	.5573558	1.28	0.200	-.3776532	1.807735
__00007L	-.0318103	.2001754	-0.16	0.874	-.4242537	.3606331
__00007M	-.4673206	.1510317	-3.09	0.002	-.7634177	-.1712234
__00007N	.2137997	.0985872	2.17	0.030	.0205197	.4070797
__00007O	-.1247982	.1654777	-0.75	0.451	-.4492167	.1996204
__00007P	.1243075	.2022549	0.61	0.539	-.2722126	.5208276
__00007Q	-.2577916	.093149	-2.77	0.006	-.44041	-.0751732
__00007R	.3120928	.341623	0.91	0.361	-.3576581	.9818436
__00007S	-.0034971	.0515272	-0.07	0.946	-.104516	.0975218
__00007T	-.1026725	.1335291	-0.77	0.442	-.364456	.159111
__00007U	-.2863888	.3525564	-0.81	0.417	-.9775747	.404797
__00007V	-.0145882	.0840067	-0.17	0.862	-.1792831	.1501067
__00007W	.0302022	.1401237	0.22	0.829	-.2445099	.3049144
__00007X	.0127266	.068249	0.19	0.852	-.1210754	.1465286
__00007Y	.011615	.0354925	0.33	0.743	-.0579679	.0811978
__00007Z	.0192594	.0243518	0.79	0.429	-.0284823	.0670011
__000080	-.0520659	.0367332	-1.42	0.156	-.1240813	.0199495
__000081	.0074844	.0458301	0.16	0.870	-.0823653	.0973341
__000082	-.0092692	.0250804	-0.37	0.712	-.0584393	.039901
__000083	-.6187548	.6460129	-0.96	0.338	-1.885261	.6477517
__000084	-.0112214	.085224	-0.13	0.895	-.1783029	.15586
__000085	.3525767	.2320116	1.52	0.129	-.1022813	.8074348
__000086	-.1801003	.5851383	-0.31	0.758	-1.327262	.9670617
__000087	.1227549	.1270836	0.97	0.334	-.1263922	.371902
__000088	.1560992	.2096002	0.74	0.456	-.2548215	.5670199
__000089	.0646371	.0879419	0.73	0.462	-.1077727	.2370469
__00008A	.0633778	.0580976	1.09	0.275	-.0505224	.177278
__00008B	-.1364833	.0372561	-3.66	0.000	-.2095238	-.0634429
__00008C	-.0783793	.0598937	-1.31	0.191	-.1958007	.0390421
__00008D	-.0575332	.0742469	-0.77	0.438	-.203094	.0880275
__00008E	.0254045	.0380941	0.67	0.505	-.0492789	.100088
__00008F	-.6177178	.369632	-1.67	0.095	-1.34238	.1069446
__00008G	-.0317726	.0673616	-0.47	0.637	-.1638349	.1002897
__00008H	.2688117	.1644492	1.63	0.102	-.0535905	.5912138
__00008I	.2100456	.3698826	0.57	0.570	-.5151082	.9351995
__00008J	.2603779	.0915817	2.84	0.004	.0808323	.4399236
__00008K	.0164434	.1478299	0.11	0.911	-.2733767	.3062636
__00008L	-.0003342	.0584807	-0.01	0.995	-.1149855	.1143171
__00008M	.0033843	.0414695	0.08	0.935	-.0779166	.0846851
__00008N	.0976856	.0290578	3.36	0.001	.0407179	.1546533
__00008O	.036825	.0414185	0.89	0.374	-.0443759	.1180259
__00008P	.0241119	.0499587	0.48	0.629	-.0738321	.1220559
__00008Q	.093957	.0266353	3.53	0.000	.0417387	.1461753
__00008R	.0289838	.3065858	0.09	0.925	-.5720767	.6300443
__00008S	.0134996	.0506631	0.27	0.790	-.0858252	.1128245
__00008T	-.0568547	.1261228	-0.45	0.652	-.3041181	.1904087
__00008U	.2188436	.299767	0.73	0.465	-.3688487	.806536
__00008V	-.1466508	.068239	-2.15	0.032	-.2804331	-.0128685
__00008W	.1108966	.1205566	0.92	0.358	-.1254542	.3472474

__00008X	.0182658	.0433641	0.42	0.674	-.0667494	.103281
__00008Y	.0265336	.0382644	0.69	0.488	-.0484836	.1015509
__00008Z	.0164918	.0236746	0.70	0.486	-.0299221	.0629057
__000090	.020586	.0356514	0.58	0.564	-.0493084	.0904804
__000091	-.0047838	.0427423	-0.11	0.911	-.0885799	.0790123
__000092	-.0829605	.0210101	-3.95	0.000	-.1241507	-.0417702
__000093	-.4094298	.4696249	-0.87	0.383	-1.330128	.5112683
__000094	.1192809	.0783881	1.52	0.128	-.0343987	.2729606
__000095	-.0880112	.1921223	-0.46	0.647	-.4646664	.288644
__000096	.7191293	.4843677	1.48	0.138	-.2304721	1.668731
__000097	-.1575725	.1068288	-1.48	0.140	-.36701	.0518649
__000098	.0486714	.1805269	0.27	0.787	-.3052511	.4025939
__000099	.0163231	.0667358	0.24	0.807	-.1145122	.1471584
__00009A	-.0097641	.0512516	-0.19	0.849	-.1102427	.0907145
__00009B	.0211698	.0333373	0.64	0.525	-.0441878	.0865275
__00009C	.0299615	.0519262	0.58	0.564	-.0718397	.1317627
__00009D	.0421154	.0629107	0.67	0.503	-.0812208	.1654515
__00009E	-.231622	.0328905	-7.04	0.000	-.2961038	-.1671403
__00009F	1.392713	.6843995	2.03	0.042	.0509498	2.734476
__00009G	.3063354	.103631	2.96	0.003	.1031671	.5095038
__00009H	-.7879098	.2599709	-3.03	0.002	-1.297582	-.2782377
__00009I	-1.290035	.6443401	-2.00	0.045	-2.553262	-.0268078
__00009J	.1245667	.126381	0.99	0.324	-.1232029	.3723362
__00009K	-.1676645	.2185815	-0.77	0.443	-.596193	.260864
__00009L	-.0242079	.0903493	-0.27	0.789	-.2013374	.1529215
__00009M	-.0516026	.0652663	-0.79	0.429	-.1795569	.0763518
__00009N	.0427224	.0410102	1.04	0.298	-.0376779	.1231228
__00009O	-.1030896	.063182	-1.63	0.103	-.2269577	.0207784
__00009P	.0562261	.0737912	0.76	0.446	-.0884413	.2008935
__00009Q	.0866804	.0393798	2.20	0.028	.0094763	.1638844
__00009S	10.3489	23.34413	0.44	0.658	-35.41721	56.115
__00009U	3.070541	11.62813	0.26	0.792	-19.72637	25.86745
__00009X	.5036958	.551354	0.91	0.361	-.5772321	1.584624
__00009Z	.6429531	.3818525	1.68	0.092	-.1056675	1.391574
__0000A2	-2.467956	4.518966	-0.55	0.585	-11.32738	6.391464
__0000A4	-1.42847	2.261506	-0.63	0.528	-5.862146	3.005207
__0000A7	-13.95919	28.83248	-0.48	0.628	-70.48517	42.5668
__0000A9	-3.131204	14.65646	-0.21	0.831	-31.86515	25.60275
__0000AC	.1632369	.7070409	0.23	0.817	-1.222915	1.549388
__0000AE	-1.001614	.681084	-1.47	0.141	-2.336877	.3336495
__0000AH	-.3093675	1.175594	-0.26	0.792	-2.614116	1.995381
__0000AJ	-1.620338	.9876035	-1.64	0.101	-3.556531	.3158562
__0000AM	-.653672	.4804686	-1.36	0.174	-1.595629	.2882854
__0000AO	.4930217	.389469	1.27	0.206	-.2705312	1.256575
__0000AR	.686208	.280469	2.45	0.014	.1363493	1.236067
__0000AT	-.3679982	.2432242	-1.51	0.130	-.8448385	.1088421
__0000AW	-.8216576	.2719544	-3.02	0.003	-1.354823	-.2884919
__0000AY	-.076271	.2205878	-0.35	0.730	-.5087327	.3561906
__0000B1	1.208734	.3418656	3.54	0.000	.5385077	1.878961
__0000B3	.1541703	.2939959	0.52	0.600	-.4222079	.7305484
__0000B6	.2898039	.3948855	0.73	0.463	-.484368	1.063976
__0000B8	-.34282	.3487392	-0.98	0.326	-1.026522	.3408822
__0000BB	-.0921309	.2260317	-0.41	0.684	-.5352654	.3510036
__0000BD	.2020962	.1774955	1.14	0.255	-.1458831	.5500756
xbarlag	-15.12295	26.44198	-0.57	0.567	-66.96238	36.71649
x2barlag	-1.673219	.6244478	-2.68	0.007	-2.897447	-.4489909
xbar2lag	5.364017	5.250167	1.02	0.307	-4.92892	15.65695
rootxbarlag	17.96884	32.40327	0.55	0.579	-45.55768	81.49536
z1barlag	1.013972	1.055821	0.96	0.337	-1.055963	3.083907
z2barlag	1.519529	1.618294	0.94	0.348	-1.653131	4.692189
z3barlag	.1649241	.642994	0.26	0.798	-1.095664	1.425512
z4barlag	-.0489663	.3140101	-0.16	0.876	-.6645822	.5666495
z5barlag	.8661587	.2777918	3.12	0.002	.3215487	1.410769
z6barlag	-1.269362	.5291088	-2.40	0.016	-2.306679	-.232046
z7barlag	-.0219233	.5580507	-0.04	0.969	-1.11598	1.072134
z8barlag	-.1370138	.2901937	-0.47	0.637	-.7059377	.4319101
_cons	-6.233623	11.09202	-0.56	0.574	-27.9795	15.51225

(option **xb** assumed; fitted values)

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000BZ |    100,756    .4496933    .1465197   .0488622   1.209362
```

```
956
957 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"
958
959 * interact qhat_hats with Alist
960 di "`qhat_hats'"
    __0000BT __0000BW __0000C2 __0000C5  __0000BU __0000BX __0000C3 __0000C6
961 if "$Alist"~="one" {
962         local qhat_hats_backup "`qhat_hats'"
963         local qhat_hats ""
964         foreach var1 of local qhat_hats_backup {
    2.                 foreach var2 of global Alist {
    3.                         g `var1'_`var2'=`var1'*`var2'
    4.                         sum `var1' `var2' `var1'_`var2'
    5.                         local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                 }
    7.         }
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000BT |    100,756    .4886646    .1019256   .1918466   1.011302
         one |    100,756           1           0          1          1
__0000BT_one |    100,756    .4886646    .1019256   .1918466   1.011302
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000BT |    100,756    .4886646    .1019256   .1918466   1.011302
         zi8 |    100,756     .755032    .4300704          0          1
__0000BT_zi8 |    100,756    .3785235    .2319327          0   1.011302
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000BW |    100,756    .5388173    .1195423    .183938   1.225449
         one |    100,756           1           0          1          1
__0000BW_one |    100,756    .5388173    .1195423    .183938   1.225449
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000BW |    100,756    .5388173    .1195423    .183938   1.225449
         zi8 |    100,756     .755032    .4300704          0          1
__0000BW_zi8 |    100,756    .4147684    .2559531          0   1.225449
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000C2 |    100,756    .4941648    .1795649   .0472068   1.301779
         one |    100,756           1           0          1          1
__0000C2_one |    100,756    .4941648    .1795649   .0472068    1.30178
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000C2 |    100,756    .4941648    .1795649   .0472068   1.301779
         zi8 |    100,756     .755032    .4300704          0          1
__0000C2_zi8 |    100,756    .3805831    .2630012          0    1.30178
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000C5 |    100,756    .5395445    .1928733   .0535019   1.632775
         one |    100,756           1           0          1          1
__0000C5_one |    100,756    .5395445    .1928733   .0535019   1.632775
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000C5 |    100,756    .5395445    .1928733   .0535019   1.632775
         zi8 |    100,756     .755032    .4300704          0          1
__0000C5_zi8 |    100,756    .4120875    .2810489          0   1.632775
```

Variable	Obs	Mean	Std. dev.	Min	Max
__0000BU	100,756	.5372392	.409914	.0211592	3.52272
one	100,756	1	0	1	1
__0000BU_one	100,756	.5372392	.409914	.0211592	3.52272

Variable	Obs	Mean	Std. dev.	Min	Max
__0000BU	100,756	.5372392	.409914	.0211592	3.52272
zi8	100,756	.755032	.4300704	0	1
__0000BU_zi8	100,756	.420579	.4287098	0	3.52272

Variable	Obs	Mean	Std. dev.	Min	Max
__0000BX	100,756	.5914551	.4571351	.0220332	4.436599
one	100,756	1	0	1	1
__0000BX_one	100,756	.5914551	.4571351	.0220332	4.436599

Variable	Obs	Mean	Std. dev.	Min	Max
__0000BX	100,756	.5914551	.4571351	.0220332	4.436599
zi8	100,756	.755032	.4300704	0	1
__0000BX_zi8	100,756	.459418	.4710957	0	4.436599

Variable	Obs	Mean	Std. dev.	Min	Max
__0000C3	100,756	.5775047	.5462573	.0113699	5.883261
one	100,756	1	0	1	1
__0000C3_one	100,756	.5775047	.5462573	.0113699	5.883261

Variable	Obs	Mean	Std. dev.	Min	Max
__0000C3	100,756	.5775047	.5462573	.0113699	5.883261
zi8	100,756	.755032	.4300704	0	1
__0000C3_zi8	100,756	.4462475	.5263852	0	5.646247

Variable	Obs	Mean	Std. dev.	Min	Max
__0000C6	100,756	.6314424	.6078898	.0140829	7.365447
one	100,756	1	0	1	1
__0000C6_one	100,756	.6314424	.6078898	.0140829	7.365447

Variable	Obs	Mean	Std. dev.	Min	Max
__0000C6	100,756	.6314424	.6078898	.0140829	7.365447
zi8	100,756	.755032	.4300704	0	1
__0000C6_zi8	100,756	.4832137	.5753456	0	7.068721

```
965 }

966
967 local r_RE      "`r_RE' `qhat_hats'"

968
969 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
970
971 if simple_model==1      {
972         local r_RE "xbarlag `xixbarlag' `budget'"
973         local r_RE1 ""
```

```
974
975         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
976
977         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
978 }

979
980 if `J'==2 & simple_model==0     {
981         g y = p1*qi1
982         g ybar_q1 = p1*qikbar1
983         g ybar_q2 = p2*qikbar2
984         g ybar = p1*qikbar1 + p2*qikbar2
985         g ybar2 = ybar^2
986         g ybar_q1_2 = ybar_q1^2
987         g ybar_q2_2 = ybar_q2^2
988         g ybar_q1_q2 = ybar_q1*ybar_q2
989         g ybarx = ybar*xi
990         g ybarx_1 = ybar_q1*xi
991         g ybarx_2 = ybar_q2*xi
992         g xi2 = xi^2
993
994         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
995
996         if same_spillover==1 {
997                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS	Number of obs	=	100,756
				F(14, 100741)	=	40651.53
Model	4624.65549	14	330.332535	Prob > F	=	0.0000
Residual	818.616882	100,741	.008125955	R-squared	=	0.8496
				Adj R-squared	=	0.8496
Total	5443.27237	100,755	.054024836	Root MSE	=	.09014

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	.0156383	.0010278	15.22	0.000	.0136239	.0176527
ybarx	.0065851	.0011903	5.53	0.000	.004252	.0089182
xi2	-.0400426	.0006612	-60.57	0.000	-.0413384	-.0387467
ybar	-.1905239	.0029227	-65.19	0.000	-.1962524	-.1847954
xi	.4171471	.0018919	220.49	0.000	.413439	.4208552
ybar_q1	.4161366	.0035201	118.22	0.000	.4092372	.4230359
pzi1	.2300041	.0014979	153.55	0.000	.2270683	.23294
pzi2	.0136424	.0024858	5.49	0.000	.0087702	.0185146
pzi3	-.001903	.0007557	-2.52	0.012	-.0033841	-.000422
pzi4	.0105911	.0007648	13.85	0.000	.0090921	.0120902
pzi5	-.0146319	.0008205	-17.83	0.000	-.0162401	-.0130236
pzi6	.002414	.000675	3.58	0.000	.001091	.0037371
pzi7	-.0041455	.0008694	-4.77	0.000	-.0058494	-.0024415
pzi8	.0069854	.0006433	10.86	0.000	.0057246	.0082462
_cons	-.02647	.0018333	-14.44	0.000	-.0300634	-.0228767

```
998                 local b = _b[xi]
999                 local a = _b[ybar_q1]
1000                if same_spillover==1 {
```

```
1001                        local a = logit(_b[ybar_q1]/2 + .5)
1002                        if `a'==. local a = .5
1003                }
1004                local d = _b[xi2]
1005                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1006        }
1007        else {
1008                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1009                local b = _b[xi]
1010                local a1 = _b[ybar_q1]/(1-`b')
1011                local a2 = _b[ybar_q2]/`b'
1012                if same_spillover==1 {
1013                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1014                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1015
1016                        if `a1'==. local a1 = .5
1017                        if `a2'==. local a2 = .5
1018                }
1019                local d = _b[xi2]
1020
1021                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1022        }
1023
1024        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1025        forv j = 1(1)`T' {
    2.                local coef = _b[pzi`j']
    3.                local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1026}

1027
1028if `J'==3 & simple_model==0
  >                         local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1030if `J'==4 & simple_model==0
  >                         local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1032
1033if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1034
1035if `J'==2 & estimate_RE==1 {
1036
1037        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00523085
  Iteration 1:  GMM criterion Q(b) =  .00452293
  Iteration 2:  GMM criterion Q(b) =  .00180291
  Iteration 3:  GMM criterion Q(b) =  .00013367
  Iteration 4:  GMM criterion Q(b) =  .00013264

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00635676
  Iteration 1:  GMM criterion Q(b) =  .00521786
  Iteration 2:  GMM criterion Q(b) =  .00516293
```

GMM estimation

Number of parameters = **23**
Number of moments = **38**
Initial weight matrix: **Unadjusted** Number of obs = **100,756**
GMM weight matrix: **Robust**

(Std. err. adjusted for **562** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
A11						
one	.5997267	.0565394	10.61	0.000	.4889115	.7105418
zi8	.0313067	.0539903	0.58	0.562	-.0745124	.1371258
C1						
zi1	.4859712	.0462101	10.52	0.000	.3954012	.5765413
zi2	-.2240093	.0844086	-2.65	0.008	-.3894471	-.0585715
zi3	.0092745	.024226	0.38	0.702	-.0382075	.0567566
zi4	.0700128	.0284179	2.46	0.014	.0143148	.1257109
zi5	.0522695	.0250307	2.09	0.037	.0032102	.1013288
zi6	.0106846	.0192342	0.56	0.579	-.0270137	.0483829
zi7	-.0061114	.0237643	-0.26	0.797	-.0526886	.0404659
zi8	.008244	.0173261	0.48	0.634	-.0257146	.0422026
C2						
zi1	.2744366	.0727377	3.77	0.000	.1318733	.4169998
zi2	-.4161143	.1386717	-3.00	0.003	-.6879058	-.1443229
zi3	.017851	.0386764	0.46	0.644	-.0579532	.0936553
zi4	.1157418	.0467464	2.48	0.013	.0241206	.2073629
zi5	.0999585	.032205	3.10	0.002	.0368377	.1630792
zi6	.0053367	.0286845	0.19	0.852	-.0508839	.0615572
zi7	-.0143508	.0398411	-0.36	0.719	-.0924379	.0637364
zi8	.013247	.0195049	0.68	0.497	-.0249819	.0514759
/d1	-.0873402	.007877	-11.09	0.000	-.1027787	-.0719016
/b1	.3923849	.0222033	17.67	0.000	.3488673	.4359026
/AVA11	.6999286	.6346878	1.10	0.270	-.5440366	1.943894
/AVA12	-.683515	.5488444	-1.25	0.213	-1.75923	.3922002
/AVA22	.3113742	.5164636	0.60	0.547	-.7008758	1.323624

Instruments for equation **eq1_RE**: **__0000BG __0000BI __0000BL xi __000002 __00000E __00000Q __000019 __00001L __000024 __00002G __00002Z __00003B __00003U __000046 __00004P __000051 __00005K __00005W __00006F __00006R __0000BT_one __0000BT_zi8 __0000BW_one __0000BW_zi8 __0000C2_one __0000C2_zi8 __0000C5_one __0000C5_zi8 __0000BU_one __0000BU_zi8 __0000BX_one __0000BX_zi8 __0000C3_one __0000C3_zi8 __0000C6_one __0000C6_zi8 _cons**

```
1038
1039        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1040        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1041
1042        if simple_model==1 {
1043                // calculate derivative and save
1044                local RE_acz_drv ""
1045                tempvar junk
1046                g `junk'=.
1047                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
```

```
1048                replace `junk'=p1*qikbar1
1049                su `junk' $GMM_weight
1050                local mean_y = r(mean)
1051                su xi $GMM_weight
1052                local mean_x = r(mean)
1053
1054                local c_term ""
1055
1056                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1057                estimates store gmm_est
1058                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1059                estimates save "${filename_RE}_dF", replace
1060                estimates restore gmm_est
1061        }
1062}

1063
1064if `J'==3 & estimate_RE==1 {
1065        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1066}

1067if `J'==4 & estimate_RE==1 {
1068        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1069}

1070esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXownsland_drop3_all
      > _exactInst_noPPXdInst_14_zlist3_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not
      found)
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXownsl
  > and_drop3_all_exactInst_noPPXdInst_14_zlist3_updateExp_urbOnly_clus_Dist_dropPre_RE.
  > tex)

1071
1072
1073// save coefs
1074matrix est_b=e(b)

1075unique hhidi
  Number of unique values of hhidi is  21696
  Number of records is  100756

1076local N_hh = r(sum)

1077estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  21696
```

```
1078unique group_round
  Number of unique values of group_round is  4452
  Number of records is  100756

1079estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  4452

1080estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
         e(avg_grp_size) =  4.8733154

1081if estimate_RE==1 {
1082        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXownsland_dro
  > p3_all_exactInst_noPPXdInst_14_zlist3_updateExp_urbOnly_clus_Dist_dropPre_RE.ster no
  > t found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXownsland_drop3_all_
  > exactInst_noPPXdInst_14_zlist3_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1083        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1084}

1085
1086estimates store temp1_FE

1087
1088/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1089
1090if simple_model==0 {
1091
1092        // make FE instruments
1093
1094        // denoting ximxk as x and zimzk of individual zs as zi:
1095        //   x,x2,p,zi,px,zi,pzi
1096        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1097        local xdiff "`ximxk' "
1098        local xdiffxdiff ""
1099        local zindivdiff ""
1100        local pzindivdiff ""
1101        local pzindivdiffpqhat ""
1102        local pxzindivdiff ""
1103        local p2zindiv2diff ""
1104        local zg ""
1105        local pzg ""
1106        local zindivdiffxdiff ""
1107        local pzindivdiffxdiff ""
1108        local pzindivdiffp ""
1109        local pzindivdiffpzg ""
1110        local zgxdiff ""
1111        local pzgxdiff ""
1112        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1113        g double `ximxkxdiff'=`ximxk'*`ximxk'
1114        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1115        g double `xixkxdiff'=xi*xk*`ximxk'
1116
```

```
1117        local xdiffxdiff "`x2imx2k'"
1118
1119        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
1120                                                 gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                  local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                          }
   30.                                  }
   31.                          }
   32.                  }
   33.          }
1121        if `Tindivp1'<=`T' {
1122                forval t=`Tindivp1'/`T' {
    2.                          tempvar zi`t'xdiff
    3.                          g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                          local zg "`zg' zi`t'"
    5.                          forval j=1/`J' {
    6.                                  tempvar zi`t'p`j'xdiff
    7.                                  g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                  local pzg "`pzg' `zi`t'p`j''"
    9.                                  local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                          }
   11.                  }
1123        }
```

```
1124
1125        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1126        // denoting zg as group-zs
1127        //  zg,pzgx,rprp,rprpx
1128        local rootprootp ""
1129        local rootprootpxdiff ""
1130
1131        forval j=1/`J' {
    2.                  forval s=`j'/`J' {
    3.                          tempvar xdiffrp`j'rp`s'
    4.                          g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1132                        if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1133                        local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                  }
    8.          }
1134
1135        local pqhatxdiff ""
1136        forval j=1/`J' {
    2.                  forval s=1/`J' {
    3.                          foreach name of global Alist {
    4.                                  tempvar   p`s'qikhat`j'xdiff`name'
    5.                                  g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
1137                                local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
1138
1139        // set instruments
1140        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1141}

1142
1143
1144if simple_model==1 {
1145        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1146
1147        /******************
  >         Initial values
  >         ******************/
1148
1149        if init_FE_from_RE==1 {
1150                estimates use "$filename_RE"
1151
1152                // clear init values
1153                local initial_values_FE ""
1154
1155                // extract vector of coefs and paramter names from
1156                matrix coefs = e(b)
1157                local paramlist = e(params)
```

```
1158
1159                // iterate through paramter name list, taking
1160                local t=0
1161                foreach p of local paramlist {
    2.                          local `++t'
    3.                          local p_nice = subinstr("`p'",":_cons","",.)
    4.                          local est = coefs[1,`t']
    5.                          if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                  }
1162        }
1163
1164        // make z diff instruments
1165        local pzindivdiff ""
1166        local pz2indivdiff ""
1167        local pzXzindivdiff ""
1168        local pxzindivdiff ""
1169
1170        forv i=1/`T' {
    2.
1171                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1172                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1173                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1174                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                 }
   16.         }
1175
1176        // p z interactions
1177        local pzg ""
1178        local pzgxdiff ""
1179        if `Tindivp1'<=`T' {
1180                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1181        }
1182
1183        tempvar p1ximxk  p1x2imx2k
1184        g `p1ximxk' = p1*`ximxk'
1185        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1186
1187        // define instruments
1188        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
```

```
1189
1190        // add prediction of quantity as extra instrument
1191        local pqhat ""
1192        local qhat ""
1193        local pqhatxdiff ""
1194        local pqhatxdiff2 ""
1195        local pzindivdiffpqhat ""
1196        local pzindivdiffpxbar_lag ""
1197
1198        tempvar qik_hat1
1199         reg qikbar1    `r_RE' `qhat_insts', vce(cluster group_round)
1200        predict `qik_hat1'
1201        local qhat "`qhat' `qik_hat1'"
1202
1203        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1204
1205        g double `p1_qikbar1' = p1*qikbar1
1206        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1207        predict `p1_qikhat1'
1208        local pqhat "`pqhat' `p1_qikhat1'"
1209
1210        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1211        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1212
1213        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1214
1215        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1216        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1217        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1218
1219        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1220                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1221
1222        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1223}

1224
1225// starting values
1226if init_FE_from_RE==1 & "$init_FE_file"!="" {
1227        estimates use "$init_FE_file"
1228        local initial_values_FE ""
1229
1230        // extract vector of coefs and paramter names from
1231        matrix coefs = e(b)
1232        local paramlist = e(params)
1233
1234        // iterate through paramter name list, taking
1235        local t=0
1236        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
```

```
1237}

1238
1239if estimate_FE==1 {
1240        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique
  note: instruments __00006I __00006U __0000E2 __0000E5 __0000EB __0000EE __0000G5
    __0000G6 __0000G8 __0000G9 __0000GE __0000GF __0000GH __0000GI omitted because
    of collinearity.

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00070884
  Iteration 1:  GMM criterion Q(b) =  .00028059
  Iteration 2:  GMM criterion Q(b) =  .00021899

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00914217
  Iteration 1:  GMM criterion Q(b) =  .00869521
  Iteration 2:  GMM criterion Q(b) =  .00863179
  Iteration 3:  GMM criterion Q(b) =  .00862047

  GMM estimation

  Number of parameters =  20
  Number of moments    =  93
  Initial weight matrix: Unadjusted                 Number of obs   =    100,756
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 562 clusters in state_district_round)
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11						
_cons	.7276711	.0317684	22.91	0.000	.6654062	.7899361
C12						
_cons	-.1082217	.0620922	-1.74	0.081	-.2299202	.0134767
C13						
_cons	-.0210604	.0142257	-1.48	0.139	-.0489422	.0068214
C14						
_cons	-.00476	.0142225	-0.33	0.738	-.0326356	.0231155
C15						
_cons	.0015925	.0285773	0.06	0.956	-.0544179	.0576029
C16						
_cons	-.0658933	.0198405	-3.32	0.001	-.10478	-.0270066
C17						
_cons	-.0662636	.0226008	-2.93	0.003	-.1105604	-.0219667
C18						
_cons	.8037025	.1672929	4.80	0.000	.4758144	1.131591
C21						
_cons	.6004541	.0470885	12.75	0.000	.5081623	.692746
C22						
_cons	-.2015186	.0909562	-2.22	0.027	-.3797894	-.0232477
C23						
_cons	-.027852	.0193644	-1.44	0.150	-.0658055	.0101014
C24						
_cons	-.0271514	.0250598	-1.08	0.279	-.0762676	.0219648
C25						

	_cons	.0280668	.0350132	0.80	0.423	-.0405578	.0966915
C26							
	_cons	-.0892115	.0255936	-3.49	0.000	-.1393741	-.0390489
C27							
	_cons	-.0874545	.0320685	-2.73	0.006	-.1503076	-.0246013
C28							
	_cons	-1.039803	.154336	-6.74	0.000	-1.342296	-.7373101
d1							
	_cons	-.0914192	.0068345	-13.38	0.000	-.1048145	-.0780239
b1							
	_cons	.4567872	.0255328	17.89	0.000	.4067437	.5068306
A11							
	one	.0779625	.1247198	0.63	0.532	-.1664838	.3224088
	zi8	.4455289	.1363851	3.27	0.001	.1782191	.7128388

Instruments for equation **eq1_FE:** **__000004 __00000H __00000T __00001C __00001O __000027 __00002J __000032 __00003E __00003X __000049 __00004S __000054 __00005N __00005Z o.__00006I o.__00006U __000005 __0000GM __0000GN __0000GO __0000GP __0000GQ __0000GR __0000GS __0000GT __0000CG __0000CH __0000CJ __0000CK __0000CP __0000CQ __0000CS __0000CT __0000CZ __0000D0 __0000D2 __0000D3 __0000D8 __0000D9 __0000DB __0000DC __0000DI __0000DJ __0000DL __0000DM __0000DR __0000DS __0000DU __0000DV __0000E1 o.__0000E2 __0000E4 o.__0000E5 __0000EA o.__0000EB __0000ED o.__0000EE __0000EK __0000EL __0000EN __0000EO __0000ET __0000EU __0000EW __0000EX __0000F3 __0000F4 __0000F6 __0000F7 __0000FC __0000FD __0000FF __0000FG __0000FM __0000FN __0000FP __0000FQ __0000FV __0000FW __0000FY __0000FZ o.__0000G5 o.__0000G6 o.__0000G8 o.__0000G9 o.__0000GE o.__0000GF o.__0000GH o.__0000GI __0000CD __0000CM __0000CW __0000D5 __0000DF __0000DO __0000DY __0000E7 __0000EH __0000EQ __0000F0 __0000F9 __0000FJ __0000FS __0000G2 __0000GB _cons**

```
1241
1242        esttab using "${filename_FE}.tex", se replace //added this
```

(file **diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXownsland_drop3_all > _exactInst_noPPXdInst_14_zlist3_updateExp_urbOnly_clus_Dist_dropPre_FE.tex** not found)

(output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXownsl > and_drop3_all_exactInst_noPPXdInst_14_zlist3_updateExp_urbOnly_clus_Dist_dropPre_FE. > tex)

```
1243
1244        estat overid
```

Test of overidentifying restriction:

Hansen's J chi2(**73**) = **868.565** (p = **0.0000**)

```
1245        di "$S_DATE $S_TIME"
```

7 Dec 2023 20:50:16

```
1246        unique hhidi
```

Number of unique values of hhidi is **21696**
Number of records is **100756**

```
1247        local N_hh = r(sum)
1248        estadd scalar N_hh = r(sum)
```

added scalar:
e(N_hh) = **21696**

```
1249        unique group_round
  Number of unique values of group_round is  4452
  Number of records is  100756
1250        estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  4452
1251        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
         e(avg_grp_size) =  4.8733154
1252        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXownsland_dro
  > p3_all_exactInst_noPPXdInst_14_zlist3_updateExp_urbOnly_clus_Dist_dropPre_FE.ster no
  > t found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXownsland_drop3_all_
  > exactInst_noPPXdInst_14_zlist3_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1253
1254        if simple_model==1 {
1255                local RE_acz_drv ""
1256                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1257                noi di "`RE_acz_drv'"
1258
1259                cap g junk=.
1260                replace junk=p1*qikbar1
1261                su junk $GMM_weight
1262                local mean_y = r(mean)
1263                su xi $GMM_weight
1264                local mean_x = r(mean)
1265
1266                estimates store gmm_est
1267                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1268                estimates save "${filename_FE}_dF", replace
1269                estimates restore gmm_est
1270        }
1271}

1272
1273
  end of do-file
1274}

1275else {
1276        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1277        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
```

```
1278
1279        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1280
1281        drop if size_group_round<min_group_size
1282        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1283
1284        do "$CODE/estim program.do"
1285}

1286
1287
1288log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table4_37.smcl
   log type:  smcl
  closed on:   7 Dec 2023, 20:50:16
```

A.4.3 Columns 4 and 8

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table4_48.smcl
  log type:  smcl
 opened on:   7 Dec 2023, 20:55:24

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=4

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
. local grp_type ""

. if $rand_grp==1 local grp_type "_Random"

.
. // make summary stats table
. local make_sumstats = 0

.
. // measure expenditure in per capita terms
. global meas_pc = 0

. local typeExp ""

. if $meas_pc==1 local typeExp "_ExpPC"

.
. // number of goods
. scalar J=2

. local J = J

.
. // group def options
. *local group_def "dist"
. *local group_def "fsuXseg"
. *local group_def "fsuXsegXreligXsched"
. local group_def "fsuXsegXreligXschedXehigh"

.
. // comparision group
. local groupComp_def ""

. local desc_compGrp ""

. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

.
. // load data
. include "$CODE/prep demand system data.do"

. //clear all
. pause on

. set more off

. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

.
. scalar short_zlist=$short_zlist_val

.
. cap prog drop bysmeanw

. prog def bysmeanw
  1.         syntax varname, weight(varname) by(varlist) [rename(name)]
  2.
```

```
71.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
72.         tempvar totweight totsum varXweight
    4.
73.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
74. end

75.
76. // load prices
77. use "$DATA/laspeyres_state_core.dta", clear

78. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

79. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)        expend_type   ->   (dropped)
  xij variables:
                          laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

80. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

81. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

82.
83. use "$DATA/laspeyres_state_core_2cats.dta", clear

84. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

85. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

86. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

87. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)        expend_type   ->   (dropped)
  xij variables:
                          laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

88. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
89. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

90.
91. if $use_norm==1 {
92.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
93.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations               840   ->   280
  Number of variables                    5   ->   6
  j variable (3 values)       expend_type   ->   (dropped)
  xij variables:
                         laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

94.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.             }
95.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
96. }

97.
98. // read in consumption data
99. use "$DATA/years1999_2004.dta", clear

100 *use "$DATA/years1999_2004_sm.dta", clear
101 g hhid = _n

102
103 // update expenditure if normal goods
104 if $use_norm==1 {
105         egen update_exp = rowtotal(food fuel clothing other_nondur)
106         replace expenditure = update_exp
  (197,188 real changes made)
107 }

108
109 // possible groups
110 egen group_district=group(state district)

111 egen group_round_district=group(state district round)

112 egen group_round_districtUrb=group(state district round urban)

113
114 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
115 drop _merge

116 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
117 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

118 drop _merge

119
120 //make weights comparable across rounds
121 bys round: egen meanweight=mean(weight)

122 bys round: replace weight=weight/meanweight
  (234590 real changes made)

123 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
124
125 // clean hh chars
126 replace religion=1 if religion==.
  (20 real changes made)

127 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
128 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

129 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
130 g scheduled = socgroup==1 | socgroup==2

131 g hindu = religion==1

132 g muslim = religion==2

133 egen hinduXscheduled = group(religion hindu)

134 g schednh = scheduled==1 & hindu==0

135 g schedh  = scheduled==1 & hindu==1

136 g nschedh = scheduled==0 & hindu==1

137 g nschednh = scheduled==0 & hindu==0

138 g nonHindu=religion>1

139
140 // simplify education categories
141 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

142
143 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
144 local cutoff_group_size=10

145 scalar cutoff_expenditure_percentile=99

146 scalar scale_factor=1

147
```

```
148 // generate variables
149 g low_educ = z9==0 & z10==0

150
151 // drop top and bottom percentiles of expenditure in each round/state
152 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

153 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

154 drop expenditure_cutoff

155 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

156 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

157
158 // keep urban houselds smaller than 13 and with head older than 20
159 keep if urban==1
  (141,042 observations deleted)

160 drop if hhsize>12
  (656 observations deleted)

161 gen married=marstat==2

162 drop if age<20
  (1,182 observations deleted)

163 g hhsizem1=hhsize-1

164
165 // winsorize weights
166 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

167 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

168 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

169
170 g lux=ilux+vlux
```

```
171 g necc=inec+vnec

172
173 // education variables
174 drop if zone==.
  (799 observations deleted)

175 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
176 g educmed=educsimp==2

177 g educhigh=educsimp==3

178
179 // scale age (helps numerical performance)
180 drop if age==.
  (1 observation deleted)

181 replace age=age/40
  (86,380 real changes made)

182
183 // rename z's so that they can be used in each file
184 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

185
186 // landowner dummy
187 g owns_land = landowned>.005 & landowned<.

188
189 egen group_inst=group(state district urban)

190
191 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

192 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

193 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)

194 if "`group_def'"=="fsuXsegXreligXschedXownsland"       egen group = group(geogroup_
  > seg religion scheduled owns_land)

195 if "`group_def'"=="fsuXsegXreligXschedXehigh"                  egen group = group(g
  > eogroup_seg religion scheduled educhigh)
  (43,637 missing values generated)
```

```
196
197 if $rand_grp==1 {
198         sort round
199         sort round state district
200         g shuffle = _n
201         g rand = runiform()
202         sort round rand
203         sort round state district rand
204         g group_new = group[shuffle]
205         replace group = group_new
206         drop group_new shuffle rand
207 }

208
209 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

210 egen district_round=group(state district round urban)

211
212 // make comparision group, so can restrict to sample in other regs
213 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

214 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

215 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)

216 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

217
218 if short_zlist==1 {
219         global T=7
220
221         g z1=hhsizem1/10
222         g z2=age/3
223         g z3=married
224         g z4=.
225         g land=exp(demog23)
226         replace z4=0 if land<=0.005
227         replace z4=ln(land+1) if land>0.005
228         g z5=demog24
229         g z6=educmed
230         g z7=educhigh
231         order land, after(z7)
232
233         global zlist "z1 z2 z3 z4 z5 z6 z7"
234         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
235         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
236 }

237
238 if short_zlist==2 {
239         global T=10
240
241         g z1=hhsizem1/10
242         g z2=age/3
```

```
243         g z3=married
244         g z4=.
245         g land=exp(demog23)
246         replace z4=0 if land<=0.005
247         replace z4=ln(land+1) if land>0.005
248         g z5=demog24
249         g z6=educmed
250         g z7=educhigh
251         g z8 = scheduled==1 & hindu==0
252         g z9 = scheduled==0 & hindu==0
253         g z10 = scheduled==0 & hindu==1
254
255         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
257         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
258 }

259
260 if short_zlist==3 {
261         global T=8
262
263         g z1=hhsizem1/10
264         g z2=age/3
265         g z3=married
266         g z4=.
267         g land=exp(demog23)
268         replace z4=0 if land<=0.005
269         replace z4=ln(land+1) if land>0.005
270         g z5=demog24
271         g z6=educmed
272         g z7=educhigh
273         g z8 = owns_land
274
275         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
276         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
277         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
278 }

279
280 if short_zlist==4 {
281         global T=7
282
283         g z1=hhsizem1/10
284         g z2=age/3
285         g z3=married
286         g z4=.
  (86,380 missing values generated)
287         g land=exp(demog23)
  (12,776 missing values generated)
288         replace z4=0 if land<=0.005
  (29,502 real changes made)
289         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
290         g z5=demog24
  (29 missing values generated)
291         g z6=educmed
292         g z7=educhigh
293
294         global zlist "z1 z2 z3 z4 z5 z6 z7"
295         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
296         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
297 }
298
299 if short_zlist==5 {
300         global T=6
301
302         g z1=age/3
303         g z2=married
304         g z3=.
305         g land=exp(demog23)
306         replace z3=0 if land<=0.005
307         replace z3=ln(land+1) if land>0.005
308         g z4=demog24
309         g z5=educmed
310         g z6=educhigh
311         order land, after(z6)
312
313         global zlist "z1 z2 z3 z4 z5 z6"
314         global zlist_indiv "z1 z2 z3 z4 z5 z6"
315         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
316 }
317
318 if short_zlist==6 {
319         global T=5
320
321         g z1=hhsizem1/10
322         g z2=p1
323         g z3=p2
324         g z4=p3
325         g z5=p4
326         g land=exp(demog23)
327
328         global zlist "z1 z2 z3 z4 z5"
329         global zlist_indiv "z1"
330         global xzlist_indiv "xz1"
331
332         drop if land==. | demog24==.
333 }
334
335 if short_zlist==7 {
336         global T=11
337
338         g z1=hhsizem1/10
339         g z2=age/3
340         g z3=married
341         g z4=.
342         g land=exp(demog23)
343         replace z4=0 if land<=0.005
344         replace z4=ln(land+1) if land>0.005
345         g z5=demog24
346         g z6=educmed
347         g z7=educhigh
348         g z8=p1
349         g z9=p2
350         g z10=p3
351         g z11=p4
352
```

```
353         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
354         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
355         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
356 }

357
358
359
360 // drop missings and calculate size of grp
361 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

362 bys group_round: egen size_group_round=count(group_round)

363
364 // size of comparision group
365 if "`groupComp_def'"!="" {
366         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
367 }

368
369 // turn things int oper capita terms if ption flagged
370 if $meas_pc==1 {
371         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
372 }

373
374 // normalize expenditure by overall mean
375 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |   73,576   45441.646    4270.163   2537.534   308.7667   27910.14

376 scalar expenditure_mean=r(mean)

377 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

378
379 if $use_norm==0 {
380         g pq1=vlux_norm
381         g pq2=ilux_norm
382         g pq3=vnec_norm
383         g pq4=inec_norm
384 }
```

```
385 else {
386         g pq1=food_norm
387         g pq2=fuel_norm
388         g pq3=clothing_norm
389         g pq4=other_nondur_norm
390 }

391 g x=expenditure_norm

392 g x2=x*x

393
394 // make q averages
395 qui bysort group_round: egen weight_group_round=sum(weight)

396
397 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

398
399 // 2 category consumption
400 g PQ1=pq1+pq2

401 g PQ2=pq3+pq4

402 if $use_vis==1 {
403         drop PQ1 PQ2
404         g PQ1=pq1+pq3
405         g PQ2=pq2+pq4
406 }

407 if $use_vislux==1 {
408         drop PQ1 PQ2
409         g PQ1=pq1
410         g PQ2=pq2+pq3+pq4
411 }

412 if $use_norm==1 {
413         drop PQ1 PQ2
414         g PQ1=pq1
415         g PQ2=pq2+pq3+pq4
416         if `J'==3 {
417                 drop PQ1 PQ2
418                 g PQ1=pq1
419                 g PQ2=pq2
420                 g PQ3=pq3+pq4
421         }
422 }

423
424 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
425 if `J'==3 {
426         qui g Q`J'=PQ`J'/P`J'
427         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
428 }

429
430 // only scheduled groups (with some of other group)
431 bys geogroup_seg religion: egen share_sched = mean(scheduled)

432 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

433
434 // update group epend wiht other group
435 if $only_scheduled==2 {
436         preserve
437                 keep if scheduled==0
438                 keep geogroup_seg religion Qbar* qbar*
439                 duplicates drop
440                 tempfile update_exp
441                 save `update_exp'
442         restore
443         keep if scheduled==1 & share_sched>0 & share_sched<1
444         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
445 }

446 drop share_sched

447
448 // laysperes_cpi varies by district and round
449 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

450
451 // make instruments
452 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
453 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
454 // measure all the x-stuff in reals, using aggregate laysperes index.
455 g weight_temp=weight

456 global instlist ""

457 g xreal=x/local_cpi
  (82 missing values generated)

458 g x_temp=x

459 g x2_temp=x2

460 replace x=xreal
  (73,576 real changes made, 82 to missing)

461 replace x2=x*x
  (73,576 real changes made, 82 to missing)

462
463 // make z x interactions
464 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
465
466 levelsof round, local(roundlist)
  59 60 61 62

467 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

468
469 g xbar2lag=xbarlag^2
  (4 missing values generated)

470 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

471
472 bysmeanw x, weight(weight) by(group) rename(xbar)

473 bysmeanw x2, weight(weight) by(group) rename(x2bar)

474 g xbar2=xbar^2

475 g rootxbar = sqrt(xbar)

476 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

477
478 replace x=x_temp
  (73,576 real changes made)

479 replace x2=x2_temp
  (73,576 real changes made)

480
481 // add x terms to inst lsit
482 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

483
484 // only groups of size 3 or more
485 drop if size_group_round<3
  (53,343 observations deleted)

486 if "`groupComp_def'"!="" drop if size_groupComp_round<3

487 drop size_group_round

488 bys group_round: egen size_group_round=count(group_round)
```

```
489
490 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
491 if `J'==3 local extrakeep "Q3 P3 Qbar3"

492 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

493 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

494
495 g obs_numi=_n

496
497 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

498 rename x xi

499
500 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

501 rename weight weighti

502 rename land landi

503 rename owns_land owns_landi

504 rename schednh schednhi

505 rename schedh schedhi

506 rename nschedh nschedhi

507 rename nschednh nschednhi

508 g urbani = urban

509 rename hhid hhidi

510 rename scheduled scheduledi

511 rename hhsize hhsizei

512
513 tempfile obs_i

514 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_41e4_000001.tmp saved as .dta format

515
516 // calculate sumstats at hh level
517 if `make_sumstats'==1 {
518         count
```

```
519         local sumstats_n_obs = r(N)
520
521         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
522                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
523         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
524         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
525 }

526
527 rename obs_numi obs_numk

528
529 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

530 rename xi xk

531
532 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

533 rename weighti weightk

534 rename landi landk

535 rename owns_landi owns_landk

536 rename urbani urbank

537 rename schednhi schednhik

538 rename schedhi schedhk

539 rename nschedhi nschedhk

540 rename nschednhi nschednhk

541 rename scheduledi scheduledk

542 rename hhsizei hhsizek

543 drop geogroup

544
545 // combine into pairs within groups
546 joinby group_round using "`obs_i'"

547 drop if obs_numi==obs_numk
  (20,233 observations deleted)

548
549 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
550 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

551
552 // summary stats tables
553 if `make_sumstats'==1 {
554         count
555         local sumstats_n_obs_pair = r(N)
556
557         cap file close sumstat
558                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
559                 file write sumstat "\begin{table}[htbp]\centering" _n
560                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
561                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
562                 file write sumstat "\begin{tabular}{lcccccccc}" _n
563                 file write sumstat "\toprule" _n
564                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
565                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
566                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
567                 file write sumstat "\midrule" _n
568
569                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
570                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var'" _n
    5.                 }
571
572                 file write sumstat "\bottomrule" _n
573                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
574                 file write sumstat "\end{tabular}" _n
575                 file write sumstat "\end{table}" _n
576                 file close sumstat
577                 BREAK
578 }

579
580 // constant for estimation
581 g one=1

582 egen state_district_round=group(state district urban round)

583 egen state_district=group(state district)

584
```

```
585 // keep only thick round
586 keep if round==61
  (0 observations deleted)

587
588
589
590
591
592
593 // options
594 local output_number="14"

595
596 // dimension of A matrix
597 scalar Afull=0

598 local descA "full"

599 scalar Adiag=1

600 if Adiag==1 local descA "diag"

601 scalar same_spillover=1

602 if same_spillover==1 local descA "same"

603
604 // minimum group size (must be 3 or more)
605 local min_group_size = 3

606 scalar min_group_size=`min_group_size'

607
608 // flag for generic model
609 scalar simple_model=0

610 local simp_name ""

611 if simple_model==1 local simp_name "_simple"

612
613 // interactions with peer effects
614 //global Alist "one"
615 //global Alist "one schednhi schedhi nschednhi"
616 //global Alist "one   zi8"
617 global Alist "one   zi7"

618
619 // turn on or off measurement error correction
620 scalar noMeasError=0

621 local measError_desc ""

622 if noMeasError==1 local measError_desc "_NoMeasError"

623
624 // estimate RE and or FE
625 scalar estimate_RE=1
```

```
626 scalar estimate_FE=1

627
628 // estimation options
629 global w_initial "winit(unadjusted)"

630 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

631 global wmatrix "wmatrix(robust)"

632 global trace_level ""

633 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

634 global technique ""

635 global maxiter ""

636 global technique_RE ""

637 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

638 global derivatives "quickderivatives"

639 scalar init_FE_from_RE=1

640 global clust_var "state_district_round"

641 local clus_name "_clus_Dist"

642
643
644
645
646
647
648
649
650 // xxzx to keep names same
651 local wmatrix_desc ""

652 local sectorName "_urbOnly"

653 local dropZdesc "_dropPre"

654 local desc_struc_v0 ""

655 local inst_desc "_exactInst"

656 local inst_rootp "_noPPXdInst"

657 local weight_desc ""

658 local RE_sq_inst_desc ""

659 local cnstname ""

660 local actual_pqhat_desc ""

661 local expUpname "_updateExp"
```

```
662
663
664 if $rand_grp==0 {
665         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
666         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
667
668         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
669
670         drop if size_group_round<min_group_size
  (0 observations deleted)
671         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
672
673         //do "$CODE/estim program.do"
674         do "$CODE/estim program.do"

675 // This code estimates a QES demand system
676 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
677 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
678 // i indexes observations, k indexes other observations in the group
679 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
680 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
681 //              ensure that all zt are z-scored or otherwise have similar scales
682 // qikbarj are leave-two-out group averages
683 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
684 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
685 // note that qi and qk are actually quantities, and not spending
686 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
687
688 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
689
690 set seed 339487731

691
692 local J=J

693 local Jm1=`J'-1

694 local T=$T

695 local Tm1=`T'-1
```

```
696 local Tindiv : word count $zlist_indiv

697
698 /*****************************************************************************
  > Set weights
  > *****************************************************************************/
699
700 tempvar clust_weight num_group_round

701 bys group_round: egen `num_group_round'=count(group_round)

702 g double `clust_weight'=size_group_round/`num_group_round'

703 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

704 global GMM_weight "[aweight=`clust_weight']"

705
706 /*****************************************************************************
  >  Update data for J=2, and J=3
  > *****************************************************************************/
707
708 if `J'==2 {
709          replace p1=P1
  (84,052 real changes made)
710          replace p2=P2
  (84,052 real changes made)
711          replace qi1=Qi1
  (84,049 real changes made)
712          replace qk1=Qk1
  (84,049 real changes made)
713          replace qikbar1=Qikbar1
  (84,052 real changes made)
714          replace qi2=Qi2
  (84,052 real changes made)
715          replace qk2=Qk2
  (84,052 real changes made)
716          replace qikbar2=Qikbar2
  (84,052 real changes made)
717
718          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
719 }

720
721 if `J'==3 {
722          forv j = 1(1)3 {
    2.                 replace p`j'=P`j'
    3.                 replace qi`j'=Qi`j'
    4.                 replace qk`j'=Qk`j'
    5.                 replace qikbar`j'=Qikbar`j'
    6.          }
723
724          foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
725 }

726
```

```
727
728 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
729
730 global zilist ""

731 global zklist ""

732 forval t=1/`Tm1' {
    2.         global zilist "$zilist zi`t'"
    3.         global zklist "$zklist zk`t'"
    4. }

733 global zilistTm1 "$zilist"

734 global zklistTm1 "$zklist"

735 if `T'>0 global zilist "$zilist zi`T'"

736 if `T'>0 global zklist "$zklist zk`T'"

737
738 // create b_p and pC
739 // remember b_p doesn't vary within group
740 local b_p "exp( 0"

741 forval j=1/`Jm1' {
    2.         local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

742 local b_p "`b_p' + ln(p`J') )"

743 if simple_model==1      local b_p "1"

744 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

745
746 // pCj is the sum of these two, for use in the RE model
747 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
748 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
749 forval j=1/`J' {
    2.         local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

750 local Tindivp1=`Tindiv'+1

751 forval j=1/`J' {
    2.         local pCi`j' "( 0 "
    3.         local pCk`j' "( 0 "
    4.         local pCg`j' "( 0 "
    5.         forval t=1/`Tindiv' {
    6.                 local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                 local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.         }
    9.         forval t=`Tindivp1'/`T' {
   10.                 local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.         }
   12.         local pCi`j' "`pCi`j'' )"
   13.         local pCk`j' "`pCk`j'' )"
   14.         local pCg`j' "`pCg`j'' )"
   15. }
```

```
752
753 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
754 local pC "( 0"

755 local pCi "( 0"

756 local pCk "( 0"

757 local pCg "( 0"

758 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

759 local pC "`pC' )"

760 local pCi "`pCi' )"

761 local pCk "`pCk' )"

762 local pCg "`pCg' )"

763
764 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
765 local pC_2nd "( 0"

766 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

767 local pC_2nd "`pC_2nd' )"

768
769 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
770 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
771 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
772 local rpDrp "( 0"

773 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
774         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

775
776 local rpDrp "`rpDrp' )"

777 di "`rpDrp'"
  ( 0 )

778
779 if simple_model==1      {
780         local pC "0"
781         local pC1 "0"
782         local pCi "0"
783         local pCk "0"
784         local pC_2nd "0"
785         local rpDrp "0"
786         local Drp1 "0"
787 }

788
789 // create Aq
790 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
791 // k means use qk as the group quantity (for use in RE only)
792 local qtypelist "ikbar k bar"

793 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
794                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
795                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
796         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
797                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
798         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
799
800 // Aikbar is A'qbar-minus-ik; Ak is A'qk
801 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
802 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
803 local xi_hat            "(xi - `pCi')"

804 local xk_hat            "(xk - `pCk')"

805 local x_hat                    "(xi - `pC' - `rpDrp' )"

806 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

807
808 // construct structural vo
809 * let AVA be A'VA, and ensure its diagonals are positive.
810 local v0 "(0 "

811 forval j=1/`J' {
    2.          local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
812         local jplus1=`j'+1
    4.          if `jplus1'<=`J' {
    5.                  forval k=`jplus1'/`J' {
    6.                          local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                  }
    8.          }
    9. }

813
814 local v0 "`v0' )"

815
816 // make all equations, then put pieces together
817
818 * RE equations; first equation has first call to C, other equations use {Cj: } form
819 * note "-vj:", this is because we subtract the structural v0 term from E[q]
820 local j=1

821
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

822 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

823 if `J'>2 {
824         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                  if noMeasError==1                               local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
```

```
825 }

826
827 *       FE equations
828 forval j=1/`Jm1' {
    2.                                                                              l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                    ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

829
830 // clean up equations, and combine
831 local eqs_RE ""

832 local eqs_FE ""

833 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

834
835 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2
  > ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qikba
  > r1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d
  > 1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

836 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > }*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))        ))
```

```
837
838 * replace the first occurrence of Ajj with the Alist
839 if Afull==0 {
840         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
841 }

842 if Afull==1 {
843         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
844 }

845
846 if simple_model==1 {
847         // construct z sums and interactions
848         if `T'>0 {
849
850                 local RE_cz_2            "+ {d}*(0"
851                 local RE_acz     ""
852                 local RE_acz1     ""
853                 local RE_acz2     ""
854                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
855                 local RE_czcz    ""
856                 local FE_czi     ""
857                 local FE_czk     ""
858                 local FE_czg     ""
859                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
860                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
861                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
862                 local RE_cz_2 "`RE_cz_2')^2"
863         }
864
865         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
866
867         local xi_hat "({b}*p1*xi `FE_czi')"
868         local xk_hat "({b}*p1*xk `FE_czk')"
869
870         local eqs_FE
  >                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
871 }

872
873 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one   zi7}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikb
  > ar1) + ({A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) +
  > (p2*{C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p
  > 2*qikbar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p
  > 2*{C2:}) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) +
  > (p2*{C2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}
  > *p1*qikbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22
  > }*p2 )*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) )

874 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))*({b1} - 2*(( ({A11:
  > one   zi7}*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp
  > ( {b1}*ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C
  > 14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*z
  > k2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))
  >      ))

875
876 /*******************************************************************************
  >  construct instruments
  > *******************************************************************************/
877
878 //make squares and differences of x
879 tempvar x2i x2k ximxk x2imx2k xiP

880 g double `xiP'=xi*local_cpi

881 g double `x2i'=xi*xi

882 g double `x2k'=xi*xi

883 g double `ximxk'=xi-xk
```

```
884 g double `x2imx2k'=xi*xi-xk*xk

885
886 g xbarlag_x2barlag = xbarlag*x2barlag

887 g xbarlag2  = xbarlag^2

888 g x2barlag2 = x2barlag^2

889
890 //make interactions with z and zp
891 forval t=1/`T' {
    2.          tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.          g double `xizi`t''=xi*zi`t'
    4.          g double `xkzk`t''=xk*zk`t'
    5.          g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.          g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.          g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.          g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.          g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.          forval j=1/`J' {
   11.                  tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                  g double `zi`t'p`j''= zi`t'*p`j'
   13.                  g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                  g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                  g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                  forv s=1/`T' {
   17.                          tempvar zi`t'zi`s'p`j'
   18.                          g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                  }
   20.          }
   21. }

892
893 // make group-level instruments, and the instrument lists rg, xrg
894 if noMeasError==0       local rg "xbarlag"

895 if noMeasError==1       local rg "xbar"

896
897 local rg "`rg' x2barlag"

898 local rg "`rg' xbar2lag"

899 local rg "`rg' rootxbarlag"

900
901 local Tindivm1=`Tindiv'-1

902 // lag instruments
903 forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }

904
905 local zrg ""

906 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var''"
    6.          }
    7. }
```

```
907
908 local xrg ""

909 local prg ""

910 local xprg ""

911 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var''"
    5.
912         forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var''"
   10.                  local xprg "`xprg' `xp`j'`var''"
   11.          }
   12. }

913
914 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
915 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
916 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
917 local budget "xi `x2i'"

918
919 local rootprootp ""

920 local xrootprootp ""

921 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

922
923 // demographics multiplied by prices and budget
924 local zinsts ""

925 local xzinsts ""

926 local xPzinsts ""

927 local pzinsts ""

928 local xpzinsts ""

929 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }
```

```
930
931 // price insts
932 local roundinsts " "

933 forval j=1/`J' {
    2.         forval k=`j'/`J' {
    3.                 tempvar p`j'p`k'
    4.                 g `p`j'p`k''=p`j'*p`k'
    5.                 local roundinsts "`roundinsts' `p`j'p`k''"
    6.         }
    7. }

934
935 // instruments for all equations are ri:  x,x2,p,z,zx,
936 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
937 local qhat_insts ""

938 local qhat_insts "`qhat_insts' `zrg'"

939 local qhat_insts "`qhat_insts' `prg'"

940 local qhat_insts "`qhat_insts' `rg' "

941
942 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

943
944 local count: word count `r_RE'

945 di "total instruments: `count'"
  total instruments: 19

946
947 local qhat_hats ""

948 local pqhat ""

949 local xpqhat ""

950 local pqhat_sq ""

951 forval j=1/`J' {
    2.         if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.         if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
952         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
953         // use all prices times all qikhats as instruments, interacted with xi
954         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
955                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }

  Linear regression                               Number of obs     =     84,052
                                                  F(129, 4445)      =      23.10
                                                  Prob > F          =     0.0000
                                                  R-squared         =     0.2501
                                                  Root MSE          =     .13615
```

(Std. err. adjusted for **4,446** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	-12.73252	7.369067	-1.73	0.084	-27.17956	1.714515
__00009L	-3.908656	1.317828	-2.97	0.003	-6.492254	-1.325057
__00009O	1.952542	3.984012	0.49	0.624	-5.858104	9.763189
xi	.1079365	.0095	11.36	0.000	.0893118	.1265612
__000002	-.0151526	.0031675	-4.78	0.000	-.0213625	-.0089427
__00000E	.0894046	.1010424	0.88	0.376	-.1086888	.287498
__00000P	.0062898	.0602955	0.10	0.917	-.1119193	.1244989
__000017	.1500829	.1818022	0.83	0.409	-.2063399	.5065058
__00001I	-.06866	.1119726	-0.61	0.540	-.288182	.150862
__000020	.027154	.0541844	0.50	0.616	-.0790744	.1333823
__00002B	-.0515801	.0294315	-1.75	0.080	-.1092805	.0061203
__00002T	-.0157976	.0754667	-0.21	0.834	-.1637499	.1321547
__000034	.0108097	.0573323	0.19	0.850	-.1015902	.1232096
__00003M	.0238553	.0605589	0.39	0.694	-.0948703	.1425809
__00003X	.0122294	.0384987	0.32	0.751	-.0632473	.087706
__00004F	-.066814	.0482764	-1.38	0.166	-.1614597	.0278317
__00004Q	.0320565	.0266222	1.20	0.229	-.0201363	.0842494
__000058	-.0519014	.095193	-0.55	0.586	-.2385271	.1347243
__00005J	-.0450822	.0577163	-0.78	0.435	-.1582348	.0680705
__00005U	.4811364	.4238445	1.14	0.256	-.3498097	1.312083
__00005V	.014548	.0768475	0.19	0.850	-.1361113	.1652074
__00005W	-.1497377	.1714757	-0.87	0.383	-.4859155	.1864401
__00005X	-.4583032	.4598629	-1.00	0.319	-1.359863	.443257
__00005Y	.1026161	.1150777	0.89	0.373	-.1229934	.3282257
__00005Z	.0137161	.1871368	0.07	0.942	-.3531653	.3805974
__000060	-.0301891	.0822054	-0.37	0.713	-.1913527	.1309745
__000061	-.0101523	.0540427	-0.19	0.851	-.1161029	.0957983
__000062	.0228172	.0375025	0.61	0.543	-.0507064	.0963408
__000063	-.0645826	.0512432	-1.26	0.208	-.1650449	.0358796
__000064	-.049856	.0648913	-0.77	0.442	-.1770753	.0773633
__000065	-1.05826	1.006148	-1.05	0.293	-3.030811	.9142914
__000066	.0057092	.1329888	0.04	0.966	-.2550151	.2664334
__000067	.2675357	.3748926	0.71	0.475	-.4674403	1.002512
__000068	1.295167	.9652226	1.34	0.180	-.5971494	3.187484
__000069	-.3305621	.194341	-1.70	0.089	-.7115671	.050443
__00006A	-.4518249	.349679	-1.29	0.196	-1.13737	.2337199
__00006B	-.1281442	.1342526	-0.95	0.340	-.3913462	.1350578
__00006C	-.0705728	.1186977	-0.59	0.552	-.3032793	.1621337
__00006D	-.0221978	.0638702	-0.35	0.728	-.1474151	.1030196
__00006E	-.268217	.1015146	-2.64	0.008	-.4672361	-.0691978
__00006F	-.1645999	.1263539	-1.30	0.193	-.4123166	.0831167
__00006G	-.2515089	.2243853	-1.12	0.262	-.6914159	.188398
__00006H	.0398296	.0368432	1.08	0.280	-.0324014	.1120606
__00006I	.0138952	.0863635	0.16	0.872	-.1554202	.1832105
__00006J	.2012072	.236487	0.85	0.395	-.262425	.6648394
__00006K	.075745	.0614647	1.23	0.218	-.0447564	.1962465
__00006L	.001442	.090585	0.02	0.987	-.1761497	.1790337
__00006M	-.0204062	.045365	-0.45	0.653	-.1093442	.0685317
__00006N	.0462515	.0261846	1.77	0.077	-.0050833	.0975863
__00006O	-.0278014	.0167481	-1.66	0.097	-.060636	.0050332
__00006P	-.017407	.0262634	-0.66	0.508	-.0688963	.0340824
__00006Q	.0035924	.0317898	0.11	0.910	-.0587315	.0659164
__00006R	-.558085	.5825991	-0.96	0.338	-1.700269	.5840992
__00006S	-.0336344	.0624044	-0.54	0.590	-.1559781	.0887093
__00006T	.2740001	.2277791	1.20	0.229	-.1725604	.7205606
__00006U	.3940912	.5036755	0.78	0.434	-.5933635	1.381546
__00006V	-.1727422	.0829799	-2.08	0.037	-.335424	-.0100603
__00006W	-.0266908	.1516779	-0.18	0.860	-.3240551	.2706734
__00006X	.0434122	.0569473	0.76	0.446	-.0682329	.1550573
__00006Y	.0353139	.0337511	1.05	0.295	-.0308551	.1014829
__00006Z	-.0519732	.0256215	-2.03	0.043	-.1022041	-.0017423
__000070	-.0032762	.0411726	-0.08	0.937	-.083995	.0774427
__000071	-.0910445	.0501467	-1.82	0.070	-.189357	.0072679
__000072	-.2795439	.3363418	-0.83	0.406	-.9389413	.3798535
__000073	-.0613711	.0465087	-1.32	0.187	-.1525513	.0298091
__000074	.1964182	.1271467	1.54	0.122	-.0528527	.4456892
__000075	.1755066	.3213495	0.55	0.585	-.4544984	.8055116

__000076	-.10287	.068151	-1.51	0.131	-.2364799	.0307398
__000077	-.0212849	.1109386	-0.19	0.848	-.2387797	.1962099
__000078	.0145733	.0464277	0.31	0.754	-.076448	.1055947
__000079	.0193689	.0385896	0.50	0.616	-.0562859	.0950237
__00007A	.023343	.0215788	1.08	0.279	-.0189622	.0656483
__00007B	-.0185474	.0333235	-0.56	0.578	-.083878	.0467831
__00007C	-.0816465	.0385531	-2.12	0.034	-.1572298	-.0060633
__00007D	.1379219	.2183944	0.63	0.528	-.2902398	.5660837
__00007E	-.0090376	.0343054	-0.26	0.792	-.0762933	.0582182
__00007F	-.0779297	.0820547	-0.95	0.342	-.2387977	.0829384
__00007G	.0601434	.2347476	0.26	0.798	-.4000787	.5203656
__00007H	-.0655412	.0545923	-1.20	0.230	-.1725693	.0414869
__00007I	-.064272	.0917696	-0.70	0.484	-.244186	.1156421
__00007J	-.0275497	.0354447	-0.78	0.437	-.097039	.0419395
__00007K	-.0355562	.0295291	-1.20	0.229	-.093448	.0223356
__00007L	.0057291	.0170182	0.34	0.736	-.0276349	.0390932
__00007M	-.0103075	.0283361	-0.36	0.716	-.0658604	.0452455
__00007N	.0406568	.031951	1.27	0.203	-.021983	.1032965
__00007O	-.3553033	.4323883	-0.82	0.411	-1.203	.4923931
__00007P	.1045004	.0715063	1.46	0.144	-.0356876	.2446883
__00007Q	-.0804392	.1705848	-0.47	0.637	-.4148703	.2539919
__00007R	.4957288	.4610228	1.08	0.282	-.4081053	1.399563
__00007S	-.0405426	.1102724	-0.37	0.713	-.2567314	.1756462
__00007T	-.2696218	.1965574	-1.37	0.170	-.6549722	.1157286
__00007U	.1165778	.0754072	1.55	0.122	-.0312578	.2644135
__00007V	.0006939	.0543475	0.01	0.990	-.1058542	.107242
__00007W	.0437502	.0326678	1.34	0.181	-.0202949	.1077953
__00007X	-.0277642	.0580464	-0.48	0.632	-.141564	.0860356
__00007Y	.0669801	.0687706	0.97	0.330	-.0678445	.2018047
__000080	-34.90927	17.15201	-2.04	0.042	-68.53575	-1.282791
__000082	2.77102	8.913253	0.31	0.756	-14.70339	20.24543
__000085	-.356454	.3873145	-0.92	0.357	-1.115783	.4028752
__000087	.4267385	.2658285	1.61	0.108	-.0944176	.9478947
__00008A	7.124866	3.344713	2.13	0.033	.5675637	13.68217
__00008C	-1.771513	1.695097	-1.05	0.296	-5.094748	1.551721
__00008F	42.89889	21.2218	2.02	0.043	1.293601	84.50417
__00008H	-.7682622	11.37816	-0.07	0.946	-23.07512	21.5386
__00008K	-.4212837	.5179607	-0.81	0.416	-1.436744	.5941772
__00008M	-1.071909	.4960119	-2.16	0.031	-2.044339	-.0994784
__00008P	.358066	.9117491	0.39	0.695	-1.429416	2.145548
__00008R	-1.327817	.6602769	-2.01	0.044	-2.622289	-.0333457
__00008U	-.081808	.3440669	-0.24	0.812	-.7563504	.5927343
__00008W	.4068209	.2689278	1.51	0.130	-.1204114	.9340532
__00008Z	.7600942	.2284302	3.33	0.001	.3122573	1.207931
__000091	-.4406069	.1607908	-2.74	0.006	-.7558369	-.125377
__000094	.1844437	.177546	1.04	0.299	-.1636348	.5325222
__000096	.0574517	.1500922	0.38	0.702	-.2368037	.3517072
__000099	-.2482066	.2524644	-0.98	0.326	-.7431626	.2467494
__00009B	.0618988	.2010635	0.31	0.758	-.3322857	.4560833
__00009E	-.1201186	.3108826	-0.39	0.699	-.7296032	.489366
__00009G	-.4216688	.2265725	-1.86	0.063	-.8658637	.0225262
xbarlag	32.89004	22.04866	1.49	0.136	-10.33631	76.11639
x2barlag	-.3384075	.4240148	-0.80	0.425	-1.169688	.4928726
xbar2lag	-4.997865	4.347636	-1.15	0.250	-13.5214	3.525665
rootxbarlag	-43.33674	27.18811	-1.59	0.111	-96.63896	9.965479
z1barlag	1.728336	.8419091	2.05	0.040	.0777746	3.378896
z2barlag	1.142782	1.170059	0.98	0.329	-1.151115	3.43668
z3barlag	-.3725164	.4869302	-0.77	0.444	-1.327142	.5821092
z4barlag	-.2461954	.2698192	-0.91	0.362	-.7751753	.2827845
z5barlag	-.2633075	.1880512	-1.40	0.162	-.6319814	.1053664
z6barlag	.2532203	.348117	0.73	0.467	-.4292622	.9357029
z7barlag	.5552983	.3880471	1.43	0.152	-.2054672	1.316064
_cons	15.31784	9.309328	1.65	0.100	-2.933081	33.56876

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	84,052	.4253486	.0785636	.199297	.8855615

Linear regression

Number of obs	=	**84,052**	
F(128, 4445)	=	**.**	
Prob > F	=	**.**	
R-squared	=	**0.4130**	
Root MSE	=	**.18969**	

(Std. err. adjusted for **4,446** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**9.629519**	**8.083283**	**1.19**	**0.234**	**-6.21774**	**25.47678**
__00009L	**1.106562**	**1.793862**	**0.62**	**0.537**	**-2.4103**	**4.623423**
__00009O	**2.819742**	**4.609513**	**0.61**	**0.541**	**-6.217198**	**11.85668**
xi	**.1251284**	**.0157269**	**7.96**	**0.000**	**.0942959**	**.155961**
__000002	**-.0020993**	**.0062796**	**-0.33**	**0.738**	**-.0144104**	**.0102119**
__00000E	**.0212689**	**.1233318**	**0.17**	**0.863**	**-.2205228**	**.2630607**
__00000P	**-.0639746**	**.0779059**	**-0.82**	**0.412**	**-.2167089**	**.0887597**
__000017	**.2896356**	**.2595834**	**1.12**	**0.265**	**-.2192771**	**.7985483**
__00001I	**.1762503**	**.166318**	**1.06**	**0.289**	**-.1498157**	**.5023164**
__000020	**.0708586**	**.0653964**	**1.08**	**0.279**	**-.057351**	**.1990682**
__00002B	**.0402053**	**.0371738**	**1.08**	**0.280**	**-.0326739**	**.1130845**
__00002T	**.2554372**	**.0792663**	**3.22**	**0.001**	**.1000358**	**.4108387**
__000034	**-.053384**	**.0654696**	**-0.82**	**0.415**	**-.181737**	**.0749691**
__00003M	**.029735**	**.0723042**	**0.41**	**0.681**	**-.1120172**	**.1714872**
__00003X	**-.0018951**	**.0472524**	**-0.04**	**0.968**	**-.0945333**	**.0907431**
__00004F	**-.0634327**	**.0566019**	**-1.12**	**0.262**	**-.1744006**	**.0475351**
__00004Q	**.027829**	**.0358356**	**0.78**	**0.437**	**-.0424267**	**.0980847**
__000058	**.0150677**	**.1604314**	**0.09**	**0.925**	**-.2994577**	**.3295931**
__00005J	**.0177828**	**.0974184**	**0.18**	**0.855**	**-.1732059**	**.2087714**
__00005U	**-.326043**	**.5650175**	**-0.58**	**0.564**	**-1.433759**	**.7816726**
__00005V	**.0570096**	**.0880207**	**0.65**	**0.517**	**-.1155548**	**.229574**
__00005W	**-.0213147**	**.2226448**	**-0.10**	**0.924**	**-.4578093**	**.41518**
__00005X	**.2829804**	**.5728053**	**0.49**	**0.621**	**-.8400031**	**1.405964**
__00005Y	**.1505949**	**.1440063**	**1.05**	**0.296**	**-.1317291**	**.432919**
__00005Z	**.1165222**	**.2342244**	**0.50**	**0.619**	**-.3426741**	**.5757186**
__000060	**-.1906844**	**.1186647**	**-1.61**	**0.108**	**-.4233262**	**.0419574**
__000061	**.057797**	**.0579556**	**1.00**	**0.319**	**-.0558248**	**.1714188**
__000062	**-.0355028**	**.046257**	**-0.77**	**0.443**	**-.1261895**	**.055184**
__000063	**.0126584**	**.0588251**	**0.22**	**0.830**	**-.1026681**	**.127985**
__000064	**-.0006398**	**.0727721**	**-0.01**	**0.993**	**-.1433092**	**.1420297**
__000065	**2.08159**	**1.269419**	**1.64**	**0.101**	**-.407103**	**4.570282**
__000066	**.1521409**	**.1994681**	**0.76**	**0.446**	**-.238916**	**.5431977**
__000067	**-.8048261**	**.5069429**	**-1.59**	**0.112**	**-1.798687**	**.1890343**
__000068	**-1.313909**	**1.259324**	**-1.04**	**0.297**	**-3.782811**	**1.154993**
__000069	**-.5768386**	**.2729932**	**-2.11**	**0.035**	**-1.112041**	**-.0416359**
__00006A	**-.1026947**	**.5271538**	**-0.19**	**0.846**	**-1.136179**	**.9307892**
__00006B	**-.1269372**	**.1864405**	**-0.68**	**0.496**	**-.4924534**	**.2385791**
__00006C	**-.4209147**	**.1299051**	**-3.24**	**0.001**	**-.6755934**	**-.166236**
__00006D	**-.0107406**	**.0876206**	**-0.12**	**0.902**	**-.1825207**	**.1610394**
__00006E	**-.2439648**	**.1385943**	**-1.76**	**0.078**	**-.5156787**	**.0277491**
__00006F	**-.2456211**	**.168179**	**-1.46**	**0.144**	**-.5753356**	**.0840933**
__00006G	**.1892016**	**.3713697**	**0.51**	**0.610**	**-.5388678**	**.917271**
__00006H	**.0602881**	**.0488747**	**1.23**	**0.217**	**-.0355306**	**.1561068**
__00006I	**-.1195145**	**.1419753**	**-0.84**	**0.400**	**-.3978568**	**.1588279**
__00006J	**-.3124916**	**.3363266**	**-0.93**	**0.353**	**-.971859**	**.3468759**
__00006K	**.0562282**	**.0772067**	**0.73**	**0.466**	**-.0951355**	**.2075918**
__00006L	**.0001789**	**.1217172**	**0.00**	**0.999**	**-.2384475**	**.2388052**
__00006M	**.0226298**	**.06748**	**0.34**	**0.737**	**-.1096645**	**.1549242**
__00006N	**.0202324**	**.0294364**	**0.69**	**0.492**	**-.0374775**	**.0779424**
__00006O	**.0013999**	**.0227663**	**0.06**	**0.951**	**-.0432333**	**.0460331**
__00006P	**-.0435669**	**.0319756**	**-1.36**	**0.173**	**-.1062549**	**.0191212**
__00006Q	**.0023696**	**.0397697**	**0.06**	**0.952**	**-.0755988**	**.0803379**
__00006R	**-.028743**	**.4379357**	**-0.07**	**0.948**	**-.887315**	**.8298291**
__00006S	**.0648557**	**.0669205**	**0.97**	**0.333**	**-.0663418**	**.1960531**
__00006T	**.043945**	**.1615795**	**0.27**	**0.786**	**-.2728312**	**.3607212**
__00006U	**-.4790627**	**.4205855**	**-1.14**	**0.255**	**-1.30362**	**.3454943**
__00006V	**.0551929**	**.0915788**	**0.60**	**0.547**	**-.1243471**	**.2347329**
__00006W	**.2464766**	**.1838457**	**1.34**	**0.180**	**-.1139525**	**.6069056**
__00006X	**.097147**	**.065464**	**1.48**	**0.138**	**-.031195**	**.2254891**
__00006Y	**.1130417**	**.0411917**	**2.74**	**0.006**	**.0322854**	**.1937981**
__00006Z	**-.1040381**	**.0313626**	**-3.32**	**0.001**	**-.1655243**	**-.0425519**

__000070	-.0975131	.0491546	-1.98	0.047	-.1938805	-.0011457
__000071	-.0342095	.062284	-0.55	0.583	-.1563171	.0878981
__000072	-.5269194	.3543749	-1.49	0.137	-1.221671	.1678319
__000073	-.0468236	.0544254	-0.86	0.390	-.1535244	.0598773
__000074	.2469635	.1436979	1.72	0.086	-.034756	.5286829
__000075	.2392732	.3375853	0.71	0.478	-.422562	.9011085
__000076	.1281287	.0813462	1.58	0.115	-.0313503	.2876076
__000077	.0200856	.1290184	0.16	0.876	-.2328546	.2730258
__000078	-.0140711	.0556515	-0.25	0.800	-.1231758	.0950335
__000079	.0814369	.0374041	2.18	0.030	.0081063	.1547676
__00007A	.0421845	.0252955	1.67	0.095	-.0074073	.0917763
__00007B	-.0024611	.0375964	-0.07	0.948	-.0761689	.0712467
__00007C	.0120392	.0431916	0.28	0.780	-.0726377	.0967162
__00007D	.3125724	.2440836	1.28	0.200	-.165953	.7910978
__00007E	.025896	.0403614	0.64	0.521	-.0532324	.1050243
__00007F	-.1596487	.0964736	-1.65	0.098	-.348785	.0294876
__00007G	-.051805	.2549994	-0.20	0.839	-.5517309	.4481208
__00007H	-.0953725	.0592071	-1.61	0.107	-.211448	.0207029
__00007I	-.1163743	.1102801	-1.06	0.291	-.3325782	.0998296
__00007J	.018921	.038441	0.49	0.623	-.0564426	.0942845
__00007K	-.0869756	.0300097	-2.90	0.004	-.1458095	-.0281417
__00007L	-.0127916	.0204976	-0.62	0.533	-.0529771	.0273939
__00007M	.010521	.0282985	0.37	0.710	-.0449581	.0660001
__00007N	-.0180405	.0350262	-0.52	0.607	-.0867092	.0506282
__00007O	-.0081516	.8382152	-0.01	0.992	-1.651471	1.635167
__00007P	.3107059	.1183335	2.63	0.009	.0787133	.5426985
__00007Q	-.4109411	.3260791	-1.26	0.208	-1.050218	.2283363
__00007R	.3407095	.7987095	0.43	0.670	-1.225159	1.906578
__00007S	-.5534748	.1682856	-3.29	0.001	-.8833984	-.2235513
__00007T	-.2196563	.2969799	-0.74	0.460	-.8018847	.3625721
__00007U	.2872022	.1201019	2.39	0.017	.0517427	.5226617
__00007V	-.2009352	.0679227	-2.96	0.003	-.3340976	-.0677728
__00007W	.0140617	.0573991	0.24	0.806	-.098469	.1265924
__00007X	-.056224	.0826135	-0.68	0.496	-.2181876	.1057397
__00007Y	-.0080098	.094626	-0.08	0.933	-.1935238	.1775043
__000080	23.04426	17.99441	1.28	0.200	-12.23374	58.32226
__000082	6.739301	10.28004	0.66	0.512	-13.4147	26.8933
__000085	.0366167	.4613836	0.08	0.937	-.8679248	.9411582
__000087	.1273868	.3305412	0.39	0.700	-.5206385	.775412
__00008A	-4.520475	3.435897	-1.32	0.188	-11.25654	2.215594
__00008C	-1.253674	1.946983	-0.64	0.520	-5.070729	2.563382
__00008F	-28.44369	22.68575	-1.25	0.210	-72.91905	16.03166
__00008H	-8.673547	13.05254	-0.66	0.506	-34.26303	16.91593
__00008K	-.3171968	.5838269	-0.54	0.587	-1.461788	.8273946
__00008M	-1.378233	.5476595	-2.52	0.012	-2.451918	-.304548
__00008P	1.346432	1.090681	1.23	0.217	-.791847	3.48471
__00008R	-1.460585	.8627891	-1.69	0.091	-3.152082	.2309109
__00008U	-1.032953	.4043974	-2.55	0.011	-1.825773	-.2401328
__00008W	.5845627	.3269034	1.79	0.074	-.0563307	1.225456
__00008Z	.3738756	.2324088	1.61	0.108	-.0817613	.8295126
__000091	.0382166	.1811049	0.21	0.833	-.316839	.3932723
__000094	-.6116784	.2279377	-2.68	0.007	-1.05855	-.1648071
__000096	-.1050342	.1884987	-0.56	0.577	-.4745854	.2645171
__000099	.6643731	.2628324	2.53	0.012	.1490908	1.179656
__00009B	.2677219	.2431047	1.10	0.271	-.2088842	.7443281
__00009E	-.1939407	.3366949	-0.58	0.565	-.8540303	.4661489
__00009G	-.2679983	.2820832	-0.95	0.342	-.8210218	.2850252
xbarlag	-31.50109	20.33454	-1.55	0.121	-71.36692	8.364735
x2barlag	-.4339314	.5076501	-0.85	0.393	-1.429178	.5613155
xbar2lag	6.658686	3.973539	1.68	0.094	-1.131428	14.4488
rootxbarlag	38.90188	25.41688	1.53	0.126	-10.92786	88.73163
z1barlag	1.921348	.8663586	2.22	0.027	.2228537	3.619842
z2barlag	.04958	1.394448	0.04	0.972	-2.684232	2.783392
z3barlag	.4241796	.5396868	0.79	0.432	-.6338751	1.482234
z4barlag	-.2806967	.2681117	-1.05	0.295	-.8063291	.2449357
z5barlag	.8190618	.2280605	3.59	0.000	.3719497	1.266174
z6barlag	-.8933056	.411293	-2.17	0.030	-1.699645	-.0869666
z7barlag	.4867033	.4527264	1.08	0.282	-.4008658	1.374273
_cons	-13.94648	8.863303	-1.57	0.116	-31.32297	3.430003

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A2	84,052	.4027835	.1589901	.1248021	1.461195

```
956
957 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"
958
959 * interact qhat_hats with Alist
960 di "`qhat_hats'"
__00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9
961 if "$Alist"~="one" {
962         local qhat_hats_backup "`qhat_hats'"
963         local qhat_hats ""
964         foreach var1 of local qhat_hats_backup {
  2.                 foreach var2 of global Alist {
  3.                         g `var1'_`var2'=`var1'*`var2'
  4.                         sum `var1' `var2' `var1'_`var2'
  5.                         local qhat_hats "`qhat_hats' `var1'_`var2'"
  6.                 }
  7.         }
```

Variable	Obs	Mean	Std. dev.	Min	Max
__00009W	84,052	.4615108	.1003007	.1982114	1.064132
one	84,052	1	0	1	1
__00009W_one	84,052	.4615108	.1003007	.1982114	1.064132

Variable	Obs	Mean	Std. dev.	Min	Max
__00009W	84,052	.4615108	.1003007	.1982114	1.064132
zi7	84,052	.1309427	.3373397	0	1
__00009W_zi7	84,052	.0770198	.2018732	0	1.064132

Variable	Obs	Mean	Std. dev.	Min	Max
__00009Z	84,052	.5102133	.1199581	.1972201	1.195608
one	84,052	1	0	1	1
__00009Z_one	84,052	.5102133	.1199581	.1972201	1.195608

Variable	Obs	Mean	Std. dev.	Min	Max
__00009Z	84,052	.5102133	.1199581	.1972201	1.195608
zi7	84,052	.1309427	.3373397	0	1
__00009Z_zi7	84,052	.0857241	.225065	0	1.195608

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A5	84,052	.4407914	.189912	.1215886	1.546164
one	84,052	1	0	1	1
__0000A5_one	84,052	.4407914	.189912	.1215886	1.546164

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A5	84,052	.4407914	.189912	.1215886	1.546164
zi7	84,052	.1309427	.3373397	0	1
__0000A5_zi7	84,052	.1008931	.2711148	0	1.546164

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A8	84,052	.4829239	.2085856	.136642	1.972777
one	84,052	1	0	1	1
__0000A8_one	84,052	.4829239	.2085856	.136642	1.972777

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A8	84,052	.4829239	.2085856	.136642	1.972777
zi7	84,052	.1309427	.3373397	0	1
__0000A8_zi7	84,052	.1117997	.3010185	0	1.972777

Variable	Obs	Mean	Std. dev.	Min	Max
__00009X	84,052	.4702234	.3851815	.023333	3.618202
one	84,052	1	0	1	1
__00009X_one	84,052	.4702234	.3851815	.023333	3.618201

Variable	Obs	Mean	Std. dev.	Min	Max
__00009X	84,052	.4702234	.3851815	.023333	3.618202
zi7	84,052	.1309427	.3373397	0	1
__00009X_zi7	84,052	.1171405	.362284	0	3.618201

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A0	84,052	.5196248	.4319204	.0242968	4.40807
one	84,052	1	0	1	1
__0000A0_one	84,052	.5196248	.4319204	.0242968	4.408071

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A0	84,052	.5196248	.4319204	.0242968	4.40807
zi7	84,052	.1309427	.3373397	0	1
__0000A0_zi7	84,052	.1304855	.4063337	0	4.408071

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A6	84,052	.48585	.5328118	.0206382	6.614207
one	84,052	1	0	1	1
__0000A6_one	84,052	.48585	.5328118	.0206382	6.614207

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A6	84,052	.48585	.5328118	.0206382	6.614207
zi7	84,052	.1309427	.3373397	0	1
__0000A6_zi7	84,052	.1587206	.5187135	0	6.614207

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A9	84,052	.5337806	.599019	.0214907	8.280542
one	84,052	1	0	1	1
__0000A9_one	84,052	.5337806	.599019	.0214907	8.280542

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A9	84,052	.5337806	.599019	.0214907	8.280542
zi7	84,052	.1309427	.3373397	0	1
__0000A9_zi7	84,052	.1765088	.585031	0	8.280542

```
965 }

966
967 local r_RE      "`r_RE' `qhat_hats'"

968
969 /*******************************************************************************
  > Set instruments and start values
  > *******************************************************************************/
970
971 if simple_model==1      {
972         local r_RE "xbarlag `xixbarlag' `budget'"
973         local r_RE1 ""
```

```
974
975         forv i=1/`T' {
    2.                 local r_RE "`r_RE' zi`i'"
    3.         }
976
977         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
978 }

979
980 if `J'==2 & simple_model==0     {
981         g y = p1*qi1
982         g ybar_q1 = p1*qikbar1
983         g ybar_q2 = p2*qikbar2
984         g ybar = p1*qikbar1 + p2*qikbar2
985         g ybar2 = ybar^2
986         g ybar_q1_2 = ybar_q1^2
987         g ybar_q2_2 = ybar_q2^2
988         g ybar_q1_q2 = ybar_q1*ybar_q2
989         g ybarx = ybar*xi
990         g ybarx_1 = ybar_q1*xi
991         g ybarx_2 = ybar_q2*xi
992         g xi2 = xi^2
993
994         forv j = 1(1)`T' {
    2.                 g pzi`j' = p1*zi`j'
    3.         }
995
996         if same_spillover==1 {
997                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS	Number of obs	=	84,052
				F(13, 84038)	=	39936.27
Model	3720.99829	13	286.230637	Prob > F	=	0.0000
Residual	602.31584	84,038	.007167184	R-squared	=	0.8607
				Adj R-squared	=	0.8607
Total	4323.31413	84,051	.051436796	Root MSE	=	.08466

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	.0209725	.0010689	19.62	0.000	.0188775	.0230676
ybarx	-.0031615	.001283	-2.46	0.014	-.0056762	-.0006468
xi2	-.0408396	.0007404	-55.16	0.000	-.0422908	-.0393884
ybar	-.1934306	.0030554	-63.31	0.000	-.1994191	-.187442
xi	.4424816	.0019664	225.02	0.000	.4386274	.4463358
ybar_q1	.3958873	.0037564	105.39	0.000	.3885247	.4032498
pzi1	.2069751	.0015401	134.39	0.000	.2039564	.2099938
pzi2	.0157768	.0024828	6.35	0.000	.0109106	.0206431
pzi3	-.003048	.0007626	-4.00	0.000	-.0045427	-.0015534
pzi4	.0120017	.0008158	14.71	0.000	.0104028	.0136007
pzi5	-.0120278	.0007743	-15.53	0.000	-.0135454	-.0105103
pzi6	.0000399	.0006329	0.06	0.950	-.0012005	.0012804
pzi7	-.0055547	.0010894	-5.10	0.000	-.0076899	-.0034195
_cons	-.0186167	.0017721	-10.51	0.000	-.0220901	-.0151433

```
998                 local b = _b[xi]
999                 local a = _b[ybar_q1]
1000                if same_spillover==1 {
1001                        local a = logit(_b[ybar_q1]/2 + .5)
```

```
1002                        if `a'==. local a = .5
1003                }
1004                local d = _b[xi2]
1005                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1006        }
1007        else {
1008                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1009                local b = _b[xi]
1010                local a1 = _b[ybar_q1]/(1-`b')
1011                local a2 = _b[ybar_q2]/`b'
1012                if same_spillover==1 {
1013                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1014                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1015
1016                        if `a1'==. local a1 = .5
1017                        if `a2'==. local a2 = .5
1018                }
1019                local d = _b[xi2]
1020
1021                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1022        }
1023
1024        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1025        forv j = 1(1)`T' {
    2.                  local coef = _b[pzi`j']
    3.                  local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1026}

1027
1028if `J'==3 & simple_model==0
  >                             local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1030if `J'==4 & simple_model==0
  >                             local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1032
1033if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1034
1035if `J'==2 & estimate_RE==1 {
1036
1037        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00424632
  Iteration 1:  GMM criterion Q(b) =  .00069675
  Iteration 2:  GMM criterion Q(b) =  .00019341
  Iteration 3:  GMM criterion Q(b) =   .0001273

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00620424
  Iteration 1:  GMM criterion Q(b) =  .00547721
  Iteration 2:  GMM criterion Q(b) =  .00518403
  Iteration 3:  GMM criterion Q(b) =  .00496645
```

GMM estimation

Number of parameters = **21**
Number of moments = **36**
Initial weight matrix: **Unadjusted** Number of obs = **84,052**
GMM weight matrix: **Robust**

(Std. err. adjusted for **561** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
A11						
one	.6939831	.041191	16.85	0.000	.6132503	.774716
zi7	-.185733	.0569843	-3.26	0.001	-.2974202	-.0740458
C1						
zi1	.4600308	.0477361	9.64	0.000	.3664697	.5535919
zi2	-.2907914	.0857827	-3.39	0.001	-.4589224	-.1226604
zi3	.0182132	.0230568	0.79	0.430	-.0269772	.0634036
zi4	.1060752	.0263305	4.03	0.000	.0544684	.157682
zi5	.0462677	.024554	1.88	0.060	-.0018572	.0943925
zi6	-.0115799	.0170766	-0.68	0.498	-.0450494	.0218895
zi7	.0016236	.0307559	0.05	0.958	-.058657	.0619041
C2						
zi1	.2374361	.0722832	3.28	0.001	.0957637	.3791085
zi2	-.49054	.1327112	-3.70	0.000	-.7506491	-.2304309
zi3	.0342812	.0357279	0.96	0.337	-.0357443	.1043066
zi4	.1673947	.0398482	4.20	0.000	.0892937	.2454957
zi5	.082194	.0303608	2.71	0.007	.0226879	.1417001
zi6	-.0235384	.0237182	-0.99	0.321	-.0700252	.0229485
zi7	-.0768932	.0404636	-1.90	0.057	-.1562004	.0024141
/d1	-.0920781	.0080494	-11.44	0.000	-.1078548	-.0763015
/b1	.4188999	.0243033	17.24	0.000	.3712662	.4665335
/AVA11	.5626649	.6275175	0.90	0.370	-.6672469	1.792577
/AVA12	-.4995774	.5468859	-0.91	0.361	-1.571454	.5722992
/AVA22	.0305261	.5201783	0.06	0.953	-.9890046	1.050057

Instruments for equation **eq1_RE**: **__00009J __00009L __00009O xi __000002 __00000E __00000P __000017 __0000II __000020 __00002B __00002T __000034 __00003M __00003X __00004F __00004Q __000058 __00005J __00009W_one __00009W_zi7 __00009Z_one __00009Z_zi7 __0000A5_one __0000A5_zi7 __0000A8_one __0000A8_zi7 __00009X_one __00009X_zi7 __0000A0_one __0000A0_zi7 __0000A6_one __0000A6_zi7 __0000A9_one __0000A9_zi7 _cons**

```
1038
1039        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1040        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1041
1042        if simple_model==1 {
1043                // calculate derivative and save
1044                local RE_acz_drv ""
1045                tempvar junk
1046                g `junk'=.
1047                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
```

```
1048                replace `junk'=p1*qikbar1
1049                su `junk' $GMM_weight
1050                local mean_y = r(mean)
1051                su xi $GMM_weight
1052                local mean_x = r(mean)
1053
1054                local c_term ""
1055
1056                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1057                estimates store gmm_est
1058                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1059                estimates save "${filename_RE}_dF", replace
1060                estimates restore gmm_est
1061        }
1062}

1063
1064if `J'==3 & estimate_RE==1 {
1065        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1066}

1067if `J'==4 & estimate_RE==1 {
1068        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1069}

1070esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXehigh_drop3_all_ex
      > actInst_noPPXdInst_14_zlist4_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not
      found)
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXehigh
  > _drop3_all_exactInst_noPPXdInst_14_zlist4_updateExp_urbOnly_clus_Dist_dropPre_RE.tex
  > )

1071
1072
1073// save coefs
1074matrix est_b=e(b)

1075unique hhidi
  Number of unique values of hhidi is  20233
  Number of records is  84052

1076local N_hh = r(sum)

1077estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  20233
```

```
1078unique group_round
  Number of unique values of group_round is  4446
  Number of records is  84052

1079estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  4446

1080estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
         e(avg_grp_size) =  4.5508322

1081if estimate_RE==1 {
1082        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXehigh_drop3_
  > all_exactInst_noPPXdInst_14_zlist4_updateExp_urbOnly_clus_Dist_dropPre_RE.ster not f
  > ound)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXehigh_drop3_all_exa
  > ctInst_noPPXdInst_14_zlist4_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1083        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1084}

1085
1086estimates store temp1_FE

1087
1088/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1089
1090if simple_model==0 {
1091
1092        // make FE instruments
1093
1094        // denoting ximxk as x and zimzk of individual zs as zi:
1095        //   x,x2,p,zi,px,zi,pzi
1096        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1097        local xdiff "`ximxk' "
1098        local xdiffxdiff ""
1099        local zindivdiff ""
1100        local pzindivdiff ""
1101        local pzindivdiffpqhat ""
1102        local pxzindivdiff ""
1103        local p2zindiv2diff ""
1104        local zg ""
1105        local pzg ""
1106        local zindivdiffxdiff ""
1107        local pzindivdiffxdiff ""
1108        local pzindivdiffp ""
1109        local pzindivdiffpzg ""
1110        local zgxdiff ""
1111        local pzgxdiff ""
1112        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1113        g double `ximxkxdiff'=`ximxk'*`ximxk'
1114        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1115        g double `xixkxdiff'=xi*xk*`ximxk'
1116
```

```
1117        local xdiffxdiff "`x2imx2k'"
1118
1119        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
1120                                                gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                  local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                          }
   30.                                  }
   31.                          }
   32.                  }
   33.          }
1121        if `Tindivp1'<=`T' {
1122                forval t=`Tindivp1'/`T' {
    2.                          tempvar zi`t'xdiff
    3.                          g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                          local zg "`zg' zi`t'"
    5.                          forval j=1/`J' {
    6.                                  tempvar zi`t'p`j'xdiff
    7.                                  g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                  local pzg "`pzg' `zi`t'p`j''"
    9.                                  local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                          }
   11.                  }
1123        }
```

```
1124
1125        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1126        // denoting zg as group-zs
1127        //  zg,pzgx,rprp,rprpx
1128        local rootprootp ""
1129        local rootprootpxdiff ""
1130
1131        forval j=1/`J' {
    2.                  forval s=`j'/`J' {
    3.                          tempvar xdiffrp`j'rp`s'
    4.                          g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1132                        if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1133                        local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                  }
    8.          }
1134
1135        local pqhatxdiff ""
1136        forval j=1/`J' {
    2.                  forval s=1/`J' {
    3.                          foreach name of global Alist {
    4.                                  tempvar   p`s'qikhat`j'xdiff`name'
    5.                                  g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
1137                                local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
1138
1139        // set instruments
1140        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1141}

1142
1143
1144if simple_model==1 {
1145        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1146
1147        /******************
  >         Initial values
  >         ******************/
1148
1149        if init_FE_from_RE==1 {
1150                estimates use "$filename_RE"
1151
1152                // clear init values
1153                local initial_values_FE ""
1154
1155                // extract vector of coefs and paramter names from
1156                matrix coefs = e(b)
1157                local paramlist = e(params)
```

```
1158
1159                // iterate through paramter name list, taking
1160                local t=0
1161                foreach p of local paramlist {
    2.                          local `++t'
    3.                          local p_nice = subinstr("`p'",":_cons","",.)
    4.                          local est = coefs[1,`t']
    5.                          if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                  }
1162        }
1163
1164        // make z diff instruments
1165        local pzindivdiff ""
1166        local pz2indivdiff ""
1167        local pzXzindivdiff ""
1168        local pxzindivdiff ""
1169
1170        forv i=1/`T' {
    2.
1171                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1172                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1173                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1174                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                 }
   16.         }
1175
1176        // p z interactions
1177        local pzg ""
1178        local pzgxdiff ""
1179        if `Tindivp1'<=`T' {
1180                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1181        }
1182
1183        tempvar p1ximxk  p1x2imx2k
1184        g `p1ximxk' = p1*`ximxk'
1185        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1186
1187        // define instruments
1188        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
```

```
1189
1190        // add prediction of quantity as extra instrument
1191        local pqhat ""
1192        local qhat ""
1193        local pqhatxdiff ""
1194        local pqhatxdiff2 ""
1195        local pzindivdiffpqhat ""
1196        local pzindivdiffpxbar_lag ""
1197
1198        tempvar qik_hat1
1199         reg qikbar1    `r_RE' `qhat_insts', vce(cluster group_round)
1200        predict `qik_hat1'
1201        local qhat "`qhat' `qik_hat1'"
1202
1203        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1204
1205        g double `p1_qikbar1' = p1*qikbar1
1206        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1207        predict `p1_qikhat1'
1208        local pqhat "`pqhat' `p1_qikhat1'"
1209
1210        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1211        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1212
1213        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1214
1215        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1216        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1217        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1218
1219        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1220                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1221
1222        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1223}

1224
1225// starting values
1226if init_FE_from_RE==1 & "$init_FE_file"!="" {
1227        estimates use "$init_FE_file"
1228        local initial_values_FE ""
1229
1230        // extract vector of coefs and paramter names from
1231        matrix coefs = e(b)
1232        local paramlist = e(params)
1233
1234        // iterate through paramter name list, taking
1235        local t=0
1236        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
```

```
1237}

1238
1239if estimate_FE==1 {
1240        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique
  note: instruments __00005B __00005M __0000D7 __0000DA __0000DG __0000DJ __0000DP
    __0000DQ __0000DS __0000DT __0000DY __0000DZ __0000E1 __0000E2 omitted because
    of collinearity.

  Step 1
  Iteration 0:  GMM criterion Q(b) =   .0006442
  Iteration 1:  GMM criterion Q(b) =  .00027344
  Iteration 2:  GMM criterion Q(b) =  .00025252

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01222079
  Iteration 1:  GMM criterion Q(b) =  .01075907
  Iteration 2:  GMM criterion Q(b) =  .01056771
  Iteration 3:  GMM criterion Q(b) =  .01054598

  GMM estimation

  Number of parameters =  18
  Number of moments    =  81
  Initial weight matrix: Unadjusted                 Number of obs   =     84,052
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 561 clusters in state_district_round)
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11						
_cons	.5868161	.0268372	21.87	0.000	.5342162	.6394159
C12						
_cons	-.1281767	.0474684	-2.70	0.007	-.221213	-.0351404
C13						
_cons	-.0276503	.013544	-2.04	0.041	-.054196	-.0011046
C14						
_cons	.0068438	.0128821	0.53	0.595	-.0184046	.0320922
C15						
_cons	-.0116728	.0196171	-0.60	0.552	-.0501215	.0267759
C16						
_cons	-.0281605	.0136523	-2.06	0.039	-.0549184	-.0014025
C17						
_cons	-.0707912	.3051384	-0.23	0.817	-.6688514	.5272691
C21						
_cons	.3999368	.0356767	11.21	0.000	.3300118	.4698618
C22						
_cons	-.2010722	.0645929	-3.11	0.002	-.3276719	-.0744725
C23						
_cons	-.03224	.0171428	-1.88	0.060	-.0658394	.0013593
C24						
_cons	-.0095652	.0199939	-0.48	0.632	-.0487526	.0296221
C25						
_cons	.0040589	.0230924	0.18	0.860	-.0412014	.0493192
C26						

	_cons	-.0359734	.0165686	-2.17	0.030	-.0684473	-.0034995
C27							
	_cons	-.6190718	.2804978	-2.21	0.027	-1.168838	-.0693062
d1							
	_cons	-.0942569	.0070045	-13.46	0.000	-.1079855	-.0805284
b1							
	_cons	.4880181	.01973	24.73	0.000	.4493479	.5266882
A11							
	one	.1453645	.1426805	1.02	0.308	-.1342841	.4250131
	zi7	.4540851	.1631542	2.78	0.005	.1343087	.7738616

Instruments for equation **eq1_FE:** __000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T o.__00005B o.__00005M __000005 __0000E6 __0000E7 __0000E8 __0000E9 __0000EA __0000EB __0000EC __0000ED __0000AJ __0000AK __0000AM __0000AN __0000AS __0000AT __0000AV __0000AW __0000B2 __0000B3 __0000B5 __0000B6 __0000BB __0000BC __0000BE __0000BF __0000BL __0000BM __0000BO __0000BP __0000BU __0000BV __0000BX __0000BY __0000C4 __0000C5 __0000C7 __0000C8 __0000CD __0000CE __0000CG __0000CH __0000CN __0000CO __0000CQ __0000CR __0000CW __0000CX __0000CZ __0000D0 __0000D6 o.__0000D7 __0000D9 o.__0000DA __0000DF o.__0000DG __0000DI o.__0000DJ o.__0000DP o.__0000DQ o.__0000DS o.__0000DT o.__0000DY o.__0000DZ o.__0000E1 o.__0000E2 __0000AG __0000AP __0000AZ __0000B8 __0000BI __0000BR __0000C1 __0000CA __0000CK __0000CT __0000D3 __0000DC __0000DM __0000DV _cons

```
1241
1242        esttab using "${filename_FE}.tex", se replace //added this
```

(file **diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXehigh_drop3_all_ex > actInst_noPPXdInst_14_zlist4_updateExp_urbOnly_clus_Dist_dropPre_FE.tex** not found)

(output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXehigh > _drop3_all_exactInst_noPPXdInst_14_zlist4_updateExp_urbOnly_clus_Dist_dropPre_FE.tex >)

```
1243
1244        estat overid
```

Test of overidentifying restriction:

Hansen's J chi2(**63**) = **886.411** (p = **0.0000**)

```
1245        di "$S_DATE $S_TIME"
```

7 Dec 2023 21:00:53

```
1246        unique hhidi
```

Number of unique values of hhidi is **20233**
Number of records is **84052**

```
1247        local N_hh = r(sum)
1248        estadd scalar N_hh = r(sum)
```

added scalar:
e(N_hh) = **20233**

```
1249        unique group_round
```

Number of unique values of group_round is **4446**
Number of records is **84052**

```
1250        estadd scalar N_grp = r(sum)
```

added scalar:
e(N_grp) = **4446**

```
1251        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
         e(avg_grp_size) =  4.5508322
1252        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXehigh_drop3_
  > all_exactInst_noPPXdInst_14_zlist4_updateExp_urbOnly_clus_Dist_dropPre_FE.ster not f
  > ound)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXschedXehigh_drop3_all_exa
  > ctInst_noPPXdInst_14_zlist4_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1253
1254        if simple_model==1 {
1255                local RE_acz_drv ""
1256                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1257                noi di "`RE_acz_drv'"
1258
1259                cap g junk=.
1260                replace junk=p1*qikbar1
1261                su junk $GMM_weight
1262                local mean_y = r(mean)
1263                su xi $GMM_weight
1264                local mean_x = r(mean)
1265
1266                estimates store gmm_est
1267                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1268                estimates save "${filename_FE}_dF", replace
1269                estimates restore gmm_est
1270        }
1271}

1272
1273
  end of do-file
1274}

1275else {
1276        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1277        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1278
1279        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1280
1281        drop if size_group_round<min_group_size
1282        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1283
1284        do "$CODE/estim program.do"
1285}

1286
1287
1288log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table4_48.smcl
   log type:  smcl
  closed on:   7 Dec 2023, 21:00:53
```

A.5 Table 5

A.5.1 Columns 1 and 3

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table5_13.smcl
  log type:  smcl
 opened on:   7 Dec 2023, 21:19:27

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=1

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
50. //local group_def "dist"
51. //local group_def "fsuXseg"
52. local group_def "fsuXsegXreligXsched"

53.
54. // comparision group
55. local groupComp_def ""

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.          syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
70.          if "`rename'" == "" local rename = "`varlist'bar"
    3.
```

```
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations            1,118    ->   280
  Number of variables                   5    ->   7
  j variable (4 values)       expend_type    ->   (dropped)
  xij variables:
                         laspeyres_state    ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560    ->   280
  Number of variables                   5    ->   5
  j variable (2 values)       expend_type    ->   (dropped)
  xij variables:
                         laspeyres_state    ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                  Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations                 840   ->   280
  Number of variables                      5   ->   6
  j variable (3 values)          expend_type   ->   (dropped)
  xij variables:
                              laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,637 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
```

```
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }
297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }
316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }
333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
```

```
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1
  (61,772 observations deleted)

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (22 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (22 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (11,804 real changes made, 22 to missing)

460 replace x2=x*x
  (11,804 real changes made, 22 to missing)

461
462 // make z x interactions
```

```
. foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (22 missing values generated)
  (22 missing values generated)
  (22 missing values generated)
  (22 missing values generated)
  (22 missing values generated)
  (22 missing values generated)
  (22 missing values generated)

. 
. levelsof round, local(roundlist)
  59 60 61 62

. foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

. 
. g xbar2lag=xbarlag^2
  (149 missing values generated)

. g rootxbarlag=sqrt(xbarlag)
  (149 missing values generated)

. 
. bysmeanw x, weight(weight) by(group) rename(xbar)

. bysmeanw x2, weight(weight) by(group) rename(x2bar)

. g xbar2=xbar^2

. g rootxbar = sqrt(xbar)

. foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

. 
. replace x=x_temp
  (11,804 real changes made)

. replace x2=x2_temp
  (11,804 real changes made)

. 
. // add x terms to inst lsit
. global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"
```

```
482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (9,374 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)

488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei
```

```
511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_41e4_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521               local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup
```

```
543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (2,430 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
549 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var''" _n
    5.                 }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }
```

```
578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0
```

```
620 local measError_desc ""
621 if noMeasError==1 local measError_desc "_NoMeasError"
622
623 // estimate RE and or FE
624 scalar estimate_RE=1
625 scalar estimate_FE=1
626
627 // estimation options
628 global w_initial "winit(unadjusted)"
629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"
630 global wmatrix "wmatrix(robust)"
631 global trace_level ""
632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"
633 global technique ""
634 global maxiter ""
635 global technique_RE ""
636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"
637 global derivatives "quickderivatives"
638 scalar init_FE_from_RE=1
639 global clust_var "state_district_round"
640 local clus_name "_clus_Dist"
641
642
643
644
645
646
647
648
649 // xxzx to keep names same
650 local wmatrix_desc ""
651 local sectorName "_urbOnly"
652 local dropZdesc "_dropPre"
653 local desc_struc_v0 ""
654 local inst_desc "_exactInst"
655 local inst_rootp "_noPPXdInst"
656 local weight_desc ""
```

```
657 local RE_sq_inst_desc ""

658 local cnstname ""

659 local actual_pqhat_desc ""

660 local expUpname "_updateExp"

661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //               ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731
```

```
690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T

694 local Tm1=`T'-1

695 local Tindiv : word count $zlist_indiv

696
697 /*****************************************************************************
  > Set weights
  > *****************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*****************************************************************************
  >  Update data for J=2, and J=3
  > *****************************************************************************/
706
707 if `J'==2 {
708          replace p1=P1
  (9,072 real changes made)
709          replace p2=P2
  (9,072 real changes made)
710          replace qi1=Qi1
  (9,072 real changes made)
711          replace qk1=Qk1
  (9,072 real changes made)
712          replace qikbar1=Qikbar1
  (9,072 real changes made)
713          replace qi2=Qi2
  (9,072 real changes made)
714          replace qk2=Qk2
  (9,072 real changes made)
715          replace qikbar2=Qikbar2
  (9,072 real changes made)
716
717          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721          forv j = 1(1)3 {
    2.                 replace p`j'=P`j'
    3.                 replace qi`j'=Qi`j'
    4.                 replace qk`j'=Qk`j'
    5.                 replace qikbar`j'=Qikbar`j'
    6.          }
```

```
722
723         foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                 qui replace `var'=.
    3.         }
724 }

725
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.         global zilist "$zilist zi`t'"
    3.         global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.         local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.         local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.         local pCi`j' "( 0 "
    3.         local pCk`j' "( 0 "
    4.         local pCg`j' "( 0 "
    5.         forval t=1/`Tindiv' {
    6.                 local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                 local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.         }
    9.         forval t=`Tindivp1'/`T' {
   10.                 local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.         }
   12.         local pCi`j' "`pCi`j'' )"
   13.         local pCk`j' "`pCk`j'' )"
   14.         local pCg`j' "`pCg`j'' )"
   15. }
```

```
751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
773         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.          local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.          if `jplus1'<=`J' {
    5.                  forval k=`jplus1'/`J' {
    6.                          local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                  }
    8.          }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                  if noMeasError==1                               local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
```

```
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                              l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                    ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2
  > ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qikba
  > r1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d
  > 1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > }*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))        ))
```

```
836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) +
  >  ({A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C
  > 2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikb
  > ar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:
  > }) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C
  > 2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qi
  > kbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )
  > *{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > one}*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}
  > *ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi
  > 4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p
  > 1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         )
  > )

874
875 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi

882 g double `ximxk'=xi-xk
```

```
883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag
  (110 missing values generated)

886 g xbarlag2  = xbarlag^2
  (110 missing values generated)

887 g x2barlag2 = x2barlag^2
  (110 missing values generated)

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.         tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.         g double `xizi`t''=xi*zi`t'
    4.         g double `xkzk`t''=xk*zk`t'
    5.         g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.         g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.         g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.         g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.         g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.         forval j=1/`J' {
   11.                 tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                 g double `zi`t'p`j''= zi`t'*p`j'
   13.                 g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                 g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                 g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                 forv s=1/`T' {
   17.                         tempvar zi`t'zi`s'p`j'
   18.                         g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                 }
   20.         }
   21. }
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)

891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1
```

```
// lag instruments
forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }

local zrg ""

forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var''"
    6.          }
    7. }
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
```

```
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)

906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.         tempvar xi`var'
    3.         g double `xi`var''=xi*`var'
    4.         local xrg "`xrg' `xi`var''"
    5.
911         forval j=1/`J' {
    6.                 tempvar p`j'`var' xp`j'`var'
    7.                 g double `p`j'`var''=p`j'*`var'
    8.                 g double `xp`j'`var''=xi*p`j'*`var'
    9.                 local prg "`prg' `p`j'`var''"
   10.                 local xprg "`xprg' `xp`j'`var''"
   11.         }
   12. }
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
```

```
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.         tempvar xip`j'
    3.         g double `xip`j''=xi*p`j'
    4.         local pinsts "`pinsts' p`j'"
    5.         local xpinsts "`xpinsts' `xip`j''"
    6.         forval s=`j'/`J' {
    7.                 tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                 g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                 g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                 local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                 local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.         }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""

925 local xPzinsts ""

926 local pzinsts ""
```

```
927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t'`"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t'`"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j'`"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j'`"
    8.          }
    9. }

929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.          forval k=`j'/`J' {
    3.                  tempvar p`j'p`k'
    4.                  g `p`j'p`k''=p`j'*p`k'
    5.                  local roundinsts "`roundinsts' `p`j'p`k'`"
    6.          }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.          if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.          if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
```

```
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }
```

Linear regression

Number of obs	**8,962**
F(129, 563)	**10.22**
Prob > F	**0.0000**
R-squared	**0.2440**
Root MSE	**.11384**

(Std. err. adjusted for **564** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**5.747719**	**5.720346**	**1.00**	**0.315**	**-5.488107**	**16.98355**
__00009L	**-4.271324**	**2.60538**	**-1.64**	**0.102**	**-9.388776**	**.8461287**
__00009O	**3.80121**	**3.464076**	**1.10**	**0.273**	**-3.002882**	**10.6053**
xi	**.1409618**	**.0222163**	**6.34**	**0.000**	**.0973247**	**.1845988**
__000002	**-.0416414**	**.0072964**	**-5.71**	**0.000**	**-.055973**	**-.0273098**
__00000E	**.3590801**	**.2137539**	**1.68**	**0.094**	**-.0607724**	**.7789326**
__00000P	**-.159779**	**.1263228**	**-1.26**	**0.206**	**-.4079006**	**.0883425**
__000017	**.2534986**	**.3147365**	**0.81**	**0.421**	**-.3647027**	**.8716999**
__00001I	**-.1573448**	**.197102**	**-0.80**	**0.425**	**-.54449**	**.2298003**
__000020	**.1192467**	**.0970974**	**1.23**	**0.220**	**-.0714707**	**.3099641**
__00002B	**-.0462725**	**.0566339**	**-0.82**	**0.414**	**-.1575121**	**.0649671**
__00002T	**.5482488**	**.2011203**	**2.73**	**0.007**	**.1532111**	**.9432866**
__000034	**-.2134848**	**.1477549**	**-1.44**	**0.149**	**-.503703**	**.0767334**
__00003M	**.0446375**	**.0941473**	**0.47**	**0.636**	**-.1402854**	**.2295604**
__00003X	**-.075696**	**.0569539**	**-1.33**	**0.184**	**-.187564**	**.0361721**
__00004F	**-.1085238**	**.0738198**	**-1.47**	**0.142**	**-.2535196**	**.036472**
__00004Q	**-.0497479**	**.0489783**	**-1.02**	**0.310**	**-.1459505**	**.0464546**
__000058	**-.1285494**	**.1368329**	**-0.94**	**0.348**	**-.3973148**	**.140216**
__00005J	**-.0626132**	**.0807349**	**-0.78**	**0.438**	**-.2211916**	**.0959651**
__00005U	**1.16441**	**.6665571**	**1.75**	**0.081**	**-.1448321**	**2.473653**
__00005V	**.2848836**	**.1463783**	**1.95**	**0.052**	**-.0026307**	**.5723979**
__00005W	**-.6931706**	**.2365403**	**-2.93**	**0.004**	**-1.15778**	**-.2285614**
__00005X	**-.894838**	**.6883225**	**-1.30**	**0.194**	**-2.246832**	**.4571558**
__00005Y	**.1055045**	**.1306049**	**0.81**	**0.420**	**-.1510279**	**.3620369**
__00005Z	**-.317496**	**.2354067**	**-1.35**	**0.178**	**-.7798786**	**.1448866**
__000060	**-.0562026**	**.0953142**	**-0.59**	**0.556**	**-.2434175**	**.1310123**
__000061	**.1664415**	**.1093609**	**1.52**	**0.129**	**-.0483637**	**.3812467**
__000062	**-.0237775**	**.0608635**	**-0.39**	**0.696**	**-.1433249**	**.0957698**
__000063	**.0051224**	**.0521611**	**0.10**	**0.922**	**-.0973317**	**.1075766**
__000064	**-.047712**	**.0627829**	**-0.76**	**0.448**	**-.1710293**	**.0756053**
__000065	**.4992916**	**1.269608**	**0.39**	**0.694**	**-1.994455**	**2.993039**
__000066	**.0248558**	**.2329406**	**0.11**	**0.915**	**-.432683**	**.4823945**
__000067	**-.09665**	**.4521117**	**-0.21**	**0.831**	**-.9846818**	**.7913818**
__000068	**-.3672573**	**1.336104**	**-0.27**	**0.784**	**-2.991614**	**2.2571**
__000069	**.1676445**	**.222131**	**0.75**	**0.451**	**-.2686622**	**.6039513**
__00006A	**-.5265561**	**.3981856**	**-1.32**	**0.187**	**-1.308667**	**.2555547**
__00006B	**.0940075**	**.1381593**	**0.68**	**0.497**	**-.177363**	**.3653781**
__00006C	**.0719998**	**.1824399**	**0.39**	**0.693**	**-.2863462**	**.4303457**
__00006D	**.0091813**	**.1017359**	**0.09**	**0.928**	**-.190647**	**.2090097**
__00006E	**-.1374774**	**.0968261**	**-1.42**	**0.156**	**-.327662**	**.0527072**
__00006F	**-.0976161**	**.1332172**	**-0.73**	**0.464**	**-.3592795**	**.1640473**
__00006G	**.1200228**	**.3717333**	**0.32**	**0.747**	**-.6101307**	**.8501763**
__00006H	**.0323949**	**.0638603**	**0.51**	**0.612**	**-.0930386**	**.1578283**

00006I	-.0569992	.1416989	-0.40	0.688	-.3353222	.2213237
00006J	-.2216452	.3760438	-0.59	0.556	-.9602653	.516975
00006K	.0102401	.0799055	0.13	0.898	-.1467093	.1671894
00006L	.170364	.1436548	1.19	0.236	-.1118008	.4525288
00006M	.0107019	.0514742	0.21	0.835	-.090403	.1118068
00006N	.0465856	.0659902	0.71	0.481	-.0830315	.1762026
00006O	.0055152	.0293338	0.19	0.851	-.0521019	.0631323
00006P	-.0932843	.0326029	-2.86	0.004	-.1573224	-.0292462
00006Q	-.0260765	.0402388	-0.65	0.517	-.1051131	.05296
00006R	1.228252	.7293314	1.68	0.093	-.2042905	2.660795
00006S	.0434222	.1781383	0.24	0.808	-.3064747	.3933191
00006T	-.3660491	.2621605	-1.40	0.163	-.8809811	.148883
00006U	-1.437678	.6640729	-2.16	0.031	-2.742041	-.133315
00006V	.4508507	.1205105	3.74	0.000	.2141456	.6875557
00006W	.0014559	.2432167	0.01	0.995	-.476267	.4791787
00006X	.009136	.0746016	0.12	0.903	-.1373954	.1556675
00006Y	-.0227005	.0960214	-0.24	0.813	-.2113044	.1659034
00006Z	.0228643	.0610888	0.37	0.708	-.0971255	.1428541
000070	-.0458165	.0577304	-0.79	0.428	-.1592098	.0675768
000071	.0606965	.0625507	0.97	0.332	-.0621648	.1835578
000072	-.75893	.3455586	-2.20	0.028	-1.437672	-.0801885
000073	-.2312197	.0704219	-3.28	0.001	-.3695415	-.0928979
000074	.5142819	.1200554	4.28	0.000	.2784708	.7500931
000075	.614946	.3530301	1.74	0.082	-.078471	1.308363
000076	.0099156	.0613463	0.16	0.872	-.11058	.1304112
000077	-.251764	.1321383	-1.91	0.057	-.5113082	.0077802
000078	.0239573	.0402109	0.60	0.552	-.0550244	.1029391
000079	-.025745	.047907	-0.54	0.591	-.1198433	.0683533
00007A	.0036702	.0304154	0.12	0.904	-.0560713	.0634117
00007B	.0012043	.029832	0.04	0.968	-.0573913	.0597999
00007C	-.004745	.03626	-0.13	0.896	-.0759664	.0664764
00007D	-.5740883	.3421303	-1.68	0.094	-1.246096	.0979195
00007E	.0381456	.0601783	0.63	0.526	-.0800559	.156347
00007F	.0640986	.1078561	0.59	0.553	-.147751	.2759482
00007G	.7332626	.3264335	2.25	0.025	.0920862	1.374439
00007H	.0102552	.0608562	0.17	0.866	-.1092777	.1297882
00007I	-.1232564	.1093953	-1.13	0.260	-.3381292	.0916165
00007J	-.0569274	.0364646	-1.56	0.119	-.1285507	.0146959
00007K	.0887648	.0489852	1.81	0.071	-.0074512	.1849809
00007L	-.0551862	.0249964	-2.21	0.028	-.1042838	-.0060886
00007M	-.024072	.0242557	-0.99	0.321	-.0717146	.0235706
00007N	.0277149	.0335968	0.82	0.410	-.0382755	.0937053
00007O	-1.13101	.4848866	-2.33	0.020	-2.083418	-.1786023
00007P	-.0435355	.1001369	-0.43	0.664	-.2402231	.1531521
00007Q	.3654978	.1891627	1.93	0.054	-.0060531	.7370488
00007R	1.16244	.507497	2.29	0.022	.1656215	2.159259
00007S	.0570413	.0881808	0.65	0.518	-.1161623	.2302449
00007T	-.2036173	.145231	-1.40	0.161	-.4888781	.0816435
00007U	-.1332294	.0603627	-2.21	0.028	-.2517929	-.0146658
00007V	-.041391	.0728866	-0.57	0.570	-.1845539	.101772
00007W	-.0012864	.0358602	-0.04	0.971	-.0717225	.0691498
00007X	.0434206	.0352712	1.23	0.219	-.0258585	.1126998
00007Y	.0325858	.0478631	0.68	0.496	-.0614263	.1265979
000080	12.67072	14.52484	0.87	0.383	-15.85878	41.20021
000082	2.990678	8.82883	0.34	0.735	-14.35079	20.33215
000085	-.4291792	.7178658	-0.60	0.550	-1.839202	.9808432
000087	.8248535	.3534159	2.33	0.020	.1306788	1.519028
00008A	-2.039627	3.150398	-0.65	0.518	-8.227597	4.148343
00008C	-1.513494	1.945439	-0.78	0.437	-5.334699	2.307712
00008F	-14.41287	16.71094	-0.86	0.389	-47.23628	18.41053
00008H	-4.208317	10.5505	-0.40	0.690	-24.93147	16.51484
00008K	.192414	.7421106	0.26	0.796	-1.26523	1.650058
00008M	.0945454	.6374058	0.15	0.882	-1.157439	1.346529
00008P	.8368594	1.206148	0.69	0.488	-1.53224	3.205958
00008R	1.572858	.9757009	1.61	0.108	-.3436008	3.489316
00008U	.4124171	.471575	0.87	0.382	-.5138441	1.338678
00008W	-.5900783	.2787474	-2.12	0.035	-1.13759	-.0425665
00008Z	1.191815	.5413488	2.20	0.028	.1285051	2.255125
000091	-.4896809	.4551836	-1.08	0.282	-1.383746	.4043846
000094	.2869013	.3593708	0.80	0.425	-.4189699	.9927726
000096	.063333	.2172741	0.29	0.771	-.363434	.4900999
000099	-.6446566	.2790827	-2.31	0.021	-1.192827	-.0964861

__00009B	.2737888	.2179809	1.26	0.210	-.1543664	.7019441
__00009E	.0381564	.3483025	0.11	0.913	-.6459747	.7222875
__00009G	.9038708	.2714258	3.33	0.001	.3707398	1.437002
xbarlag	-13.59045	18.65429	-0.73	0.467	-50.23096	23.05007
x2barlag	-.6327815	.6809827	-0.93	0.353	-1.970359	.7047956
xbar2lag	3.305845	4.117163	0.80	0.422	-4.781031	11.39272
rootxbarlag	16.56054	21.04297	0.79	0.432	-24.77179	57.89287
z1barlag	-.4572849	1.161856	-0.39	0.694	-2.739386	1.824817
z2barlag	-2.298737	1.661446	-1.38	0.167	-5.562127	.9646538
z3barlag	.2537767	.4907429	0.52	0.605	-.710134	1.217687
z4barlag	-.7035252	.6926887	-1.02	0.310	-2.064095	.6570446
z5barlag	-.3764805	.3795412	-0.99	0.322	-1.12197	.3690093
z6barlag	.4811469	.378402	1.27	0.204	-.2621052	1.224399
z7barlag	-1.016984	.4484877	-2.27	0.024	-1.897897	-.1360703
_cons	-4.554202	6.715524	-0.68	0.498	-17.74474	8.63634

(option **xb** assumed; fitted values)
(110 missing values generated)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	8,962	.3564358	.0642132	.1633886	.8302536

(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)

Linear regression

Number of obs	=	8,962
F(128, 563)	=	.
Prob > F	=	.
R-squared	=	0.2333
Root MSE	=	.14393

(Std. err. adjusted for **564** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	-1.083467	5.935255	-0.18	0.855	-12.74142	10.57448
__00009L	3.939113	3.266587	1.21	0.228	-2.477074	10.3553
__00009O	6.835108	3.956711	1.73	0.085	-.9366105	14.60683
xi	.1919222	.0244721	7.84	0.000	.1438545	.2399899
__000002	-.0590336	.0084322	-7.00	0.000	-.0755961	-.0424712
__00000E	.3337955	.2204996	1.51	0.131	-.0993068	.7668978
__00000P	.0084455	.1253886	0.07	0.946	-.2378412	.2547322
__000017	1.004492	.5171704	1.94	0.053	-.011327	2.020311
__00001I	-.1980039	.2632071	-0.75	0.452	-.7149918	.318984
__000020	.0137185	.1222695	0.11	0.911	-.2264417	.2538786
__00002B	.048998	.0739083	0.66	0.508	-.0961717	.1941677
__00002T	.3169153	.2421269	1.31	0.191	-.1586671	.7924976
__000034	.1277945	.1580742	0.81	0.419	-.1826928	.4382818
__00003M	.1112933	.1061821	1.05	0.295	-.0972682	.3198548
__00003X	-.0963248	.0733973	-1.31	0.190	-.2404908	.0478413
__00004F	-.0111251	.0952395	-0.12	0.907	-.1981932	.1759431
__00004Q	-.0801516	.0592017	-1.35	0.176	-.1964349	.0361316
__000058	-.0608227	.1664915	-0.37	0.715	-.3878432	.2661977
__00005J	-.0332709	.094966	-0.35	0.726	-.2198018	.15326
__00005U	1.58886	.8918486	1.78	0.075	-.1628968	3.340617
__00005V	.3497876	.2481566	1.41	0.159	-.1376382	.8372133
__00005W	-.8218921	.3760976	-2.19	0.029	-1.560618	-.0831662
__00005X	-1.464274	.8996074	-1.63	0.104	-3.231271	.3027223
__00005Y	.0933725	.1515756	0.62	0.538	-.2043502	.3910953
__00005Z	-.1818131	.2886418	-0.63	0.529	-.7487595	.3851334
__000060	-.0940603	.1113883	-0.84	0.399	-.3128476	.1247271
__000061	.2827886	.1406333	2.01	0.045	.0065585	.5590187
__000062	-.0317301	.0706307	-0.45	0.653	-.170462	.1070017
__000063	-.0281487	.0629602	-0.45	0.655	-.1518142	.0955168
__000064	.0368189	.073209	0.50	0.615	-.1069773	.180615
__000065	1.432175	1.720323	0.83	0.405	-1.946859	4.811209
__000066	.4926219	.3540946	1.39	0.165	-.202886	1.18813

__000067	-.6336892	.6169797	-1.03	0.305	-1.845552	.578174
__000068	-2.25491	1.750511	-1.29	0.198	-5.69324	1.183419
__000069	.0349802	.3132912	0.11	0.911	-.5803822	.6503426
__00006A	-.2249349	.5847102	-0.38	0.701	-1.373415	.923545
__00006B	.1509789	.2082507	0.72	0.469	-.2580643	.5600221
__00006C	-.1932417	.2473663	-0.78	0.435	-.6791152	.2926318
__00006D	-.085186	.146859	-0.58	0.562	-.3736444	.2032724
__00006E	.0501347	.1328009	0.38	0.706	-.2107111	.3109805
__00006F	.0472086	.1735639	0.27	0.786	-.2937032	.3881204
__00006G	-.154843	.4647919	-0.33	0.739	-1.067781	.7580949
__00006H	.1235027	.0816993	1.51	0.131	-.03697	.2839754
__00006I	-.0955291	.1814898	-0.53	0.599	-.4520088	.2609507
__00006J	-.029447	.4590007	-0.06	0.949	-.93101	.872116
__00006K	.0595036	.1020602	0.58	0.560	-.1409616	.2599688
__00006L	.1743663	.1681716	1.04	0.300	-.1559541	.5046867
__00006M	-.0651254	.0601698	-1.08	0.280	-.1833102	.0530594
__00006N	-.1046789	.1117462	-0.94	0.349	-.3241692	.1148115
__00006O	.01302	.0346964	0.38	0.708	-.0551303	.0811702
__00006P	-.0307579	.0357824	-0.86	0.390	-.1010413	.0395254
__00006Q	.0407712	.0443041	0.92	0.358	-.0462504	.1277928
__00006R	.6701058	.9299785	0.72	0.471	-1.156545	2.496757
__00006S	.0200121	.2394475	0.08	0.933	-.4503075	.4903317
__00006T	-.3229109	.3253731	-0.99	0.321	-.9620043	.3161825
__00006U	-.640895	.8256587	-0.78	0.438	-2.262643	.9808527
__00006V	.5651726	.1591771	3.55	0.000	.2525191	.877826
__00006W	-.9373229	.2987795	-3.14	0.002	-1.524181	-.3504643
__00006X	-.1239941	.0913638	-1.36	0.175	-.3034497	.0554616
__00006Y	.2568817	.1278759	2.01	0.045	.0057096	.5080538
__00006Z	.0580578	.0850118	0.68	0.495	-.1089213	.2250369
__000070	-.1809693	.0641577	-2.82	0.005	-.3069871	-.0549516
__000071	.0346175	.0813357	0.43	0.671	-.125141	.194376
__000072	-.4225712	.4346616	-0.97	0.331	-1.276328	.4311853
__000073	-.1619888	.1046419	-1.55	0.122	-.3675251	.0435474
__000074	.3133865	.1773932	1.77	0.078	-.0350468	.6618198
__000075	.2694395	.4621215	0.58	0.560	-.6382533	1.177132
__000076	.0624534	.0753451	0.83	0.408	-.0855383	.2104452
__000077	-.3834297	.1482989	-2.59	0.010	-.6747163	-.092143
__000078	.1102028	.0469253	2.35	0.019	.0180327	.2023729
__000079	-.0638558	.0615924	-1.04	0.300	-.1848348	.0571231
__00007A	.0727061	.0345055	2.11	0.036	.0049308	.1404814
__00007B	.0349186	.0337849	1.03	0.302	-.0314413	.1012785
__00007C	.0102543	.0407627	0.25	0.801	-.0698113	.0903199
__00007D	-.5775402	.4053759	-1.42	0.155	-1.373774	.2186937
__00007E	.0385396	.0762789	0.51	0.614	-.1112864	.1883656
__00007F	.0837639	.1290169	0.65	0.516	-.1696493	.3371772
__00007G	.6578898	.3997218	1.65	0.100	-.1272385	1.443018
__00007H	.0509667	.0759273	0.67	0.502	-.0981688	.2001021
__00007I	-.2524079	.143923	-1.75	0.080	-.5350995	.0302836
__00007J	-.0219281	.0415383	-0.53	0.598	-.103517	.0596608
__00007K	.0467212	.0636006	0.73	0.463	-.0782023	.1716447
__00007L	-.0389733	.0326119	-1.20	0.233	-.1030291	.0250825
__00007M	-.0140627	.0300025	-0.47	0.639	-.0729933	.0448679
__00007N	-.0092208	.0381248	-0.24	0.809	-.0841051	.0656635
__00007O	-1.520885	.6224722	-2.44	0.015	-2.743537	-.2982337
__00007P	.0242651	.131144	0.19	0.853	-.2333261	.2818563
__00007Q	.4726095	.2650051	1.78	0.075	-.0479099	.993129
__00007R	1.328301	.6530431	2.03	0.042	.045603	2.611
__00007S	-.0138804	.1237031	-0.11	0.911	-.2568564	.2290956
__00007T	-.2876657	.212542	-1.35	0.176	-.7051377	.1298064
__00007U	-.1499288	.0821426	-1.83	0.068	-.3112722	.0114146
__00007V	-.0483759	.1086316	-0.45	0.656	-.2617486	.1649969
__00007W	-.0528355	.0543656	-0.97	0.332	-.1596196	.0539485
__00007X	.0854972	.0475467	1.80	0.073	-.0078934	.1788877
__00007Y	.0396064	.054197	0.73	0.465	-.0668467	.1460595
__000080	-1.853882	15.06591	-0.12	0.902	-31.44615	27.73838
__000082	19.27764	10.68323	1.80	0.072	-1.706212	40.2615
__000085	.7664116	.9649635	0.79	0.427	-1.128957	2.66178
__000087	1.251081	.5225581	2.39	0.017	.2246789	2.277482
__00008A	1.051261	3.335125	0.32	0.753	-5.499547	7.602068
__00008C	-4.632426	2.355611	-1.97	0.050	-9.259285	-.0055662
__00008F	-.6572483	17.02602	-0.04	0.969	-34.09954	32.78504
__00008H	-25.51652	12.60042	-2.03	0.043	-50.2661	-.7669364

```
      __00008K |   .1720968   .7723585     0.22   0.824    -1.344959    1.689153
      __00008M |   1.118669   .6598946     1.70   0.091    -.1774876    2.414825
      __00008P |    -1.1014   1.632029    -0.67   0.500    -4.307008    2.104209
      __00008R |   1.414843   1.050108     1.35   0.178    -.6477661    3.477452
      __00008U |  -.0631202   .5753208    -0.11   0.913    -1.193158    1.066917
      __00008W |  -.2608257   .3196552    -0.82   0.415    -.8886882    .3670367
      __00008Z |   .3385591   .6867901     0.49   0.622    -1.010425    1.687543
      __000091 |  -1.053177    .506961    -2.08   0.038    -2.048943   -.0574109
      __000094 |   .2629187   .4019763     0.65   0.513    -.5266377    1.052475
      __000096 |  -.1347651   .2203119    -0.61   0.541    -.5674988    .2979685
      __000099 |  -.3373559   .3569749    -0.95   0.345    -1.038521    .3638094
      __00009B |   .0655094   .2281191     0.29   0.774    -.3825591     .513578
      __00009E |  -.1018544   .4462692    -0.23   0.820    -.9784102    .7747015
      __00009G |   .3262177   .3251723     1.00   0.316    -.3124813    .9649168
       xbarlag |  -16.33344   20.60522    -0.79   0.428    -56.80594    24.13906
      x2barlag |  -2.554107   1.105905    -2.31   0.021    -4.726312   -.3819022
      xbar2lag |   3.672664    4.75535     0.77   0.440     -5.66773    13.01306
   rootxbarlag |   26.03771   23.10197     1.13   0.260    -19.33887    71.41428
      z1barlag |  -1.667624    1.22026    -1.37   0.172    -4.064441    .7291937
      z2barlag |  -.2560618   2.150697    -0.12   0.905    -4.480432    3.968308
      z3barlag |    .381395   .6300566     0.61   0.545    -.8561538    1.618944
      z4barlag |   1.011862   .9107246     1.11   0.267    -.7769712    2.800695
      z5barlag |    -.13724   .4524444    -0.30   0.762    -1.025925    .7514452
      z6barlag |   .3068243   .3873211     0.79   0.429    -.4539466    1.067595
      z7barlag |  -.3247348   .5432947    -0.60   0.550    -1.391867    .7423973
         _cons |  -9.775061   7.348165    -1.33   0.184    -24.20823    4.658107
------------------------------------------------------------------------------
(option xb assumed; fitted values)
(110 missing values generated)

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
     __0000A2 |      8,962    .2964597    .0788125    .0043309    .9595904
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)

955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"

957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
   __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9

960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                 foreach var2 of global Alist {
    3.                         g `var1'_`var2'=`var1'*`var2'
    4.                         sum `var1' `var2' `var1'_`var2'
    5.                         local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                 }
    7.         }
964 }
```

```
965
966 local r_RE      "`r_RE' `qhat_hats'"
967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }
978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS	Number of obs	=	9,072
				F(13, 9058)	=	4469.12
Model	270.802088	13	20.8309298	Prob > F	=	0.0000
Residual	42.2200752	9,058	.004661081	R-squared	=	0.8651
				Adj R-squared	=	0.8649
Total	313.022163	9,071	.03450801	Root MSE	=	.06827

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	.0226955	.004389	5.17	0.000	.014092	.0312989
ybarx	.0211619	.0051316	4.12	0.000	.0111028	.031221
xi2	-.0474636	.0023906	-19.85	0.000	-.0521497	-.0427775
ybar	-.2098967	.010162	-20.66	0.000	-.2298166	-.1899768
xi	.4422196	.0065099	67.93	0.000	.4294586	.4549805
ybar_q1	.3978245	.0116654	34.10	0.000	.3749577	.4206913
pzi1	.1588043	.0040866	38.86	0.000	.1507936	.166815
pzi2	.0112992	.0065624	1.72	0.085	-.0015647	.0241631
pzi3	.0037226	.0019815	1.88	0.060	-.0001617	.0076068
pzi4	.0144642	.0030027	4.82	0.000	.0085782	.0203502
pzi5	-.0073398	.0016042	-4.58	0.000	-.0104843	-.0041953
pzi6	.0014927	.0015404	0.97	0.333	-.0015268	.0045123
pzi7	-.0045483	.0025593	-1.78	0.076	-.009565	.0004685
_cons	-.0075227	.004926	-1.53	0.127	-.0171787	.0021334

```
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = _b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                local coef = _b[pzi`j']
    3.                local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1025}

1026
1027if `J'==3 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00242521
  Iteration 1:  GMM criterion Q(b) =  .00224482
  Iteration 2:  GMM criterion Q(b) =  .00017258
  Iteration 3:  GMM criterion Q(b) =  .00009564
```

```
Step 2
Iteration 0:  GMM criterion Q(b) =  .00623091
Iteration 1:  GMM criterion Q(b) =  .00542216
Iteration 2:  GMM criterion Q(b) =  .00494292
Iteration 3:  GMM criterion Q(b) =   .0048292
Iteration 4:  GMM criterion Q(b) =  .00429571
Iteration 5:  GMM criterion Q(b) =  .00379339

GMM estimation

Number of parameters =  20
Number of moments    =  28
Initial weight matrix: Unadjusted                 Number of obs   =      8,962
GMM weight matrix:     Robust

                (Std. err. adjusted for 296 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
A11          |
         one |   .8021776   .0701823    11.43   0.000     .6646229    .9397323
-------------+----------------------------------------------------------------
C1           |
         zi1 |   .0467984   .1679322     0.28   0.780    -.2823426    .3759394
         zi2 |   .1255687   .2164547     0.58   0.562    -.2986746     .549812
         zi3 |  -.1068863   .0720994    -1.48   0.138    -.2481986     .034426
         zi4 |   .0707929   .1299236     0.54   0.586    -.1838527    .3254385
         zi5 |   .0194576   .0482024     0.40   0.686    -.0750173    .1139326
         zi6 |   .0679434   .0488244     1.39   0.164    -.0277507    .1636375
         zi7 |   .0485509   .1166637     0.42   0.677    -.1801057    .2772075
-------------+----------------------------------------------------------------
C2           |
         zi1 |  -.3228909   .2570678    -1.26   0.209    -.8267346    .1809527
         zi2 |   .1519223   .2994175     0.51   0.612    -.4349252    .7387698
         zi3 |  -.1378033   .0958623    -1.44   0.151    -.3256899    .0500833
         zi4 |   .1159333   .1799759     0.64   0.519    -.2368129    .4686796
         zi5 |   .0459748   .0673664     0.68   0.495    -.0860609    .1780105
         zi6 |   .0921307   .0710008     1.30   0.194    -.0470283    .2312898
         zi7 |   .0809897   .1789914     0.45   0.651     -.269827    .4318064
-------------+----------------------------------------------------------------
         /d1 |  -.0496782   .0169471    -2.93   0.003    -.0828938   -.0164625
         /b1 |   .4076038    .028808    14.15   0.000      .351141    .4640665
      /AVA11 |  -3.719783   2.154605    -1.73   0.084    -7.942731     .503165
      /AVA12 |   3.866866   2.154105     1.80   0.073    -.3551017    8.088833
      /AVA22 |  -4.611361   2.582003    -1.79   0.074    -9.671993    .4492711
------------------------------------------------------------------------------
Instruments for equation eq1_RE: __00009J __00009L __00009O xi __000002 __00000E
    __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M
    __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5
    __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
```

```
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactIns
      > t_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_onlySched_RE.tex
      not found)
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_onlySched_RE
  > .tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  2430
  Number of records is  9072
```

```
1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  2430

1077unique group_round
  Number of unique values of group_round is  572
  Number of records is  9072

1078estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  572

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  4.2482517

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_onlySched_RE.ster n
  > ot found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_onlySched_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
```

```
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*`p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
1119                                                gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                             local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                          }
   30.                                  }
   31.                          }
   32.                  }
   33.          }
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
```

```
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
1120        if `Tindivp1'<=`T' {
1121                forval t=`Tindivp1'/`T' {
    2.                          tempvar zi`t'xdiff
    3.                          g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                          local zg "`zg' zi`t'"
    5.                          forval j=1/`J' {
    6.                                  tempvar zi`t'p`j'xdiff
    7.                                  g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                  local pzg "`pzg' `zi`t'p`j''"
    9.                                  local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                          }
   11.                 }
1122        }
1123
1124        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1125        // denoting zg as group-zs
1126        //  zg,pzgx,rprp,rprpx
1127        local rootprootp ""
1128        local rootprootpxdiff ""
1129
1130        forval j=1/`J' {
    2.                  forval s=`j'/`J' {
    3.                          tempvar xdiffrp`j'rp`s'
    4.                          g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1131                        if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1132                        local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                  }
    8.          }
1133
1134        local pqhatxdiff ""
1135        forval j=1/`J' {
    2.                  forval s=1/`J' {
    3.                          foreach name of global Alist {
    4.                                  tempvar   p`s'qikhat`j'xdiff`name'
    5.                                  g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
1136                                    local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
```

```
1137
1138        // set instruments
1139        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1140}

1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                tempvar ximxk`var'
    3.                g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                        local `++t'
    3.                        local p_nice = subinstr("`p'",":_cons","",.)
    4.                        local est = coefs[1,`t']
    5.                        if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                        if `j'>`i' {
   11.                                tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                        }
   15.                }
   16.         }
```

```
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >     `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1     `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
```

```
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00124886
  Iteration 1:  GMM criterion Q(b) =  .00102958
  Iteration 2:  GMM criterion Q(b) =  .00082021
  Iteration 3:  GMM criterion Q(b) =  .00075637

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .03281875
  Iteration 1:  GMM criterion Q(b) =  .03257365
  Iteration 2:  GMM criterion Q(b) =  .02800218
  Iteration 3:  GMM criterion Q(b) =  .02792573

  GMM estimation

  Number of parameters =  17
  Number of moments    =  63
  Initial weight matrix: Unadjusted                 Number of obs   =      8,962
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 296 clusters in state_district_round)
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11						
_cons	.4574244	.067084	6.82	0.000	.3259422	.5889065
C12						
_cons	.0721966	.11839	0.61	0.542	-.1598435	.3042367
C13						
_cons	.0088391	.0344689	0.26	0.798	-.0587186	.0763969
C14						
_cons	.0891957	.0403647	2.21	0.027	.0100823	.1683092
C15						
_cons	.0244887	.0402265	0.61	0.543	-.0543538	.1033312

C16 _cons	**-.0612593**	**.0322849**	**-1.90**	**0.058**	**-.1245366**	**.0020179**
C17 _cons	**.0018363**	**.0429446**	**0.04**	**0.966**	**-.0823336**	**.0860061**
C21 _cons	**.2561478**	**.0790532**	**3.24**	**0.001**	**.1012065**	**.4110892**
C22 _cons	**.0771325**	**.143993**	**0.54**	**0.592**	**-.2050886**	**.3593537**
C23 _cons	**.0206618**	**.0415237**	**0.50**	**0.619**	**-.0607232**	**.1020468**
C24 _cons	**.1072742**	**.0476154**	**2.25**	**0.024**	**.0139497**	**.2005987**
C25 _cons	**.0450682**	**.0457567**	**0.98**	**0.325**	**-.0446134**	**.1347498**
C26 _cons	**-.0784689**	**.0381594**	**-2.06**	**0.040**	**-.1532601**	**-.0036778**
C27 _cons	**.0218685**	**.0511284**	**0.43**	**0.669**	**-.0783414**	**.1220784**
d1 _cons	**-.1025596**	**.0136392**	**-7.52**	**0.000**	**-.129292**	**-.0758272**
b1 _cons	**.4162985**	**.0375007**	**11.10**	**0.000**	**.3427984**	**.4897985**
A11 one	**.4447294**	**.1382796**	**3.22**	**0.001**	**.1737064**	**.7157523**

Instruments for equation **eq1_FE**: **__000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7 __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1 __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY __0000D5 _cons**

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
```

(file **diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactIns > t_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_onlySched_FE.tex** not found)
(output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3 > _all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_onlySched_FE > .tex)

```
1242
1243        estat overid
```

Test of overidentifying restriction:

Hansen's J chi2(**46**) = **250.27** (p = **0.0000**)

```
1244        di "$S_DATE $S_TIME"
   7 Dec 2023 21:19:51
1245        unique hhidi
  Number of unique values of hhidi is  2430
  Number of records is  9072
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  2430
1248        unique group_round
  Number of unique values of group_round is  572
  Number of records is  9072
1249        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  572
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  4.2482517
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_onlySched_FE.ster n
  > ot found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_onlySched_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}
```

```
1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table5_13.smcl
   log type:  smcl
  closed on:   7 Dec 2023, 21:19:51
```

A.5.2 Columns 2 and 4

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table5_24.smcl
  log type:  smcl
 opened on:   7 Dec 2023, 21:14:30

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
  C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=2

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
50. //local group_def "dist"
51. //local group_def "fsuXseg"
52. local group_def "fsuXsegXreligXsched"

53.
54. // comparision group
55. local groupComp_def ""

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.          syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
70.          if "`rename'" == "" local rename = "`varlist'bar"
    3.
```

```
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations            1,118    ->   280
  Number of variables                   5    ->   7
  j variable (4 values)       expend_type    ->   (dropped)
  xij variables:
                         laspeyres_state    ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560    ->   280
  Number of variables                   5    ->   5
  j variable (2 values)       expend_type    ->   (dropped)
  xij variables:
                         laspeyres_state    ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                  Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations                 840   ->   280
  Number of variables                      5   ->   6
  j variable (3 values)          expend_type   ->   (dropped)
  xij variables:
                                laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,637 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
```

```
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }
297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }
316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }
333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
```

```
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
  (14,430 observations deleted)
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop

  Duplicates in terms of all variables

  (53,013 observations deleted)
439                 tempfile update_exp
440                 save `update_exp'
  file C:\Users\pousim2\AppData\Local\Temp\ST_41e4_000002.tmp saved as .dta format
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
  (61,772 observations deleted)
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
  (label religion already defined)

    Result                      Number of obs
    -----------------------------------------
    Not matched                             0

    Matched                            11,804
        not updated                     7,790
        missing updated                     0
        nonmissing conflict             4,014
    -----------------------------------------
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (22 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight
```

```
. global instlist ""

. g xreal=x/local_cpi
  (22 missing values generated)

. g x_temp=x

. g x2_temp=x2

. replace x=xreal
  (11,804 real changes made, 22 to missing)

. replace x2=x*x
  (11,804 real changes made, 22 to missing)

.
. // make z x interactions
. foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (22 missing values generated)
  (22 missing values generated)
  (22 missing values generated)
  (22 missing values generated)
  (22 missing values generated)
  (22 missing values generated)
  (22 missing values generated)

.
. levelsof round, local(roundlist)
  59 60 61 62

. foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

.
. g xbar2lag=xbarlag^2
  (149 missing values generated)

. g rootxbarlag=sqrt(xbarlag)
  (149 missing values generated)

.
. bysmeanw x, weight(weight) by(group) rename(xbar)

. bysmeanw x2, weight(weight) by(group) rename(x2bar)

. g xbar2=xbar^2

. g rootxbar = sqrt(xbar)
```

```
475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.          bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.          global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (11,804 real changes made)

478 replace x2=x2_temp
  (11,804 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (9,374 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)

488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi
```

```
503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_41e4_000003.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk
```

```
535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (2,430 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
549 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
```

```
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                          file write sumstat "`sumstats_row_`var'"" _n
    5.                 }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0
```

```
609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=1

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1

625 scalar estimate_FE=1

626
627 // estimation options
628 global w_initial "winit(unadjusted)"

629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

630 global wmatrix "wmatrix(robust)"

631 global trace_level ""

632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

633 global technique ""

634 global maxiter ""

635 global technique_RE ""

636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

637 global derivatives "quickderivatives"

638 scalar init_FE_from_RE=1

639 global clust_var "state_district_round"

640 local clus_name "_clus_Dist"

641
642
643
644
645
646
647
```

```
648
649 // xxzx to keep names same
650 local wmatrix_desc ""
651 local sectorName "_urbOnly"
652 local dropZdesc "_dropPre"
653 local desc_struc_v0 ""
654 local inst_desc "_exactInst"
655 local inst_rootp "_noPPXdInst"
656 local weight_desc ""
657 local RE_sq_inst_desc ""
658 local cnstname ""
659 local actual_pqhat_desc ""
660 local expUpname "_updateExp"
661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //              ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
```

```
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T

694 local Tm1=`T'-1

695 local Tindiv : word count $zlist_indiv

696
697 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
706
707 if `J'==2 {
708          replace p1=P1
  (9,072 real changes made)
709          replace p2=P2
  (9,072 real changes made)
710          replace qi1=Qi1
  (9,072 real changes made)
711          replace qk1=Qk1
  (9,072 real changes made)
712          replace qikbar1=Qikbar1
  (9,072 real changes made)
713          replace qi2=Qi2
  (9,072 real changes made)
714          replace qk2=Qk2
  (9,072 real changes made)
```

```
715         replace qikbar2=Qikbar2
  (9,072 real changes made)
716
717         foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                  qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721         forv j = 1(1)3 {
    2.                  replace p`j'=P`j'
    3.                  replace qi`j'=Qi`j'
    4.                  replace qk`j'=Qk`j'
    5.                  replace qikbar`j'=Qikbar`j'
    6.          }
722
723         foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                  qui replace `var'=.
    3.          }
724 }

725
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )
```

```
744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }

751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }
```

```
766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
773         local Drp`j' "`Drp`j'' )"
   15.          di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.          if Afull==1 {
    3.                  forval j=1/`J' {
    4.                          local A`qtype'`j' "( 0"
    5.
```

```
793                         forval l=1/`J' {
    6.                                  if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                  if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                  if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                          }
   11.                          local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.            forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }

798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.         local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.            if `jplus1'<=`J' {
    5.                 forval k=`jplus1'/`J' {
    6.                         local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                 }
    8.         }
    9. }
```

```
812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >            local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                               local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.              if noMeasError==1                       local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'              - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                              l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                ))"
    3.          if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                       )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.          local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.          local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.          local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.          local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.          local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.          local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.          local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.          local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }
```

```
833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2) )*( ({A11:}*p1*qbar1) +
  > ({A11:}*p2*qbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2) )
  >    + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:}) ) -
  >  ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}) )
  >  - ( 0 ) )-( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2) ) )*{b1} + ({A11:}*p1*qbar1)  + (
  > p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d1}/exp(
  >  {b1}*ln(p1/p2) + ln(p2) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))*({b1} - 2*(( ({A11:
  > }*p1*qbar1) + ({A11:}*p2*qbar2) )  + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1
  > /p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1
  > *{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13
  > }*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))            ))

836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2             "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
```

```
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                              "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}     `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                          "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qbar1) + ({A11:}*p2*qbar2) )*( ({A11:}*p1*qbar1)
  >  + ({A11:}*p2*qbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C
  > 2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2)
  >  )    + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qbar1) + ({A11:}*p2*qbar2) ) )*{b1} + ({A11:}*p1*qbar1)
  > + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d1}/e
  > xp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))*({b1} - 2*(( ({A11:
  > one}*p1*qbar1) + ({A11:}*p2*qbar2) )  + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
```

```
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))            )
  > )

874
875 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi

882 g double `ximxk'=xi-xk

883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag
  (110 missing values generated)

886 g xbarlag2  = xbarlag^2
  (110 missing values generated)

887 g x2barlag2 = x2barlag^2
  (110 missing values generated)

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.          tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.          g double `xizi`t''=xi*zi`t'
    4.          g double `xkzk`t''=xk*zk`t'
    5.          g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.          g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.          g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.          g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.          g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.          forval j=1/`J' {
   11.                  tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                  g double `zi`t'p`j''= zi`t'*p`j'
   13.                  g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                  g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                  g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                  forv s=1/`T' {
   17.                          tempvar zi`t'zi`s'p`j'
   18.                          g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                  }
   20.          }
   21. }
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
```

```
// make group-level instruments, and the instrument lists rg, xrg
if noMeasError==0       local rg "xbarlag"

if noMeasError==1       local rg "xbar"


local rg "`rg' x2barlag"

local rg "`rg' xbar2lag"

local rg "`rg' rootxbarlag"


local Tindivm1=`Tindiv'-1

// lag instruments
forval t=1/`Tindiv' {
  2.                  local rg "`rg' z`t'barlag"
  3. }


local zrg ""

forval t=1/`Tindiv' {
  2.          foreach var of varlist `rg' {
  3.                  tempvar zi`t'`var'
  4.                  g double `zi`t'`var''=zi`t'*`var'
  5.                  local zrg "`zrg' `zi`t'`var''"
  6.          }
  7. }
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
```

```
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)

906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
  2.         tempvar xi`var'
  3.         g double `xi`var''=xi*`var'
  4.         local xrg "`xrg' `xi`var''"
  5.
911         forval j=1/`J' {
  6.                 tempvar p`j'`var' xp`j'`var'
  7.                 g double `p`j'`var''=p`j'*`var'
  8.                 g double `xp`j'`var''=xi*p`j'*`var'
  9.                 local prg "`prg' `p`j'`var''"
 10.                 local xprg "`xprg' `xp`j'`var''"
 11.         }
 12. }
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
```

```
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.         tempvar xip`j'
    3.         g double `xip`j''=xi*p`j'
    4.         local pinsts "`pinsts' p`j'"
    5.         local xpinsts "`xpinsts' `xip`j''"
    6.         forval s=`j'/`J' {
    7.                 tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                 g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                 g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                 local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                 local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.         }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""
```

```
925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }

929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.          forval k=`j'/`J' {
    3.                  tempvar p`j'p`k'
    4.                  g `p`j'p`k''=p`j'*p`k'
    5.                  local roundinsts "`roundinsts' `p`j'p`k''"
    6.          }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.          if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.          if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
```

```
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }
```

Linear regression

Number of obs	**8,962**
F(129, 563)	**3.27**
Prob > F	**0.0000**
R-squared	**0.2990**
Root MSE	**.1553**

(Std. err. adjusted for **564** clusters in **group_round**)

qbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**2.542667**	**3.273211**	**0.78**	**0.438**	**-3.886529**	**8.971864**
__00009L	**-4.387037**	**4.670963**	**-0.94**	**0.348**	**-13.56168**	**4.787605**
__00009O	**3.480729**	**2.483559**	**1.40**	**0.162**	**-1.397444**	**8.358901**
__xi	**-.0572937**	**.0299694**	**-1.91**	**0.056**	**-.1161591**	**.0015718**
__000002	**.0065492**	**.0099549**	**0.66**	**0.511**	**-.013004**	**.0261025**
__00000E	**-.001793**	**.2723988**	**-0.01**	**0.995**	**-.5368351**	**.5332491**
__00000P	**-.1502676**	**.1609963**	**-0.93**	**0.351**	**-.4664944**	**.1659593**
__000017	**-.1732237**	**.5636693**	**-0.31**	**0.759**	**-1.280375**	**.9339279**
__00001I	**.4030505**	**.3922173**	**1.03**	**0.305**	**-.3673375**	**1.173439**
__000020	**.1590822**	**.1461047**	**1.09**	**0.277**	**-.1278947**	**.4460592**
__00002B	**.0578698**	**.0831987**	**0.70**	**0.487**	**-.1055479**	**.2212876**
__00002T	**.5229233**	**.3487644**	**1.50**	**0.134**	**-.162115**	**1.207962**
__000034	**-.1988477**	**.2747368**	**-0.72**	**0.470**	**-.7384819**	**.3407866**
__00003M	**.1525375**	**.1429227**	**1.07**	**0.286**	**-.1281893**	**.4332643**
__00003X	**.0929891**	**.0955825**	**0.97**	**0.331**	**-.0947528**	**.280731**
__00004F	**.1134823**	**.1113323**	**1.02**	**0.308**	**-.1051951**	**.3321598**
__00004Q	**-.014257**	**.0762068**	**-0.19**	**0.852**	**-.1639413**	**.1354274**
__000058	**.0325621**	**.1809896**	**0.18**	**0.857**	**-.3229352**	**.3880594**
__00005J	**.1049484**	**.101456**	**1.03**	**0.301**	**-.0943302**	**.3042269**
__00005U	**.0754329**	**.0981391**	**0.77**	**0.442**	**-.1173305**	**.2681964**
__00005V	**.4356615**	**.2404419**	**1.81**	**0.071**	**-.0366112**	**.9079343**
__00005W	**-.5705788**	**.2512206**	**-2.27**	**0.024**	**-1.064023**	**-.0771347**
__00005X	**.3471922**	**.4378966**	**0.79**	**0.428**	**-.5129184**	**1.207303**
__00005Y	**-.1595458**	**.1926146**	**-0.83**	**0.408**	**-.5378769**	**.2187853**
__00005Z	**.30586**	**.3160052**	**0.97**	**0.334**	**-.3148332**	**.9265531**
__000060	**-.0136905**	**.1386722**	**-0.10**	**0.921**	**-.2860686**	**.2586876**
__000061	**-.0797003**	**.1951269**	**-0.41**	**0.683**	**-.4629658**	**.3035653**
__000062	**.0075445**	**.1003051**	**0.08**	**0.940**	**-.1894733**	**.2045623**
__000063	**-.172392**	**.0747857**	**-2.31**	**0.022**	**-.319285**	**-.0254991**
__000064	**-.1394473**	**.0878507**	**-1.59**	**0.113**	**-.3120025**	**.033108**
__000065	**-.007391**	**.164069**	**-0.05**	**0.964**	**-.3296531**	**.3148712**
__000066	**.0827947**	**.4132365**	**0.20**	**0.841**	**-.7288788**	**.8944682**
__000067	**.0325851**	**.517925**	**0.06**	**0.950**	**-.9847161**	**1.049886**
__000068	**-.1718651**	**.7039322**	**-0.24**	**0.807**	**-1.554519**	**1.210789**
__000069	**-.0563636**	**.3840236**	**-0.15**	**0.883**	**-.8106577**	**.6979304**
__00006A	**.133462**	**.6364745**	**0.21**	**0.834**	**-1.116693**	**1.383617**
__00006B	**-.1866922**	**.2428413**	**-0.77**	**0.442**	**-.6636778**	**.2902934**
__00006C	**.1315534**	**.2664752**	**0.49**	**0.622**	**-.3918535**	**.6549603**
__00006D	**-.2509659**	**.1672029**	**-1.50**	**0.134**	**-.5793836**	**.0774518**
__00006E	**-.1659274**	**.17909**	**-0.93**	**0.355**	**-.5176937**	**.1858389**
__00006F	**.0208256**	**.2012652**	**0.10**	**0.918**	**-.3744968**	**.4161481**
__00006G	**-.0496178**	**.0409856**	**-1.21**	**0.227**	**-.130121**	**.0308855**
__00006H	**.0477413**	**.1035703**	**0.46**	**0.645**	**-.1556901**	**.2511727**

00006I	.0041783	.1276058	0.03	0.974	-.2464632	.2548198
00006J	-.3084528	.2055527	-1.50	0.134	-.7121967	.0952911
00006K	.2650946	.1255137	2.11	0.035	.0185623	.5116268
00006L	-.2079665	.2119877	-0.98	0.327	-.6243498	.2084169
00006M	-.0210831	.0671455	-0.31	0.754	-.1529695	.1108032
00006N	.0570596	.120642	0.47	0.636	-.1799039	.2940231
00006O	-.0494915	.0424161	-1.17	0.244	-.1328047	.0338217
00006P	.0645058	.0462704	1.39	0.164	-.0263778	.1553895
00006Q	.0388598	.0497575	0.78	0.435	-.0588733	.1365929
00006R	-.1796421	.0949493	-1.89	0.059	-.3661403	.0068561
00006S	1.430955	.3815698	3.75	0.000	.6814806	2.180429
00006T	-1.221454	.3411657	-3.58	0.000	-1.891567	-.5513409
00006U	-.88928	.4958602	-1.79	0.073	-1.863242	.0846819
00006V	.0404546	.2156393	0.19	0.851	-.3831013	.4640105
00006W	.1927992	.3427973	0.56	0.574	-.4805186	.866117
00006X	.1964108	.1153	1.70	0.089	-.0300599	.4228815
00006Y	.0361738	.153	0.24	0.813	-.2643467	.3366944
00006Z	-.1847018	.1113522	-1.66	0.098	-.4034183	.0340146
000070	.0642566	.079212	0.81	0.418	-.0913306	.2198438
000071	.2512844	.1373218	1.83	0.068	-.0184412	.52101
000072	-.156422	.0502475	-3.11	0.002	-.2551174	-.0577266
000073	-.2836934	.1567306	-1.81	0.071	-.5915415	.0241548
000074	.3958412	.182161	2.17	0.030	.0380431	.7536393
000075	-.3187829	.2242648	-1.42	0.156	-.7592808	.121715
000076	-.0954816	.1053738	-0.91	0.365	-.3024555	.1114922
000077	.1675074	.200161	0.84	0.403	-.225646	.5606609
000078	.0210659	.0654517	0.32	0.748	-.1074934	.1496253
000079	.1359131	.0888148	1.53	0.127	-.0385357	.3103619
00007A	-.1285628	.0479081	-2.68	0.007	-.2226632	-.0344623
00007B	.0078033	.0428533	0.18	0.856	-.0763687	.0919752
00007C	.0580967	.0545025	1.07	0.287	-.0489564	.1651498
00007D	-.0284851	.0329662	-0.86	0.388	-.0932368	.0362666
00007E	-.0203925	.0976533	-0.21	0.835	-.2122018	.1714168
00007F	.0526313	.0947668	0.56	0.579	-.1335083	.238771
00007G	-.0778159	.2180834	-0.36	0.721	-.5061724	.3505406
00007H	.1154023	.1043068	1.11	0.269	-.0894756	.3202802
00007I	-.1038892	.1748183	-0.59	0.553	-.4472651	.2394866
00007J	-.0430768	.0602943	-0.71	0.475	-.161506	.0753525
00007K	.1779178	.1058066	1.68	0.093	-.0299062	.3857418
00007L	-.0752743	.0411253	-1.83	0.068	-.1560521	.0055035
00007M	-.0149201	.0389504	-0.38	0.702	-.0914261	.0615858
00007N	-.0254606	.0453205	-0.56	0.574	-.1144786	.0635573
00007O	-.0552127	.0432114	-1.28	0.202	-.1400879	.0296626
00007P	-.2570085	.1416023	-1.82	0.070	-.5351418	.0211248
00007Q	.3314477	.1645213	2.01	0.044	.008297	.6545983
00007R	-.1297698	.2788797	-0.47	0.642	-.6775415	.4180019
00007S	-.1035108	.1203956	-0.86	0.390	-.3399901	.1329685
00007T	.0498916	.2027806	0.25	0.806	-.3484073	.4481906
00007U	.001642	.0798602	0.02	0.984	-.1552184	.1585024
00007V	-.0414081	.0787144	-0.53	0.599	-.1960178	.1132016
00007W	-.1003045	.0552144	-1.82	0.070	-.2087559	.0081469
00007X	.0646334	.0503714	1.28	0.200	-.0343053	.1635722
00007Y	.0019131	.0592353	0.03	0.974	-.1144361	.1182624
000080	-.7505589	.4463104	-1.68	0.093	-1.627196	.126078
000082	.3575823	.3228215	1.11	0.268	-.2764993	.9916638
000085	-1.641933	1.231656	-1.33	0.183	-4.061135	.7772678
000087	.697406	.978383	0.71	0.476	-1.224321	2.619133
00008A	1.261184	1.509821	0.84	0.404	-1.704386	4.226754
00008C	-.3585904	1.204069	-0.30	0.766	-2.723607	2.006426
00008F	1.688796	2.803338	0.60	0.547	-3.817482	7.195075
00008H	-1.701331	2.019259	-0.84	0.400	-5.667532	2.264871
00008K	-.4255853	1.241055	-0.34	0.732	-2.863249	2.012078
00008M	.0527258	1.026465	0.05	0.959	-1.963443	2.068895
00008P	-3.394917	2.202164	-1.54	0.124	-7.720378	.9305434
00008R	.6908691	1.529042	0.45	0.652	-2.312454	3.694192
00008U	.4975809	.6500815	0.77	0.444	-.7793003	1.774462
00008W	-.6856832	.4966749	-1.38	0.168	-1.661245	.2898789
00008Z	1.229616	.7547933	1.63	0.104	-.2529391	2.712171
000091	-.6123937	.7107441	-0.86	0.389	-2.008428	.7836404
000094	.3089398	.5193699	0.59	0.552	-.7111995	1.329079
000096	.7481854	.3423434	2.19	0.029	.0757591	1.420612
000099	-1.085889	.4069645	-2.67	0.008	-1.885243	-.2865346

__00009B	-.6268771	.2902439	-2.16	0.031	-1.19697	-.0567839
__00009E	-.0502447	.4990568	-0.10	0.920	-1.030485	.929996
__00009G	-.1979399	.3710783	-0.53	0.594	-.9268069	.5309271
xbar	.6489024	.5326783	1.22	0.224	-.3973772	1.695182
x2barlag	.8709937	1.848123	0.47	0.638	-2.759064	4.501051
xbar2lag	-1.022401	2.300926	-0.44	0.657	-5.541849	3.497046
rootxbarlag	.9753137	3.849605	0.25	0.800	-6.586029	8.536656
z1barlag	.1440083	1.855394	0.08	0.938	-3.500332	3.788348
z2barlag	2.61456	3.033885	0.86	0.389	-3.344555	8.573676
z3barlag	.3765959	.8616754	0.44	0.662	-1.315895	2.069087
z4barlag	-.5966162	1.046277	-0.57	0.569	-2.651699	1.458467
z5barlag	-.8922035	.6224268	-1.43	0.152	-2.114766	.3303588
z6barlag	1.831222	.4874443	3.76	0.000	.873791	2.788654
z7barlag	.1736821	.6277521	0.28	0.782	-1.05934	1.406704
_cons	-2.139894	2.846801	-0.75	0.453	-7.731542	3.451754

(option **xb** assumed; fitted values)
(110 missing values generated)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	8,962	.4313214	.1007003	.0602194	1.192568

(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)

Linear regression

Number of obs	=	8,962
F(129, 563)	=	6.36
Prob > F	=	0.0000
R-squared	=	0.2664
Root MSE	=	.03271

(Std. err. adjusted for **564** clusters in **group_round**)

qbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	-.0341188	.690451	-0.05	0.961	-1.390293	1.322056
__00009L	.4070717	.9493386	0.43	0.668	-1.457606	2.27175
__00009O	-.8022366	.4643586	-1.73	0.085	-1.714323	.1098502
xi	-.005772	.0060421	-0.96	0.340	-.0176398	.0060959
__000002	.0017224	.0020301	0.85	0.397	-.0022651	.0057098
__00000E	-.0049568	.0489346	-0.10	0.919	-.1010734	.0911599
__00000P	-.0025615	.0321059	-0.08	0.936	-.0656235	.0605005
__000017	-.0313852	.1297284	-0.24	0.809	-.2861961	.2234256
__00001I	.0786851	.0791846	0.99	0.321	-.0768482	.2342184
__000020	.0326666	.028622	1.14	0.254	-.0235524	.0888855
__00002B	.0031903	.0180702	0.18	0.860	-.0323029	.0386835
__00002T	.0901403	.0681822	1.32	0.187	-.0437823	.2240629
__000034	-.1478633	.0606483	-2.44	0.015	-.2669879	-.0287387
__00003M	.0262147	.0293709	0.89	0.372	-.0314752	.0839046
__00003X	-.0146086	.0198197	-0.74	0.461	-.0535381	.0243209
__00004F	.0140846	.0222275	0.63	0.527	-.0295744	.0577435
__00004Q	-.0142264	.0160344	-0.89	0.375	-.045721	.0172682
__000058	-.0085669	.0393383	-0.22	0.828	-.0858346	.0687008
__00005J	.000447	.0227288	0.02	0.984	-.0441966	.0450906
__00005U	.0075578	.0159199	0.47	0.635	-.0237118	.0388274
__00005V	.0529627	.0470043	1.13	0.260	-.0393625	.1452879
__00005W	-.0590386	.0493593	-1.20	0.232	-.1559895	.0379122
__00005X	.0585425	.0778301	0.75	0.452	-.0943304	.2114153
__00005Y	.0012544	.0384691	0.03	0.974	-.0743061	.076815
__00005Z	-.0543246	.0691702	-0.79	0.433	-.1901877	.0815386
__000060	-.0091848	.0254034	-0.36	0.718	-.0590818	.0407122
__000061	.0111994	.0260922	0.43	0.668	-.0400506	.0624494
__000062	.0114994	.0201155	0.57	0.568	-.0280112	.0510099
__000063	-.0298683	.0159297	-1.88	0.061	-.0611572	.0014205
__000064	-.029028	.0188929	-1.54	0.125	-.0661372	.0080811
__000065	-.0057894	.0345117	-0.17	0.867	-.0735767	.061998
__000066	.1713412	.0801333	2.14	0.033	.0139443	.328738

__000067	-.1925996	.1039719	-1.85	0.064	-.3968197	.0116205
__000068	-.1401419	.1587524	-0.88	0.378	-.4519611	.1716774
__000069	.1068272	.0849699	1.26	0.209	-.0600695	.2737238
__00006A	.2416682	.159787	1.51	0.131	-.0721833	.5555198
__00006B	-.1081304	.0558389	-1.94	0.053	-.2178084	.0015476
__00006C	-.0051583	.0557521	-0.09	0.926	-.1146659	.1043493
__00006D	-.0300668	.0417028	-0.72	0.471	-.1119789	.0518452
__00006E	-.0019598	.0434706	-0.05	0.964	-.0873443	.0834246
__00006F	.0932545	.0510802	1.83	0.068	-.0070764	.1935855
__00006G	-.0017613	.0084987	-0.21	0.836	-.0184544	.0149318
__00006H	.0061492	.0195585	0.31	0.753	-.0322673	.0445657
__00006I	.0013242	.024594	0.05	0.957	-.046983	.0496313
__00006J	-.0614577	.0438801	-1.40	0.162	-.1476465	.024731
__00006K	.0110156	.0276374	0.40	0.690	-.0432694	.0653005
__00006L	-.0005999	.0457767	-0.01	0.990	-.0905138	.089314
__00006M	.0048582	.0149706	0.32	0.746	-.024547	.0342633
__00006N	-.0046235	.0208259	-0.22	0.824	-.0455294	.0362825
__00006O	-.0133426	.0095854	-1.39	0.164	-.0321702	.005485
__00006P	.0030855	.0100204	0.31	0.758	-.0165965	.0227675
__00006Q	.0099788	.0115301	0.87	0.387	-.0126685	.0326261
__00006R	-.0324969	.0203955	-1.59	0.112	-.0725575	.0075636
__00006S	.1015647	.0743022	1.37	0.172	-.0443787	.2475081
__00006T	-.1556057	.0672986	-2.31	0.021	-.2877927	-.0234186
__00006U	.1594458	.1179245	1.35	0.177	-.0721799	.3910715
__00006V	.0354376	.0446494	0.79	0.428	-.0522622	.1231375
__00006W	-.1043122	.0915411	-1.14	0.255	-.284116	.0754916
__00006X	.0387947	.0245499	1.58	0.115	-.0094259	.0870153
__00006Y	.0476773	.0320531	1.49	0.137	-.015281	.1106355
__00006Z	-.0325869	.0264098	-1.23	0.218	-.0844607	.0192869
__000070	-.0444832	.0236352	-1.88	0.060	-.0909071	.0019408
__000071	-.0106114	.0264515	-0.40	0.688	-.062567	.0413443
__000072	-.0052922	.0099923	-0.53	0.597	-.024919	.0143347
__000073	.0117912	.0322726	0.37	0.715	-.0515982	.0751805
__000074	-.0122454	.0384802	-0.32	0.750	-.0878277	.063337
__000075	-.024849	.0453754	-0.55	0.584	-.1139747	.0642767
__000076	-.0022169	.0201426	-0.11	0.912	-.0417807	.0373469
__000077	-.0180076	.0484608	-0.37	0.710	-.1131937	.0771785
__000078	.0155227	.0133675	1.16	0.246	-.0107335	.041779
__000079	.0134312	.0146827	0.91	0.361	-.0154084	.0422709
__00007A	-.0196079	.0104577	-1.87	0.061	-.0401488	.000933
__00007B	.0087165	.0100127	0.87	0.384	-.0109503	.0283832
__00007C	.0192468	.0124439	1.55	0.123	-.0051953	.0436888
__00007D	.0053045	.0067236	0.79	0.430	-.007902	.0185109
__00007E	.0237456	.0165388	1.44	0.152	-.0087397	.0562309
__00007F	-.028425	.017965	-1.58	0.114	-.0637117	.0068617
__00007G	-.0393543	.0469467	-0.84	0.402	-.1315664	.0528578
__00007H	.0394149	.0216638	1.82	0.069	-.0031369	.0819666
__00007I	.0356437	.0408752	0.87	0.384	-.0446429	.1159303
__00007J	.0011803	.0119341	0.10	0.921	-.0222604	.024621
__00007K	.0264595	.0154164	1.72	0.087	-.0038212	.0567402
__00007L	-.0094736	.0092273	-1.03	0.305	-.0275978	.0086505
__00007M	.0017312	.0090912	0.19	0.849	-.0161256	.019588
__00007N	.0029805	.0105547	0.28	0.778	-.0177508	.0237118
__00007O	.0135426	.0102436	1.32	0.187	-.0065777	.0336628
__00007P	-.0110638	.0278207	-0.40	0.691	-.0657088	.0435813
__00007Q	.0037326	.0339204	0.11	0.912	-.0628934	.0703586
__00007R	-.0194374	.0574683	-0.34	0.735	-.1323159	.0934411
__00007S	.0103637	.0270276	0.38	0.702	-.0427236	.063451
__00007T	.0018623	.05051	0.04	0.971	-.0973486	.1010733
__00007U	-.0031765	.0182115	-0.17	0.862	-.0389473	.0325942
__00007V	-.0028281	.0186982	-0.15	0.880	-.0395549	.0338987
__00007W	.0190065	.0139192	1.37	0.173	-.0083334	.0463463
__00007X	.0333194	.0136872	2.43	0.015	.0064351	.0602037
__00007Y	.0225602	.0146241	1.54	0.123	-.0061643	.0512847
__000080	-.239221	.0819929	-2.92	0.004	-.4002703	-.0781717
__000082	.0687532	.0672044	1.02	0.307	-.0632487	.2007552
__000085	-.2557173	.2419027	-1.06	0.291	-.7308593	.2194246
__000087	.2214301	.1623069	1.36	0.173	-.0973708	.5402311
__00008A	-.025322	.3098749	-0.08	0.935	-.6339742	.5833301
__00008C	-.3907113	.216533	-1.80	0.072	-.8160226	.0345999
__00008F	1.329229	.6515908	2.04	0.042	.049383	2.609075
__00008H	.4727846	.4025823	1.17	0.241	-.3179621	1.263531

```
    __00008K |   -.187002   .2907563    -0.64   0.520    -.7581016    .3840976
    __00008M |    .0046829   .2136349     0.02   0.983    -.4149359    .4243016
    __00008P |   -1.391372   .6045978    -2.30   0.022    -2.578915   -.2038291
    __00008R |      .32389   .3605488     0.90   0.369     -.384295    1.032075
    __00008U |   -.2682841   .1389603    -1.93   0.054    -.5412281    .0046598
    __00008W |    .0824807   .0964205     0.86   0.393    -.1069071    .2718686
    __00008Z |    .3252096   .1476343     2.20   0.028     .0352284    .6151909
    __000091 |    .0229479   .1397396     0.16   0.870    -.2515267    .2974226
    __000094 |   -.1558374   .1189012    -1.31   0.191    -.3893815    .0777066
    __000096 |    .2940566   .0796238     3.69   0.000     .1376605    .4504526
    __000099 |   -.1714246   .1014534    -1.69   0.092     -.370698    .0278489
    __00009B |   -.1215481   .0647936    -1.88   0.061    -.2488149    .0057186
    __00009E |   -.3214703   .1217646    -2.64   0.009    -.5606388   -.0823019
    __00009G |   -.0971609   .0870289    -1.12   0.265     -.268102    .0737801
        xbar |    .209015   .1032493     2.02   0.043     .0062142    .4118158
    x2barlag |   -.0732115   .3356975    -0.22   0.827    -.7325839     .586161
    xbar2lag |    .5626247   .4515083     1.25   0.213    -.3242218    1.449471
  rootxbarlag |   -1.783549   .8047653    -2.22   0.027    -3.364258   -.2028396
    z1barlag |    .1192817    .408972     0.29   0.771    -.6840155     .922579
    z2barlag |    .9038338   .8144993     1.11   0.268    -.6959946    2.503662
    z3barlag |    .2223513   .1750022     1.27   0.204    -.1213857    .5660884
    z4barlag |   -.3662981    .195503    -1.87   0.062    -.7503025    .0177063
    z5barlag |   -.1362971   .1572647    -0.87   0.386    -.4451944    .1726002
    z6barlag |     .302317    .126073     2.40   0.017      .054686    .5499479
    z7barlag |    .3899952   .1638959     2.38   0.018      .068073    .7119173
       _cons |    .5299702    .583705     0.91   0.364    -.6165353    1.676476
------------------------------------------------------------------------------
(option xb assumed; fitted values)
(110 missing values generated)

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
    __0000A2 |      8,962    .0840991    .0195696  -.0065237   .1807836
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)
(110 missing values generated)

955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"

957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9

960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                 foreach var2 of global Alist {
    3.                         g `var1'_`var2'=`var1'*`var2'
    4.                         sum `var1' `var2' `var1'_`var2'
    5.                         local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                 }
    7.         }
964 }
```

```
965
966 local r_RE      "`r_RE' `qhat_hats'"
967
968 /*******************************************************************************
  > Set instruments and start values
  > *******************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }
978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS	Number of obs	=	9,072
				F(13, 9058)	=	3871.94
Model	265.283472	13	20.4064209	Prob > F	=	0.0000
Residual	47.7386914	9,058	.005270335	R-squared	=	0.8475
				Adj R-squared	=	0.8473
Total	313.022163	9,071	.03450801	Root MSE	=	.0726

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	-.0000405	.0000223	-1.82	0.069	-.0000842	3.15e-06
ybarx	-.0005265	.0012546	-0.42	0.675	-.0029858	.0019328
xi2	-.0448792	.0024031	-18.68	0.000	-.04959	-.0401685
ybar	.0044931	.001688	2.66	0.008	.0011843	.0078019
xi	.4542414	.0063598	71.42	0.000	.4417747	.4667081
ybar_q1	-.0139312	.0026363	-5.28	0.000	-.019099	-.0087635
pzi1	.1595646	.0043178	36.95	0.000	.1511007	.1680285
pzi2	.0189942	.0069637	2.73	0.006	.0053438	.0326447
pzi3	.0071562	.0021054	3.40	0.001	.0030292	.0112832
pzi4	.022049	.0031787	6.94	0.000	.0158181	.0282799
pzi5	-.0094989	.0017041	-5.57	0.000	-.0128393	-.0061585
pzi6	.0011517	.0016369	0.70	0.482	-.0020569	.0043603
pzi7	-.0029922	.0027226	-1.10	0.272	-.0083292	.0023448
_cons	.0022742	.0042143	0.54	0.589	-.0059867	.0105352

```
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = _b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                local coef = _b[pzi`j']
    3.                local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1025}

1026
1027if `J'==3 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                               local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00102183
  Iteration 1:  GMM criterion Q(b) =  .00060791
  Iteration 2:  GMM criterion Q(b) =  .00023826
  Iteration 3:  GMM criterion Q(b) =  .00015294
```

Step **2**
Iteration 0: GMM criterion Q(b) = **.01233673**
Iteration 1: GMM criterion Q(b) = **.01200687**
Iteration 2: GMM criterion Q(b) = **.01168807**
Iteration 3: GMM criterion Q(b) = **.01130952**
Iteration 4: GMM criterion Q(b) = **.01086462**
Iteration 5: GMM criterion Q(b) = **.01037298**
Iteration 6: GMM criterion Q(b) = **.00986152**
Iteration 7: GMM criterion Q(b) = **.00969151**
Iteration 8: GMM criterion Q(b) = **.00903236**
Iteration 9: GMM criterion Q(b) = **.00880646**
Iteration 10: GMM criterion Q(b) = **.00739297**
Iteration 11: GMM criterion Q(b) = **.00593206**
Iteration 12: GMM criterion Q(b) = **.00592957**

GMM estimation

Number of parameters = **20**
Number of moments = **28**
Initial weight matrix: **Unadjusted** Number of obs = **8,962**
GMM weight matrix: **Robust**

(Std. err. adjusted for **296** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
A11						
one	**.0523264**	**.0387695**	**1.35**	**0.177**	**-.0236604**	**.1283132**
C1						
zi1	**.273601**	**.1742129**	**1.57**	**0.116**	**-.0678501**	**.6150521**
zi2	**.2071944**	**.2311745**	**0.90**	**0.370**	**-.2458994**	**.6602881**
zi3	**-.0168841**	**.0938206**	**-0.18**	**0.857**	**-.2007691**	**.1670009**
zi4	**.318963**	**.1382022**	**2.31**	**0.021**	**.0480916**	**.5898344**
zi5	**-.0061869**	**.0452848**	**-0.14**	**0.891**	**-.0949435**	**.0825696**
zi6	**.0987097**	**.0648179**	**1.52**	**0.128**	**-.028331**	**.2257504**
zi7	**.0337345**	**.104112**	**0.32**	**0.746**	**-.1703213**	**.2377903**
C2						
zi1	**.0194626**	**.2412798**	**0.08**	**0.936**	**-.4534371**	**.4923622**
zi2	**.2802641**	**.3053298**	**0.92**	**0.359**	**-.3181713**	**.8786996**
zi3	**-.0333947**	**.1232315**	**-0.27**	**0.786**	**-.274924**	**.2081346**
zi4	**.3616451**	**.1400084**	**2.58**	**0.010**	**.0872336**	**.6360566**
zi5	**.0066137**	**.0603786**	**0.11**	**0.913**	**-.1117261**	**.1249535**
zi6	**.1364136**	**.0870853**	**1.57**	**0.117**	**-.0342704**	**.3070976**
zi7	**.0614613**	**.1625785**	**0.38**	**0.705**	**-.2571866**	**.3801093**
/d1	**-.0735326**	**.0192562**	**-3.82**	**0.000**	**-.111274**	**-.0357913**
/b1	**.4279066**	**.0553557**	**7.73**	**0.000**	**.3194113**	**.5364019**
/AVA11	**1.023858**	**2.307065**	**0.44**	**0.657**	**-3.497907**	**5.545623**
/AVA12	**-.8440659**	**1.911314**	**-0.44**	**0.659**	**-4.590172**	**2.90204**
/AVA22	**.8441163**	**1.707532**	**0.49**	**0.621**	**-2.502585**	**4.190818**

Instruments for equation **eq1_RE**: **__00009J __00009L __00009O xi __000002 __00000E __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5 __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons**

```
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
> _b[/a]+1.19*_b[/b])
```

```
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactIns
      > t_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dropPre_onlySch
      > edCrossGrp_RE.tex not found)
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dropPre_
  > onlySchedCrossGrp_RE.tex)
```

```
1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  2430
  Number of records is  9072

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  2430

1077unique group_round
  Number of unique values of group_round is  572
  Number of records is  9072

1078estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  572

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  4.2482517

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dropPre_onlySch
  > edCrossGrp_RE.ster not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dropPre_onlySchedCross
  > Grp_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
```

```
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*`p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
1119                                                 gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                  local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                          }
   30.                                  }
   31.                          }
   32.                  }
   33.          }
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
```

```
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
1120        if `Tindivp1'<=`T' {
1121                forval t=`Tindivp1'/`T' {
    2.                          tempvar zi`t'xdiff
    3.                          g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                          local zg "`zg' zi`t'"
    5.                          forval j=1/`J' {
    6.                                  tempvar zi`t'p`j'xdiff
    7.                                  g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                  local pzg "`pzg' `zi`t'p`j''"
    9.                                  local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                          }
   11.                 }
1122        }
1123
1124        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1125        // denoting zg as group-zs
1126        //  zg,pzgx,rprp,rprpx
1127        local rootprootp ""
1128        local rootprootpxdiff ""
1129
1130        forval j=1/`J' {
    2.                  forval s=`j'/`J' {
    3.                          tempvar xdiffrp`j'rp`s'
    4.                          g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1131                        if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1132                        local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                  }
    8.          }
1133
1134        local pqhatxdiff ""
1135        forval j=1/`J' {
    2.                  forval s=1/`J' {
    3.                          foreach name of global Alist {
    4.                                  tempvar   p`s'qikhat`j'xdiff`name'
    5.                                  g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
```

```
1136                                local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
  (110 missing values generated)
1137
1138        // set instruments
1139        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1140}

1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                          local `++t'
    3.                          local p_nice = subinstr("`p'",":_cons","",.)
    4.                          local est = coefs[1,`t']
    5.                          if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                  }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
```

```
1172                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                  }
   16.         }
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1    `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
```

```
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00098892
  Iteration 1:  GMM criterion Q(b) =   .0005983
  Iteration 2:  GMM criterion Q(b) =  .00055023
  Iteration 3:  GMM criterion Q(b) =  .00053145

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .03433075
  Iteration 1:  GMM criterion Q(b) =  .03055171
  Iteration 2:  GMM criterion Q(b) =  .02918518
  Iteration 3:  GMM criterion Q(b) =  .02898124

  GMM estimation

  Number of parameters =  17
  Number of moments    =  63
  Initial weight matrix: Unadjusted                 Number of obs   =      8,962
  GMM weight matrix:     Robust
```

(Std. err. adjusted for **296** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11						
_cons	**.4699463**	**.0675135**	**6.96**	**0.000**	**.3376223**	**.6022704**
C12						
_cons	**.0546345**	**.1291707**	**0.42**	**0.672**	**-.1985356**	**.3078045**
C13						
_cons	**-.0218117**	**.0405536**	**-0.54**	**0.591**	**-.1012953**	**.0576719**
C14						
_cons	**.0837672**	**.0476018**	**1.76**	**0.078**	**-.0095306**	**.177065**
C15						
_cons	**-.0001059**	**.0373699**	**-0.00**	**0.998**	**-.0733496**	**.0731377**
C16						
_cons	**-.057386**	**.0343492**	**-1.67**	**0.095**	**-.1247091**	**.0099372**
C17						
_cons	**-.0267603**	**.0429085**	**-0.62**	**0.533**	**-.1108593**	**.0573388**
C21						
_cons	**.2691405**	**.0813533**	**3.31**	**0.001**	**.1096908**	**.4285901**
C22						
_cons	**.0638409**	**.1582172**	**0.40**	**0.687**	**-.246259**	**.3739409**
C23						
_cons	**-.014236**	**.0477996**	**-0.30**	**0.766**	**-.1079215**	**.0794495**
C24						
_cons	**.1079371**	**.0571862**	**1.89**	**0.059**	**-.0041458**	**.22002**
C25						
_cons	**.0163033**	**.0431025**	**0.38**	**0.705**	**-.068176**	**.1007827**
C26						
_cons	**-.0753167**	**.0409297**	**-1.84**	**0.066**	**-.1555375**	**.0049041**
C27						
_cons	**-.0117553**	**.0528997**	**-0.22**	**0.824**	**-.1154368**	**.0919261**
d1						
_cons	**-.0896548**	**.0137025**	**-6.54**	**0.000**	**-.1165112**	**-.0627983**
b1						
_cons	**.478547**	**.040122**	**11.93**	**0.000**	**.3999094**	**.5571847**
A11						
one	**.0105709**	**.3016704**	**0.04**	**0.972**	**-.5806922**	**.601834**

Instruments for equation **eq1_FE**: **__000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7 __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1 __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY __0000D5 _cons**

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactIns
      > t_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dropPre_onlySch
      > edCrossGrp_FE.tex not found)
  (output written to diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dropPre_
  > onlySchedCrossGrp_FE.tex)
1242
1243        estat overid

    Test of overidentifying restriction:

    Hansen's J chi2(46) =  259.73 (p = 0.0000)
1244        di "$S_DATE $S_TIME"
   7 Dec 2023 21:14:57
1245        unique hhidi
  Number of unique values of hhidi is  2430
  Number of records is  9072
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  2430
1248        unique group_round
  Number of unique values of group_round is  572
  Number of records is  9072
1249        estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  572
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  4.2482517
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dropPre_onlySch
  > edCrossGrp_FE.ster not found)
  file diag_2good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_NoMeasError_urbOnly_clus_Dist_dropPre_onlySchedCross
  > Grp_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
```

```
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}

1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table5_24.smcl
   log type:  smcl
  closed on:   7 Dec 2023, 21:14:57
```

A.6 Table 6

A.6.1 Columns 1 and 4

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table6_14.smcl
  log type:  smcl
 opened on:   7 Dec 2023, 21:26:23

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
  C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""
35. if $rand_grp==1 local grp_type "_Random"
36.
37. // make summary stats table
38. local make_sumstats = 0
39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0
42. local typeExp ""
43. if $meas_pc==1 local typeExp "_ExpPC"
44.
45. // number of goods
46. scalar J=2
47. local J = J
48.
49. // group def options
50. local group_def "dist"
51. local group_def "fsuXseg"
52. local group_def "fsuXsegXreligXsched"
53.
54. // comparision group
55. local groupComp_def ""
56. local desc_compGrp ""
57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"
58.
59. // load data
60. include "$CODE/prep demand system data.do"
61. //clear all
62. pause on
63. set more off
64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.
65.
66. scalar short_zlist=$short_zlist_val
67.
68. cap prog drop bysmeanw
69. prog def bysmeanw
    1.         syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
70.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations               840   ->   280
  Number of variables                    5   ->   6
  j variable (3 values)       expend_type   ->   (dropped)
  xij variables:
                           laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.             }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,637 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                    egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
```

```
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }
297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }
316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }
333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
```

```
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (82 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,576 real changes made, 82 to missing)

460 replace x2=x*x
  (73,576 real changes made, 82 to missing)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.          g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,576 real changes made)

478 replace x2=x2_temp
  (73,576 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (48,819 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)
```

```
488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_41e4_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
```

```
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,757 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
549 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var'" _n
    5.                 }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
```

```
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=1

597 local descA "full"

598 scalar Adiag=0

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=0

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1
```

```
scalar estimate_FE=1

// estimation options
global w_initial "winit(unadjusted)"
if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"
global wmatrix "wmatrix(robust)"
global trace_level ""
global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"
global technique ""
global maxiter ""
global technique_RE ""
if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"
global derivatives "quickderivatives"
scalar init_FE_from_RE=1
global clust_var "state_district_round"
local clus_name "_clus_Dist"









// xxzx to keep names same
local wmatrix_desc ""
local sectorName "_urbOnly"
local dropZdesc "_dropPre"
local desc_struc_v0 ""
local inst_desc "_exactInst"
local inst_rootp "_noPPXdInst"
local weight_desc ""
local RE_sq_inst_desc ""
local cnstname ""
local actual_pqhat_desc ""
local expUpname "_updateExp"
```

```
661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //              ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T

694 local Tm1=`T'-1
```

```
695 local Tindiv : word count $zlist_indiv

696
697 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
706
707 if `J'==2 {
708          replace p1=P1
  (128,640 real changes made)
709          replace p2=P2
  (128,640 real changes made)
710          replace qi1=Qi1
  (128,632 real changes made)
711          replace qk1=Qk1
  (128,632 real changes made)
712          replace qikbar1=Qikbar1
  (128,640 real changes made)
713          replace qi2=Qi2
  (128,640 real changes made)
714          replace qk2=Qk2
  (128,640 real changes made)
715          replace qikbar2=Qikbar2
  (128,640 real changes made)
716
717          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721          forv j = 1(1)3 {
    2.                 replace p`j'=P`j'
    3.                 replace qi`j'=Qi`j'
    4.                 replace qk`j'=Qk`j'
    5.                 replace qikbar`j'=Qikbar`j'
    6.          }
722
723          foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
724 }

725
```

```
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }
```

```
751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
773         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j'"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.          local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.          if `jplus1'<=`J' {
    5.                  forval k=`jplus1'/`J' {
    6.                          local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                  }
    8.          }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                  if noMeasError==1                               local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
```

```
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                      l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                    )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ( {A11:}*p1*qk1 + {A12:}*p1*qk2 ) + ( {A21:}*p2*qk1 + {A22:}*p
  > 2*qk2 ) )*( ( {A11:}*p1*qikbar1 + {A12:}*p1*qikbar2 ) + ( {A21:}*p2*qikbar1 + {A22:}
  > *p2*qikbar2 ) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2: zi1
  > zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ( {A11:}*p1*qikbar1 + {A12:}*p1*qikbar2 ) +
  > ( {A21:}*p2*qikbar1 + {A22:}*p2*qikbar2 ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) -
  > ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln
  > (p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )-( ( {A11:}*p1*qikbar1 + {A12:}
  > *p1*qikbar2 ) + ( {A21:}*p2*qikbar1 + {A22:}*p2*qikbar2 ) )*{b1} + ( {A11:}*p1*qikb
  > ar1 + {A12:}*p1*qikbar2 ) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p
  > 2 + p2*{AVA22}*p2 )*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))*({b1} - 2*(( ( {A11
  > :}*p1*qikbar1 + {A12:}*p1*qikbar2 ) + ( {A21:}*p2*qikbar1 + {A22:}*p2*qikbar2 ) ) +
  > ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 +
  > p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*
  > zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 +
  >  p1*{C16}*zk6 + p1*{C17}*zk7 ))         ))
```

```
836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ( {A11:one}*p1*qk1 + {A12:one}*p1*qk2 ) + ( {A21:one}*p2*qk1 +
  >  {A22:one}*p2*qk2 ) )*( ( {A11:}*p1*qikbar1 + {A12:}*p1*qikbar2 ) + ( {A21:}*p2*qikb
  > ar1 + {A22:}*p2*qikbar2 ) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (
  > p2*{C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ( {A11:}*p1*qikbar1 + {A12:}*p1*
  > qikbar2 ) + ( {A21:}*p2*qikbar1 + {A22:}*p2*qikbar2 ) )   + (xi - ( (p1*{C1:}) + (p2
  > *{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )) * {d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )-( ( {A11:}*p1*qikb
  > ar1 + {A12:}*p1*qikbar2 ) + ( {A21:}*p2*qikbar1 + {A22:}*p2*qikbar2 ) )*{b1} + ( {A
  > 11:}*p1*qikbar1 + {A12:}*p1*qikbar2 ) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*
  > p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))*({b1} - 2*(( ( {A11
  > :one}*p1*qikbar1 + {A12:one}*p1*qikbar2 ) + ( {A21:one}*p2*qikbar1 + {A22:one}*p2*qi
  > kbar2 ) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) )) - (( p1*
  > {C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6
  >  + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1
  > *{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         ))

874
875 /*******************************************************************************
  >  construct instruments
  > *******************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi
```

```
882 g double `ximxk'=xi-xk

883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.         tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.         g double `xizi`t''=xi*zi`t'
    4.         g double `xkzk`t''=xk*zk`t'
    5.         g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.         g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.         g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.         g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.         g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.         forval j=1/`J' {
   11.                 tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                 g double `zi`t'p`j''= zi`t'*p`j'
   13.                 g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                 g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                 g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                 forv s=1/`T' {
   17.                         tempvar zi`t'zi`s'p`j'
   18.                         g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                 }
   20.         }
   21. }

891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                 local rg "`rg' z`t'barlag"
    3. }

903
904 local zrg ""
```

```
905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var'"
    6.          }
    7. }

906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var'"
    5.
911         forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var'"
   10.                  local xprg "`xprg' `xp`j'`var'"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""
```

```
925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }

929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.          forval k=`j'/`J' {
    3.                  tempvar p`j'p`k'
    4.                  g `p`j'p`k''=p`j'*p`k'
    5.                  local roundinsts "`roundinsts' `p`j'p`k''"
    6.          }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.          if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.          if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
```

```
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }
```

Linear regression

Number of obs	**128,640**
F(129, 4598)	**14.99**
Prob > F	**0.0000**
R-squared	**0.2238**
Root MSE	**.131**

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**-12.79512**	**7.260345**	**-1.76**	**0.078**	**-27.02888**	**1.438647**
__00009L	**-3.637138**	**1.251887**	**-2.91**	**0.004**	**-6.091436**	**-1.182839**
__00009O	**1.004898**	**3.873148**	**0.26**	**0.795**	**-6.588331**	**8.598128**
xi	**.1090911**	**.0077876**	**14.01**	**0.000**	**.0938237**	**.1243585**
__000002	**-.0157421**	**.0025123**	**-6.27**	**0.000**	**-.0206674**	**-.0108169**
__00000E	**-.1156487**	**.0903327**	**-1.28**	**0.201**	**-.2927442**	**.0614468**
__00000P	**-.0139978**	**.0521173**	**-0.27**	**0.788**	**-.1161727**	**.0881771**
__000017	**.0509528**	**.1735472**	**0.29**	**0.769**	**-.289283**	**.3911886**
__00001I	**-.0239111**	**.1060934**	**-0.23**	**0.822**	**-.2319051**	**.184083**
__000020	**.0171582**	**.04793**	**0.36**	**0.720**	**-.0768076**	**.111124**
__00002B	**-.025461**	**.0245826**	**-1.04**	**0.300**	**-.0736547**	**.0227327**
__00002T	**.047008**	**.0663415**	**0.71**	**0.479**	**-.0830531**	**.1770692**
__000034	**.0018229**	**.0474534**	**0.04**	**0.969**	**-.0912086**	**.0948543**
__00003M	**.0130708**	**.0588349**	**0.22**	**0.824**	**-.1022739**	**.1284156**
__00003X	**.0098183**	**.0379708**	**0.26**	**0.796**	**-.0646228**	**.0842594**
__00004F	**-.025421**	**.0466786**	**-0.54**	**0.586**	**-.1169335**	**.0660916**
__00004Q	**.0074369**	**.0250442**	**0.30**	**0.767**	**-.0416617**	**.0565356**
__000058	**-.0217155**	**.0609657**	**-0.36**	**0.722**	**-.1412376**	**.0978066**
__00005J	**.0055081**	**.0340618**	**0.16**	**0.872**	**-.0612694**	**.0722856**
__00005U	**.0797255**	**.4007228**	**0.20**	**0.842**	**-.7058835**	**.8653345**
__00005V	**-.0346695**	**.0711757**	**-0.49**	**0.626**	**-.174208**	**.1048689**
__00005W	**-.0306542**	**.1635038**	**-0.19**	**0.851**	**-.3512**	**.2898916**
__00005X	**.1385994**	**.4167248**	**0.33**	**0.739**	**-.6783813**	**.9555801**
__00005Y	**-.0224787**	**.1028475**	**-0.22**	**0.827**	**-.2241091**	**.1791517**
__00005Z	**.0485407**	**.1674104**	**0.29**	**0.772**	**-.279664**	**.3767454**
__000060	**-.0305375**	**.0696472**	**-0.44**	**0.661**	**-.1670794**	**.1060045**
__000061	**.0243364**	**.0499235**	**0.49**	**0.626**	**-.0735376**	**.1222104**
__000062	**.0703686**	**.0341366**	**2.06**	**0.039**	**.0034444**	**.1372928**
__000063	**-.0376336**	**.047996**	**-0.78**	**0.433**	**-.1317289**	**.0564616**
__000064	**-.0701443**	**.057982**	**-1.21**	**0.226**	**-.1838168**	**.0435282**
__000065	**-.1665861**	**.877927**	**-0.19**	**0.850**	**-1.887744**	**1.554572**
__000066	**.2283964**	**.1442115**	**1.58**	**0.113**	**-.0543274**	**.5111203**
__000067	**-.3199538**	**.3445552**	**-0.93**	**0.353**	**-.9954474**	**.3555397**
__000068	**.5394784**	**.8576131**	**0.63**	**0.529**	**-1.141855**	**2.220812**
__000069	**-.1708187**	**.1878736**	**-0.91**	**0.363**	**-.5391411**	**.1975038**
__00006A	**-.424151**	**.3283207**	**-1.29**	**0.196**	**-1.067817**	**.2195151**
__00006B	**-.0463852**	**.1238777**	**-0.37**	**0.708**	**-.289245**	**.1964745**
__00006C	**-.2113415**	**.1076302**	**-1.96**	**0.050**	**-.4223483**	**-.0003347**
__00006D	**.0467054**	**.0589613**	**0.79**	**0.428**	**-.068887**	**.1622978**
__00006E	**-.1444075**	**.0994178**	**-1.45**	**0.146**	**-.339314**	**.0504991**
__00006F	**.030443**	**.1210695**	**0.25**	**0.801**	**-.2069113**	**.2677973**
__00006G	**-.1629455**	**.1849126**	**-0.88**	**0.378**	**-.5254629**	**.199572**
__00006H	**.0332459**	**.032772**	**1.01**	**0.310**	**-.031003**	**.0974948**

00006I	-.0177702	.0723866	-0.25	0.806	-.1596827	.1241422
00006J	.1731219	.2093027	0.83	0.408	-.2372119	.5834557
00006K	.0504318	.0524078	0.96	0.336	-.0523128	.1531763
00006L	-.0507186	.0851838	-0.60	0.552	-.2177197	.1162825
00006M	-.0292887	.0358816	-0.82	0.414	-.0996338	.0410563
00006N	.0309542	.0234174	1.32	0.186	-.0149551	.0768636
00006O	-.014908	.0145441	-1.03	0.305	-.0434214	.0136054
00006P	-.0290572	.0253252	-1.15	0.251	-.0787067	.0205924
00006Q	.0028381	.030826	0.09	0.927	-.0575957	.0632719
00006R	-.6012457	.495913	-1.21	0.225	-1.573473	.3709819
00006S	-.0369337	.0565668	-0.65	0.514	-.1478319	.0739644
00006T	.2871989	.1903878	1.51	0.131	-.0860527	.6604504
00006U	.3847497	.4323705	0.89	0.374	-.4629041	1.232403
00006V	-.1362284	.0767153	-1.78	0.076	-.2866273	.0141705
00006W	-.0523872	.1430883	-0.37	0.714	-.332909	.2281346
00006X	.0125271	.0552197	0.23	0.821	-.0957301	.1207843
00006Y	.0345059	.0349869	0.99	0.324	-.0340851	.103097
00006Z	-.0725883	.0241739	-3.00	0.003	-.1199808	-.0251958
000070	-.0109506	.0430715	-0.25	0.799	-.0953914	.0734902
000071	-.0582328	.0494466	-1.18	0.239	-.1551719	.0387062
000072	-.4534528	.305937	-1.48	0.138	-1.053236	.1463307
000073	-.0906118	.0458174	-1.98	0.048	-.180436	-.0007877
000074	.2961195	.1206212	2.45	0.014	.0596441	.5325949
000075	.2853582	.2940211	0.97	0.332	-.2910644	.8617807
000076	-.076137	.0648958	-1.17	0.241	-.2033639	.0510898
000077	.0084989	.1095353	0.08	0.938	-.2062429	.2232407
000078	-.0029301	.0439453	-0.07	0.947	-.0890839	.0832237
000079	-.0005896	.0346299	-0.02	0.986	-.0684808	.0673017
00007A	.0293777	.0211752	1.39	0.165	-.0121359	.0708912
00007B	-.0056913	.0314004	-0.18	0.856	-.0672512	.0558686
00007C	-.0796128	.0372845	-2.14	0.033	-.1527083	-.0065174
00007D	.1277082	.2134185	0.60	0.550	-.2906944	.5461109
00007E	.0131872	.0355408	0.37	0.711	-.0564899	.0828643
00007F	-.0882595	.083501	-1.06	0.291	-.2519616	.0754427
00007G	.006318	.2231374	0.03	0.977	-.4311384	.4437744
00007H	-.0032218	.0521663	-0.06	0.951	-.1054929	.0990492
00007I	-.0735209	.0881929	-0.83	0.405	-.2464212	.0993795
00007J	-.0161166	.0327932	-0.49	0.623	-.0804071	.0481739
00007K	-.0226722	.0278317	-0.81	0.415	-.0772358	.0318913
00007L	.0113972	.0160408	0.71	0.477	-.0200505	.0428448
00007M	-.0074927	.0281454	-0.27	0.790	-.0626712	.0476857
00007N	.0512257	.0317663	1.61	0.107	-.0110516	.113503
00007O	-.0216683	.3475372	-0.06	0.950	-.7030081	.6596714
00007P	.0879331	.0479654	1.83	0.067	-.0061021	.1819684
00007Q	-.123569	.1328924	-0.93	0.353	-.3841018	.1369638
00007R	.1056987	.3234608	0.33	0.744	-.5284398	.7398371
00007S	.0276477	.0679204	0.41	0.684	-.1055088	.1608043
00007T	-.2030032	.1161087	-1.75	0.080	-.430632	.0246256
00007U	.0181395	.0416299	0.44	0.663	-.063475	.0997541
00007V	-.0771625	.0330746	-2.33	0.020	-.1420047	-.0123204
00007W	-.0141401	.0196985	-0.72	0.473	-.0527587	.0244784
00007X	.0035144	.0352185	0.10	0.921	-.0655307	.0725594
00007Y	.0380758	.0423369	0.90	0.369	-.0449249	.1210764
000080	-36.81121	17.02884	-2.16	0.031	-70.19592	-3.42651
000082	.0760369	8.506323	0.01	0.993	-16.60044	16.75251
000085	-.6294823	.4059554	-1.55	0.121	-1.42535	.1663852
000087	.4964106	.2690436	1.85	0.065	-.0310441	1.023865
00008A	8.027308	3.333333	2.41	0.016	1.492375	14.56224
00008C	-1.201791	1.594737	-0.75	0.451	-4.32824	1.924659
00008F	44.59295	20.99826	2.12	0.034	3.426279	85.75962
00008H	2.168832	10.94407	0.20	0.843	-19.2868	23.62447
00008K	-.4921474	.5029834	-0.98	0.328	-1.478236	.4939415
00008M	-.8762901	.4582283	-1.91	0.056	-1.774638	.0220574
00008P	-1.190171	.9196827	-1.29	0.196	-2.99319	.612849
00008R	-1.132574	.642393	-1.76	0.078	-2.391973	.1268243
00008U	.1009246	.3299369	0.31	0.760	-.5459101	.7477593
00008W	.1590723	.2320867	0.69	0.493	-.295929	.6140737
00008Z	.7462393	.2282715	3.27	0.001	.2987175	1.193761
000091	-.4374243	.1583699	-2.76	0.006	-.7479053	-.1269433
000094	.0122267	.1794758	0.07	0.946	-.339632	.3640855
000096	.1116296	.1418125	0.79	0.431	-.1663908	.3896501
000099	-.0594555	.2718099	-0.22	0.827	-.5923333	.4734223

__00009B	-.0418895	.1972203	-0.21	0.832	-.4285359	.3447569
__00009E	.0027748	.3171447	0.01	0.993	-.618981	.6245306
__00009G	-.4676324	.2253747	-2.07	0.038	-.9094749	-.0257898
xbarlag	38.11984	21.35656	1.78	0.074	-3.749277	79.98895
x2barlag	-.1896805	.4397159	-0.43	0.666	-1.051735	.6723737
xbar2lag	-6.519805	4.180344	-1.56	0.119	-14.71529	1.675676
rootxbarlag	-48.72697	26.43457	-1.84	0.065	-100.5514	3.097471
z1barlag	1.544271	.7682996	2.01	0.044	.0380348	3.050507
z2barlag	2.57067	1.188866	2.16	0.031	.2399211	4.901418
z3barlag	-.2976042	.4371373	-0.68	0.496	-1.154603	.5593949
z4barlag	-.1807586	.2681451	-0.67	0.500	-.7064517	.3449345
z5barlag	-.1795307	.1890717	-0.95	0.342	-.5502021	.1911407
z6barlag	.1391592	.3756895	0.37	0.711	-.5973725	.8756909
z7barlag	.430906	.4037833	1.07	0.286	-.3607032	1.222515
_cons	16.33104	9.106066	1.79	0.073	-1.521223	34.1833

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	128,640	.4424145	.0703047	.2011188	.8127233

Linear regression

Number of obs	=	128,640
F(129, 4598)	=	46.28
Prob > F	=	0.0000
R-squared	=	0.3053
Root MSE	=	.20268

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	4.564747	8.325438	0.55	0.584	-11.75711	20.8866
__00009L	.5937963	2.171845	0.27	0.785	-3.664063	4.851655
__00009O	-.8608651	4.900916	-0.18	0.861	-10.46901	8.747282
xi	.1450715	.0142385	10.19	0.000	.1171571	.1729859
__000002	-.0081835	.0053966	-1.52	0.129	-.0187634	.0023964
__00000E	-.1996901	.1318225	-1.51	0.130	-.4581254	.0587452
__00000P	-.1306909	.0773583	-1.69	0.091	-.2823502	.0209685
__000017	.3219988	.2747868	1.17	0.241	-.2167153	.8607128
__00001I	.2451433	.1716831	1.43	0.153	-.091438	.5817245
__000020	.0465067	.0713123	0.65	0.514	-.0932996	.186313
__00002B	.0499324	.0392375	1.27	0.203	-.026992	.1268568
__00002T	.3317416	.0880092	3.77	0.000	.1592013	.5042819
__000034	-.0424598	.0611807	-0.69	0.488	-.1624034	.0774838
__00003M	-.0824665	.0821755	-1.00	0.316	-.2435698	.0786368
__00003X	.0562813	.0513967	1.10	0.274	-.0444809	.1570436
__00004F	-.0214064	.0649749	-0.33	0.742	-.1487884	.1059755
__00004Q	-.0033885	.0382735	-0.09	0.929	-.0784229	.0716459
__000058	-.0616114	.095554	-0.64	0.519	-.248943	.1257203
__00005J	.0962271	.0576901	1.67	0.095	-.0168731	.2093274
__00005U	-1.072842	.6175735	-1.74	0.082	-2.283583	.1378984
__00005V	-.0686079	.112662	-0.61	0.543	-.2894794	.1522637
__00005W	.2956557	.2555724	1.16	0.247	-.2053889	.7967004
__00005X	1.107247	.6463172	1.71	0.087	-.1598453	2.374339
__00005Y	.006665	.1525531	0.04	0.965	-.2924124	.3057424
__00005Z	.4177337	.2272567	1.84	0.066	-.0277986	.863266
__000060	-.2003814	.1148141	-1.75	0.081	-.4254721	.0247093
__000061	.0627776	.0573305	1.10	0.274	-.0496177	.1751728
__000062	-.0079371	.0479	-0.17	0.868	-.101844	.0859698
__000063	.0234324	.0606999	0.39	0.699	-.0955686	.1424334
__000064	-.0476537	.0743598	-0.64	0.522	-.1934346	.0981272
__000065	4.601567	1.273473	3.61	0.000	2.104948	7.098186
__000066	.5353632	.231593	2.31	0.021	.0813297	.9893968
__000067	-2.104046	.5308536	-3.96	0.000	-3.144774	-1.063318
__000068	-3.209865	1.309425	-2.45	0.014	-5.776967	-.6427636
__000069	-.5733963	.2998063	-1.91	0.056	-1.16116	.0143679
__00006A	-.3073944	.5099257	-0.60	0.547	-1.307093	.6923047
__00006B	-.0013787	.1805182	-0.01	0.994	-.3552811	.3525236
__00006C	-.6917272	.1201482	-5.76	0.000	-.9272752	-.4561791
__00006D	.0978464	.0930479	1.05	0.293	-.0845721	.280265

__00006E	-.1126402	.1459926	-0.77	0.440	-.3988558	.1735754
__00006F	.0293601	.1813666	0.16	0.871	-.3262055	.3849256
__00006G	.228589	.3106349	0.74	0.462	-.3804045	.8375825
__00006H	.0378065	.0490674	0.77	0.441	-.0583892	.1340023
__00006I	-.1229966	.1220438	-1.01	0.314	-.362261	.1162679
__00006J	-.2612815	.3194111	-0.82	0.413	-.8874806	.3649176
__00006K	.042626	.077334	0.55	0.582	-.1089857	.1942377
__00006L	-.0857245	.1255653	-0.68	0.495	-.3318928	.1604438
__00006M	.0143417	.0615696	0.23	0.816	-.1063642	.1350476
__00006N	.0142187	.0300003	0.47	0.636	-.0445962	.0730337
__00006O	.0108951	.0238843	0.46	0.648	-.0359295	.0577198
__00006P	-.051378	.0341179	-1.51	0.132	-.1182654	.0155094
__00006Q	.0061354	.0401718	0.15	0.879	-.0726205	.0848914
__00006R	-.3954866	.5294751	-0.75	0.455	-1.433512	.6425387
__00006S	.0027485	.070072	0.04	0.969	-.1346264	.1401233
__00006T	.2422518	.1891001	1.28	0.200	-.1284752	.6129788
__00006U	-.2749271	.4753902	-0.58	0.563	-1.20692	.657066
__00006V	.1610918	.0976758	1.65	0.099	-.0303997	.3525833
__00006W	.1256103	.177919	0.71	0.480	-.2231964	.474417
__00006X	.0319531	.0688723	0.46	0.643	-.1030697	.1669758
__00006Y	.0856173	.0433777	1.97	0.048	.0005762	.1706584
__00006Z	-.1346033	.0315098	-4.27	0.000	-.1963778	-.0728289
__000070	-.0896637	.0497818	-1.80	0.072	-.18726	.0079325
__000071	-.0242239	.0620782	-0.39	0.696	-.145927	.0974792
__000072	-.817985	.3639092	-2.25	0.025	-1.531422	-.1045482
__000073	-.1058926	.0632791	-1.67	0.094	-.2299501	.0181648
__000074	.4129247	.154788	2.67	0.008	.109466	.7163835
__000075	.4604809	.3667051	1.26	0.209	-.258437	1.179399
__000076	.1273306	.0859626	1.48	0.139	-.0411973	.2958585
__000077	.1296378	.1391035	0.93	0.351	-.1430718	.4023475
__000078	-.0674194	.0576484	-1.17	0.242	-.1804379	.0455991
__000079	.0872997	.0407137	2.14	0.032	.0074813	.167118
__00007A	.0612698	.02713	2.26	0.024	.008082	.1144575
__00007B	.0328333	.0413449	0.79	0.427	-.0482225	.1138891
__00007C	.0255229	.0482849	0.53	0.597	-.0691387	.1201845
__00007D	.354494	.2770601	1.28	0.201	-.1886768	.8976647
__00007E	.0890569	.0489815	1.82	0.069	-.0069704	.1850841
__00007F	-.2285898	.1151561	-1.99	0.047	-.4543511	-.0028285
__00007G	-.1664105	.2810873	-0.59	0.554	-.7174767	.3846556
__00007H	-.0569692	.0661062	-0.86	0.389	-.186569	.0726307
__00007I	-.0611722	.1122723	-0.54	0.586	-.2812799	.1589354
__00007J	.0315184	.0411415	0.77	0.444	-.0491387	.1121755
__00007K	-.0752832	.0302422	-2.49	0.013	-.1345724	-.015994
__00007L	.0125053	.0227207	0.55	0.582	-.0320382	.0570487
__00007M	.0286478	.0317049	0.90	0.366	-.0335091	.0908047
__00007N	.0174569	.0380542	0.46	0.646	-.0571476	.0920614
__00007O	.2851267	.4558075	0.63	0.532	-.6084747	1.178728
__00007P	.2207786	.0723679	3.05	0.002	.0789028	.3626543
__00007Q	-.3915989	.1823773	-2.15	0.032	-.7491459	-.0340519
__00007R	-.012365	.4576998	-0.03	0.978	-.9096763	.8849463
__00007S	-.1570488	.1017663	-1.54	0.123	-.3565595	.042462
__00007T	-.4053457	.1710711	-2.37	0.018	-.7407271	-.0699642
__00007U	.0972112	.0620585	1.57	0.117	-.0244534	.2188757
__00007V	-.2339363	.0420127	-5.57	0.000	-.3163014	-.1515711
__00007W	-.0218837	.0324529	-0.67	0.500	-.085507	.0417396
__00007X	.0212272	.0497507	0.43	0.670	-.076308	.1187624
__00007Y	.005633	.0584831	0.10	0.923	-.109022	.1202879
__000080	8.445766	18.96278	0.45	0.656	-28.73038	45.62192
__000082	-2.538594	11.18003	-0.23	0.820	-24.45681	19.37962
__000085	-.0754758	.5249525	-0.14	0.886	-1.104635	.9536832
__000087	.2107624	.3791253	0.56	0.578	-.5325052	.9540299
__00008A	-1.497372	3.684009	-0.41	0.684	-8.719797	5.725054
__00008C	.2927278	2.177494	0.13	0.893	-3.976205	4.561661
__00008F	-10.63528	23.58523	-0.45	0.652	-56.87365	35.6031
__00008H	3.123759	14.11997	0.22	0.825	-24.55816	30.80567
__00008K	-.1941854	.6776997	-0.29	0.774	-1.522802	1.134431
__00008M	-1.39456	.6534375	-2.13	0.033	-2.675611	-.1135085
__00008P	-1.343119	1.111617	-1.21	0.227	-3.522422	.8361837
__00008R	-1.271986	.9151353	-1.39	0.165	-3.066091	.5221185
__00008U	-.7730286	.4412188	-1.75	0.080	-1.638029	.091972
__00008W	.2278244	.3407778	0.67	0.504	-.4402638	.8959125
__00008Z	.601064	.2397331	2.51	0.012	.1310721	1.071056

__000091	-.0640346	.2041853	-0.31	0.754	-.4643358	.3362665
__000094	-.7268351	.2604519	-2.79	0.005	-1.237446	-.2162243
__000096	-.228038	.2082336	-1.10	0.274	-.6362758	.1801998
__000099	.7647842	.3113365	2.46	0.014	.1544152	1.375153
__00009B	.1977191	.261654	0.76	0.450	-.3152483	.7106865
__00009E	-.0660116	.3747056	-0.18	0.860	-.8006145	.6685913
__00009G	-.243422	.302943	-0.80	0.422	-.8373358	.3504917
xbarlag	-4.769469	22.18503	-0.21	0.830	-48.26277	38.72383
x2barlag	-.4538931	.5918067	-0.77	0.443	-1.614118	.7063322
xbar2lag	1.725566	4.421587	0.39	0.696	-6.942867	10.394
rootxbarlag	5.399595	27.44809	0.20	0.844	-48.41183	59.21102
z1barlag	1.81121	1.002554	1.81	0.071	-.1542766	3.776696
z2barlag	2.690187	1.511498	1.78	0.075	-.2730747	5.653448
z3barlag	.552162	.563691	0.98	0.327	-.5529429	1.657267
z4barlag	-.2955395	.2702011	-1.09	0.274	-.8252634	.2341844
z5barlag	1.024167	.2637745	3.88	0.000	.5070427	1.541292
z6barlag	-.9630149	.4686881	-2.05	0.040	-1.881869	-.0441613
z7barlag	.2824898	.5031912	0.56	0.575	-.7040064	1.268986
_cons	-3.250004	9.429703	-0.34	0.730	-21.73675	15.23674

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A2	128,640	.4438109	.13429	.0976942	1.111178

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"

957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
__00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9

960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
964 }

965
966 local r_RE      "`r_RE' `qhat_hats'"

967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
```

```
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }

978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                 g pzi`j' = p1*zi`j'
    3.          }
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
```

Source	SS	df	MS		
Model	**5976.12589**	**16**	**373.507868**	Number of obs	= **128,640**
Residual	**981.927241**	**128,623**	**.00763415**	F(16, 128623)	= **48925.93**
				Prob > F	= **0.0000**
Total	**6958.05313**	**128,639**	**.054089764**	R-squared	= **0.8589**
				Adj R-squared	= **0.8589**
				Root MSE	= **.08737**

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
xi2	**-.0374445**	**.0005646**	**-66.33**	**0.000**	**-.038551**	**-.0363379**
ybar_q1_2	**-.1066234**	**.0092279**	**-11.55**	**0.000**	**-.1247099**	**-.0885368**
ybar_q2_2	**.1108799**	**.0024778**	**44.75**	**0.000**	**.1060235**	**.1157364**
ybarx_1	**.236722**	**.0034032**	**69.56**	**0.000**	**.2300518**	**.2433922**
ybarx_2	**-.1057571**	**.0018199**	**-58.11**	**0.000**	**-.1093241**	**-.1021902**
ybar_q1_q2	**-.1052302**	**.008448**	**-12.46**	**0.000**	**-.121788**	**-.0886723**
xi	**.3630039**	**.0018141**	**200.11**	**0.000**	**.3594484**	**.3665594**
ybar_q1	**.2163611**	**.0076981**	**28.11**	**0.000**	**.2012729**	**.2314493**
ybar_q2	**-.1332122**	**.0038871**	**-34.27**	**0.000**	**-.1408309**	**-.1255936**
pzi1	**.2218601**	**.0012929**	**171.59**	**0.000**	**.2193259**	**.2243942**
pzi2	**.0173906**	**.0021069**	**8.25**	**0.000**	**.0132612**	**.0215201**
pzi3	**-.002822**	**.0006512**	**-4.33**	**0.000**	**-.0040983**	**-.0015456**
pzi4	**.0085446**	**.000649**	**13.17**	**0.000**	**.0072725**	**.0098166**
pzi5	**-.0138749**	**.0006967**	**-19.91**	**0.000**	**-.0152405**	**-.0125093**
pzi6	**.0026067**	**.0005751**	**4.53**	**0.000**	**.0014796**	**.0037338**
pzi7	**-.0041551**	**.000745**	**-5.58**	**0.000**	**-.0056152**	**-.0026949**
_cons	**-.0081675**	**.0020377**	**-4.01**	**0.000**	**-.0121614**	**-.0041736**

```
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = _b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                  local coef = _b[pzi`j']
    3.                  local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1025}

1026
1027if `J'==3 & simple_model==0
  >                             local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                             local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00131158
  Iteration 1:  GMM criterion Q(b) =  .00069554
  Iteration 2:  GMM criterion Q(b) =   .0001106
  Iteration 3:  GMM criterion Q(b) =  .00008084

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00574197
  Iteration 1:  GMM criterion Q(b) =  .00490383
  Iteration 2:  GMM criterion Q(b) =  .00405535
  Iteration 3:  GMM criterion Q(b) =  .00399976
  Iteration 4:  GMM criterion Q(b) =  .00395707

  GMM estimation

  Number of parameters =  23
  Number of moments    =  28
  Initial weight matrix: Unadjusted                 Number of obs   =    128,640
  GMM weight matrix:     Robust
```

(Std. err. adjusted for **564** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
A11						
one	**.4111038**	**.1706418**	**2.41**	**0.016**	**.076652**	**.7455557**
A12						
one	**-.0945049**	**.1017405**	**-0.93**	**0.353**	**-.2939125**	**.1049027**
A21						
one	**-.3974851**	**.2748697**	**-1.45**	**0.148**	**-.9362197**	**.1412496**
A22						
one	**.4524226**	**.1710336**	**2.65**	**0.008**	**.1172028**	**.7876424**
C1						
zi1	**.515034**	**.0549295**	**9.38**	**0.000**	**.4073742**	**.6226938**
zi2	**-.1915335**	**.0812917**	**-2.36**	**0.018**	**-.3508623**	**-.0322047**
zi3	**-.0261924**	**.0226472**	**-1.16**	**0.247**	**-.07058**	**.0181953**
zi4	**.0618551**	**.0306593**	**2.02**	**0.044**	**.0017639**	**.1219463**
zi5	**.0371967**	**.0242122**	**1.54**	**0.124**	**-.0102583**	**.0846518**
zi6	**.0180741**	**.019833**	**0.91**	**0.362**	**-.0207978**	**.056946**
zi7	**.0145687**	**.0253287**	**0.58**	**0.565**	**-.0350746**	**.064212**
C2						
zi1	**.3294254**	**.0871532**	**3.78**	**0.000**	**.1586082**	**.5002426**
zi2	**-.3644719**	**.1323404**	**-2.75**	**0.006**	**-.6238543**	**-.1050894**
zi3	**-.0394469**	**.0352659**	**-1.12**	**0.263**	**-.1085667**	**.029673**
zi4	**.1048784**	**.052928**	**1.98**	**0.048**	**.0011415**	**.2086153**
zi5	**.0776347**	**.0315229**	**2.46**	**0.014**	**.015851**	**.1394183**
zi6	**.0207083**	**.029813**	**0.69**	**0.487**	**-.0377241**	**.0791407**
zi7	**.0246201**	**.041906**	**0.59**	**0.557**	**-.0575141**	**.1067543**
/d1	**-.0698487**	**.0080853**	**-8.64**	**0.000**	**-.0856956**	**-.0540017**
/b1	**.4403671**	**.0234757**	**18.76**	**0.000**	**.3943557**	**.4863786**
/AVA11	**2.146432**	**.9492138**	**2.26**	**0.024**	**.2860072**	**4.006857**
/AVA12	**-1.646061**	**.8751478**	**-1.88**	**0.060**	**-3.36132**	**.0691969**
/AVA22	**.3783027**	**.9293939**	**0.41**	**0.684**	**-1.443276**	**2.199881**

Instruments for equation **eq1_RE: __00009J __00009L __00009O xi __000002 __00000E __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5 __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons**

```
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
```

```
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Afull_fsuXsegXreligXsched_drop3_all_exactIns
      > t_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not found)
  (output written to diag_2good_norm_main_nopcross_state_Afull_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  24757

1077unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640
```

```
1078estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4599

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  5.3831268

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Afull_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster not found)
  file diag_2good_norm_main_nopcross_state_Afull_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
```

```
   13.                         local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                         local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                         forval l=1/`J' {
   16.                                 tempvar zdiff`t'p`j'p`l'
   17.                                 g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                 local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                 foreach name of global Alist {
   20.                                         tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                         g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*`p`l'qikhat`l''*`name'
   22.                                         local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                 }
   24.                                 if `Tindivp1'<=`T' {
   25.                                         forval s=`Tindivp1'/`T' {
   26.                                                 tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                 * don't generate these to save me
  > mory!
1119                                                 gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                 local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                         }
   30.                                 }
   31.                         }
   32.                 }
   33.         }
1120        if `Tindivp1'<=`T' {
1121                forval t=`Tindivp1'/`T' {
    2.                         tempvar zi`t'xdiff
    3.                         g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                         local zg "`zg' zi`t'"
    5.                         forval j=1/`J' {
    6.                                 tempvar zi`t'p`j'xdiff
    7.                                 g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                 local pzg "`pzg' `zi`t'p`j''"
    9.                                 local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                         }
   11.                 }
1122        }
1123
1124        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1125        // denoting zg as group-zs
1126        //  zg,pzgx,rprp,rprpx
1127        local rootprootp ""
1128        local rootprootpxdiff ""
1129
1130        forval j=1/`J' {
    2.                 forval s=`j'/`J' {
    3.                         tempvar xdiffrp`j'rp`s'
    4.                         g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1131                        if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1132                        local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                 }
    8.         }
```

```
1133
1134        local pqhatxdiff ""
1135        forval j=1/`J' {
    2.                  forval s=1/`J' {
    3.                          foreach name of global Alist {
    4.                                  tempvar   p`s'qikhat`j'xdiff`name'
    5.                                  g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
1136                                    local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
1137
1138        // set instruments
1139        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1140}

1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                          local `++t'
    3.                          local p_nice = subinstr("`p'",":_cons","",.)
    4.                          local est = coefs[1,`t']
    5.                          if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                  }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
```

```
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                  }
   16.         }
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1    `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
```

```
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00058897
  Iteration 1:  GMM criterion Q(b) =  .00018738
  Iteration 2:  GMM criterion Q(b) =  .00015359

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00696504
  Iteration 1:  GMM criterion Q(b) =  .00644637
  Iteration 2:  GMM criterion Q(b) =  .00627527
  Iteration 3:  GMM criterion Q(b) =  .00625857
  Iteration 4:  GMM criterion Q(b) =  .00625347

  GMM estimation
```

Number of parameters = **20**
Number of moments = **63**
Initial weight matrix: **Unadjusted** Number of obs = **128,640**
GMM weight matrix: **Robust**

(Std. err. adjusted for **564** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11						
_cons	**.7360898**	**.0347387**	**21.19**	**0.000**	**.6680031**	**.8041764**
C12						
_cons	**-.0820169**	**.0556717**	**-1.47**	**0.141**	**-.1911314**	**.0270976**
C13						
_cons	**-.0143318**	**.0153995**	**-0.93**	**0.352**	**-.0445143**	**.0158507**
C14						
_cons	**-.0188115**	**.012734**	**-1.48**	**0.140**	**-.0437698**	**.0061467**
C15						
_cons	**.0184477**	**.0342141**	**0.54**	**0.590**	**-.0486107**	**.0855061**
C16						
_cons	**-.069564**	**.0242593**	**-2.87**	**0.004**	**-.1171114**	**-.0220166**
C17						
_cons	**-.0735083**	**.0270067**	**-2.72**	**0.006**	**-.1264405**	**-.020576**
C21						
_cons	**.6219649**	**.0498514**	**12.48**	**0.000**	**.5242579**	**.7196719**
C22						
_cons	**-.1607215**	**.0814032**	**-1.97**	**0.048**	**-.3202688**	**-.0011741**
C23						
_cons	**-.0194283**	**.0212791**	**-0.91**	**0.361**	**-.0611347**	**.022278**
C24						
_cons	**-.0491318**	**.0225701**	**-2.18**	**0.029**	**-.0933683**	**-.0048953**
C25						
_cons	**.0446603**	**.0417255**	**1.07**	**0.284**	**-.0371202**	**.1264409**
C26						
_cons	**-.093816**	**.0314606**	**-2.98**	**0.003**	**-.1554777**	**-.0321543**
C27						
_cons	**-.0987043**	**.038497**	**-2.56**	**0.010**	**-.174157**	**-.0232517**
d1						
_cons	**-.078649**	**.0063264**	**-12.43**	**0.000**	**-.0910485**	**-.0662496**
b1						
_cons	**.4057761**	**.0189585**	**21.40**	**0.000**	**.3686181**	**.4429341**
A11						
one	**9.740779**	**2.06629**	**4.71**	**0.000**	**5.690926**	**13.79063**
A12						
one	**-6.382982**	**1.860039**	**-3.43**	**0.001**	**-10.02859**	**-2.737372**
A21						
one	**-7.694638**	**1.827703**	**-4.21**	**0.000**	**-11.27687**	**-4.112407**
A22						
one	**5.399721**	**1.577499**	**3.42**	**0.001**	**2.307881**	**8.491562**

Instruments for equation **eq1_FE**: **__000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T**

```
      __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ
      __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD
      __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7
      __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1
      __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA
      __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY
      __0000D5 _cons
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Afull_fsuXsegXreligXsched_drop3_all_exactIns
      > t_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex not found)
  (output written to diag_2good_norm_main_nopcross_state_Afull_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex)
1242
1243        estat overid

    Test of overidentifying restriction:

    Hansen's J chi2(43) = 804.447 (p = 0.0000)
1244        di "$S_DATE $S_TIME"
   7 Dec 2023 21:35:44
1245        unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  24757
1248        unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640
1249        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4599
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  5.3831268
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Afull_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster not found)
  file diag_2good_norm_main_nopcross_state_Afull_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
```

```
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}

1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table6_14.smcl
   log type:  smcl
  closed on:   7 Dec 2023, 21:35:44
```

A.6.2 Columns 2 and 5

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table6_25.smcl
  log type:  smcl
 opened on:   7 Dec 2023, 21:38:39

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
  C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""
35. if $rand_grp==1 local grp_type "_Random"
36.
37. // make summary stats table
38. local make_sumstats = 0
39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0
42. local typeExp ""
43. if $meas_pc==1 local typeExp "_ExpPC"
44.
45. // number of goods
46. scalar J=2
47. local J = J
48.
49. // group def options
50. local group_def "dist"
51. local group_def "fsuXseg"
52. local group_def "fsuXsegXreligXsched"
53.
54. // comparision group
55. local groupComp_def ""
56. local desc_compGrp ""
57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"
58.
59. // load data
60. include "$CODE/prep demand system data.do"
61. //clear all
62. pause on
63. set more off
64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.
65.
66. scalar short_zlist=$short_zlist_val
67.
68. cap prog drop bysmeanw
69. prog def bysmeanw
    1.         syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
70.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                  Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations                 840   ->   280
  Number of variables                      5   ->   6
  j variable (3 values)          expend_type   ->   (dropped)
  xij variables:
                                laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.             }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,637 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
```

```
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }
297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }
316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }
333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
```

```
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (82 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,576 real changes made, 82 to missing)

460 replace x2=x*x
  (73,576 real changes made, 82 to missing)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,576 real changes made)

478 replace x2=x2_temp
  (73,576 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (48,819 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)
```

```
488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_41e4_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
```

```
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,757 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
549 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var'" _n
    5.                 }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
```

```
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=0

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1
```

```
625 scalar estimate_FE=1

626
627 // estimation options
628 global w_initial "winit(unadjusted)"

629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

630 global wmatrix "wmatrix(robust)"

631 global trace_level ""

632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

633 global technique ""

634 global maxiter ""

635 global technique_RE ""

636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

637 global derivatives "quickderivatives"

638 scalar init_FE_from_RE=1

639 global clust_var "state_district_round"

640 local clus_name "_clus_Dist"

641
642
643
644
645
646
647
648
649 // xxzx to keep names same
650 local wmatrix_desc ""

651 local sectorName "_urbOnly"

652 local dropZdesc "_dropPre"

653 local desc_struc_v0 ""

654 local inst_desc "_exactInst"

655 local inst_rootp "_noPPXdInst"

656 local weight_desc ""

657 local RE_sq_inst_desc ""

658 local cnstname ""

659 local actual_pqhat_desc ""

660 local expUpname "_updateExp"
```

```
661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //              ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T

694 local Tm1=`T'-1
```

```
695 local Tindiv : word count $zlist_indiv

696
697 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
706
707 if `J'==2 {
708          replace p1=P1
  (128,640 real changes made)
709          replace p2=P2
  (128,640 real changes made)
710          replace qi1=Qi1
  (128,632 real changes made)
711          replace qk1=Qk1
  (128,632 real changes made)
712          replace qikbar1=Qikbar1
  (128,640 real changes made)
713          replace qi2=Qi2
  (128,640 real changes made)
714          replace qk2=Qk2
  (128,640 real changes made)
715          replace qikbar2=Qikbar2
  (128,640 real changes made)
716
717          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721          forv j = 1(1)3 {
    2.                replace p`j'=P`j'
    3.                replace qi`j'=Qi`j'
    4.                replace qk`j'=Qk`j'
    5.                replace qikbar`j'=Qikbar`j'
    6.          }
722
723          foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                qui replace `var'=.
    3.          }
724 }

725
```

```
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }
```

```
751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
773         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j'"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.          local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.          if `jplus1'<=`J' {
    5.                  forval k=`jplus1'/`J' {
    6.                          local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                  }
    8.          }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                  if noMeasError==1                               local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
```

```
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                      l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                    ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A22:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A22:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2
  > ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qikba
  > r1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d
  > 1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > }*p1*qikbar1) + ({A22:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))        ))
```

```
836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qk1) + ({A22:one}*p2*qk2) )*( ({A11:}*p1*qikbar1
  > ) + ({A22:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2
  > *{C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A22:}*p2*q
  > ikbar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{
  > C2:}) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2
  > *{C2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) ))*{b1} + ({A11:}*p1
  > *qikbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p
  > 2 )*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))*({b1} - 2*(( ({A11:
  > one}*p1*qikbar1) + ({A22:one}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}
  > *zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2
  > + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))
  >   ))

874
875 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi

882 g double `ximxk'=xi-xk
```

```
883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.          tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.          g double `xizi`t''=xi*zi`t'
    4.          g double `xkzk`t''=xk*zk`t'
    5.          g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.          g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.          g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.          g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.          g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.          forval j=1/`J' {
   11.                  tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                  g double `zi`t'p`j''= zi`t'*p`j'
   13.                  g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                  g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                  g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                  forv s=1/`T' {
   17.                          tempvar zi`t'zi`s'p`j'
   18.                          g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                  }
   20.          }
   21. }

891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }

903
904 local zrg ""

905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var''"
    6.          }
    7. }
```

```
906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var''"
    5.
911          forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var''"
   10.                  local xprg "`xprg' `xp`j'`var''"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""

925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }
```

```
929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.         forval k=`j'/`J' {
    3.                 tempvar p`j'p`k'
    4.                 g `p`j'p`k''=p`j'*p`k'
    5.                 local roundinsts "`roundinsts' `p`j'p`k''"
    6.         }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.         if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.         if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }

  Linear regression                               Number of obs     =    128,640
                                                  F(128, 4598)      =          .
                                                  Prob > F          =          .
                                                  R-squared         =     0.2238
                                                  Root MSE          =       .131
```

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	-12.79512	7.260343	-1.76	0.078	-27.02887	1.438643
__00009L	-3.637138	1.251887	-2.91	0.004	-6.091436	-1.182839
__00009O	1.004898	3.87315	0.26	0.795	-6.588336	8.598132
xi	.1090911	.0077876	14.01	0.000	.0938237	.1243585
__000002	-.0157421	.0025123	-6.27	0.000	-.0206674	-.0108169
__00000E	-.1156487	.0903327	-1.28	0.201	-.2927442	.0614468
__00000P	-.0139978	.0521173	-0.27	0.788	-.1161727	.0881771
__000017	.0509528	.1735472	0.29	0.769	-.289283	.3911886
__00001I	-.0239111	.1060934	-0.23	0.822	-.2319051	.184083
__000020	.0171582	.04793	0.36	0.720	-.0768076	.111124
__00002B	-.025461	.0245826	-1.04	0.300	-.0736547	.0227327
__00002T	.047008	.0663415	0.71	0.479	-.0830531	.1770692
__000034	.0018229	.0474534	0.04	0.969	-.0912086	.0948543
__00003M	.0130708	.0588349	0.22	0.824	-.1022739	.1284156
__00003X	.0098183	.0379708	0.26	0.796	-.0646228	.0842594
__00004F	-.025421	.0466786	-0.54	0.586	-.1169335	.0660916
__00004Q	.0074369	.0250442	0.30	0.767	-.0416617	.0565356
__000058	-.0217155	.0609657	-0.36	0.722	-.1412376	.0978066
__00005J	.0055081	.0340618	0.16	0.872	-.0612694	.0722856
__00005U	.0797255	.4007228	0.20	0.842	-.7058835	.8653344
__00005V	-.0346695	.0711757	-0.49	0.626	-.174208	.1048689
__00005W	-.0306542	.1635037	-0.19	0.851	-.3512	.2898916
__00005X	.1385994	.4167248	0.33	0.739	-.6783813	.9555801
__00005Y	-.0224787	.1028475	-0.22	0.827	-.2241091	.1791517
__00005Z	.0485407	.1674104	0.29	0.772	-.279664	.3767454
__000060	-.0305375	.0696472	-0.44	0.661	-.1670794	.1060045
__000061	.0243364	.0499235	0.49	0.626	-.0735376	.1222104
__000062	.0703686	.0341366	2.06	0.039	.0034444	.1372928
__000063	-.0376336	.047996	-0.78	0.433	-.1317289	.0564616
__000064	-.0701443	.057982	-1.21	0.226	-.1838168	.0435282
__000065	-.1665861	.877927	-0.19	0.850	-1.887744	1.554572
__000066	.2283964	.1442115	1.58	0.113	-.0543274	.5111203
__000067	-.3199538	.3445552	-0.93	0.353	-.9954474	.3555397
__000068	.5394784	.8576131	0.63	0.529	-1.141855	2.220812
__000069	-.1708187	.1878736	-0.91	0.363	-.5391411	.1975038
__00006A	-.424151	.3283207	-1.29	0.196	-1.067817	.2195151
__00006B	-.0463852	.1238777	-0.37	0.708	-.289245	.1964745
__00006C	-.2113415	.1076302	-1.96	0.050	-.4223483	-.0003347
__00006D	.0467054	.0589613	0.79	0.428	-.068887	.1622978
__00006E	-.1444075	.0994178	-1.45	0.146	-.339314	.0504991
__00006F	.030443	.1210695	0.25	0.801	-.2069113	.2677973
__00006G	-.1629455	.1849126	-0.88	0.378	-.5254629	.199572
__00006H	.0332459	.032772	1.01	0.310	-.031003	.0974948
__00006I	-.0177702	.0723866	-0.25	0.806	-.1596827	.1241422
__00006J	.1731219	.2093027	0.83	0.408	-.2372119	.5834557
__00006K	.0504318	.0524078	0.96	0.336	-.0523128	.1531763
__00006L	-.0507186	.0851838	-0.60	0.552	-.2177197	.1162825
__00006M	-.0292887	.0358816	-0.82	0.414	-.0996338	.0410563
__00006N	.0309542	.0234174	1.32	0.186	-.0149551	.0768636
__00006O	-.014908	.0145441	-1.03	0.305	-.0434214	.0136054
__00006P	-.0290572	.0253252	-1.15	0.251	-.0787067	.0205924
__00006Q	.0028381	.030826	0.09	0.927	-.0575957	.0632719
__00006R	-.6012457	.495913	-1.21	0.225	-1.573473	.3709819
__00006S	-.0369337	.0565668	-0.65	0.514	-.1478319	.0739644
__00006T	.2871989	.1903878	1.51	0.131	-.0860527	.6604504
__00006U	.3847497	.4323705	0.89	0.374	-.4629041	1.232403
__00006V	-.1362284	.0767153	-1.78	0.076	-.2866273	.0141705
__00006W	-.0523872	.1430883	-0.37	0.714	-.332909	.2281346
__00006X	.0125271	.0552197	0.23	0.821	-.0957301	.1207843
__00006Y	.0345059	.0349869	0.99	0.324	-.0340851	.103097
__00006Z	-.0725883	.0241739	-3.00	0.003	-.1199808	-.0251958
__000070	-.0109506	.0430715	-0.25	0.799	-.0953914	.0734902
__000071	-.0582328	.0494466	-1.18	0.239	-.1551719	.0387062
__000072	-.4534528	.305937	-1.48	0.138	-1.053236	.1463307
__000073	-.0906118	.0458174	-1.98	0.048	-.180436	-.0007877
__000074	.2961195	.1206212	2.45	0.014	.0596441	.5325949
__000075	.2853582	.2940211	0.97	0.332	-.2910644	.8617807

__000076	-.076137	.0648958	-1.17	0.241	-.2033639	.0510898
__000077	.0084989	.1095353	0.08	0.938	-.2062429	.2232407
__000078	-.0029301	.0439453	-0.07	0.947	-.0890839	.0832237
__000079	-.0005896	.0346299	-0.02	0.986	-.0684808	.0673017
__00007A	.0293777	.0211752	1.39	0.165	-.0121359	.0708912
__00007B	-.0056913	.0314004	-0.18	0.856	-.0672512	.0558686
__00007C	-.0796128	.0372845	-2.14	0.033	-.1527083	-.0065174
__00007D	.1277082	.2134185	0.60	0.550	-.2906944	.5461109
__00007E	.0131872	.0355408	0.37	0.711	-.0564899	.0828643
__00007F	-.0882595	.083501	-1.06	0.291	-.2519616	.0754427
__00007G	.006318	.2231374	0.03	0.977	-.4311384	.4437744
__00007H	-.0032218	.0521663	-0.06	0.951	-.1054929	.0990492
__00007I	-.0735209	.0881929	-0.83	0.405	-.2464212	.0993795
__00007J	-.0161166	.0327932	-0.49	0.623	-.0804071	.0481739
__00007K	-.0226722	.0278317	-0.81	0.415	-.0772358	.0318913
__00007L	.0113972	.0160408	0.71	0.477	-.0200505	.0428448
__00007M	-.0074927	.0281454	-0.27	0.790	-.0626712	.0476857
__00007N	.0512257	.0317663	1.61	0.107	-.0110516	.113503
__00007O	-.0216683	.3475372	-0.06	0.950	-.7030081	.6596714
__00007P	.0879331	.0479654	1.83	0.067	-.0061021	.1819684
__00007Q	-.123569	.1328924	-0.93	0.353	-.3841018	.1369638
__00007R	.1056987	.3234608	0.33	0.744	-.5284398	.7398371
__00007S	.0276477	.0679204	0.41	0.684	-.1055088	.1608043
__00007T	-.2030032	.1161087	-1.75	0.080	-.430632	.0246256
__00007U	.0181395	.0416299	0.44	0.663	-.063475	.0997541
__00007V	-.0771625	.0330746	-2.33	0.020	-.1420047	-.0123204
__00007W	-.0141401	.0196985	-0.72	0.473	-.0527587	.0244784
__00007X	.0035144	.0352185	0.10	0.921	-.0655307	.0725594
__00007Y	.0380758	.0423369	0.90	0.369	-.0449249	.1210764
__000080	-36.81121	17.02884	-2.16	0.031	-70.19591	-3.426515
__000082	.0760368	8.506328	0.01	0.993	-16.60045	16.75252
__000085	-.6294823	.4059554	-1.55	0.121	-1.42535	.1663852
__000087	.4964106	.2690436	1.85	0.065	-.0310441	1.023865
__00008A	8.027308	3.333333	2.41	0.016	1.492376	14.56224
__00008C	-1.201791	1.594738	-0.75	0.451	-4.328242	1.924661
__00008F	44.59295	20.99826	2.12	0.034	3.426287	85.75961
__00008H	2.168832	10.94408	0.20	0.843	-19.28682	23.62448
__00008K	-.4921474	.5029834	-0.98	0.328	-1.478236	.4939415
__00008M	-.8762901	.4582283	-1.91	0.056	-1.774638	.0220574
__00008P	-1.190171	.9196827	-1.29	0.196	-2.99319	.612849
__00008R	-1.132574	.642393	-1.76	0.078	-2.391973	.1268243
__00008U	.1009246	.3299369	0.31	0.760	-.5459101	.7477593
__00008W	.1590723	.2320867	0.69	0.493	-.295929	.6140737
__00008Z	.7462393	.2282715	3.27	0.001	.2987175	1.193761
__000091	-.4374243	.1583699	-2.76	0.006	-.7479053	-.1269433
__000094	.0122267	.1794758	0.07	0.946	-.339632	.3640855
__000096	.1116296	.1418125	0.79	0.431	-.1663908	.3896501
__000099	-.0594555	.2718099	-0.22	0.827	-.5923333	.4734223
__00009B	-.0418895	.1972203	-0.21	0.832	-.4285359	.3447569
__00009E	.0027748	.3171447	0.01	0.993	-.618981	.6245306
__00009G	-.4676324	.2253747	-2.07	0.038	-.9094749	-.0257898
xbarlag	38.11984	21.35658	1.78	0.074	-3.749306	79.98898
x2barlag	-.1896805	.4397159	-0.43	0.666	-1.051735	.6723737
xbar2lag	-6.519805	4.180346	-1.56	0.119	-14.71529	1.675681
rootxbarlag	-48.72697	26.43458	-1.84	0.065	-100.5514	3.097502
z1barlag	1.544271	.7682996	2.01	0.044	.0380348	3.050507
z2barlag	2.57067	1.188866	2.16	0.031	.2399211	4.901418
z3barlag	-.2976042	.4371373	-0.68	0.496	-1.154603	.5593949
z4barlag	-.1807586	.2681451	-0.67	0.500	-.7064517	.3449345
z5barlag	-.1795307	.1890717	-0.95	0.342	-.5502021	.1911407
z6barlag	.1391592	.3756895	0.37	0.711	-.5973725	.8756909
z7barlag	.430906	.4037833	1.07	0.286	-.3607032	1.222515
_cons	16.33104	9.106072	1.79	0.073	-1.521234	34.18331

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	128,640	.4424145	.0703047	.2011188	.8127233

Linear regression

Number of obs	= **128,640**
F(128, 4598)	= .
Prob > F	= .
R-squared	= **0.3053**
Root MSE	= **.20268**

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**4.564747**	**8.325433**	**0.55**	**0.584**	**-11.7571**	**20.88659**
__00009L	**.5937963**	**2.171845**	**0.27**	**0.785**	**-3.664063**	**4.851655**
__00009O	**-.8608651**	**4.900914**	**-0.18**	**0.861**	**-10.46901**	**8.747278**
xi	**.1450715**	**.0142385**	**10.19**	**0.000**	**.1171571**	**.1729859**
__000002	**-.0081835**	**.0053966**	**-1.52**	**0.129**	**-.0187634**	**.0023964**
__00000E	**-.1996901**	**.1318225**	**-1.51**	**0.130**	**-.4581254**	**.0587452**
__00000P	**-.1306909**	**.0773583**	**-1.69**	**0.091**	**-.2823502**	**.0209685**
__000017	**.3219988**	**.2747868**	**1.17**	**0.241**	**-.2167153**	**.8607128**
__00001I	**.2451433**	**.1716831**	**1.43**	**0.153**	**-.091438**	**.5817245**
__000020	**.0465067**	**.0713123**	**0.65**	**0.514**	**-.0932996**	**.186313**
__00002B	**.0499324**	**.0392375**	**1.27**	**0.203**	**-.026992**	**.1268568**
__00002T	**.3317416**	**.0880092**	**3.77**	**0.000**	**.1592013**	**.5042819**
__000034	**-.0424598**	**.0611807**	**-0.69**	**0.488**	**-.1624034**	**.0774838**
__00003M	**-.0824665**	**.0821755**	**-1.00**	**0.316**	**-.2435698**	**.0786368**
__00003X	**.0562813**	**.0513967**	**1.10**	**0.274**	**-.0444809**	**.1570436**
__00004F	**-.0214064**	**.0649749**	**-0.33**	**0.742**	**-.1487884**	**.1059755**
__00004Q	**-.0033885**	**.0382735**	**-0.09**	**0.929**	**-.0784229**	**.0716459**
__000058	**-.0616114**	**.095554**	**-0.64**	**0.519**	**-.248943**	**.1257203**
__00005J	**.0962271**	**.0576901**	**1.67**	**0.095**	**-.0168731**	**.2093274**
__00005U	**-1.072842**	**.6175735**	**-1.74**	**0.082**	**-2.283583**	**.1378984**
__00005V	**-.0686079**	**.112662**	**-0.61**	**0.543**	**-.2894794**	**.1522637**
__00005W	**.2956557**	**.2555724**	**1.16**	**0.247**	**-.2053889**	**.7967004**
__00005X	**1.107247**	**.6463172**	**1.71**	**0.087**	**-.1598453**	**2.374339**
__00005Y	**.006665**	**.1525531**	**0.04**	**0.965**	**-.2924124**	**.3057424**
__00005Z	**.4177337**	**.2272567**	**1.84**	**0.066**	**-.0277986**	**.863266**
__000060	**-.2003814**	**.1148141**	**-1.75**	**0.081**	**-.4254721**	**.0247093**
__000061	**.0627776**	**.0573305**	**1.10**	**0.274**	**-.0496177**	**.1751728**
__000062	**-.0079371**	**.0479**	**-0.17**	**0.868**	**-.101844**	**.0859698**
__000063	**.0234324**	**.0606999**	**0.39**	**0.699**	**-.0955686**	**.1424334**
__000064	**-.0476537**	**.0743598**	**-0.64**	**0.522**	**-.1934346**	**.0981272**
__000065	**4.601567**	**1.273473**	**3.61**	**0.000**	**2.104948**	**7.098186**
__000066	**.5353632**	**.231593**	**2.31**	**0.021**	**.0813297**	**.9893968**
__000067	**-2.104046**	**.5308536**	**-3.96**	**0.000**	**-3.144774**	**-1.063318**
__000068	**-3.209865**	**1.309425**	**-2.45**	**0.014**	**-5.776967**	**-.6427636**
__000069	**-.5733963**	**.2998063**	**-1.91**	**0.056**	**-1.16116**	**.0143679**
__00006A	**-.3073944**	**.5099257**	**-0.60**	**0.547**	**-1.307093**	**.6923047**
__00006B	**-.0013787**	**.1805182**	**-0.01**	**0.994**	**-.3552811**	**.3525236**
__00006C	**-.6917272**	**.1201482**	**-5.76**	**0.000**	**-.9272752**	**-.4561791**
__00006D	**.0978464**	**.0930479**	**1.05**	**0.293**	**-.0845721**	**.280265**
__00006E	**-.1126402**	**.1459926**	**-0.77**	**0.440**	**-.3988558**	**.1735754**
__00006F	**.0293601**	**.1813666**	**0.16**	**0.871**	**-.3262055**	**.3849256**
__00006G	**.228589**	**.3106349**	**0.74**	**0.462**	**-.3804045**	**.8375825**
__00006H	**.0378065**	**.0490674**	**0.77**	**0.441**	**-.0583892**	**.1340023**
__00006I	**-.1229966**	**.1220438**	**-1.01**	**0.314**	**-.362261**	**.1162679**
__00006J	**-.2612815**	**.3194111**	**-0.82**	**0.413**	**-.8874806**	**.3649176**
__00006K	**.042626**	**.077334**	**0.55**	**0.582**	**-.1089857**	**.1942377**
__00006L	**-.0857245**	**.1255653**	**-0.68**	**0.495**	**-.3318928**	**.1604438**
__00006M	**.0143417**	**.0615696**	**0.23**	**0.816**	**-.1063642**	**.1350476**
__00006N	**.0142187**	**.0300003**	**0.47**	**0.636**	**-.0445962**	**.0730337**
__00006O	**.0108951**	**.0238843**	**0.46**	**0.648**	**-.0359295**	**.0577198**
__00006P	**-.051378**	**.0341179**	**-1.51**	**0.132**	**-.1182654**	**.0155094**
__00006Q	**.0061354**	**.0401718**	**0.15**	**0.879**	**-.0726205**	**.0848914**
__00006R	**-.3954866**	**.5294751**	**-0.75**	**0.455**	**-1.433512**	**.6425387**
__00006S	**.0027485**	**.070072**	**0.04**	**0.969**	**-.1346264**	**.1401233**
__00006T	**.2422518**	**.1891001**	**1.28**	**0.200**	**-.1284752**	**.6129788**
__00006U	**-.2749271**	**.4753902**	**-0.58**	**0.563**	**-1.20692**	**.657066**
__00006V	**.1610918**	**.0976758**	**1.65**	**0.099**	**-.0303997**	**.3525833**
__00006W	**.1256103**	**.177919**	**0.71**	**0.480**	**-.2231964**	**.474417**
__00006X	**.0319531**	**.0688723**	**0.46**	**0.643**	**-.1030697**	**.1669758**
__00006Y	**.0856173**	**.0433777**	**1.97**	**0.048**	**.0005762**	**.1706584**
__00006Z	**-.1346033**	**.0315098**	**-4.27**	**0.000**	**-.1963778**	**-.0728289**

__000070	-.0896637	.0497818	-1.80	0.072	-.18726	.0079325
__000071	-.0242239	.0620782	-0.39	0.696	-.145927	.0974792
__000072	-.817985	.3639092	-2.25	0.025	-1.531422	-.1045482
__000073	-.1058926	.0632791	-1.67	0.094	-.22995	.0181648
__000074	.4129247	.154788	2.67	0.008	.109466	.7163835
__000075	.4604809	.366705	1.26	0.209	-.258437	1.179399
__000076	.1273306	.0859626	1.48	0.139	-.0411973	.2958585
__000077	.1296378	.1391035	0.93	0.351	-.1430718	.4023475
__000078	-.0674194	.0576484	-1.17	0.242	-.1804379	.0455991
__000079	.0872997	.0407137	2.14	0.032	.0074813	.167118
__00007A	.0612698	.02713	2.26	0.024	.008082	.1144575
__00007B	.0328333	.0413449	0.79	0.427	-.0482225	.1138891
__00007C	.0255229	.0482849	0.53	0.597	-.0691387	.1201845
__00007D	.354494	.2770601	1.28	0.201	-.1886768	.8976647
__00007E	.0890569	.0489815	1.82	0.069	-.0069704	.1850841
__00007F	-.2285898	.1151561	-1.99	0.047	-.4543511	-.0028285
__00007G	-.1664105	.2810873	-0.59	0.554	-.7174767	.3846556
__00007H	-.0569692	.0661062	-0.86	0.389	-.186569	.0726307
__00007I	-.0611722	.1122723	-0.54	0.586	-.2812799	.1589354
__00007J	.0315184	.0411415	0.77	0.444	-.0491387	.1121755
__00007K	-.0752832	.0302422	-2.49	0.013	-.1345724	-.015994
__00007L	.0125053	.0227207	0.55	0.582	-.0320382	.0570487
__00007M	.0286478	.0317049	0.90	0.366	-.0335091	.0908047
__00007N	.0174569	.0380542	0.46	0.646	-.0571476	.0920614
__00007O	.2851267	.4558075	0.63	0.532	-.6084747	1.178728
__00007P	.2207786	.0723679	3.05	0.002	.0789028	.3626543
__00007Q	-.3915989	.1823773	-2.15	0.032	-.7491459	-.0340519
__00007R	-.012365	.4576998	-0.03	0.978	-.9096763	.8849463
__00007S	-.1570488	.1017663	-1.54	0.123	-.3565595	.042462
__00007T	-.4053457	.1710711	-2.37	0.018	-.7407271	-.0699642
__00007U	.0972112	.0620585	1.57	0.117	-.0244534	.2188757
__00007V	-.2339363	.0420127	-5.57	0.000	-.3163014	-.1515711
__00007W	-.0218837	.0324529	-0.67	0.500	-.085507	.0417396
__00007X	.0212272	.0497507	0.43	0.670	-.076308	.1187624
__00007Y	.005633	.0584831	0.10	0.923	-.109022	.1202879
__000080	8.445766	18.96277	0.45	0.656	-28.73037	45.6219
__000082	-2.538594	11.18002	-0.23	0.820	-24.4568	19.37962
__000085	-.0754758	.5249525	-0.14	0.886	-1.104635	.9536832
__000087	.2107624	.3791253	0.56	0.578	-.5325052	.9540299
__00008A	-1.497372	3.684006	-0.41	0.684	-8.719792	5.725049
__00008C	.2927278	2.177493	0.13	0.893	-3.976204	4.56166
__00008F	-10.63528	23.58522	-0.45	0.652	-56.87362	35.60307
__00008H	3.123759	14.11996	0.22	0.825	-24.55815	30.80566
__00008K	-.1941854	.6776996	-0.29	0.774	-1.522802	1.134431
__00008M	-1.39456	.6534375	-2.13	0.033	-2.675611	-.1135085
__00008P	-1.343119	1.111617	-1.21	0.227	-3.522422	.8361837
__00008R	-1.271986	.9151353	-1.39	0.165	-3.066091	.5221185
__00008U	-.7730286	.4412188	-1.75	0.080	-1.638029	.091972
__00008W	.2278244	.3407778	0.67	0.504	-.4402638	.8959125
__00008Z	.601064	.2397331	2.51	0.012	.1310721	1.071056
__000091	-.0640346	.2041853	-0.31	0.754	-.4643358	.3362665
__000094	-.7268351	.2604519	-2.79	0.005	-1.237446	-.2162243
__000096	-.228038	.2082336	-1.10	0.274	-.6362758	.1801998
__000099	.7647842	.3113365	2.46	0.014	.1544152	1.375153
__00009B	.1977191	.261654	0.76	0.450	-.3152483	.7106865
__00009E	-.0660116	.3747056	-0.18	0.860	-.8006145	.6685913
__00009G	-.243422	.302943	-0.80	0.422	-.8373358	.3504917
xbarlag	-4.769469	22.185	-0.21	0.830	-48.26273	38.72379
x2barlag	-.4538931	.5918067	-0.77	0.443	-1.614118	.7063322
xbar2lag	1.725566	4.421583	0.39	0.696	-6.94286	10.39399
rootxbarlag	5.399595	27.44806	0.20	0.844	-48.41177	59.21096
z1barlag	1.81121	1.002554	1.81	0.071	-.1542766	3.776696
z2barlag	2.690187	1.511498	1.78	0.075	-.2730747	5.653448
z3barlag	.552162	.563691	0.98	0.327	-.5529429	1.657267
z4barlag	-.2955395	.2702011	-1.09	0.274	-.8252634	.2341844
z5barlag	1.024167	.2637745	3.88	0.000	.5070427	1.541292
z6barlag	-.9630149	.4686881	-2.05	0.040	-1.881869	-.0441613
z7barlag	.2824898	.5031912	0.56	0.575	-.7040064	1.268986
_cons	-3.250004	9.429695	-0.34	0.730	-21.73673	15.23672

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A2	**128,640**	**.4438109**	**.13429**	**.0976942**	**1.111178**

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"
957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9
960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
964 }
965
966 local r_RE      "`r_RE' `qhat_hats'"
967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }
978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
```

```
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
```

Source	SS	df	MS	Number of obs	=	128,640
				F(16, 128623)	=	48925.93
Model	5976.12589	16	373.507868	Prob > F	=	0.0000
Residual	981.927241	128,623	.00763415	R-squared	=	0.8589
				Adj R-squared	=	0.8589
Total	6958.05313	128,639	.054089764	Root MSE	=	.08737

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
xi2	-.0374445	.0005646	-66.33	0.000	-.038551	-.0363379
ybar_q1_2	-.1066234	.0092279	-11.55	0.000	-.1247099	-.0885368
ybar_q2_2	.1108799	.0024778	44.75	0.000	.1060235	.1157364
ybarx_1	.236722	.0034032	69.56	0.000	.2300518	.2433922
ybarx_2	-.1057571	.0018199	-58.11	0.000	-.1093241	-.1021902
ybar_q1_q2	-.1052302	.008448	-12.46	0.000	-.121788	-.0886723
xi	.3630039	.0018141	200.11	0.000	.3594484	.3665594
ybar_q1	.2163611	.0076981	28.11	0.000	.2012729	.2314493
ybar_q2	-.1332122	.0038871	-34.27	0.000	-.1408309	-.1255936
pzi1	.2218601	.0012929	171.59	0.000	.2193259	.2243942
pzi2	.0173906	.0021069	8.25	0.000	.0132612	.0215201
pzi3	-.002822	.0006512	-4.33	0.000	-.0040983	-.0015456
pzi4	.0085446	.000649	13.17	0.000	.0072725	.0098166
pzi5	-.0138749	.0006967	-19.91	0.000	-.0152405	-.0125093
pzi6	.0026067	.0005751	4.53	0.000	.0014796	.0037338
pzi7	-.0041551	.000745	-5.58	0.000	-.0056152	-.0026949
_cons	-.0081675	.0020377	-4.01	0.000	-.0121614	-.0041736

```
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = -_b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                       local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                       local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                       if `a1'==. local a1 = .5
1016                       if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
```

```
1024        forv j = 1(1)`T' {
    2.                local coef = _b[pzi`j']
    3.                local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.         }
1025}

1026
1027if `J'==3 & simple_model==0
  >                           local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                           local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00131158
  Iteration 1:  GMM criterion Q(b) =  .00129941
  Iteration 2:  GMM criterion Q(b) =  .00019501
  Iteration 3:  GMM criterion Q(b) =  .00012324

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00628677
  Iteration 1:  GMM criterion Q(b) =  .00473085
  Iteration 2:  GMM criterion Q(b) =  .00458086
  Iteration 3:  GMM criterion Q(b) =  .00456875

  GMM estimation

  Number of parameters =  21
  Number of moments    =  28
  Initial weight matrix: Unadjusted                 Number of obs   =    128,640
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
A11          |
         one |   .6386334    .035983    17.75   0.000     .5681081    .7091587
-------------+----------------------------------------------------------------
A22          |
         one |    .572239   .0335989    17.03   0.000     .5063864    .6380916
-------------+----------------------------------------------------------------
C1           |
         zi1 |    .480775   .0468969    10.25   0.000     .3888588    .5726911
         zi2 |  -.2133117   .0741682    -2.88   0.004    -.3586787   -.0679447
         zi3 |  -.0057888   .0216874    -0.27   0.790    -.0482953    .0367177
         zi4 |   .1005197    .032089     3.13   0.002     .0376264     .163413
         zi5 |   .0562264   .0231198     2.43   0.015     .0109125    .1015404
         zi6 |   .0105685   .0186388     0.57   0.571    -.0259629    .0470998
         zi7 |   .0017484   .0230712     0.08   0.940    -.0434704    .0469671
------------------------------------------------------------------------------
```

C2						
zi1	**.2725508**	**.0751344**	**3.63**	**0.000**	**.1252902**	**.4198115**
zi2	**-.3951905**	**.1206088**	**-3.28**	**0.001**	**-.6315794**	**-.1588015**
zi3	**-.0081602**	**.0338998**	**-0.24**	**0.810**	**-.0746026**	**.0582822**
zi4	**.1663959**	**.0511606**	**3.25**	**0.001**	**.066123**	**.2666687**
zi5	**.1020354**	**.0295905**	**3.45**	**0.001**	**.0440391**	**.1600316**
zi6	**.0077339**	**.027699**	**0.28**	**0.780**	**-.0465552**	**.0620229**
zi7	**.0057401**	**.0384689**	**0.15**	**0.881**	**-.0696577**	**.0811378**
/d1	**-.0900836**	**.0076088**	**-11.84**	**0.000**	**-.1049965**	**-.0751707**
/b1	**.3999796**	**.0213464**	**18.74**	**0.000**	**.3581415**	**.4418177**
/AVA11	**1.211787**	**.6297775**	**1.92**	**0.054**	**-.0225546**	**2.446128**
/AVA12	**-1.208932**	**.5618459**	**-2.15**	**0.031**	**-2.31013**	**-.1077342**
/AVA22	**.737902**	**.5302183**	**1.39**	**0.164**	**-.3013067**	**1.777111**

Instruments for equation **eq1_RE: __00009J __00009L __00009O xi __000002 __00000E __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5 __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons**

```
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
```

```
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3_all_exactIns
      > t_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not found)
  (output written to diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  24757

1077unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640

1078estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  4599

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  5.3831268

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster not found)
  file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE
```

```
1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*`p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
```

```
1119                                                    gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                     local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                          }
   30.                                    }
   31.                           }
   32.                   }
   33.           }
1120        if `Tindivp1'<=`T' {
1121                forval t=`Tindivp1'/`T' {
    2.                          tempvar zi`t'xdiff
    3.                          g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                          local zg "`zg' zi`t'"
    5.                          forval j=1/`J' {
    6.                                  tempvar zi`t'p`j'xdiff
    7.                                  g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                  local pzg "`pzg' `zi`t'p`j''"
    9.                                  local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                          }
   11.                   }
1122        }
1123
1124        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1125        // denoting zg as group-zs
1126        //  zg,pzgx,rprp,rprpx
1127        local rootprootp ""
1128        local rootprootpxdiff ""
1129
1130        forval j=1/`J' {
    2.                  forval s=`j'/`J' {
    3.                          tempvar xdiffrp`j'rp`s'
    4.                          g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1131                        if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1132                        local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                  }
    8.          }
1133
1134        local pqhatxdiff ""
1135        forval j=1/`J' {
    2.                  forval s=1/`J' {
    3.                          foreach name of global Alist {
    4.                                  tempvar   p`s'qikhat`j'xdiff`name'
    5.                                  g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
1136                                local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
1137
1138        // set instruments
1139        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1140}
```

```
1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                tempvar ximxk`var'
    3.                g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                        local `++t'
    3.                        local p_nice = subinstr("`p'",":_cons","",.)
    4.                        local est = coefs[1,`t']
    5.                        if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                        if `j'>`i' {
   11.                                tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                        }
   15.                }
   16.         }
```

```
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >     `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1     `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
```

```
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00073774
  Iteration 1:  GMM criterion Q(b) =  .00024787
  Iteration 2:  GMM criterion Q(b) =  .00019178

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00927676
  Iteration 1:  GMM criterion Q(b) =  .00836217
  Iteration 2:  GMM criterion Q(b) =  .00823763
  Iteration 3:  GMM criterion Q(b) =  .00822026

  GMM estimation

  Number of parameters =  18
  Number of moments    =  63
  Initial weight matrix: Unadjusted                 Number of obs   =    128,640
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
C11          |
       _cons |   .6282793   .0325287    19.31   0.000     .5645242    .6920344
-------------+----------------------------------------------------------------
C12          |
       _cons |  -.0531358    .044604    -1.19   0.234     -.140558    .0342864
-------------+----------------------------------------------------------------
C13          |
       _cons |  -.0111727   .0128836    -0.87   0.386    -.0364241    .0140787
-------------+----------------------------------------------------------------
C14          |
       _cons |  -.0193327   .0122219    -1.58   0.114    -.0432873    .0046218
-------------+----------------------------------------------------------------
C15          |
       _cons |  -.0112344    .020477    -0.55   0.583    -.0513685    .0288997
------------------------------------------------------------------------------
```

C16 _cons	**-.0205004**	**.0142143**	**-1.44**	**0.149**	**-.0483598**	**.0073591**
C17 _cons	**-.0179014**	**.0166358**	**-1.08**	**0.282**	**-.050507**	**.0147041**
C21 _cons	**.4718626**	**.046017**	**10.25**	**0.000**	**.3816709**	**.5620542**
C22 _cons	**-.1140294**	**.0648999**	**-1.76**	**0.079**	**-.2412308**	**.013172**
C23 _cons	**-.0146051**	**.0177636**	**-0.82**	**0.411**	**-.049421**	**.0202109**
C24 _cons	**-.0481249**	**.0214564**	**-2.24**	**0.025**	**-.0901787**	**-.0060711**
C25 _cons	**.0087781**	**.0260062**	**0.34**	**0.736**	**-.042193**	**.0597493**
C26 _cons	**-.0292134**	**.0189598**	**-1.54**	**0.123**	**-.066374**	**.0079471**
C27 _cons	**-.0158767**	**.0243514**	**-0.65**	**0.514**	**-.0636046**	**.0318511**
d1 _cons	**-.0794863**	**.0061876**	**-12.85**	**0.000**	**-.0916137**	**-.0673588**
b1 _cons	**.3739456**	**.0182256**	**20.52**	**0.000**	**.3382241**	**.4096671**
A11 one	**2.228055**	**.381634**	**5.84**	**0.000**	**1.480066**	**2.976044**
A22 one	**-.9112117**	**.2755403**	**-3.31**	**0.001**	**-1.451261**	**-.3711627**

Instruments for equation **eq1_FE: __000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7 __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1 __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY __0000D5 _cons**

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3_all_exactIns
      > t_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex not found)
  (output written to diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex)
1242
1243        estat overid

    Test of overidentifying restriction:

    Hansen's J chi2(45) = 1057.45 (p = 0.0000)
```

```
1244        di "$S_DATE $S_TIME"
   7 Dec 2023 21:45:21
1245        unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  24757
1248        unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640
1249        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4599
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  5.3831268
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster not found)
  file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}

1274else {
```

```
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table6_25.smcl
   log type:  smcl
  closed on:   7 Dec 2023, 21:45:21
-------------------------------------------------------------------------------
```

A.7 Table 7

A.7.1 Columns 1 and 3

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table7_13.smcl
  log type:  smcl
 opened on:   8 Dec 2023, 09:29:37

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=3

47. local J = J

48.
49. // group def options
50. local group_def "dist"

51. local group_def "fsuXseg"

52. local group_def "fsuXsegXreligXsched"

53.
54. // comparision group
55. local groupComp_def ""

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.         syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
70.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                  Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations                 840   ->   280
  Number of variables                      5   ->   6
  j variable (3 values)          expend_type   ->   (dropped)
  xij variables:
                              laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.             }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "

116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,637 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                    egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
```

```
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }
297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }
316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }
333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
```

```
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (82 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,576 real changes made, 82 to missing)

460 replace x2=x*x
  (73,576 real changes made, 82 to missing)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.          g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,576 real changes made)

478 replace x2=x2_temp
  (73,576 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (48,819 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)
```

```
488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_4630_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
```

```
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,757 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
549 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var'" _n
    5.                 }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
```

```
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=0

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1
```

```
625 scalar estimate_FE=1

626
627 // estimation options
628 global w_initial "winit(unadjusted)"

629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

630 global wmatrix "wmatrix(robust)"

631 global trace_level ""

632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

633 global technique ""

634 global maxiter ""

635 global technique_RE ""

636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

637 global derivatives "quickderivatives"

638 scalar init_FE_from_RE=1

639 global clust_var "state_district_round"

640 local clus_name "_clus_Dist"

641
642
643
644
645
646
647
648
649 // xxzx to keep names same
650 local wmatrix_desc ""

651 local sectorName "_urbOnly"

652 local dropZdesc "_dropPre"

653 local desc_struc_v0 ""

654 local inst_desc "_exactInst"

655 local inst_rootp "_noPPXdInst"

656 local weight_desc ""

657 local RE_sq_inst_desc ""

658 local cnstname ""

659 local actual_pqhat_desc ""

660 local expUpname "_updateExp"
```

```
661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //              ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T

694 local Tm1=`T'-1
```

```
695 local Tindiv : word count $zlist_indiv

696
697 /*****************************************************************************
  > Set weights
  > *****************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*****************************************************************************
  >  Update data for J=2, and J=3
  > *****************************************************************************/
706
707 if `J'==2 {
708         replace p1=P1
709         replace p2=P2
710         replace qi1=Qi1
711         replace qk1=Qk1
712         replace qikbar1=Qikbar1
713         replace qi2=Qi2
714         replace qk2=Qk2
715         replace qikbar2=Qikbar2
716
717         foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                 qui replace `var'=.
    3.         }
718 }

719
720 if `J'==3 {
721         forv j = 1(1)3 {
    2.                 replace p`j'=P`j'
    3.                 replace qi`j'=Qi`j'
    4.                 replace qk`j'=Qk`j'
    5.                 replace qikbar`j'=Qikbar`j'
    6.         }
  (128,640 real changes made)
  (128,632 real changes made)
  (128,632 real changes made)
  (128,640 real changes made)
  (0 real changes made)
  (0 real changes made)
  (0 real changes made)
  (0 real changes made)
  (128,640 real changes made)
  (128,637 real changes made)
  (128,637 real changes made)
  (128,640 real changes made)
722
723         foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                 qui replace `var'=.
    3.         }
```

```
724 }

725
726
727 /*****************************************************************************
  > Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }
```

```
751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
773         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.          if Afull==1 {
    3.                  forval j=1/`J' {
    4.                          local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                  if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                  if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                  if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                          }
   11.                          local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                  forval j=1/`J' {
   16.                          if same_spillover==0 local A_term "{A`j'`j':}"
   17.                          if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
```

```
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j'"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }

798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.         local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.         if `jplus1'<=`J' {
    5.                 forval k=`jplus1'/`J' {
    6.                         local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                 }
    8.         }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                             local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                 if noMeasError==1                       local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'              - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.         }
```

```
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                    l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                    ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A22:}*p2*qk2) + ({A33:}*p3*qk3) )*( ({A11:
  > }*p1*qikbar1) + ({A22:}*p2*qikbar2) + ({A33:}*p3*qikbar3) )   - 2*(xi - ( (p1*{C1: z
  > i1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p3*{C3: zi1
  >  zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) +
  >  ({A33:}*p3*qikbar3) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )*
  > (xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p3)
  >  + {b2}*ln(p2/p3) + ln(p3) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0
  >  ) )-( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) + ({A33:}*p3*qikbar3) ))*{b1} + ({A
  > 11:}*p1*qikbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + 2*p1*
  > {AVA13}*p3 + p2*{AVA22}*p2 + 2*p2*{AVA23}*p3 + p3*{AVA33}*p3 )*{d1}/exp( {b1}*ln(p1/
  > p3) + {b2}*ln(p2/p3) + ln(p3) ) ) ) (eq2_RE: (p2*qi2 - ((( ({A11:}*p1*qk1) + ({A22
  > :}*p2*qk2) + ({A33:}*p3*qk3) )*( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) + ({A33:}
  > *p3*qikbar3) )        - 2*(xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )*(
  >  ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) + ({A33:}*p3*qikbar3) ) + (xi - ( (p1*{C1
  > :}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:
  > }) ) - ( 0 ) )) * {d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3) ) + ((xi - ( (p
  > 1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A22:}*p2*qi
  > kbar2) + ({A33:}*p3*qikbar3) ) )*{b2} + ({A22:}*p2*qikbar2) + (p2*{C2:}) + ( 0 ) - (
  > 0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + 2*p1*{AVA13}*p3 + p2*{AVA22}*p2 + 2*p2*{AVA23}
  > *p3 + p3*{AVA33}*p3 )*{d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) + ( p3*{C31}*zi1 + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*
  > {C36}*zi6 + p3*{C37}*zi7 ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*z
  > k3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 +
  > p2*{C22}*zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*
  > zk7 ) + ( p3*{C31}*zk1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 +
  >  p3*{C36}*zk6 + p3*{C37}*zk7 ) ))^2) * {d1}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + l
  > n(p3) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*
  > {C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}
  > *zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) + ( p3*{C31}*zi1
  > + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*{C36}*zi6 + p3*{C37
  > }*zi7 ) )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p
  > 1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C2
  > 3}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) + ( p3*{C31}*zk
```

```
  > 1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 + p3*{C36}*zk6 + p3*{C
  > 37}*zk7 ) )))*({b1} - 2*(( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) + ({A33:}*p3*qi
  > kbar3) ) + ( ( 0 ) + ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/
  > p3) + ln(p3) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*
  > {C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}
  > *zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         )) (eq2_
  > FE: ((p2*qi2 - p2*qk2) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*
  > {C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C22}
  > *zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) +
  > ( p3*{C31}*zi1 + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*{C36
  > }*zi6 + p3*{C37}*zi7 ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 +
  >  p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{
  > C22}*zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7
  > ) + ( p3*{C31}*zk1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 + p3*
  > {C36}*zk6 + p3*{C37}*zk7 ) ))^2) * {d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3
  > ) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{C15
  > }*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}*zi3
  >  + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) + ( p3*{C31}*zi1 + p3
  > *{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*{C36}*zi6 + p3*{C37}*zi
  > 7 ) )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) + ( p3*{C31}*zk1 +
  > p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 + p3*{C36}*zk6 + p3*{C37}*
  > zk7 ) )))*({b2} - 2*(( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) + ({A33:}*p3*qikbar
  > 3) ) + ( ( 0 ) + ( 0 ) + ( 0 ) ) + ( 0 ))*{d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3)
  > + ln(p3) )) - (( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25
  > }*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) - ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3
  >  + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ))         ))

836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2           "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
```

```
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                              "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}     `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                          "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qk1) + ({A22:one}*p2*qk2) + ({A33:one}*p3*qk3) )
  > *( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) + ({A33:}*p3*qikbar3) )   - 2*(xi - ( (
  > p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p3
  > *{C3: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A22:}*p2*q
  > ikbar2) + ({A33:}*p3*qikbar3) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) -
  >  ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )) * {d1}/exp( {b1}*
  > ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) + ({A33:}*p3*qikbar3) ))*{
  > b1} + ({A11:}*p1*qikbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p
  > 2 + 2*p1*{AVA13}*p3 + p2*{AVA22}*p2 + 2*p2*{AVA23}*p3 + p3*{AVA33}*p3 )*{d1}/exp( {b
  > 1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3) ) ) ) (eq2_RE: (p2*qi2 - ((( ({A11:}*p1*qk1
  > ) + ({A22:}*p2*qk2) + ({A33:}*p3*qk3) )*( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2)
  > + ({A33:}*p3*qikbar3) )        - 2*(xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) -
  > ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) + ({A33:}*p3*qikbar3) ) + (xi -
  > ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:}) +
  >  (p3*{C3:}) ) - ( 0 ) )) * {d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3) ) + ((
  > xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A2
  > 2:}*p2*qikbar2) + ({A33:}*p3*qikbar3) ) )*{b2} + ({A22:}*p2*qikbar2) + (p2*{C2:}) +
  > ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + 2*p1*{AVA13}*p3 + p2*{AVA22}*p2 + 2*p
  > 2*{AVA23}*p3 + p3*{AVA33}*p3 )*{d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3) )
  > ) ) )
```

```
873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) + ( p3*{C31}*zi1 + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*
  > {C36}*zi6 + p3*{C37}*zi7 ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*z
  > k3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 +
  > p2*{C22}*zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*
  > zk7 ) + ( p3*{C31}*zk1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 +
  >  p3*{C36}*zk6 + p3*{C37}*zk7 ) ))^2) * {d1}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + l
  > n(p3) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*
  > {C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}
  > *zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) + ( p3*{C31}*zi1
  > + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*{C36}*zi6 + p3*{C37
  > }*zi7 ) )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p
  > 1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C2
  > 3}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) + ( p3*{C31}*zk
  > 1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 + p3*{C36}*zk6 + p3*{C
  > 37}*zk7 ) )))*({b1} - 2*(( ({A11:one}*p1*qikbar1) + ({A22:one}*p2*qikbar2) + ({A33:o
  > ne}*p3*qikbar3) ) + ( ( 0 ) + ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1/p3) + {b
  > 2}*ln(p2/p3) + ln(p3) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*
  > zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 +
  >  p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))
  >  )) (eq2_FE: ((p2*qi2 - p2*qk2) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*
  > zi3 + p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 +
  >  p2*{C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}
  > *zi7 ) + ( p3*{C31}*zi1 + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5
  > + p3*{C36}*zi6 + p3*{C37}*zi7 ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C
  > 13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*z
  > k1 + p2*{C22}*zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{
  > C27}*zk7 ) + ( p3*{C31}*zk1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*
  > zk5 + p3*{C36}*zk6 + p3*{C37}*zk7 ) ))^2) * {d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3
  > ) + ln(p3) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4
  > + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*
  > {C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) + ( p3*{C31}
  > *zi1 + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*{C36}*zi6 + p3
  > *{C37}*zi7 ) ) ) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk
  > 4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p
  > 2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) + ( p3*{C3
  > 1}*zk1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 + p3*{C36}*zk6 +
  > p3*{C37}*zk7 ) )))*({b2} - 2*(( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) + ({A33:}*
  > p3*qikbar3) ) + ( ( 0 ) + ( 0 ) + ( 0 ) ) + ( 0 ))*{d2}/exp( {b1}*ln(p1/p3) + {b2}*l
  > n(p2/p3) + ln(p3) )) - (( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4
  > + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) - ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*
  > {C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ))         ))

874
875 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi
```

```
882 g double `ximxk'=xi-xk

883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.         tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.         g double `xizi`t''=xi*zi`t'
    4.         g double `xkzk`t''=xk*zk`t'
    5.         g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.         g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.         g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.         g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.         g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.         forval j=1/`J' {
   11.                 tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                 g double `zi`t'p`j''= zi`t'*p`j'
   13.                 g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                 g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                 g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                 forv s=1/`T' {
   17.                         tempvar zi`t'zi`s'p`j'
   18.                         g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                 }
   20.         }
   21. }

891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                 local rg "`rg' z`t'barlag"
    3. }

903
904 local zrg ""
```

```
905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var'"
    6.          }
    7. }

906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var'"
    5.
911         forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var'"
   10.                  local xprg "`xprg' `xp`j'`var'"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""
```

```
925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }

929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.          forval k=`j'/`J' {
    3.                  tempvar p`j'p`k'
    4.                  g `p`j'p`k''=p`j'*p`k'
    5.                  local roundinsts "`roundinsts' `p`j'p`k''"
    6.          }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 29

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.          if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.          if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
```

```
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }
```

Linear regression

Number of obs	= **128,640**
F(148, 4598)	= .
Prob > F	= .
R-squared	= **0.2305**
Root MSE	= **.13046**

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__0000CA	**15.8624**	**12.19746**	**1.30**	**0.194**	**-8.050478**	**39.77528**
__0000CC	**.2763886**	**.8341895**	**0.33**	**0.740**	**-1.359023**	**1.9118**
__0000CE	**-1.458342**	**2.051914**	**-0.71**	**0.477**	**-5.481078**	**2.564395**
__0000CH	**-.261114**	**1.983154**	**-0.13**	**0.895**	**-4.149048**	**3.62682**
__0000CJ	**-1.4065**	**.6557057**	**-2.15**	**0.032**	**-2.691998**	**-.1210025**
__0000CM	**-9.797743**	**7.400675**	**-1.32**	**0.186**	**-24.30662**	**4.711132**
xi	**.1067121**	**.007683**	**13.89**	**0.000**	**.0916498**	**.1217744**
__000002	**-.0150922**	**.0024866**	**-6.07**	**0.000**	**-.0199672**	**-.0102172**
__00000E	**-.1979284**	**.1017616**	**-1.95**	**0.052**	**-.39743**	**.0015733**
__00000P	**-.0186921**	**.0303069**	**-0.62**	**0.537**	**-.0781082**	**.040724**
__000010	**.038788**	**.0475737**	**0.82**	**0.415**	**-.0544792**	**.1320553**
__00001I	**.0878379**	**.2069769**	**0.42**	**0.671**	**-.3179361**	**.4936119**
__00001T	**.0026737**	**.0645317**	**0.04**	**0.967**	**-.1238394**	**.1291868**
__000024	**-.0417135**	**.0917764**	**-0.45**	**0.649**	**-.2216393**	**.1382123**
__00002M	**.0245382**	**.0517151**	**0.47**	**0.635**	**-.0768482**	**.1259245**
__00002X	**-.0151116**	**.0145696**	**-1.04**	**0.300**	**-.043675**	**.0134517**
__000038	**-.0073205**	**.0207005**	**-0.35**	**0.724**	**-.0479034**	**.0332624**
__00003Q	**.1498987**	**.0716781**	**2.09**	**0.037**	**.0093752**	**.2904223**
__000041	**.0326157**	**.0288179**	**1.13**	**0.258**	**-.0238812**	**.0891126**
__00004C	**-.0762122**	**.0338384**	**-2.25**	**0.024**	**-.1425517**	**-.0098728**
__00004U	**-.0159037**	**.0641748**	**-0.25**	**0.804**	**-.1417171**	**.1099098**
__000055	**.0012639**	**.02123**	**0.06**	**0.953**	**-.040357**	**.0428848**
__00005G	**.0311996**	**.0329209**	**0.95**	**0.343**	**-.0333412**	**.0957404**
__00005Y	**-.0655302**	**.0501822**	**-1.31**	**0.192**	**-.1639113**	**.032851**
__000069	**.0028**	**.014815**	**0.19**	**0.850**	**-.0262445**	**.0318445**
__00006K	**.0058762**	**.023205**	**0.25**	**0.800**	**-.0396168**	**.0513692**
__000072	**-.0420747**	**.067466**	**-0.62**	**0.533**	**-.1743404**	**.090191**
__00007D	**.009275**	**.0211865**	**0.44**	**0.662**	**-.0322608**	**.0508107**
__00007O	**-.0120165**	**.0295094**	**-0.41**	**0.684**	**-.0698691**	**.045836**
__00007Z	**.0247669**	**.4025898**	**0.06**	**0.951**	**-.7645023**	**.8140362**
__000080	**-.0256414**	**.0718753**	**-0.36**	**0.721**	**-.1665515**	**.1152688**
__000081	**-.0459346**	**.1644982**	**-0.28**	**0.780**	**-.36843**	**.2765608**
__000082	**.221121**	**.419613**	**0.53**	**0.598**	**-.6015218**	**1.043764**
__000083	**-.0412492**	**.1019986**	**-0.40**	**0.686**	**-.2412154**	**.1587169**
__000084	**.0949755**	**.1636718**	**0.58**	**0.562**	**-.2258999**	**.4158508**
__000085	**-.0185792**	**.0678023**	**-0.27**	**0.784**	**-.1515042**	**.1143458**
__000086	**.0215846**	**.0528749**	**0.41**	**0.683**	**-.0820755**	**.1252447**
__000087	**.0385419**	**.0347249**	**1.11**	**0.267**	**-.0295356**	**.1066194**
__000088	**-.0219194**	**.0475841**	**-0.46**	**0.645**	**-.1152071**	**.0713683**
__000089	**-.0619666**	**.0573835**	**-1.08**	**0.280**	**-.1744657**	**.0505326**
__00008A	**-.1717907**	**.8876696**	**-0.19**	**0.847**	**-1.912049**	**1.568468**
__00008B	**.2371389**	**.1441006**	**1.65**	**0.100**	**-.0453675**	**.5196453**
__00008C	**-.3483116**	**.3459745**	**-1.01**	**0.314**	**-1.026588**	**.3299644**

__00008D	.6149689	.871486	0.71	0.480	-1.093562	2.3235
__00008E	-.1561689	.1865518	-0.84	0.403	-.5218999	.2095622
__00008F	-.5748937	.3248082	-1.77	0.077	-1.211674	.0618863
__00008G	-.0773958	.1282103	-0.60	0.546	-.3287495	.1739579
__00008H	-.2206576	.1096055	-2.01	0.044	-.4355371	-.0057782
__00008I	.0952415	.0626305	1.52	0.128	-.0275443	.2180273
__00008J	-.1571377	.0986506	-1.59	0.111	-.3505402	.0362649
__00008K	.0434661	.1178775	0.37	0.712	-.1876304	.2745627
__00008L	-.1385752	.1857159	-0.75	0.456	-.5026674	.2255171
__00008M	.0323174	.032756	0.99	0.324	-.0319001	.0965349
__00008N	-.0264521	.0719491	-0.37	0.713	-.1675069	.1146027
__00008O	.1577994	.210457	0.75	0.453	-.2547973	.5703962
__00008P	.0684999	.0516808	1.33	0.185	-.0328193	.1698191
__00008Q	-.0699087	.0836604	-0.84	0.403	-.2339233	.0941059
__00008R	-.0367281	.03464	-1.06	0.289	-.1046392	.031183
__00008S	.01828	.0247779	0.74	0.461	-.0302967	.0668567
__00008T	-.0208759	.0154785	-1.35	0.177	-.0512211	.0094693
__00008U	-.0298464	.0248785	-1.20	0.230	-.0786201	.0189274
__00008V	.0034188	.030401	0.11	0.910	-.0561818	.0630194
__00008W	-.6815638	.4631033	-1.47	0.141	-1.589469	.2263411
__00008X	-.0595999	.0543503	-1.10	0.273	-.1661526	.0469528
__00008Y	.3617518	.1778597	2.03	0.042	.0130614	.7104423
__00008Z	.3564088	.4096254	0.87	0.384	-.4466536	1.159471
__000090	-.1103219	.0776549	-1.42	0.155	-.2625629	.041919
__000091	-.0550339	.135551	-0.41	0.685	-.3207789	.210711
__000092	-.0331743	.057593	-0.58	0.565	-.1460843	.0797357
__000093	.0572334	.0366143	1.56	0.118	-.0145482	.1290151
__000094	-.0481142	.0254041	-1.89	0.058	-.0979185	.00169
__000095	.0195355	.0434165	0.45	0.653	-.0655816	.1046526
__000096	-.030765	.0489128	-0.63	0.529	-.1266576	.0651276
__000097	-.5444064	.3003944	-1.81	0.070	-1.133324	.0445109
__000098	-.0938369	.0454564	-2.06	0.039	-.1829533	-.0047206
__000099	.3246808	.1205504	2.69	0.007	.0883441	.5610174
__00009A	.3673008	.2900397	1.27	0.205	-.2013162	.9359178
__00009B	-.0799506	.0636141	-1.26	0.209	-.2046648	.0447635
__00009C	-.009825	.1083546	-0.09	0.928	-.2222519	.202602
__00009D	-.0053584	.0453438	-0.12	0.906	-.094254	.0835371
__00009E	-.0143521	.0337705	-0.42	0.671	-.0805585	.0518544
__00009F	.0266307	.022425	1.19	0.235	-.017333	.0705944
__00009G	-.0067169	.0314455	-0.21	0.831	-.0683652	.0549314
__00009H	-.0661386	.0368349	-1.80	0.073	-.1383526	.0060754
__00009I	.0408486	.2100307	0.19	0.846	-.3709124	.4526096
__00009J	.0095971	.035194	0.27	0.785	-.0594	.0785941
__00009K	-.0725834	.0830405	-0.87	0.382	-.2353826	.0902158
__00009L	.1443063	.2176461	0.66	0.507	-.2823845	.5709971
__00009M	-.03505	.0515204	-0.68	0.496	-.1360548	.0659548
__00009N	-.0913499	.0873698	-1.05	0.296	-.2626367	.079937
__00009O	-.0170904	.0328973	-0.52	0.603	-.0815849	.047404
__00009P	-.0169291	.028856	-0.59	0.557	-.0735007	.0396425
__00009Q	.0172245	.0164509	1.05	0.295	-.0150273	.0494762
__00009R	-.009666	.0274487	-0.35	0.725	-.0634787	.0441467
__00009S	.0452256	.0311324	1.45	0.146	-.0158088	.10626
__00009T	-.0615899	.3425282	-0.18	0.857	-.7331097	.6099299
__00009U	.077691	.0477497	1.63	0.104	-.0159213	.1713034
__00009V	-.1074309	.1309426	-0.82	0.412	-.3641412	.1492795
__00009W	.1845007	.3207713	0.58	0.565	-.444365	.8133664
__00009X	-.0115453	.0670409	-0.17	0.863	-.1429776	.1198869
__00009Y	-.2080519	.1151702	-1.81	0.071	-.4338409	.0177371
__00009Z	.0189436	.0427652	0.44	0.658	-.0648967	.1027839
__0000A0	-.0594547	.0348877	-1.70	0.088	-.1278514	.008942
__0000A1	-.0041966	.0210033	-0.20	0.842	-.0453732	.03698
__0000A2	.0081189	.0345884	0.23	0.814	-.0596909	.0759287
__0000A3	.0358123	.0417068	0.86	0.391	-.045953	.1175777
__0000A5	40.51572	29.21938	1.39	0.166	-16.76829	97.79974
__0000A7	-1.193598	4.443261	-0.27	0.788	-9.904523	7.517326
__0000A9	-27.64726	15.94144	-1.73	0.083	-58.90014	3.605625
__0000AC	.4094557	.5765131	0.71	0.478	-.7207868	1.539698
__0000AE	.2383025	.1406811	1.69	0.090	-.0375	.514105
__0000AG	-.3241458	.3952156	-0.82	0.412	-1.098958	.4506664
__0000AJ	-9.32148	5.851725	-1.59	0.111	-20.79367	2.150711
__0000AL	-.3222155	.853455	-0.38	0.706	-1.995397	1.350966
__0000AN	5.88066	2.977866	1.97	0.048	.0426117	11.71871

__0000AQ	-46.8338	35.69909	-1.31	0.190	-116.8212	23.15356
__0000AS	2.62199	5.65325	0.46	0.643	-8.461094	13.70507
__0000AU	33.70891	20.41349	1.65	0.099	-6.311322	73.72915
__0000AX	-.3450033	1.009231	-0.34	0.732	-2.32358	1.633574
__0000AZ	-.587507	.2535504	-2.32	0.021	-1.084587	-.0904265
__0000B1	-.7093808	.6092835	-1.16	0.244	-1.903869	.4851074
__0000B4	-.8938142	1.366006	-0.65	0.513	-3.571841	1.784213
__0000B6	-.7915218	.364239	-2.17	0.030	-1.505605	-.0774385
__0000B8	-.1342555	.7615952	-0.18	0.860	-1.627348	1.358837
__0000BB	-.4342364	.5363172	-0.81	0.418	-1.485676	.6172027
__0000BD	.1340458	.1350906	0.99	0.321	-.1307966	.3988882
__0000BF	.65156	.2948491	2.21	0.027	.0735142	1.229606
__0000BI	1.346408	.3487018	3.86	0.000	.6627851	2.030031
__0000BK	-.0705683	.1250638	-0.56	0.573	-.3157534	.1746168
__0000BM	-.469576	.1462335	-3.21	0.001	-.7562638	-.1828881
__0000BP	-.117023	.3489928	-0.34	0.737	-.8012163	.5671704
__0000BR	-.0283586	.1092474	-0.26	0.795	-.242536	.1858188
__0000BT	-.1336941	.1360998	-0.98	0.326	-.4005151	.1331269
__0000BW	.4687894	.4794096	0.98	0.328	-.4710835	1.408662
__0000BY	-.0751834	.1136129	-0.66	0.508	-.2979193	.1475525
__0000C0	-.378959	.2521978	-1.50	0.133	-.8733877	.1154697
__0000C3	.7321354	.5733746	1.28	0.202	-.3919541	1.856225
__0000C5	-.3223762	.1252385	-2.57	0.010	-.5679038	-.0768486
__0000C7	-.775386	.3240042	-2.39	0.017	-1.41059	-.1401824
xbarlag	-10.32228	23.20301	-0.44	0.656	-55.81133	35.16677
x2barlag	-.6247984	.4391064	-1.42	0.155	-1.485658	.2360611
xbar2lag	4.067035	4.682742	0.87	0.385	-5.113388	13.24746
rootxbarlag	8.489736	28.11775	0.30	0.763	-46.63455	63.61402
z1barlag	1.84462	.8722483	2.11	0.035	.134595	3.554646
z2barlag	2.154337	1.266077	1.70	0.089	-.3277809	4.636455
z3barlag	-.3966135	.4999116	-0.79	0.428	-1.37668	.5834533
z4barlag	-.7071445	.3394741	-2.08	0.037	-1.372677	-.0416123
z5barlag	.2425151	.3603317	0.67	0.501	-.463908	.9489383
z6barlag	.0428925	.4246848	0.10	0.920	-.7896937	.8754786
z7barlag	.3672959	.4434833	0.83	0.408	-.5021443	1.236736
_cons	-2.318915	9.48782	-0.24	0.807	-20.9196	16.28177

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000CU	128,640	.4424744	.0713668	.2244349	.8278759

Linear regression

Number of obs	=	128,640
F(148, 4598)	=	.
Prob > F	=	.
R-squared	=	0.2988
Root MSE	=	.03143

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__0000CA	4.677875	3.092828	1.51	0.130	-1.385553	10.7413
__0000CC	-1.51303	.2391049	-6.33	0.000	-1.98179	-1.044269
__0000CE	-2.153439	.5541039	-3.89	0.000	-3.239748	-1.067129
__0000CH	-.7392562	.4496332	-1.64	0.100	-1.620753	.1422407
__0000CJ	1.053623	.1871643	5.63	0.000	.6866908	1.420554
__0000CM	-2.669284	1.845161	-1.45	0.148	-6.286685	.9481181
xi	.0170337	.00271	6.29	0.000	.0117208	.0223467
__000002	-.0010007	.0010908	-0.92	0.359	-.0031392	.0011378
__00000E	-.019419	.0235414	-0.82	0.409	-.0655714	.0267335
__00000P	-.0102415	.0075348	-1.36	0.174	-.0250133	.0045303
__000010	-.0120784	.0103652	-1.17	0.244	-.0323992	.0082424
__00001I	.0815464	.0520438	1.57	0.117	-.0204844	.1835772
__00001T	.0210307	.0177048	1.19	0.235	-.0136792	.0557405
__000024	-.0000722	.0215525	-0.00	0.997	-.0423254	.042181
__00002M	.0193565	.0120741	1.60	0.109	-.0043145	.0430275
__00002X	.0053853	.0040066	1.34	0.179	-.0024697	.0132403
__000038	-.0005198	.0052951	-0.10	0.922	-.0109007	.0098611
__00003Q	.0248189	.0159515	1.56	0.120	-.0064537	.0560916

__000041	.0062229	.0065405	0.95	0.341	-.0065995	.0190454
__00004C	-.0191381	.0080141	-2.39	0.017	-.0348496	-.0034266
__00004U	-.0093908	.0150746	-0.62	0.533	-.0389443	.0201626
__000055	.0033442	.0049122	0.68	0.496	-.006286	.0129745
__00005G	.0113813	.0072846	1.56	0.118	-.0028999	.0256626
__00005Y	-.0154265	.0121828	-1.27	0.205	-.0393106	.0084576
__000069	.0008894	.0036126	0.25	0.806	-.006193	.0079718
__00006K	-.0048485	.0061735	-0.79	0.432	-.0169516	.0072546
__000072	.0022782	.0173399	0.13	0.895	-.0317163	.0362727
__00007D	.0104635	.005651	1.85	0.064	-.0006152	.0215423
__00007O	-.0025683	.0083613	-0.31	0.759	-.0189605	.0138239
__00007Z	-.0681979	.1007445	-0.68	0.498	-.2657054	.1293096
__000080	-.0110638	.0215427	-0.51	0.608	-.0532977	.0311701
__000081	.0215672	.0453249	0.48	0.634	-.0672914	.1104258
__000082	.0837318	.1078717	0.78	0.438	-.1277485	.295212
__000083	-.0097742	.0258815	-0.38	0.706	-.0605144	.0409661
__000084	.0802533	.0369765	2.17	0.030	.0077616	.1527449
__000085	-.0124454	.0184734	-0.67	0.501	-.0486622	.0237714
__000086	.0105039	.009937	1.06	0.291	-.0089773	.0299852
__000087	.0079296	.0084764	0.94	0.350	-.0086881	.0245474
__000088	-.0024098	.0104541	-0.23	0.818	-.0229048	.0180852
__000089	-.016453	.0116881	-1.41	0.159	-.0393672	.0064611
__00008A	.4000876	.2191735	1.83	0.068	-.0295976	.8297729
__00008B	.0841541	.0375219	2.24	0.025	.0105932	.157715
__00008C	-.2411249	.0896225	-2.69	0.007	-.4168279	-.0654218
__00008D	-.2069087	.2261006	-0.92	0.360	-.6501743	.2363569
__00008E	-.1169442	.0492232	-2.38	0.018	-.2134453	-.0204431
__00008F	-.1813671	.0899292	-2.02	0.044	-.3576716	-.0050626
__00008G	-.0108419	.0339126	-0.32	0.749	-.0773269	.0556432
__00008H	-.0551308	.0268128	-2.06	0.040	-.1076968	-.0025648
__00008I	.0015426	.0172301	0.09	0.929	-.0322366	.0353219
__00008J	-.0450347	.0246133	-1.83	0.067	-.0932886	.0032193
__00008K	-.0039367	.0295443	-0.13	0.894	-.0618577	.0539843
__00008L	-.0404095	.049328	-0.82	0.413	-.137116	.0562971
__00008M	.001021	.007579	0.13	0.893	-.0138375	.0158796
__00008N	.0131892	.0193792	0.68	0.496	-.0248033	.0511816
__00008O	.0114961	.0502268	0.23	0.819	-.0869726	.1099647
__00008P	.0192111	.0125691	1.53	0.126	-.0054304	.0438526
__00008Q	-.0325345	.019847	-1.64	0.101	-.0714441	.0063751
__00008R	-.0096409	.009686	-1.00	0.320	-.02863	.0093483
__00008S	-.0001567	.0059184	-0.03	0.979	-.0117597	.0114462
__00008T	-.0001145	.0042702	-0.03	0.979	-.0084862	.0082571
__00008U	-.006079	.0052961	-1.15	0.251	-.0164619	.004304
__00008V	.0015051	.0067101	0.22	0.823	-.01165	.0146601
__00008W	-.0659617	.0806505	-0.82	0.413	-.2240755	.0921521
__00008X	-.0247029	.0120841	-2.04	0.041	-.0483934	-.0010123
__00008Y	.0580503	.0318451	1.82	0.068	-.0043813	.120482
__00008Z	.0110772	.0756801	0.15	0.884	-.1372921	.1594464
__000090	.0032114	.0148403	0.22	0.829	-.0258827	.0323056
__000091	-.0007712	.0309768	-0.02	0.980	-.0615007	.0599582
__000092	.0003623	.0134755	0.03	0.979	-.0260561	.0267807
__000093	.0312665	.0084329	3.71	0.000	.014734	.0477989
__000094	.0012232	.0065024	0.19	0.851	-.0115247	.0139711
__000095	.0022841	.0079556	0.29	0.774	-.0133128	.0178809
__000096	.006253	.0097397	0.64	0.521	-.0128414	.0253474
__000097	-.068417	.0665222	-1.03	0.304	-.1988324	.0619984
__000098	-.0321976	.011108	-2.90	0.004	-.0539746	-.0104206
__000099	.0769573	.0278348	2.76	0.006	.0223876	.131527
__00009A	-.0094448	.0631059	-0.15	0.881	-.1331627	.114273
__00009B	.006836	.0159187	0.43	0.668	-.0243723	.0380444
__00009C	.0466064	.024625	1.89	0.058	-.0016705	.0948832
__00009D	.0096626	.0105985	0.91	0.362	-.0111155	.0304408
__00009E	.0081096	.008809	0.92	0.357	-.0091603	.0253795
__00009F	-.0027182	.0052673	-0.52	0.606	-.0130445	.0076082
__00009G	.0040504	.0076259	0.53	0.595	-.0109	.0190007
__00009H	-.0011467	.0091941	-0.12	0.901	-.0191714	.0168781
__00009I	-.0238561	.0430183	-0.55	0.579	-.1081925	.0604803
__00009J	.0035423	.0080495	0.44	0.660	-.0122386	.0193232
__00009K	-.0070524	.0176384	-0.40	0.689	-.041632	.0275273
__00009L	.0517459	.0454622	1.14	0.255	-.0373818	.1408736
__00009M	-.0200109	.0121946	-1.64	0.101	-.0439182	.0038963
__00009N	-.0079232	.0196472	-0.40	0.687	-.0464411	.0305947

__00009O	.0056391	.007526	0.75	0.454	-.0091156	.0203937
__00009P	-.0080001	.0067477	-1.19	0.236	-.0212288	.0052286
__00009Q	.0016867	.0046181	0.37	0.715	-.007367	.0107403
__00009R	.007372	.0052378	1.41	0.159	-.0028967	.0176407
__00009S	.0093889	.0071346	1.32	0.188	-.0045983	.0233761
__00009T	.1166889	.0696247	1.68	0.094	-.0198089	.2531868
__00009U	.0353334	.0105196	3.36	0.001	.01471	.0559568
__00009V	-.0820606	.026895	-3.05	0.002	-.1347877	-.0293335
__00009W	-.0802119	.0698383	-1.15	0.251	-.2171284	.0567047
__00009X	-.0381151	.0170619	-2.23	0.026	-.0715646	-.0046656
__00009Y	-.0208003	.0284381	-0.73	0.465	-.0765526	.0349519
__00009Z	.0278719	.0105935	2.63	0.009	.0071036	.0486402
__0000A0	-.0219439	.007529	-2.91	0.004	-.0367044	-.0071834
__0000A1	.0001324	.0063097	0.02	0.983	-.0122376	.0125025
__0000A2	-.0053828	.0076607	-0.70	0.482	-.0204014	.0096359
__0000A3	-.0034739	.0096997	-0.36	0.720	-.0224899	.0155421
__0000A5	6.773098	7.375606	0.92	0.359	-7.686631	21.23283
__0000A7	-1.782925	1.046319	-1.70	0.088	-3.834213	.2683628
__0000A9	-7.572242	3.852731	-1.97	0.049	-15.12544	-.0190393
__0000AC	.0840278	.1516048	0.55	0.579	-.2131904	.381246
__0000AE	.0179718	.0390401	0.46	0.645	-.0585656	.0945092
__0000AG	-.0798162	.1044962	-0.76	0.445	-.2846789	.1250465
__0000AJ	-1.356543	1.47459	-0.92	0.358	-4.247447	1.53436
__0000AL	.2464558	.2129111	1.16	0.247	-.1709522	.6638638
__0000AN	1.374378	.711347	1.93	0.053	-.0202036	2.76896
__0000AQ	-8.792693	9.065879	-0.97	0.332	-26.56617	8.980781
__0000AS	2.53114	1.321734	1.92	0.056	-.0600921	5.122373
__0000AU	10.13155	4.983802	2.03	0.042	.360909	19.9022
__0000AX	.5976473	.268429	2.23	0.026	.0713975	1.123897
__0000AZ	-.1952569	.0664728	-2.94	0.003	-.3255754	-.0649383
__0000B1	-.3015137	.1654969	-1.82	0.069	-.6259669	.0229396
__0000B4	-.4059903	.3144514	-1.29	0.197	-1.022466	.2104855
__0000B6	-.1248952	.1113195	-1.12	0.262	-.3431349	.0933444
__0000B8	-.5371562	.1873321	-2.87	0.004	-.9044171	-.1698953
__0000BB	.2472246	.1393013	1.77	0.076	-.0258729	.520322
__0000BD	.0440087	.0413466	1.06	0.287	-.0370506	.1250679
__0000BF	-.2197992	.0898567	-2.45	0.014	-.3959615	-.0436369
__0000BI	.1604154	.0807685	1.99	0.047	.0020703	.3187604
__0000BK	-.027681	.027041	-1.02	0.306	-.0806943	.0253323
__0000BM	-.040153	.0378546	-1.06	0.289	-.1143663	.0340603
__0000BP	-.3622991	.0847784	-4.27	0.000	-.5285055	-.1960928
__0000BR	-.0345706	.0255464	-1.35	0.176	-.0846538	.0155125
__0000BT	.1507979	.0366801	4.11	0.000	.0788872	.2227085
__0000BW	.3318459	.1048489	3.16	0.002	.1262916	.5374002
__0000BY	-.025565	.0272362	-0.94	0.348	-.0789609	.027831
__0000C0	-.194259	.0630614	-3.08	0.002	-.3178897	-.0706283
__0000C3	.2079287	.1330361	1.56	0.118	-.0528859	.4687434
__0000C5	-.0638962	.0317678	-2.01	0.044	-.1261763	-.0016161
__0000C7	-.1780398	.0824397	-2.16	0.031	-.3396612	-.0164184
xbarlag	4.04385	5.56316	0.73	0.467	-6.862615	14.95031
x2barlag	-.0563375	.1029798	-0.55	0.584	-.2582273	.1455522
xbar2lag	-.4429217	1.123839	-0.39	0.694	-2.646186	1.760343
rootxbarlag	-5.830931	6.796825	-0.86	0.391	-19.15597	7.494109
z1barlag	-.0100134	.2133328	-0.05	0.963	-.428248	.4082213
z2barlag	1.164475	.3074819	3.79	0.000	.5616632	1.767288
z3barlag	-.0785922	.121137	-0.65	0.517	-.3160789	.1588945
z4barlag	-.0807978	.0768448	-1.05	0.293	-.2314504	.0698548
z5barlag	.2424491	.0850521	2.85	0.004	.075706	.4091921
z6barlag	-.0856014	.0913681	-0.94	0.349	-.2647267	.093524
z7barlag	.0440779	.1017553	0.43	0.665	-.1554113	.2435672
_cons	2.067702	2.311112	0.89	0.371	-2.463186	6.59859

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000D6	128,640	.0911297	.0205071	.0369996	.1767478

Linear regression

Number of obs	= **128,640**
F(148, 4598)	= .
Prob > F	= .
R-squared	= **0.2997**
Root MSE	= **.19143**

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar3	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__0000CA	19.72416	17.96959	1.10	0.272	-15.50486	54.95319
__0000CC	1.515332	1.224382	1.24	0.216	-.8850453	3.915709
__0000CE	4.474477	2.534192	1.77	0.078	-.4937567	9.442711
__0000CH	.0198028	2.952779	0.01	0.995	-5.769062	5.808667
__0000CJ	-.7064709	.8619186	-0.82	0.412	-2.396245	.9833034
__0000CM	-16.81897	9.484308	-1.77	0.076	-35.41276	1.774831
xi	.1307497	.0132927	9.84	0.000	.1046896	.1568099
__000002	-.0065885	.005017	-1.31	0.189	-.0164243	.0032473
__00000E	-.2662238	.1475316	-1.80	0.071	-.5554565	.023009
__00000P	-.0626842	.0438756	-1.43	0.153	-.1487014	.023333
__000010	-.0446399	.0615601	-0.73	0.468	-.1653273	.0760476
__00001I	.5580734	.3052427	1.83	0.068	-.0403487	1.156496
__00001T	.2032963	.0982546	2.07	0.039	.0106701	.3959226
__000024	.004304	.1170992	0.04	0.971	-.2252666	.2338747
__00002M	.1174028	.0735566	1.60	0.111	-.0268033	.261609
__00002X	.0405673	.0223973	1.81	0.070	-.0033422	.0844768
__000038	-.0238957	.0269501	-0.89	0.375	-.0767309	.0289394
__00003Q	.3655848	.0975183	3.75	0.000	.1744021	.5567675
__000041	.0186472	.036321	0.51	0.608	-.0525593	.0898538
__00004C	-.0762788	.0403286	-1.89	0.059	-.1553422	.0027846
__00004U	-.1390218	.0839474	-1.66	0.098	-.3035989	.0255554
__000055	-.0040062	.0261039	-0.15	0.878	-.0551823	.0471699
__00005G	.0629784	.0367464	1.71	0.087	-.0090621	.1350189
__00005Y	.0081864	.068496	0.12	0.905	-.1260985	.1424714
__000069	.0152629	.0209964	0.73	0.467	-.0259001	.056426
__00006K	-.0131241	.0281625	-0.47	0.641	-.0683362	.042088
__000072	.0282828	.1116791	0.25	0.800	-.1906619	.2472274
__00007D	.1054299	.0341831	3.08	0.002	.0384145	.1724453
__00007O	-.0334945	.040568	-0.83	0.409	-.1130272	.0460382
__00007Z	-1.100565	.6209016	-1.77	0.076	-2.31783	.1167004
__000080	-.0717327	.1080631	-0.66	0.507	-.2835883	.1401229
__000081	.296013	.250848	1.18	0.238	-.1957695	.7877955
__000082	1.180619	.6408524	1.84	0.066	-.0757589	2.436998
__000083	-.0639623	.1376454	-0.46	0.642	-.3338133	.2058887
__000084	.3954916	.2037375	1.94	0.052	-.0039317	.794915
__000085	-.1616121	.0933081	-1.73	0.083	-.3445408	.0213166
__000086	.0724187	.0552103	1.31	0.190	-.0358199	.1806574
__000087	-.0126977	.0413373	-0.31	0.759	-.0937387	.0683432
__000088	.0454524	.0587907	0.77	0.439	-.0698056	.1607104
__000089	-.047427	.071801	-0.66	0.509	-.1881914	.0933375
__00008A	5.205917	1.24617	4.18	0.000	2.762825	7.64901
__00008B	.5147718	.2236692	2.30	0.021	.0762727	.9532709
__00008C	-2.197021	.5136496	-4.28	0.000	-3.204021	-1.190021
__00008D	-3.957773	1.286192	-3.08	0.002	-6.479328	-1.436219
__00008E	-.3726791	.2844432	-1.31	0.190	-.9303243	.1849661
__00008F	-.3360271	.4582303	-0.73	0.463	-1.234378	.5623242
__00008G	-.0450734	.1716467	-0.26	0.793	-.3815833	.2914365
__00008H	-.6024521	.1154902	-5.22	0.000	-.8288684	-.3760358
__00008I	.0855242	.0870317	0.98	0.326	-.0850996	.2561481
__00008J	-.1278781	.139773	-0.91	0.360	-.4019003	.1461441
__00008K	.0330238	.1711934	0.19	0.847	-.3025976	.3686451
__00008L	.3999629	.3069919	1.30	0.193	-.2018887	1.001814
__00008M	.0071741	.0486119	0.15	0.883	-.0881286	.1024768
__00008N	-.1259232	.119961	-1.05	0.294	-.3611044	.1092579
__00008O	-.4172946	.3086467	-1.35	0.176	-1.02239	.187801
__00008P	.0405607	.0695364	0.58	0.560	-.0957641	.1768854
__00008Q	-.0789205	.1101536	-0.72	0.474	-.2948743	.1370333
__00008R	-.0130462	.048443	-0.27	0.788	-.1080177	.0819254
__00008S	.0322568	.0276893	1.16	0.244	-.0220275	.0865412
__00008T	.0232551	.0209743	1.11	0.268	-.0178646	.0643747
__00008U	-.040239	.0324168	-1.24	0.215	-.1037914	.0233135

__00008V	.0231589	.0382811	0.60	0.545	-.0518903	.0982081
__00008W	-.2745451	.5748501	-0.48	0.633	-1.401527	.852437
__00008X	-.0194482	.0688652	-0.28	0.778	-.1544572	.1155607
__00008Y	.2598803	.1996452	1.30	0.193	-.1315202	.6512808
__00008Z	-.4279608	.4822335	-0.89	0.375	-1.37337	.5174485
__000090	.1102851	.0937013	1.18	0.239	-.0734145	.2939847
__000091	.2280905	.1674791	1.36	0.173	-.1002489	.5564299
__000092	.0177484	.0651225	0.27	0.785	-.1099229	.1454197
__000093	.0775329	.0395368	1.96	0.050	.0000219	.1550439
__000094	-.0816406	.030453	-2.68	0.007	-.1413431	-.0219381
__000095	-.0624167	.0476672	-1.31	0.190	-.1558673	.0310339
__000096	-.0270314	.0610218	-0.44	0.658	-.1466634	.0926005
__000097	-.9576348	.3356529	-2.85	0.004	-1.615676	-.299594
__000098	-.097994	.059489	-1.65	0.100	-.2146211	.018633
__000099	.4264023	.1466334	2.91	0.004	.1389305	.7138741
__00009A	.6581535	.3378038	1.95	0.051	-.0041041	1.320411
__00009B	.1287284	.079011	1.63	0.103	-.0261711	.2836279
__00009C	.0653398	.1246325	0.52	0.600	-.1789998	.3096794
__00009D	-.0527494	.0537871	-0.98	0.327	-.1581979	.0526991
__00009E	.0434567	.0352713	1.23	0.218	-.0256919	.1126054
__00009F	.0434495	.0256297	1.70	0.090	-.0067971	.0936961
__00009G	.0366056	.0393242	0.93	0.352	-.0404888	.1136999
__00009H	.0200094	.0444559	0.45	0.653	-.0671454	.1071642
__00009I	.38181	.2874188	1.33	0.184	-.1816688	.9452888
__00009J	.0893858	.0475992	1.88	0.060	-.0039314	.182703
__00009K	-.2279116	.1181648	-1.93	0.054	-.4595713	.0037481
__00009L	-.2065209	.282176	-0.73	0.464	-.7597214	.3466795
__00009M	-.051354	.0621276	-0.83	0.409	-.1731539	.0704459
__00009N	-.0784311	.1055602	-0.74	0.458	-.2853798	.1285176
__00009O	.0225483	.0397456	0.57	0.571	-.0553721	.1004687
__00009P	-.0541108	.0297525	-1.82	0.069	-.1124399	.0042184
__00009Q	.0059482	.0204775	0.29	0.771	-.0341976	.0460939
__00009R	-.0007794	.0310464	-0.03	0.980	-.0616452	.0600863
__00009S	-.0091118	.0366086	-0.25	0.803	-.0808821	.0626586
__00009T	.2274097	.5242879	0.43	0.664	-.8004463	1.255266
__00009U	.1833137	.0700922	2.62	0.009	.0458993	.320728
__00009V	-.2983737	.1989009	-1.50	0.134	-.6883149	.0915676
__00009W	-.0473205	.5044688	-0.09	0.925	-1.036322	.9416806
__00009X	-.129888	.0969943	-1.34	0.181	-.3200434	.0602674
__00009Y	-.4228877	.1629702	-2.59	0.009	-.7423874	-.103388
__00009Z	.0492556	.0609166	0.81	0.419	-.0701702	.1686815
__0000A0	-.1776752	.0408015	-4.35	0.000	-.2576656	-.0976848
__0000A1	.0007761	.030305	0.03	0.980	-.0586363	.0601884
__0000A2	.0101742	.0500508	0.20	0.839	-.0879494	.1082978
__0000A3	.009322	.0577687	0.16	0.872	-.1039323	.1225763
__0000A5	48.30421	42.82146	1.13	0.259	-35.64641	132.2548
__0000A7	-.2093139	6.598813	-0.03	0.975	-13.14615	12.72753
__0000A9	-32.75008	20.81856	-1.57	0.116	-73.56445	8.064296
__0000AC	-.6845105	.8598735	-0.80	0.426	-2.370275	1.001254
__0000AE	.1974236	.1931704	1.02	0.307	-.1812831	.5761304
__0000AG	.6032529	.5427348	1.11	0.266	-.4607677	1.667274
__0000AJ	-7.565893	8.534276	-0.89	0.375	-24.29717	9.165386
__0000AL	-.0157107	1.240131	-0.01	0.990	-2.446962	2.415541
__0000AN	4.844677	3.802121	1.27	0.203	-2.609304	12.29866
__0000AQ	-62.6009	52.19937	-1.20	0.230	-164.9367	39.73493
__0000AS	-.0671822	8.391197	-0.01	0.994	-16.51796	16.38359
__0000AU	42.68588	26.79165	1.59	0.111	-9.838612	95.21037
__0000AX	-1.419298	1.275562	-1.11	0.266	-3.920012	1.081417
__0000AZ	-.6156358	.3230132	-1.91	0.057	-1.248897	.0176251
__0000B1	-.0913644	.7531515	-0.12	0.903	-1.567903	1.385174
__0000B4	-.4159367	1.717006	-0.24	0.809	-3.782093	2.95022
__0000B6	-.6452967	.4988886	-1.29	0.196	-1.623358	.3327645
__0000B8	-.7539992	.9232988	-0.82	0.414	-2.564108	1.05611
__0000BB	.0545919	.6476225	0.08	0.933	-1.215059	1.324243
__0000BD	.0355227	.1856828	0.19	0.848	-.3285047	.3995501
__0000BF	-.3028585	.3515291	-0.86	0.389	-.9920242	.3863072
__0000BI	.7494021	.4381024	1.71	0.087	-.1094888	1.608293
__0000BK	.0739261	.1494253	0.49	0.621	-.2190193	.3668715
__0000BM	-.0481613	.1582952	-0.30	0.761	-.3584958	.2621733
__0000BP	-.118559	.4278898	-0.28	0.782	-.9574284	.7203103
__0000BR	-.0700705	.1324916	-0.53	0.597	-.3298175	.1896766
__0000BT	-.2108873	.1574077	-1.34	0.180	-.5194818	.0977073

```
------------------------------------------------------------------------------
     __0000BW |   1.072662   .5617766     1.91   0.056    -.0286894    2.174014
     __0000BY |   .2178293   .1558139     1.40   0.162    -.0876408    .5232993
     __0000C0 |   .0250061   .2838408     0.09   0.930    -.5314582    .5814703
     __0000C3 |  -.7304097   .6755667    -1.08   0.280    -2.054845    .5940254
     __0000C5 |  -.1803978    .162804    -1.11   0.268    -.4995718    .1387761
     __0000C7 |   .2306071   .3540227     0.65   0.515    -.4634473    .9246615
      xbarlag |  -12.94394   31.17683    -0.42   0.678     -74.0655    48.17762
     x2barlag |  -.4140207   .6058787    -0.68   0.494    -1.601834    .7737923
     xbar2lag |   2.991847    6.51545     0.46   0.646    -9.781562    15.76526
  rootxbarlag |   16.20586   37.20812     0.44   0.663    -56.73992    89.15164
     z1barlag |   2.315195   1.074194     2.16   0.031     .2092591    4.421131
     z2barlag |   1.851417   1.614456     1.15   0.252    -1.313693    5.016526
     z3barlag |   .2882603   .5974322     0.48   0.629    -.8829935    1.459514
     z4barlag |  -.6319162   .4364033    -1.45   0.148    -1.487476    .2236437
     z5barlag |   .4220089   .4257129     0.99   0.322    -.4125928    1.256611
     z6barlag |  -1.353201   .5464895    -2.48   0.013    -2.424583   -.2818194
     z7barlag |   .6597472    .576698     1.14   0.253    -.4708577    1.790352
        _cons |  -6.590962    12.3033    -0.54   0.592    -30.71134    17.52942
------------------------------------------------------------------------------
(option xb assumed; fitted values)

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
     __0000DI |    128,640    .3629647    .1251629   .0785182   1.174995
```

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"

957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
  __0000CX __0000D0 __0000D3 __0000D9 __0000DC __0000DF __0000DL __0000DO __0000DR __
> 0000CY __0000D1 __0000D4 __0000DA __0000DD __0000DG __0000DM __0000DP __0000DS

960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
964 }

965
966 local r_RE      "`r_RE' `qhat_hats'"

967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
```

```
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }

978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                 g pzi`j' = p1*zi`j'
    3.          }
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = _b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                 local coef = _b[pzi`j']
    3.                 local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1025}
```

```
1026
1027if `J'==3 & simple_model==0
  >                           local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                           local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
```

```
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  1.1638934
  Iteration 1:  GMM criterion Q(b) =  .44023929
  Iteration 2:  GMM criterion Q(b) =  .02062725
  Iteration 3:  GMM criterion Q(b) =  .00053415
  Iteration 4:  GMM criterion Q(b) =  .00041835
  Iteration 5:  GMM criterion Q(b) =  1.819e-06

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .06098919
  Iteration 1:  GMM criterion Q(b) =  .03189499
  Iteration 2:  GMM criterion Q(b) =  .02004896
  Iteration 3:  GMM criterion Q(b) =  .01922349
  Iteration 4:  GMM criterion Q(b) =  .01905453
  Iteration 5:  GMM criterion Q(b) =  .01896644
  Iteration 6:  GMM criterion Q(b) =   .0189122

  GMM estimation

  Number of parameters =  34
  Number of moments    =  96
  Initial weight matrix: Identity                   Number of obs   =    128,640
  GMM weight matrix:     Robust

                (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
A11          |
         one |   .8477685    .023085    36.72   0.000     .8025227    .8930143
-------------+----------------------------------------------------------------
A22          |
         one |   .9384791   .0184351    50.91   0.000     .9023469    .9746113
-------------+----------------------------------------------------------------
A33          |
         one |   .7402855    .023226    31.87   0.000     .6947634    .7858077
-------------+----------------------------------------------------------------
C1           |
         zi1 |   .4128517   .0286691    14.40   0.000     .3566613    .4690422
         zi2 |  -.1808429   .0352966    -5.12   0.000     -.250023   -.1116629
         zi3 |  -.0427246   .0119281    -3.58   0.000    -.0661034   -.0193459
         zi4 |   .0004484   .0131141     0.03   0.973    -.0252547    .0261516
         zi5 |  -.0258895   .0190792    -1.36   0.175    -.0632841     .011505
         zi6 |  -.0194077   .0125219    -1.55   0.121    -.0439501    .0051346
         zi7 |   .0203808   .0148585     1.37   0.170    -.0087413    .0495029
-------------+----------------------------------------------------------------
C2           |
         zi1 |   .0518003   .0051028    10.15   0.000      .041799    .0618016
         zi2 |  -.0278795   .0055637    -5.01   0.000     -.038784   -.0169749
         zi3 |   -.007754   .0021976    -3.53   0.000    -.0120613   -.0034466
         zi4 |   .0010638   .0022372     0.48   0.634    -.0033209    .0054486
         zi5 |  -.0042634   .0030058    -1.42   0.156    -.0101546    .0016278
         zi6 |  -.0008804   .0020448    -0.43   0.667    -.0048881    .0031273
         zi7 |   .0056357   .0024184     2.33   0.020     .0008957    .0103757
-------------+----------------------------------------------------------------
C3           |
         zi1 |    .127602    .044214     2.89   0.004      .040944    .2142599
         zi2 |  -.3627695   .0578225    -6.27   0.000    -.4760996   -.2494394
         zi3 |  -.0613373   .0170221    -3.60   0.000    -.0946999   -.0279747
         zi4 |  -.0034274   .0244492    -0.14   0.889    -.0513469    .0444921
```

zi5	-.0053927	.0266282	-0.20	0.840	-.0575831	.0467977
zi6	-.0432959	.0179346	-2.41	0.016	-.0784471	-.0081448
zi7	.0268131	.0224152	1.20	0.232	-.0171198	.070746
/d1	-.0699983	.0057969	-12.08	0.000	-.0813601	-.0586366
/b1	.3857831	.0065288	59.09	0.000	.3729868	.3985794
/b2	.0729753	.0022274	32.76	0.000	.0686096	.077341
/AVA11	.387771	.959711	0.40	0.686	-1.493228	2.26877
/AVA12	-.0063663	.2887487	-0.02	0.982	-.5723032	.5595707
/AVA13	-.3952161	.6755914	-0.58	0.559	-1.719351	.9289187
/AVA22	-.5051306	.1672903	-3.02	0.003	-.8330137	-.1772476
/AVA23	.3894893	.1656212	2.35	0.019	.0648776	.7141009
/AVA33	-.7965634	.4917645	-1.62	0.105	-1.760404	.1672774
/d2	-.0149335	.001607	-9.29	0.000	-.0180831	-.0117838

Instruments for equation **eq1_RE:** **__0000CA __0000CC __0000CE __0000CH __0000CJ __0000CM xi __000002 __00000E __00000P __000010 __00001I __00001T __000024 __00002M __00002X __000038 __00003Q __000041 __00004C __00004U __000055 __00005G __00005Y __000069 __00006K __000072 __00007D __00007O __0000CX __0000D0 __0000D3 __0000D9 __0000DC __0000DF __0000DL __0000DO __0000DR __0000CY __0000D1 __0000D4 __0000DA __0000DD __0000DG __0000DM __0000DP __0000DS _cons**

Instruments for equation **eq2_RE:** **__0000CA __0000CC __0000CE __0000CH __0000CJ __0000CM xi __000002 __00000E __00000P __000010 __00001I __00001T __000024 __00002M __00002X __000038 __00003Q __000041 __00004C __00004U __000055 __00005G __00005Y __000069 __00006K __000072 __00007D __00007O __0000CX __0000D0 __0000D3 __0000D9 __0000DC __0000DF __0000DL __0000DO __0000DR __0000CY __0000D1 __0000D4 __0000DA __0000DD __0000DG __0000DM __0000DP __0000DS _cons**

1065}

```
1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_3good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3_all_exactIns
      > t_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not found)
  (output written to diag_3good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
              e(N_hh) =  24757
```

```
1077unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640

1078estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  4599

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
         e(avg_grp_size) =  5.3831268

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_3good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster not found)
  file diag_3good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
```

```
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
1119                                                       gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                  local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                          }
   30.                                  }
   31.                          }
   32.                  }
   33.          }
1120        if `Tindivp1'<=`T' {
1121                forval t=`Tindivp1'/`T' {
    2.                          tempvar zi`t'xdiff
    3.                          g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                          local zg "`zg' zi`t'"
    5.                          forval j=1/`J' {
    6.                                  tempvar zi`t'p`j'xdiff
    7.                                  g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                  local pzg "`pzg' `zi`t'p`j''"
    9.                                  local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                          }
   11.                  }
1122        }
1123
```

```
1124        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1125        // denoting zg as group-zs
1126        //  zg,pzgx,rprp,rprpx
1127        local rootprootp ""
1128        local rootprootpxdiff ""
1129
1130        forval j=1/`J' {
    2.                  forval s=`j'/`J' {
    3.                          tempvar xdiffrp`j'rp`s'
    4.                          g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1131                        if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1132                        local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                  }
    8.          }
1133
1134        local pqhatxdiff ""
1135        forval j=1/`J' {
    2.                  forval s=1/`J' {
    3.                          foreach name of global Alist {
    4.                                  tempvar   p`s'qikhat`j'xdiff`name'
    5.                                  g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
1136                                local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
1137
1138        // set instruments
1139        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1140}

1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
```

```
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                          local `++t'
    3.                          local p_nice = subinstr("`p'",":_cons","",.)
    4.                          local est = coefs[1,`t']
    5.                          if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                  }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                 }
   16.         }
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
1188
```

```
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1     `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
```

```
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00064867
  Iteration 1:  GMM criterion Q(b) =  .00022306
  Iteration 2:  GMM criterion Q(b) =  .00007815
  Iteration 3:  GMM criterion Q(b) =  8.400e-06

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .05561585
  Iteration 1:  GMM criterion Q(b) =  .03820004
  Iteration 2:  GMM criterion Q(b) =  .03488044
  Iteration 3:  GMM criterion Q(b) =  .03438514
  Iteration 4:  GMM criterion Q(b) =   .0343208
  Iteration 5:  GMM criterion Q(b) =  .03430441

  GMM estimation

  Number of parameters =  28
  Number of moments    = 234
  Initial weight matrix: Identity                   Number of obs   =    128,640
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
C11          |
       _cons |   .5550712   .0273816    20.27   0.000     .5014043    .6087381
-------------+----------------------------------------------------------------
C12          |
       _cons |   .0232233   .0402007     0.58   0.563    -.0555686    .1020152
-------------+----------------------------------------------------------------
C13          |
       _cons |   .0062723   .0096671     0.65   0.516    -.0126747    .0252194
-------------+----------------------------------------------------------------
C14          |
       _cons |   -.017762   .0105596    -1.68   0.093    -.0384585    .0029346
-------------+----------------------------------------------------------------
C15          |
       _cons |  -.0389831   .0167745    -2.32   0.020    -.0718605   -.0061057
-------------+----------------------------------------------------------------
C16          |
       _cons |  -.0247941   .0110353    -2.25   0.025     -.046423   -.0031652
-------------+----------------------------------------------------------------
C17          |
       _cons |  -.0038763   .0138382    -0.28   0.779    -.0309986     .023246
-------------+----------------------------------------------------------------
C21          |
       _cons |   .0722936   .0046297    15.62   0.000     .0632196    .0813676
-------------+----------------------------------------------------------------
C22          |
       _cons |   .0160534   .0063717     2.52   0.012     .0035652    .0285416
-------------+----------------------------------------------------------------
C23          |
       _cons |   .0022983   .0015748     1.46   0.144    -.0007882    .0053848
-------------+----------------------------------------------------------------
C24          |
       _cons |  -.0048485   .0017178    -2.82   0.005    -.0082153   -.0014817
-------------+----------------------------------------------------------------
C25          |
       _cons |  -.0068386   .0026181    -2.61   0.009    -.0119699   -.0017073
-------------+----------------------------------------------------------------
C26          |
```

_cons	-.0010482	.0017436	-0.60	0.548	-.0044656	.0023692
C27						
_cons	.0036896	.0020574	1.79	0.073	-.0003428	.0077221
C31						
_cons	.3134851	.0352502	8.89	0.000	.244396	.3825743
C32						
_cons	-.0174504	.054314	-0.32	0.748	-.1239039	.089003
C33						
_cons	.0095049	.0123296	0.77	0.441	-.0146607	.0336706
C34						
_cons	-.0437524	.0174917	-2.50	0.012	-.0780355	-.0094693
C35						
_cons	-.0208093	.0206122	-1.01	0.313	-.0612086	.0195899
C36						
_cons	-.0364778	.0137081	-2.66	0.008	-.0633452	-.0096104
C37						
_cons	.0036297	.0182183	0.20	0.842	-.0320775	.039337
d1						
_cons	-.0808603	.0056923	-14.21	0.000	-.0920169	-.0697037
b1						
_cons	.3261876	.0194966	16.73	0.000	.287975	.3644002
b2						
_cons	.061592	.0030052	20.50	0.000	.055702	.0674821
A11						
one	2.393436	.4256053	5.62	0.000	1.559265	3.227607
A22						
one	2.819832	.9127232	3.09	0.002	1.030927	4.608736
A33						
one	-1.386578	.3340923	-4.15	0.000	-2.041387	-.7317692
/d2	-.0114743	.0021564	-5.32	0.000	-.0157008	-.0072478

Instruments for equation **eq1_FE**: __000004 __00000H __00000S __000013 __00001L
__00001W __000027 __00002P __000030 __00003B __00003T __000044 __00004F
__00004X __000058 __00005J __000061 __00006C __00006N __000075 __00007G
__00007R __000005 __0000JJ __0000JK __0000JL __0000JM __0000JN __0000JO
__0000JP __0000JQ __0000JR __0000E2 __0000E4 __0000E6 __0000EB __0000ED
__0000EF __0000EK __0000EM __0000EO __0000EU __0000EW __0000EY __0000F3
__0000F5 __0000F7 __0000FC __0000FE __0000FG __0000FM __0000FO __0000FQ
__0000FV __0000FX __0000FZ __0000G4 __0000G6 __0000G8 __0000GE __0000GG
__0000GI __0000GN __0000GP __0000GR __0000GW __0000GY __0000H0 __0000H6
__0000H8 __0000HA __0000HF __0000HH __0000HJ __0000HO __0000HQ __0000HS
__0000HY __0000I0 __0000I2 __0000I7 __0000I9 __0000IB __0000IG __0000II
__0000IK __0000IQ __0000IS __0000IU __0000IZ __0000J1 __0000J3 __0000J8
__0000JA __0000JC __0000DZ __0000E8 __0000EH __0000ER __0000F0 __0000F9
__0000FJ __0000FS __0000G1 __0000GB __0000GK __0000GT __0000H3 __0000HC
__0000HL __0000HV __0000I4 __0000ID __0000IN __0000IW __0000J5 _cons
Instruments for equation **eq2_FE**: __000004 __00000H __00000S __000013 __00001L
__00001W __000027 __00002P __000030 __00003B __00003T __000044 __00004F
__00004X __000058 __00005J __000061 __00006C __00006N __000075 __00007G
__00007R __000005 __0000JJ __0000JK __0000JL __0000JM __0000JN __0000JO
__0000JP __0000JQ __0000JR __0000E2 __0000E4 __0000E6 __0000EB __0000ED
__0000EF __0000EK __0000EM __0000EO __0000EU __0000EW __0000EY __0000F3
__0000F5 __0000F7 __0000FC __0000FE __0000FG __0000FM __0000FO __0000FQ
__0000FV __0000FX __0000FZ __0000G4 __0000G6 __0000G8 __0000GE __0000GG
__0000GI __0000GN __0000GP __0000GR __0000GW __0000GY __0000H0 __0000H6
__0000H8 __0000HA __0000HF __0000HH __0000HJ __0000HO __0000HQ __0000HS
__0000HY __0000I0 __0000I2 __0000I7 __0000I9 __0000IB __0000IG __0000II

```
      __0000IK __0000IQ __0000IS __0000IU __0000IZ __0000J1 __0000J3 __0000J8
      __0000JA __0000JC __0000DZ __0000E8 __0000EH __0000ER __0000F0 __0000F9
      __0000FJ __0000FS __0000G1 __0000GB __0000GK __0000GT __0000H3 __0000HC
      __0000HL __0000HV __0000I4 __0000ID __0000IN __0000IW __0000J5 _cons
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
  (file
      diag_3good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3_all_exactIns
      > t_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex not found)
  (output written to diag_3good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex)
1242
1243        estat overid

    Test of overidentifying restriction:

    Hansen's J chi2(206) = 4412.92 (p = 0.0000)
1244        di "$S_DATE $S_TIME"
   8 Dec 2023 10:57:46
1245        unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  24757
1248        unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640
1249        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4599
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  5.3831268
1251        estimates save "$filename_FE", replace
  (note: file diag_3good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster not found)
  file diag_3good_norm_main_nopcross_state_Adiag_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
```

```
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}

1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table7_13.smcl
   log type:  smcl
  closed on:   8 Dec 2023, 10:57:46
```

A.7.2 Columns 2 and 4

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table7_24.smcl
  log type:  smcl
 opened on:   9 Dec 2023, 16:43:07

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
  C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=3

47. local J = J

48.
49. // group def options
50. local group_def "dist"

51. local group_def "fsuXseg"

52. local group_def "fsuXsegXreligXsched"

53.
54. // comparision group
55. local groupComp_def ""

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.         syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
70.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)        expend_type   ->   (dropped)
  xij variables:
                          laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)        expend_type   ->   (dropped)
  xij variables:
                          laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations               840   ->   280
  Number of variables                    5   ->   6
  j variable (3 values)        expend_type   ->   (dropped)
  xij variables:
                          laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.             }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
  (197,188 real changes made)
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "

116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
```

```
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,042 observations deleted)

159 drop if hhsize>12
  (656 observations deleted)

160 gen married=marstat==2

161 drop if age<20
  (1,182 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

168
169 g lux=ilux+vlux
```

```
170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,380 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,637 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)
```

```
195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                    egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,380 missing values generated)
224         g land=exp(demog23)
  (12,776 missing values generated)
225         replace z4=0 if land<=0.005
  (29,502 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
227         g z5=demog24
  (29 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
```

```
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }
297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }
316
317 if short_zlist==6 {
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }
333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
```

```
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }
```

```
384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi
  (82 missing values generated)

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,576 real changes made, 82 to missing)

460 replace x2=x*x
  (73,576 real changes made, 82 to missing)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,576 real changes made)

478 replace x2=x2_temp
  (73,576 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (48,819 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)
```

```
488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_3ffc_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
```

```
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk

530
531 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,757 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
549 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var'" _n
    5.                 }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
```

```
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1

601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1
```

```
625 scalar estimate_FE=1

626
627 // estimation options
628 global w_initial "winit(unadjusted)"

629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

630 global wmatrix "wmatrix(robust)"

631 global trace_level ""

632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

633 global technique ""

634 global maxiter ""

635 global technique_RE ""

636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

637 global derivatives "quickderivatives"

638 scalar init_FE_from_RE=1

639 global clust_var "state_district_round"

640 local clus_name "_clus_Dist"

641
642
643
644
645
646
647
648
649 // xxzx to keep names same
650 local wmatrix_desc ""

651 local sectorName "_urbOnly"

652 local dropZdesc "_dropPre"

653 local desc_struc_v0 ""

654 local inst_desc "_exactInst"

655 local inst_rootp "_noPPXdInst"

656 local weight_desc ""

657 local RE_sq_inst_desc ""

658 local cnstname ""

659 local actual_pqhat_desc ""

660 local expUpname "_updateExp"
```

```
661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //              ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T

694 local Tm1=`T'-1
```

```
695 local Tindiv : word count $zlist_indiv

696
697 /*****************************************************************************
  > Set weights
  > *****************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*****************************************************************************
  >  Update data for J=2, and J=3
  > *****************************************************************************/
706
707 if `J'==2 {
708         replace p1=P1
709         replace p2=P2
710         replace qi1=Qi1
711         replace qk1=Qk1
712         replace qikbar1=Qikbar1
713         replace qi2=Qi2
714         replace qk2=Qk2
715         replace qikbar2=Qikbar2
716
717         foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                 qui replace `var'=.
    3.         }
718 }

719
720 if `J'==3 {
721         forv j = 1(1)3 {
    2.                 replace p`j'=P`j'
    3.                 replace qi`j'=Qi`j'
    4.                 replace qk`j'=Qk`j'
    5.                 replace qikbar`j'=Qikbar`j'
    6.         }
  (128,640 real changes made)
  (128,632 real changes made)
  (128,632 real changes made)
  (128,640 real changes made)
  (0 real changes made)
  (0 real changes made)
  (0 real changes made)
  (0 real changes made)
  (128,640 real changes made)
  (128,637 real changes made)
  (128,637 real changes made)
  (128,640 real changes made)
722
723         foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                 qui replace `var'=.
    3.         }
```

```
724 }

725
726
727 /*****************************************************************************
  > Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }
```

```
751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"

759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
773         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
```

```
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j'"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }

798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"

805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.         local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.         if `jplus1'<=`J' {
    5.                 forval k=`jplus1'/`J' {
    6.                         local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                 }
    8.         }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                             local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                 if noMeasError==1                       local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'              - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.         }
```

```
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                      l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                    ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A11:}*p2*qk2) + ({A11:}*p3*qk3) )*( ({A11:
  > }*p1*qikbar1) + ({A11:}*p2*qikbar2) + ({A11:}*p3*qikbar3) )   - 2*(xi - ( (p1*{C1: z
  > i1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p3*{C3: zi1
  >  zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) +
  >  ({A11:}*p3*qikbar3) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )*
  > (xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p3)
  >  + {b2}*ln(p2/p3) + ln(p3) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0
  >  ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) + ({A11:}*p3*qikbar3) ))*{b1} + ({A
  > 11:}*p1*qikbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + 2*p1*
  > {AVA13}*p3 + p2*{AVA22}*p2 + 2*p2*{AVA23}*p3 + p3*{AVA33}*p3 )*{d1}/exp( {b1}*ln(p1/
  > p3) + {b2}*ln(p2/p3) + ln(p3) ) ) ) (eq2_RE: (p2*qi2 - ((( ({A11:}*p1*qk1) + ({A11
  > :}*p2*qk2) + ({A11:}*p3*qk3) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) + ({A11:}
  > *p3*qikbar3) )        - 2*(xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )*(
  >  ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) + ({A11:}*p3*qikbar3) ) + (xi - ( (p1*{C1
  > :}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:
  > }) ) - ( 0 ) )) * {d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3) ) + ((xi - ( (p
  > 1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qi
  > kbar2) + ({A11:}*p3*qikbar3) ) )*{b2} + ({A11:}*p2*qikbar2) + (p2*{C2:}) + ( 0 ) - (
  > 0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + 2*p1*{AVA13}*p3 + p2*{AVA22}*p2 + 2*p2*{AVA23}
  > *p3 + p3*{AVA33}*p3 )*{d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) + ( p3*{C31}*zi1 + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*
  > {C36}*zi6 + p3*{C37}*zi7 ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*z
  > k3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 +
  > p2*{C22}*zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*
  > zk7 ) + ( p3*{C31}*zk1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 +
  >  p3*{C36}*zk6 + p3*{C37}*zk7 ) ))^2) * {d1}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + l
  > n(p3) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*
  > {C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}
  > *zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) + ( p3*{C31}*zi1
  > + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*{C36}*zi6 + p3*{C37
  > }*zi7 ) )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p
  > 1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C2
  > 3}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) + ( p3*{C31}*zk
```

```
  > 1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 + p3*{C36}*zk6 + p3*{C
  > 37}*zk7 ) )))*({b1} - 2*(( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) + ({A11:}*p3*qi
  > kbar3) ) + ( ( 0 ) + ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/
  > p3) + ln(p3) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*
  > {C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}
  > *zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         )) (eq2_
  > FE: ((p2*qi2 - p2*qk2) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*
  > {C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C22}
  > *zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) +
  > ( p3*{C31}*zi1 + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*{C36
  > }*zi6 + p3*{C37}*zi7 ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 +
  >  p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{
  > C22}*zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7
  > ) + ( p3*{C31}*zk1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 + p3*
  > {C36}*zk6 + p3*{C37}*zk7 ) ))^2) * {d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3
  > ) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{C15
  > }*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}*zi3
  >  + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) + ( p3*{C31}*zi1 + p3
  > *{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*{C36}*zi6 + p3*{C37}*zi
  > 7 ) )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) + ( p3*{C31}*zk1 +
  > p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 + p3*{C36}*zk6 + p3*{C37}*
  > zk7 ) )))*({b2} - 2*(( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) + ({A11:}*p3*qikbar
  > 3) ) + ( ( 0 ) + ( 0 ) + ( 0 ) ) + ( 0 ))*{d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3)
  > + ln(p3) )) - (( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25
  > }*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) - ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3
  >  + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ))         ))
```

```
836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2           "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
```

```
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                              "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}     `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                          "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qk1) + ({A11:}*p2*qk2) + ({A11:}*p3*qk3) )*( ({A
  > 11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) + ({A11:}*p3*qikbar3) )   - 2*(xi - ( (p1*{C1
  > : zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p3*{C3:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2
  > ) + ({A11:}*p3*qikbar3) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 )
  >  )*(xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/
  > p3) + {b2}*ln(p2/p3) + ln(p3) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) -
  > ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) + ({A11:}*p3*qikbar3) ))*{b1} +
  > ({A11:}*p1*qikbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + 2*
  > p1*{AVA13}*p3 + p2*{AVA22}*p2 + 2*p2*{AVA23}*p3 + p3*{AVA33}*p3 )*{d1}/exp( {b1}*ln(
  > p1/p3) + {b2}*ln(p2/p3) + ln(p3) ) ) ) (eq2_RE: (p2*qi2 - ((( ({A11:}*p1*qk1) + ({
  > A11:}*p2*qk2) + ({A11:}*p3*qk3) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) + ({A1
  > 1:}*p3*qikbar3) )        - 2*(xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 )
  > )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) + ({A11:}*p3*qikbar3) ) + (xi - ( (p1*
  > {C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:}) + (p3*{
  > C3:}) ) - ( 0 ) )) * {d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3) ) + ((xi - (
  >  (p1*{C1:}) + (p2*{C2:}) + (p3*{C3:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2
  > *qikbar2) + ({A11:}*p3*qikbar3) ) )*{b2} + ({A11:}*p2*qikbar2) + (p2*{C2:}) + ( 0 )
  > - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + 2*p1*{AVA13}*p3 + p2*{AVA22}*p2 + 2*p2*{AVA
  > 23}*p3 + p3*{AVA33}*p3 )*{d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln(p3) ) ) )
```

```
873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) + ( p3*{C31}*zi1 + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*
  > {C36}*zi6 + p3*{C37}*zi7 ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*z
  > k3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 +
  > p2*{C22}*zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*
  > zk7 ) + ( p3*{C31}*zk1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 +
  >  p3*{C36}*zk6 + p3*{C37}*zk7 ) ))^2) * {d1}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + l
  > n(p3) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*
  > {C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}
  > *zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) + ( p3*{C31}*zi1
  > + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*{C36}*zi6 + p3*{C37
  > }*zi7 ) )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p
  > 1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C2
  > 3}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) + ( p3*{C31}*zk
  > 1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 + p3*{C36}*zk6 + p3*{C
  > 37}*zk7 ) )))*({b1} - 2*(( ({A11:one}*p1*qikbar1) + ({A11:}*p2*qikbar2) + ({A11:}*p3
  > *qikbar3) ) + ( ( 0 ) + ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln(p1/p3) + {b2}*ln(
  > p2/p3) + ln(p3) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  > p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C
  > 13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))       )) (e
  > q2_FE: ((p2*qi2 - p2*qk2) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  > p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C
  > 22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  >  + ( p3*{C31}*zi1 + p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*{
  > C36}*zi6 + p3*{C37}*zi7 ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk
  > 3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p
  > 2*{C22}*zk2 + p2*{C23}*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*z
  > k7 ) + ( p3*{C31}*zk1 + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 +
  > p3*{C36}*zk6 + p3*{C37}*zk7 ) ))^2) * {d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p3) + ln
  > (p3) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 + p1*{
  > C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}*
  > zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) + ( p3*{C31}*zi1 +
  >  p3*{C32}*zi2 + p3*{C33}*zi3 + p3*{C34}*zi4 + p3*{C35}*zi5 + p3*{C36}*zi6 + p3*{C37}
  > *zi7 ) )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1
  > *{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23
  > }*zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) + ( p3*{C31}*zk1
  >  + p3*{C32}*zk2 + p3*{C33}*zk3 + p3*{C34}*zk4 + p3*{C35}*zk5 + p3*{C36}*zk6 + p3*{C3
  > 7}*zk7 ) )))*({b2} - 2*(( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) + ({A11:}*p3*qik
  > bar3) ) + ( ( 0 ) + ( 0 ) + ( 0 ) ) + ( 0 ))*{d2}/exp( {b1}*ln(p1/p3) + {b2}*ln(p2/p
  > 3) + ln(p3) )) - (( p2*{C21}*zi1 + p2*{C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{
  > C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 ) - ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*
  > zk3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ))         ))

874
875 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi
```

```
882 g double `ximxk'=xi-xk

883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.         tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.         g double `xizi`t''=xi*zi`t'
    4.         g double `xkzk`t''=xk*zk`t'
    5.         g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.         g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.         g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.         g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.         g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.         forval j=1/`J' {
   11.                 tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                 g double `zi`t'p`j''= zi`t'*p`j'
   13.                 g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                 g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                 g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                 forv s=1/`T' {
   17.                         tempvar zi`t'zi`s'p`j'
   18.                         g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                 }
   20.         }
   21. }

891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                 local rg "`rg' z`t'barlag"
    3. }

903
904 local zrg ""
```

```
905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var'"
    6.          }
    7. }

906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var'"
    5.
911         forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var'"
   10.                  local xprg "`xprg' `xp`j'`var'"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"

917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""
```

```
925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }

929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.          forval k=`j'/`J' {
    3.                  tempvar p`j'p`k'
    4.                  g `p`j'p`k''=p`j'*p`k'
    5.                  local roundinsts "`roundinsts' `p`j'p`k''"
    6.          }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"

938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 29

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.          if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.          if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
```

```
951         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
952         // use all prices times all qikhats as instruments, interacted with xi
953         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }
```

Linear regression

Number of obs	= **128,640**
F(148, 4598)	= .
Prob > F	= .
R-squared	= **0.2305**
Root MSE	= **.13046**

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__0000CA	**15.8624**	**12.19746**	**1.30**	**0.194**	**-8.050478**	**39.77528**
__0000CC	**.2763886**	**.8341895**	**0.33**	**0.740**	**-1.359023**	**1.9118**
__0000CE	**-1.458342**	**2.051914**	**-0.71**	**0.477**	**-5.481078**	**2.564395**
__0000CH	**-.261114**	**1.983154**	**-0.13**	**0.895**	**-4.149048**	**3.62682**
__0000CJ	**-1.4065**	**.6557057**	**-2.15**	**0.032**	**-2.691998**	**-.1210025**
__0000CM	**-9.797743**	**7.400675**	**-1.32**	**0.186**	**-24.30662**	**4.711132**
xi	**.1067121**	**.007683**	**13.89**	**0.000**	**.0916498**	**.1217744**
__000002	**-.0150922**	**.0024866**	**-6.07**	**0.000**	**-.0199672**	**-.0102172**
__00000E	**-.1979284**	**.1017616**	**-1.95**	**0.052**	**-.39743**	**.0015733**
__00000P	**-.0186921**	**.0303069**	**-0.62**	**0.537**	**-.0781082**	**.040724**
__000010	**.038788**	**.0475737**	**0.82**	**0.415**	**-.0544792**	**.1320553**
__00001I	**.0878379**	**.2069769**	**0.42**	**0.671**	**-.3179361**	**.4936119**
__00001T	**.0026737**	**.0645317**	**0.04**	**0.967**	**-.1238394**	**.1291868**
__000024	**-.0417135**	**.0917764**	**-0.45**	**0.649**	**-.2216393**	**.1382123**
__00002M	**.0245382**	**.0517151**	**0.47**	**0.635**	**-.0768482**	**.1259245**
__00002X	**-.0151116**	**.0145696**	**-1.04**	**0.300**	**-.043675**	**.0134517**
__000038	**-.0073205**	**.0207005**	**-0.35**	**0.724**	**-.0479034**	**.0332624**
__00003Q	**.1498987**	**.0716781**	**2.09**	**0.037**	**.0093752**	**.2904223**
__000041	**.0326157**	**.0288179**	**1.13**	**0.258**	**-.0238812**	**.0891126**
__00004C	**-.0762122**	**.0338384**	**-2.25**	**0.024**	**-.1425517**	**-.0098728**
__00004U	**-.0159037**	**.0641748**	**-0.25**	**0.804**	**-.1417171**	**.1099098**
__000055	**.0012639**	**.02123**	**0.06**	**0.953**	**-.040357**	**.0428848**
__00005G	**.0311996**	**.0329209**	**0.95**	**0.343**	**-.0333412**	**.0957404**
__00005Y	**-.0655302**	**.0501822**	**-1.31**	**0.192**	**-.1639113**	**.032851**
__000069	**.0028**	**.014815**	**0.19**	**0.850**	**-.0262445**	**.0318445**
__00006K	**.0058762**	**.023205**	**0.25**	**0.800**	**-.0396168**	**.0513692**
__000072	**-.0420747**	**.067466**	**-0.62**	**0.533**	**-.1743404**	**.090191**
__00007D	**.009275**	**.0211865**	**0.44**	**0.662**	**-.0322608**	**.0508107**
__00007O	**-.0120165**	**.0295094**	**-0.41**	**0.684**	**-.0698691**	**.045836**
__00007Z	**.0247669**	**.4025898**	**0.06**	**0.951**	**-.7645023**	**.8140362**
__000080	**-.0256414**	**.0718753**	**-0.36**	**0.721**	**-.1665515**	**.1152688**
__000081	**-.0459346**	**.1644982**	**-0.28**	**0.780**	**-.36843**	**.2765608**
__000082	**.221121**	**.419613**	**0.53**	**0.598**	**-.6015218**	**1.043764**
__000083	**-.0412492**	**.1019986**	**-0.40**	**0.686**	**-.2412154**	**.1587169**
__000084	**.0949755**	**.1636718**	**0.58**	**0.562**	**-.2258999**	**.4158508**
__000085	**-.0185792**	**.0678023**	**-0.27**	**0.784**	**-.1515042**	**.1143458**
__000086	**.0215846**	**.0528749**	**0.41**	**0.683**	**-.0820755**	**.1252447**
__000087	**.0385419**	**.0347249**	**1.11**	**0.267**	**-.0295356**	**.1066194**
__000088	**-.0219194**	**.0475841**	**-0.46**	**0.645**	**-.1152071**	**.0713683**
__000089	**-.0619666**	**.0573835**	**-1.08**	**0.280**	**-.1744657**	**.0505326**
__00008A	**-.1717907**	**.8876696**	**-0.19**	**0.847**	**-1.912049**	**1.568468**
__00008B	**.2371389**	**.1441006**	**1.65**	**0.100**	**-.0453675**	**.5196453**
__00008C	**-.3483116**	**.3459745**	**-1.01**	**0.314**	**-1.026588**	**.3299644**

00008D	.6149689	.871486	0.71	0.480	-1.093562	2.3235
00008E	-.1561689	.1865518	-0.84	0.403	-.5218999	.2095622
00008F	-.5748937	.3248082	-1.77	0.077	-1.211674	.0618863
00008G	-.0773958	.1282103	-0.60	0.546	-.3287495	.1739579
00008H	-.2206576	.1096055	-2.01	0.044	-.4355371	-.0057782
00008I	.0952415	.0626305	1.52	0.128	-.0275443	.2180273
00008J	-.1571377	.0986506	-1.59	0.111	-.3505402	.0362649
00008K	.0434661	.1178775	0.37	0.712	-.1876304	.2745627
00008L	-.1385752	.1857159	-0.75	0.456	-.5026674	.2255171
00008M	.0323174	.032756	0.99	0.324	-.0319001	.0965349
00008N	-.0264521	.0719491	-0.37	0.713	-.1675069	.1146027
00008O	.1577994	.210457	0.75	0.453	-.2547973	.5703962
00008P	.0684999	.0516808	1.33	0.185	-.0328193	.1698191
00008Q	-.0699087	.0836604	-0.84	0.403	-.2339233	.0941059
00008R	-.0367281	.03464	-1.06	0.289	-.1046392	.031183
00008S	.01828	.0247779	0.74	0.461	-.0302967	.0668567
00008T	-.0208759	.0154785	-1.35	0.177	-.0512211	.0094693
00008U	-.0298464	.0248785	-1.20	0.230	-.0786201	.0189274
00008V	.0034188	.030401	0.11	0.910	-.0561818	.0630194
00008W	-.6815638	.4631033	-1.47	0.141	-1.589469	.2263411
00008X	-.0595999	.0543503	-1.10	0.273	-.1661526	.0469528
00008Y	.3617518	.1778597	2.03	0.042	.0130614	.7104423
00008Z	.3564088	.4096254	0.87	0.384	-.4466536	1.159471
000090	-.1103219	.0776549	-1.42	0.155	-.2625629	.041919
000091	-.0550339	.135551	-0.41	0.685	-.3207789	.210711
000092	-.0331743	.057593	-0.58	0.565	-.1460843	.0797357
000093	.0572334	.0366143	1.56	0.118	-.0145482	.1290151
000094	-.0481142	.0254041	-1.89	0.058	-.0979185	.00169
000095	.0195355	.0434165	0.45	0.653	-.0655816	.1046526
000096	-.030765	.0489128	-0.63	0.529	-.1266576	.0651276
000097	-.5444064	.3003944	-1.81	0.070	-1.133324	.0445109
000098	-.0938369	.0454564	-2.06	0.039	-.1829533	-.0047206
000099	.3246808	.1205504	2.69	0.007	.0883441	.5610174
00009A	.3673008	.2900397	1.27	0.205	-.2013162	.9359178
00009B	-.0799506	.0636141	-1.26	0.209	-.2046648	.0447635
00009C	-.009825	.1083546	-0.09	0.928	-.2222519	.202602
00009D	-.0053584	.0453438	-0.12	0.906	-.094254	.0835371
00009E	-.0143521	.0337705	-0.42	0.671	-.0805585	.0518544
00009F	.0266307	.022425	1.19	0.235	-.017333	.0705944
00009G	-.0067169	.0314455	-0.21	0.831	-.0683652	.0549314
00009H	-.0661386	.0368349	-1.80	0.073	-.1383526	.0060754
00009I	.0408486	.2100307	0.19	0.846	-.3709124	.4526096
00009J	.0095971	.035194	0.27	0.785	-.0594	.0785941
00009K	-.0725834	.0830405	-0.87	0.382	-.2353826	.0902158
00009L	.1443063	.2176461	0.66	0.507	-.2823845	.5709971
00009M	-.03505	.0515204	-0.68	0.496	-.1360548	.0659548
00009N	-.0913499	.0873698	-1.05	0.296	-.2626367	.079937
00009O	-.0170904	.0328973	-0.52	0.603	-.0815849	.047404
00009P	-.0169291	.028856	-0.59	0.557	-.0735007	.0396425
00009Q	.0172245	.0164509	1.05	0.295	-.0150273	.0494762
00009R	-.009666	.0274487	-0.35	0.725	-.0634787	.0441467
00009S	.0452256	.0311324	1.45	0.146	-.0158088	.10626
00009T	-.0615899	.3425282	-0.18	0.857	-.7331097	.6099299
00009U	.077691	.0477497	1.63	0.104	-.0159213	.1713034
00009V	-.1074309	.1309426	-0.82	0.412	-.3641412	.1492795
00009W	.1845007	.3207713	0.58	0.565	-.444365	.8133664
00009X	-.0115453	.0670409	-0.17	0.863	-.1429776	.1198869
00009Y	-.2080519	.1151702	-1.81	0.071	-.4338409	.0177371
00009Z	.0189436	.0427652	0.44	0.658	-.0648967	.1027839
0000A0	-.0594547	.0348877	-1.70	0.088	-.1278514	.008942
0000A1	-.0041966	.0210033	-0.20	0.842	-.0453732	.03698
0000A2	.0081189	.0345884	0.23	0.814	-.0596909	.0759287
0000A3	.0358123	.0417068	0.86	0.391	-.045953	.1175777
0000A5	40.51572	29.21938	1.39	0.166	-16.76829	97.79974
0000A7	-1.193598	4.443261	-0.27	0.788	-9.904523	7.517326
0000A9	-27.64726	15.94144	-1.73	0.083	-58.90014	3.605625
0000AC	.4094557	.5765131	0.71	0.478	-.7207868	1.539698
0000AE	.2383025	.1406811	1.69	0.090	-.0375	.514105
0000AG	-.3241458	.3952156	-0.82	0.412	-1.098958	.4506664
0000AJ	-9.32148	5.851725	-1.59	0.111	-20.79367	2.150711
0000AL	-.3222155	.853455	-0.38	0.706	-1.995397	1.350966
0000AN	5.88066	2.977866	1.97	0.048	.0426117	11.71871

__0000AQ	-46.8338	35.69909	-1.31	0.190	-116.8212	23.15356
__0000AS	2.62199	5.65325	0.46	0.643	-8.461094	13.70507
__0000AU	33.70891	20.41349	1.65	0.099	-6.311322	73.72915
__0000AX	-.3450033	1.009231	-0.34	0.732	-2.32358	1.633574
__0000AZ	-.587507	.2535504	-2.32	0.021	-1.084587	-.0904265
__0000B1	-.7093808	.6092835	-1.16	0.244	-1.903869	.4851074
__0000B4	-.8938142	1.366006	-0.65	0.513	-3.571841	1.784213
__0000B6	-.7915218	.364239	-2.17	0.030	-1.505605	-.0774385
__0000B8	-.1342555	.7615952	-0.18	0.860	-1.627348	1.358837
__0000BB	-.4342364	.5363172	-0.81	0.418	-1.485676	.6172027
__0000BD	.1340458	.1350906	0.99	0.321	-.1307966	.3988882
__0000BF	.65156	.2948491	2.21	0.027	.0735142	1.229606
__0000BI	1.346408	.3487018	3.86	0.000	.6627851	2.030031
__0000BK	-.0705683	.1250638	-0.56	0.573	-.3157534	.1746168
__0000BM	-.469576	.1462335	-3.21	0.001	-.7562638	-.1828881
__0000BP	-.117023	.3489928	-0.34	0.737	-.8012163	.5671704
__0000BR	-.0283586	.1092474	-0.26	0.795	-.242536	.1858188
__0000BT	-.1336941	.1360998	-0.98	0.326	-.4005151	.1331269
__0000BW	.4687894	.4794096	0.98	0.328	-.4710835	1.408662
__0000BY	-.0751834	.1136129	-0.66	0.508	-.2979193	.1475525
__0000C0	-.378959	.2521978	-1.50	0.133	-.8733877	.1154697
__0000C3	.7321354	.5733746	1.28	0.202	-.3919541	1.856225
__0000C5	-.3223762	.1252385	-2.57	0.010	-.5679038	-.0768486
__0000C7	-.775386	.3240042	-2.39	0.017	-1.41059	-.1401824
xbarlag	-10.32228	23.20301	-0.44	0.656	-55.81133	35.16677
x2barlag	-.6247984	.4391064	-1.42	0.155	-1.485658	.2360611
xbar2lag	4.067035	4.682742	0.87	0.385	-5.113388	13.24746
rootxbarlag	8.489736	28.11775	0.30	0.763	-46.63455	63.61402
z1barlag	1.84462	.8722483	2.11	0.035	.134595	3.554646
z2barlag	2.154337	1.266077	1.70	0.089	-.3277809	4.636455
z3barlag	-.3966135	.4999116	-0.79	0.428	-1.37668	.5834533
z4barlag	-.7071445	.3394741	-2.08	0.037	-1.372677	-.0416123
z5barlag	.2425151	.3603317	0.67	0.501	-.463908	.9489383
z6barlag	.0428925	.4246848	0.10	0.920	-.7896937	.8754786
z7barlag	.3672959	.4434833	0.83	0.408	-.5021443	1.236736
_cons	-2.318915	9.48782	-0.24	0.807	-20.9196	16.28177

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000CU	128,640	.4424744	.0713668	.2244349	.8278759

Linear regression

Number of obs	=	128,640
F(148, 4598)	=	.
Prob > F	=	.
R-squared	=	0.2988
Root MSE	=	.03143

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__0000CA	4.677875	3.092828	1.51	0.130	-1.385553	10.7413
__0000CC	-1.51303	.2391049	-6.33	0.000	-1.98179	-1.044269
__0000CE	-2.153439	.5541039	-3.89	0.000	-3.239748	-1.067129
__0000CH	-.7392562	.4496332	-1.64	0.100	-1.620753	.1422407
__0000CJ	1.053623	.1871643	5.63	0.000	.6866908	1.420554
__0000CM	-2.669284	1.845161	-1.45	0.148	-6.286685	.9481181
xi	.0170337	.00271	6.29	0.000	.0117208	.0223467
__000002	-.0010007	.0010908	-0.92	0.359	-.0031392	.0011378
__00000E	-.019419	.0235414	-0.82	0.409	-.0655714	.0267335
__00000P	-.0102415	.0075348	-1.36	0.174	-.0250133	.0045303
__000010	-.0120784	.0103652	-1.17	0.244	-.0323992	.0082424
__00001I	.0815464	.0520438	1.57	0.117	-.0204844	.1835772
__00001T	.0210307	.0177048	1.19	0.235	-.0136792	.0557405
__000024	-.0000722	.0215525	-0.00	0.997	-.0423254	.042181
__00002M	.0193565	.0120741	1.60	0.109	-.0043145	.0430275
__00002X	.0053853	.0040066	1.34	0.179	-.0024697	.0132403
__000038	-.0005198	.0052951	-0.10	0.922	-.0109007	.0098611
__00003Q	.0248189	.0159515	1.56	0.120	-.0064537	.0560916

__000041	.0062229	.0065405	0.95	0.341	-.0065995	.0190454
__00004C	-.0191381	.0080141	-2.39	0.017	-.0348496	-.0034266
__00004U	-.0093908	.0150746	-0.62	0.533	-.0389443	.0201626
__000055	.0033442	.0049122	0.68	0.496	-.006286	.0129745
__00005G	.0113813	.0072846	1.56	0.118	-.0028999	.0256626
__00005Y	-.0154265	.0121828	-1.27	0.205	-.0393106	.0084576
__000069	.0008894	.0036126	0.25	0.806	-.006193	.0079718
__00006K	-.0048485	.0061735	-0.79	0.432	-.0169516	.0072546
__000072	.0022782	.0173399	0.13	0.895	-.0317163	.0362727
__00007D	.0104635	.005651	1.85	0.064	-.0006152	.0215423
__00007O	-.0025683	.0083613	-0.31	0.759	-.0189605	.0138239
__00007Z	-.0681979	.1007445	-0.68	0.498	-.2657054	.1293096
__000080	-.0110638	.0215427	-0.51	0.608	-.0532977	.0311701
__000081	.0215672	.0453249	0.48	0.634	-.0672914	.1104258
__000082	.0837318	.1078717	0.78	0.438	-.1277485	.295212
__000083	-.0097742	.0258815	-0.38	0.706	-.0605144	.0409661
__000084	.0802533	.0369765	2.17	0.030	.0077616	.1527449
__000085	-.0124454	.0184734	-0.67	0.501	-.0486622	.0237714
__000086	.0105039	.009937	1.06	0.291	-.0089773	.0299852
__000087	.0079296	.0084764	0.94	0.350	-.0086881	.0245474
__000088	-.0024098	.0104541	-0.23	0.818	-.0229048	.0180852
__000089	-.016453	.0116881	-1.41	0.159	-.0393672	.0064611
__00008A	.4000876	.2191735	1.83	0.068	-.0295976	.8297729
__00008B	.0841541	.0375219	2.24	0.025	.0105932	.157715
__00008C	-.2411249	.0896225	-2.69	0.007	-.4168279	-.0654218
__00008D	-.2069087	.2261006	-0.92	0.360	-.6501743	.2363569
__00008E	-.1169442	.0492232	-2.38	0.018	-.2134453	-.0204431
__00008F	-.1813671	.0899292	-2.02	0.044	-.3576716	-.0050626
__00008G	-.0108419	.0339126	-0.32	0.749	-.0773269	.0556432
__00008H	-.0551308	.0268128	-2.06	0.040	-.1076968	-.0025648
__00008I	.0015426	.0172301	0.09	0.929	-.0322366	.0353219
__00008J	-.0450347	.0246133	-1.83	0.067	-.0932886	.0032193
__00008K	-.0039367	.0295443	-0.13	0.894	-.0618577	.0539843
__00008L	-.0404095	.049328	-0.82	0.413	-.137116	.0562971
__00008M	.001021	.007579	0.13	0.893	-.0138375	.0158796
__00008N	.0131892	.0193792	0.68	0.496	-.0248033	.0511816
__00008O	.0114961	.0502268	0.23	0.819	-.0869726	.1099647
__00008P	.0192111	.0125691	1.53	0.126	-.0054304	.0438526
__00008Q	-.0325345	.019847	-1.64	0.101	-.0714441	.0063751
__00008R	-.0096409	.009686	-1.00	0.320	-.02863	.0093483
__00008S	-.0001567	.0059184	-0.03	0.979	-.0117597	.0114462
__00008T	-.0001145	.0042702	-0.03	0.979	-.0084862	.0082571
__00008U	-.006079	.0052961	-1.15	0.251	-.0164619	.004304
__00008V	.0015051	.0067101	0.22	0.823	-.01165	.0146601
__00008W	-.0659617	.0806505	-0.82	0.413	-.2240755	.0921521
__00008X	-.0247029	.0120841	-2.04	0.041	-.0483934	-.0010123
__00008Y	.0580503	.0318451	1.82	0.068	-.0043813	.120482
__00008Z	.0110772	.0756801	0.15	0.884	-.1372921	.1594464
__000090	.0032114	.0148403	0.22	0.829	-.0258827	.0323056
__000091	-.0007712	.0309768	-0.02	0.980	-.0615007	.0599582
__000092	.0003623	.0134755	0.03	0.979	-.0260561	.0267807
__000093	.0312665	.0084329	3.71	0.000	.014734	.0477989
__000094	.0012232	.0065024	0.19	0.851	-.0115247	.0139711
__000095	.0022841	.0079556	0.29	0.774	-.0133128	.0178809
__000096	.006253	.0097397	0.64	0.521	-.0128414	.0253474
__000097	-.068417	.0665222	-1.03	0.304	-.1988324	.0619984
__000098	-.0321976	.011108	-2.90	0.004	-.0539746	-.0104206
__000099	.0769573	.0278348	2.76	0.006	.0223876	.131527
__00009A	-.0094448	.0631059	-0.15	0.881	-.1331627	.114273
__00009B	.006836	.0159187	0.43	0.668	-.0243723	.0380444
__00009C	.0466064	.024625	1.89	0.058	-.0016705	.0948832
__00009D	.0096626	.0105985	0.91	0.362	-.0111155	.0304408
__00009E	.0081096	.008809	0.92	0.357	-.0091603	.0253795
__00009F	-.0027182	.0052673	-0.52	0.606	-.0130445	.0076082
__00009G	.0040504	.0076259	0.53	0.595	-.0109	.0190007
__00009H	-.0011467	.0091941	-0.12	0.901	-.0191714	.0168781
__00009I	-.0238561	.0430183	-0.55	0.579	-.1081925	.0604803
__00009J	.0035423	.0080495	0.44	0.660	-.0122386	.0193232
__00009K	-.0070524	.0176384	-0.40	0.689	-.041632	.0275273
__00009L	.0517459	.0454622	1.14	0.255	-.0373818	.1408736
__00009M	-.0200109	.0121946	-1.64	0.101	-.0439182	.0038963
__00009N	-.0079232	.0196472	-0.40	0.687	-.0464411	.0305947

__00009O	.0056391	.007526	0.75	0.454	-.0091156	.0203937
__00009P	-.0080001	.0067477	-1.19	0.236	-.0212288	.0052286
__00009Q	.0016867	.0046181	0.37	0.715	-.007367	.0107403
__00009R	.007372	.0052378	1.41	0.159	-.0028967	.0176407
__00009S	.0093889	.0071346	1.32	0.188	-.0045983	.0233761
__00009T	.1166889	.0696247	1.68	0.094	-.0198089	.2531868
__00009U	.0353334	.0105196	3.36	0.001	.01471	.0559568
__00009V	-.0820606	.026895	-3.05	0.002	-.1347877	-.0293335
__00009W	-.0802119	.0698383	-1.15	0.251	-.2171284	.0567047
__00009X	-.0381151	.0170619	-2.23	0.026	-.0715646	-.0046656
__00009Y	-.0208003	.0284381	-0.73	0.465	-.0765526	.0349519
__00009Z	.0278719	.0105935	2.63	0.009	.0071036	.0486402
__0000A0	-.0219439	.007529	-2.91	0.004	-.0367044	-.0071834
__0000A1	.0001324	.0063097	0.02	0.983	-.0122376	.0125025
__0000A2	-.0053828	.0076607	-0.70	0.482	-.0204014	.0096359
__0000A3	-.0034739	.0096997	-0.36	0.720	-.0224899	.0155421
__0000A5	6.773098	7.375606	0.92	0.359	-7.686631	21.23283
__0000A7	-1.782925	1.046319	-1.70	0.088	-3.834213	.2683628
__0000A9	-7.572242	3.852731	-1.97	0.049	-15.12544	-.0190393
__0000AC	.0840278	.1516048	0.55	0.579	-.2131904	.381246
__0000AE	.0179718	.0390401	0.46	0.645	-.0585656	.0945092
__0000AG	-.0798162	.1044962	-0.76	0.445	-.2846789	.1250465
__0000AJ	-1.356543	1.47459	-0.92	0.358	-4.247447	1.53436
__0000AL	.2464558	.2129111	1.16	0.247	-.1709522	.6638638
__0000AN	1.374378	.711347	1.93	0.053	-.0202036	2.76896
__0000AQ	-8.792693	9.065879	-0.97	0.332	-26.56617	8.980781
__0000AS	2.53114	1.321734	1.92	0.056	-.0600921	5.122373
__0000AU	10.13155	4.983802	2.03	0.042	.360909	19.9022
__0000AX	.5976473	.268429	2.23	0.026	.0713975	1.123897
__0000AZ	-.1952569	.0664728	-2.94	0.003	-.3255754	-.0649383
__0000B1	-.3015137	.1654969	-1.82	0.069	-.6259669	.0229396
__0000B4	-.4059903	.3144514	-1.29	0.197	-1.022466	.2104855
__0000B6	-.1248952	.1113195	-1.12	0.262	-.3431349	.0933444
__0000B8	-.5371562	.1873321	-2.87	0.004	-.9044171	-.1698953
__0000BB	.2472246	.1393013	1.77	0.076	-.0258729	.520322
__0000BD	.0440087	.0413466	1.06	0.287	-.0370506	.1250679
__0000BF	-.2197992	.0898567	-2.45	0.014	-.3959615	-.0436369
__0000BI	.1604154	.0807685	1.99	0.047	.0020703	.3187604
__0000BK	-.027681	.027041	-1.02	0.306	-.0806943	.0253323
__0000BM	-.040153	.0378546	-1.06	0.289	-.1143663	.0340603
__0000BP	-.3622991	.0847784	-4.27	0.000	-.5285055	-.1960928
__0000BR	-.0345706	.0255464	-1.35	0.176	-.0846538	.0155125
__0000BT	.1507979	.0366801	4.11	0.000	.0788872	.2227085
__0000BW	.3318459	.1048489	3.16	0.002	.1262916	.5374002
__0000BY	-.025565	.0272362	-0.94	0.348	-.0789609	.027831
__0000C0	-.194259	.0630614	-3.08	0.002	-.3178897	-.0706283
__0000C3	.2079287	.1330361	1.56	0.118	-.0528859	.4687434
__0000C5	-.0638962	.0317678	-2.01	0.044	-.1261763	-.0016161
__0000C7	-.1780398	.0824397	-2.16	0.031	-.3396612	-.0164184
xbarlag	4.04385	5.56316	0.73	0.467	-6.862615	14.95031
x2barlag	-.0563375	.1029798	-0.55	0.584	-.2582273	.1455522
xbar2lag	-.4429217	1.123839	-0.39	0.694	-2.646186	1.760343
rootxbarlag	-5.830931	6.796825	-0.86	0.391	-19.15597	7.494109
z1barlag	-.0100134	.2133328	-0.05	0.963	-.428248	.4082213
z2barlag	1.164475	.3074819	3.79	0.000	.5616632	1.767288
z3barlag	-.0785922	.121137	-0.65	0.517	-.3160789	.1588945
z4barlag	-.0807978	.0768448	-1.05	0.293	-.2314504	.0698548
z5barlag	.2424491	.0850521	2.85	0.004	.075706	.4091921
z6barlag	-.0856014	.0913681	-0.94	0.349	-.2647267	.093524
z7barlag	.0440779	.1017553	0.43	0.665	-.1554113	.2435672
_cons	2.067702	2.311112	0.89	0.371	-2.463186	6.59859

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000D6	128,640	.0911297	.0205071	.0369996	.1767478

Linear regression

Number of obs	=	**128,640**
F(148, 4598)	=	**.**
Prob > F	=	**.**
R-squared	=	**0.2997**
Root MSE	=	**.19143**

(Std. err. adjusted for **4,599** clusters in **group_round**)

qikbar3	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__0000CA	**19.72416**	**17.96959**	**1.10**	**0.272**	**-15.50486**	**54.95319**
__0000CC	**1.515332**	**1.224382**	**1.24**	**0.216**	**-.8850453**	**3.915709**
__0000CE	**4.474477**	**2.534192**	**1.77**	**0.078**	**-.4937567**	**9.442711**
__0000CH	**.0198028**	**2.952779**	**0.01**	**0.995**	**-5.769062**	**5.808667**
__0000CJ	**-.7064709**	**.8619186**	**-0.82**	**0.412**	**-2.396245**	**.9833034**
__0000CM	**-16.81897**	**9.484308**	**-1.77**	**0.076**	**-35.41276**	**1.774831**
xi	**.1307497**	**.0132927**	**9.84**	**0.000**	**.1046896**	**.1568099**
__000002	**-.0065885**	**.005017**	**-1.31**	**0.189**	**-.0164243**	**.0032473**
__00000E	**-.2662238**	**.1475316**	**-1.80**	**0.071**	**-.5554565**	**.023009**
__00000P	**-.0626842**	**.0438756**	**-1.43**	**0.153**	**-.1487014**	**.023333**
__000010	**-.0446399**	**.0615601**	**-0.73**	**0.468**	**-.1653273**	**.0760476**
__00001I	**.5580734**	**.3052427**	**1.83**	**0.068**	**-.0403487**	**1.156496**
__00001T	**.2032963**	**.0982546**	**2.07**	**0.039**	**.0106701**	**.3959226**
__000024	**.004304**	**.1170992**	**0.04**	**0.971**	**-.2252666**	**.2338747**
__00002M	**.1174028**	**.0735566**	**1.60**	**0.111**	**-.0268033**	**.261609**
__00002X	**.0405673**	**.0223973**	**1.81**	**0.070**	**-.0033422**	**.0844768**
__000038	**-.0238957**	**.0269501**	**-0.89**	**0.375**	**-.0767309**	**.0289394**
__00003Q	**.3655848**	**.0975183**	**3.75**	**0.000**	**.1744021**	**.5567675**
__000041	**.0186472**	**.036321**	**0.51**	**0.608**	**-.0525593**	**.0898538**
__00004C	**-.0762788**	**.0403286**	**-1.89**	**0.059**	**-.1553422**	**.0027846**
__00004U	**-.1390218**	**.0839474**	**-1.66**	**0.098**	**-.3035989**	**.0255554**
__000055	**-.0040062**	**.0261039**	**-0.15**	**0.878**	**-.0551823**	**.0471699**
__00005G	**.0629784**	**.0367464**	**1.71**	**0.087**	**-.0090621**	**.1350189**
__00005Y	**.0081864**	**.068496**	**0.12**	**0.905**	**-.1260985**	**.1424714**
__000069	**.0152629**	**.0209964**	**0.73**	**0.467**	**-.0259001**	**.056426**
__00006K	**-.0131241**	**.0281625**	**-0.47**	**0.641**	**-.0683362**	**.042088**
__000072	**.0282828**	**.1116791**	**0.25**	**0.800**	**-.1906619**	**.2472274**
__00007D	**.1054299**	**.0341831**	**3.08**	**0.002**	**.0384145**	**.1724453**
__00007O	**-.0334945**	**.040568**	**-0.83**	**0.409**	**-.1130272**	**.0460382**
__00007Z	**-1.100565**	**.6209016**	**-1.77**	**0.076**	**-2.31783**	**.1167004**
__000080	**-.0717327**	**.1080631**	**-0.66**	**0.507**	**-.2835883**	**.1401229**
__000081	**.296013**	**.250848**	**1.18**	**0.238**	**-.1957695**	**.7877955**
__000082	**1.180619**	**.6408524**	**1.84**	**0.066**	**-.0757589**	**2.436998**
__000083	**-.0639623**	**.1376454**	**-0.46**	**0.642**	**-.3338133**	**.2058887**
__000084	**.3954916**	**.2037375**	**1.94**	**0.052**	**-.0039317**	**.794915**
__000085	**-.1616121**	**.0933081**	**-1.73**	**0.083**	**-.3445408**	**.0213166**
__000086	**.0724187**	**.0552103**	**1.31**	**0.190**	**-.0358199**	**.1806574**
__000087	**-.0126977**	**.0413373**	**-0.31**	**0.759**	**-.0937387**	**.0683432**
__000088	**.0454524**	**.0587907**	**0.77**	**0.439**	**-.0698056**	**.1607104**
__000089	**-.047427**	**.071801**	**-0.66**	**0.509**	**-.1881914**	**.0933375**
__00008A	**5.205917**	**1.24617**	**4.18**	**0.000**	**2.762825**	**7.64901**
__00008B	**.5147718**	**.2236692**	**2.30**	**0.021**	**.0762727**	**.9532709**
__00008C	**-2.197021**	**.5136496**	**-4.28**	**0.000**	**-3.204021**	**-1.190021**
__00008D	**-3.957773**	**1.286192**	**-3.08**	**0.002**	**-6.479328**	**-1.436219**
__00008E	**-.3726791**	**.2844432**	**-1.31**	**0.190**	**-.9303243**	**.1849661**
__00008F	**-.3360271**	**.4582303**	**-0.73**	**0.463**	**-1.234378**	**.5623242**
__00008G	**-.0450734**	**.1716467**	**-0.26**	**0.793**	**-.3815833**	**.2914365**
__00008H	**-.6024521**	**.1154902**	**-5.22**	**0.000**	**-.8288684**	**-.3760358**
__00008I	**.0855242**	**.0870317**	**0.98**	**0.326**	**-.0850996**	**.2561481**
__00008J	**-.1278781**	**.139773**	**-0.91**	**0.360**	**-.4019003**	**.1461441**
__00008K	**.0330238**	**.1711934**	**0.19**	**0.847**	**-.3025976**	**.3686451**
__00008L	**.3999629**	**.3069919**	**1.30**	**0.193**	**-.2018887**	**1.001814**
__00008M	**.0071741**	**.0486119**	**0.15**	**0.883**	**-.0881286**	**.1024768**
__00008N	**-.1259232**	**.119961**	**-1.05**	**0.294**	**-.3611044**	**.1092579**
__00008O	**-.4172946**	**.3086467**	**-1.35**	**0.176**	**-1.02239**	**.187801**
__00008P	**.0405607**	**.0695364**	**0.58**	**0.560**	**-.0957641**	**.1768854**
__00008Q	**-.0789205**	**.1101536**	**-0.72**	**0.474**	**-.2948743**	**.1370333**
__00008R	**-.0130462**	**.048443**	**-0.27**	**0.788**	**-.1080177**	**.0819254**
__00008S	**.0322568**	**.0276893**	**1.16**	**0.244**	**-.0220275**	**.0865412**
__00008T	**.0232551**	**.0209743**	**1.11**	**0.268**	**-.0178646**	**.0643747**
__00008U	**-.040239**	**.0324168**	**-1.24**	**0.215**	**-.1037914**	**.0233135**

__00008V	.0231589	.0382811	0.60	0.545	-.0518903	.0982081
__00008W	-.2745451	.5748501	-0.48	0.633	-1.401527	.852437
__00008X	-.0194482	.0688652	-0.28	0.778	-.1544572	.1155607
__00008Y	.2598803	.1996452	1.30	0.193	-.1315202	.6512808
__00008Z	-.4279608	.4822335	-0.89	0.375	-1.37337	.5174485
__000090	.1102851	.0937013	1.18	0.239	-.0734145	.2939847
__000091	.2280905	.1674791	1.36	0.173	-.1002489	.5564299
__000092	.0177484	.0651225	0.27	0.785	-.1099229	.1454197
__000093	.0775329	.0395368	1.96	0.050	.0000219	.1550439
__000094	-.0816406	.030453	-2.68	0.007	-.1413431	-.0219381
__000095	-.0624167	.0476672	-1.31	0.190	-.1558673	.0310339
__000096	-.0270314	.0610218	-0.44	0.658	-.1466634	.0926005
__000097	-.9576348	.3356529	-2.85	0.004	-1.615676	-.299594
__000098	-.097994	.059489	-1.65	0.100	-.2146211	.018633
__000099	.4264023	.1466334	2.91	0.004	.1389305	.7138741
__00009A	.6581535	.3378038	1.95	0.051	-.0041041	1.320411
__00009B	.1287284	.079011	1.63	0.103	-.0261711	.2836279
__00009C	.0653398	.1246325	0.52	0.600	-.1789998	.3096794
__00009D	-.0527494	.0537871	-0.98	0.327	-.1581979	.0526991
__00009E	.0434567	.0352713	1.23	0.218	-.0256919	.1126054
__00009F	.0434495	.0256297	1.70	0.090	-.0067971	.0936961
__00009G	.0366056	.0393242	0.93	0.352	-.0404888	.1136999
__00009H	.0200094	.0444559	0.45	0.653	-.0671454	.1071642
__00009I	.38181	.2874188	1.33	0.184	-.1816688	.9452888
__00009J	.0893858	.0475992	1.88	0.060	-.0039314	.182703
__00009K	-.2279116	.1181648	-1.93	0.054	-.4595713	.0037481
__00009L	-.2065209	.282176	-0.73	0.464	-.7597214	.3466795
__00009M	-.051354	.0621276	-0.83	0.409	-.1731539	.0704459
__00009N	-.0784311	.1055602	-0.74	0.458	-.2853798	.1285176
__00009O	.0225483	.0397456	0.57	0.571	-.0553721	.1004687
__00009P	-.0541108	.0297525	-1.82	0.069	-.1124399	.0042184
__00009Q	.0059482	.0204775	0.29	0.771	-.0341976	.0460939
__00009R	-.0007794	.0310464	-0.03	0.980	-.0616452	.0600863
__00009S	-.0091118	.0366086	-0.25	0.803	-.0808821	.0626586
__00009T	.2274097	.5242879	0.43	0.664	-.8004463	1.255266
__00009U	.1833137	.0700922	2.62	0.009	.0458993	.320728
__00009V	-.2983737	.1989009	-1.50	0.134	-.6883149	.0915676
__00009W	-.0473205	.5044688	-0.09	0.925	-1.036322	.9416806
__00009X	-.129888	.0969943	-1.34	0.181	-.3200434	.0602674
__00009Y	-.4228877	.1629702	-2.59	0.009	-.7423874	-.103388
__00009Z	.0492556	.0609166	0.81	0.419	-.0701702	.1686815
__0000A0	-.1776752	.0408015	-4.35	0.000	-.2576656	-.0976848
__0000A1	.0007761	.030305	0.03	0.980	-.0586363	.0601884
__0000A2	.0101742	.0500508	0.20	0.839	-.0879494	.1082978
__0000A3	.009322	.0577687	0.16	0.872	-.1039323	.1225763
__0000A5	48.30421	42.82146	1.13	0.259	-35.64641	132.2548
__0000A7	-.2093139	6.598813	-0.03	0.975	-13.14615	12.72753
__0000A9	-32.75008	20.81856	-1.57	0.116	-73.56445	8.064296
__0000AC	-.6845105	.8598735	-0.80	0.426	-2.370275	1.001254
__0000AE	.1974236	.1931704	1.02	0.307	-.1812831	.5761304
__0000AG	.6032529	.5427348	1.11	0.266	-.4607677	1.667274
__0000AJ	-7.565893	8.534276	-0.89	0.375	-24.29717	9.165386
__0000AL	-.0157107	1.240131	-0.01	0.990	-2.446962	2.415541
__0000AN	4.844677	3.802121	1.27	0.203	-2.609304	12.29866
__0000AQ	-62.6009	52.19937	-1.20	0.230	-164.9367	39.73493
__0000AS	-.0671822	8.391197	-0.01	0.994	-16.51796	16.38359
__0000AU	42.68588	26.79165	1.59	0.111	-9.838612	95.21037
__0000AX	-1.419298	1.275562	-1.11	0.266	-3.920012	1.081417
__0000AZ	-.6156358	.3230132	-1.91	0.057	-1.248897	.0176251
__0000B1	-.0913644	.7531515	-0.12	0.903	-1.567903	1.385174
__0000B4	-.4159367	1.717006	-0.24	0.809	-3.782093	2.95022
__0000B6	-.6452967	.4988886	-1.29	0.196	-1.623358	.3327645
__0000B8	-.7539992	.9232988	-0.82	0.414	-2.564108	1.05611
__0000BB	.0545919	.6476225	0.08	0.933	-1.215059	1.324243
__0000BD	.0355227	.1856828	0.19	0.848	-.3285047	.3995501
__0000BF	-.3028585	.3515291	-0.86	0.389	-.9920242	.3863072
__0000BI	.7494021	.4381024	1.71	0.087	-.1094888	1.608293
__0000BK	.0739261	.1494253	0.49	0.621	-.2190193	.3668715
__0000BM	-.0481613	.1582952	-0.30	0.761	-.3584958	.2621733
__0000BP	-.118559	.4278898	-0.28	0.782	-.9574284	.7203103
__0000BR	-.0700705	.1324916	-0.53	0.597	-.3298175	.1896766
__0000BT	-.2108873	.1574077	-1.34	0.180	-.5194818	.0977073

__0000BW	1.072662	.5617766	1.91	0.056	-.0286894	2.174014
__0000BY	.2178293	.1558139	1.40	0.162	-.0876408	.5232993
__0000C0	.0250061	.2838408	0.09	0.930	-.5314582	.5814703
__0000C3	-.7304097	.6755667	-1.08	0.280	-2.054845	.5940254
__0000C5	-.1803978	.162804	-1.11	0.268	-.4995718	.1387761
__0000C7	.2306071	.3540227	0.65	0.515	-.4634473	.9246615
xbarlag	-12.94394	31.17683	-0.42	0.678	-74.0655	48.17762
x2barlag	-.4140207	.6058787	-0.68	0.494	-1.601834	.7737923
xbar2lag	2.991847	6.51545	0.46	0.646	-9.781562	15.76526
rootxbarlag	16.20586	37.20812	0.44	0.663	-56.73992	89.15164
z1barlag	2.315195	1.074194	2.16	0.031	.2092591	4.421131
z2barlag	1.851417	1.614456	1.15	0.252	-1.313693	5.016526
z3barlag	.2882603	.5974322	0.48	0.629	-.8829935	1.459514
z4barlag	-.6319162	.4364033	-1.45	0.148	-1.487476	.2236437
z5barlag	.4220089	.4257129	0.99	0.322	-.4125928	1.256611
z6barlag	-1.353201	.5464895	-2.48	0.013	-2.424583	-.2818194
z7barlag	.6597472	.576698	1.14	0.253	-.4708577	1.790352
_cons	-6.590962	12.3033	-0.54	0.592	-30.71134	17.52942

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000DI	128,640	.3629647	.1251629	.0785182	1.174995

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"

957
958 * interact qhat_hats with Alist
959 di "`qhat_hats'"
__0000CX __0000D0 __0000D3 __0000D9 __0000DC __0000DF __0000DL __0000DO __0000DR __
> 0000CY __0000D1 __0000D4 __0000DA __0000DD __0000DG __0000DM __0000DP __0000DS

960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                 foreach var2 of global Alist {
    3.                         g `var1'_`var2'=`var1'*`var2'
    4.                         sum `var1' `var2' `var1'_`var2'
    5.                         local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                 }
    7.         }
964 }

965
966 local r_RE      "`r_RE' `qhat_hats'"

967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                 local r_RE "`r_RE' zi`i'"
    3.         }
```

```
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }

978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                 g pzi`j' = p1*zi`j'
    3.          }
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = _b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                 local coef = _b[pzi`j']
    3.                 local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1025}
```

```
1026
1027if `J'==3 & simple_model==0
  >                           local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==4 & simple_model==0
  >                           local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
```

```
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  1.1638934
  Iteration 1:  GMM criterion Q(b) =  .19461162
  Iteration 2:  GMM criterion Q(b) =  .01071907
  Iteration 3:  GMM criterion Q(b) =  .00552606
  Iteration 4:  GMM criterion Q(b) =  .00009595
  Iteration 5:  GMM criterion Q(b) =  1.363e-06

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .06009812
  Iteration 1:  GMM criterion Q(b) =  .03175118
  Iteration 2:  GMM criterion Q(b) =  .01921118
  Iteration 3:  GMM criterion Q(b) =  .01839461
  Iteration 4:  GMM criterion Q(b) =  .01823742
  Iteration 5:  GMM criterion Q(b) =  .01816581
  Iteration 6:  GMM criterion Q(b) =   .0180267

  GMM estimation

  Number of parameters =  32
  Number of moments    =  96
  Initial weight matrix: Identity                   Number of obs   =    128,640
  GMM weight matrix:     Robust

                (Std. err. adjusted for 564 clusters in state_district_round)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
A11          |
         one |   .9324252   .0144841    64.38   0.000     .9040369    .9608135
-------------+----------------------------------------------------------------
C1           |
         zi1 |   .3715618    .028144    13.20   0.000     .3164005    .4267231
         zi2 |  -.1873433   .0358446    -5.23   0.000    -.2575975   -.1170891
         zi3 |  -.0369183   .0123087    -3.00   0.003     -.061043   -.0127936
         zi4 |  -.0150106   .0136728    -1.10   0.272    -.0418089    .0117877
         zi5 |  -.0370258   .0202828    -1.83   0.068    -.0767793    .0027278
         zi6 |  -.0235869   .0129868    -1.82   0.069    -.0490404    .0018667
         zi7 |   .0192714   .0148719     1.30   0.195     -.009877    .0484199
-------------+----------------------------------------------------------------
C2           |
         zi1 |   .0463009   .0048857     9.48   0.000     .0367252    .0558766
         zi2 |  -.0247738   .0056372    -4.39   0.000    -.0358225   -.0137251
         zi3 |   -.006028    .002174    -2.77   0.006     -.010289   -.0017669
         zi4 |  -.0018655   .0022663    -0.82   0.410    -.0063073    .0025763
         zi5 |  -.0063575   .0031037    -2.05   0.041    -.0124406   -.0002745
         zi6 |  -.0008227   .0020188    -0.41   0.684    -.0047795    .0031341
         zi7 |   .0064066   .0023665     2.71   0.007     .0017684    .0110448
-------------+----------------------------------------------------------------
C3           |
         zi1 |     .06698   .0452196     1.48   0.139    -.0216488    .1556089
         zi2 |  -.3971681   .0600181    -6.62   0.000    -.5148015   -.2795347
         zi3 |  -.0510041   .0182462    -2.80   0.005    -.0867661   -.0152421
         zi4 |  -.0271873    .024523    -1.11   0.268    -.0752514    .0208769
         zi5 |  -.0204322   .0301263    -0.68   0.498    -.0794788    .0386143
         zi6 |   -.058902   .0192186    -3.06   0.002    -.0965698   -.0212341
         zi7 |   .0123412   .0226853     0.54   0.586    -.0321211    .0568036
-------------+----------------------------------------------------------------
         /d1 |  -.0524251    .006269    -8.36   0.000     -.064712   -.0401381
         /b1 |   .3698155    .006374    58.02   0.000     .3573227    .3823084
```

/b2	.0674165	.0016266	41.45	0.000	.0642284	.0706045
/AVA11	-2.218586	1.293934	-1.71	0.086	-4.75465	.3174771
/AVA12	1.006564	.4680248	2.15	0.032	.089252	1.923875
/AVA13	1.088061	.9124028	1.19	0.233	-.7002156	2.876338
/AVA22	-1.1768	.309219	-3.81	0.000	-1.782858	-.5707422
/AVA23	.0835603	.2317331	0.36	0.718	-.3706283	.537749
/AVA33	-2.233965	.7849207	-2.85	0.004	-3.772382	-.6955488
/d2	-.0087951	.0010412	-8.45	0.000	-.0108359	-.0067543

Instruments for equation **eq1_RE**: __0000CA __0000CC __0000CE __0000CH __0000CJ __0000CM **xi** __000002 __00000E __00000P __000010 __00001I __00001T __000024 __00002M __00002X __000038 __00003Q __000041 __00004C __00004U __000055 __00005G __00005Y __000069 __00006K __000072 __00007D __00007O __0000CX __0000D0 __0000D3 __0000D9 __0000DC __0000DF __0000DL __0000DO __0000DR __0000CY __0000D1 __0000D4 __0000DA __0000DD __0000DG __0000DM __0000DP __0000DS _cons

Instruments for equation **eq2_RE**: __0000CA __0000CC __0000CE __0000CH __0000CJ __0000CM **xi** __000002 __00000E __00000P __000010 __00001I __00001T __000024 __00002M __00002X __000038 __00003Q __000041 __00004C __00004U __000055 __00005G __00005Y __000069 __00006K __000072 __00007D __00007O __0000CX __0000D0 __0000D3 __0000D9 __0000DC __0000DF __0000DL __0000DO __0000DR __0000CY __0000D1 __0000D4 __0000DA __0000DD __0000DG __0000DM __0000DP __0000DS _cons

```
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_3good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactIns
      > t_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not found)
  (output written to diag_3good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3
  > _all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  24757

1077unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640
```

```
1078estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4599

1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  5.3831268

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_3good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster not found)
  file diag_3good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
```

```
                        local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
> "
                        local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
                        forval l=1/`J' {
                                tempvar zdiff`t'p`j'p`l'
                                g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
> '
                                local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
> `l'"
                                foreach name of global Alist {
                                        tempvar  zdiff`t'p`j'p`l'q`l'`name'
                                        g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
> `t'mzk`t'p`j''*`p`l'qikhat`l''*`name'
                                        local pzindivdiffpqhat "`pzindivdiffpqhat
> ' `zdiff`t'p`j'p`l'q`l'`name'"
                                }
                                if `Tindivp1'<=`T' {
                                        forval s=`Tindivp1'/`T' {
                                                tempvar zdiff`t'p`j'zi`s'p`l'
                                                * don't generate these to save me
> mory!
                                                gen double `zdiff`t'p`j'zi`s'p`l''=
>     `zdiff`t'p`j'p`l''*zi`s'
                                                local pzindivdiffpzg " `pzindivdi
> ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
                                        }
                                }
                        }
                }
        }
        if `Tindivp1'<=`T' {
                forval t=`Tindivp1'/`T' {
                        tempvar zi`t'xdiff
                        g double `zi`t'xdiff'=zi`t'*`ximxk'
                        local zg "`zg' zi`t'"
                        forval j=1/`J' {
                                tempvar zi`t'p`j'xdiff
                                g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
                                local pzg "`pzg' `zi`t'p`j''"
                                local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
                        }
                }
        }

        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
        // denoting zg as group-zs
        //  zg,pzgx,rprp,rprpx
        local rootprootp ""
        local rootprootpxdiff ""

        forval j=1/`J' {
                forval s=`j'/`J' {
                        tempvar xdiffrp`j'rp`s'
                        g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')

                        if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
> 'rp`s'' "

                        local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
>
                }
        }
```

```
1133
1134        local pqhatxdiff ""
1135        forval j=1/`J' {
    2.                  forval s=1/`J' {
    3.                          foreach name of global Alist {
    4.                                  tempvar   p`s'qikhat`j'xdiff`name'
    5.                                  g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
1136                                    local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
1137
1138        // set instruments
1139        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1140}

1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                          local `++t'
    3.                          local p_nice = subinstr("`p'",":_cons","",.)
    4.                          local est = coefs[1,`t']
    5.                          if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                  }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
```

```
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                  }
   16.         }
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1    `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
```

```
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00127391
  Iteration 1:  GMM criterion Q(b) =  .00027555
  Iteration 2:  GMM criterion Q(b) =  .00021449
  Iteration 3:  GMM criterion Q(b) =  6.072e-06

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .05806798
  Iteration 1:  GMM criterion Q(b) =  .05668218
  Iteration 2:  GMM criterion Q(b) =  .03897436
  Iteration 3:  GMM criterion Q(b) =  .03726536
  Iteration 4:  GMM criterion Q(b) =  .03718206
  Iteration 5:  GMM criterion Q(b) =  .03716589

  GMM estimation
```

Number of parameters = **26**
Number of moments = **234**
Initial weight matrix: **Identity** Number of obs = **128,640**
GMM weight matrix: **Robust**

(Std. err. adjusted for **564** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11 _cons	**.5704033**	**.0254739**	**22.39**	**0.000**	**.5204753**	**.6203313**
C12 _cons	**-.0216795**	**.0380821**	**-0.57**	**0.569**	**-.096319**	**.05296**
C13 _cons	**-.0005473**	**.0093431**	**-0.06**	**0.953**	**-.0188594**	**.0177648**
C14 _cons	**.003775**	**.0091894**	**0.41**	**0.681**	**-.0142359**	**.021786**
C15 _cons	**-.0404524**	**.015793**	**-2.56**	**0.010**	**-.071406**	**-.0094987**
C16 _cons	**-.0310993**	**.0103238**	**-3.01**	**0.003**	**-.0513336**	**-.010865**
C17 _cons	**-.016477**	**.012616**	**-1.31**	**0.192**	**-.0412039**	**.0082499**
C21 _cons	**.0745648**	**.0043235**	**17.25**	**0.000**	**.0660909**	**.0830387**
C22 _cons	**.0091458**	**.0060803**	**1.50**	**0.133**	**-.0027714**	**.021063**
C23 _cons	**.0011757**	**.0015199**	**0.77**	**0.439**	**-.0018034**	**.0041547**
C24 _cons	**-.0014471**	**.0015356**	**-0.94**	**0.346**	**-.0044569**	**.0015627**
C25 _cons	**-.0070169**	**.0024709**	**-2.84**	**0.005**	**-.0118598**	**-.0021739**
C26 _cons	**-.0020237**	**.0016287**	**-1.24**	**0.214**	**-.0052159**	**.0011685**
C27 _cons	**.0018105**	**.0018912**	**0.96**	**0.338**	**-.0018961**	**.0055171**
C31 _cons	**.3289279**	**.0320096**	**10.28**	**0.000**	**.2661903**	**.3916655**
C32 _cons	**-.0821747**	**.0497592**	**-1.65**	**0.099**	**-.1797009**	**.0153515**
C33 _cons	**.0003437**	**.011488**	**0.03**	**0.976**	**-.0221723**	**.0228598**
C34 _cons	**-.0084326**	**.0152303**	**-0.55**	**0.580**	**-.0382835**	**.0214183**
C35 _cons	**-.021509**	**.0187276**	**-1.15**	**0.251**	**-.0582145**	**.0151965**
C36 _cons	**-.0447847**	**.0122012**	**-3.67**	**0.000**	**-.0686986**	**-.0208707**
C37 _cons	**-.0147152**	**.0161792**	**-0.91**	**0.363**	**-.0464259**	**.0169955**

d1						
	_cons	**-.0960609**	**.0061806**	**-15.54**	**0.000**	**-.1081747**	**-.0839471**
b1						
	_cons	**.4310358**	**.017085**	**25.23**	**0.000**	**.3975498**	**.4645217**
b2						
	_cons	**.0769488**	**.0031938**	**24.09**	**0.000**	**.0706891**	**.0832084**
A11						
	one	**.2963482**	**.0995589**	**2.98**	**0.003**	**.1012164**	**.4914801**
	/d2	**-.0141362**	**.0023787**	**-5.94**	**0.000**	**-.0187984**	**-.0094739**

Instruments for equation **eq1_FE:** **__000004 __00000H __00000S __000013 __00001L**
__00001W __000027 __00002P __000030 __00003B __00003T __000044 __00004F
__00004X __000058 __00005J __000061 __00006C __00006N __000075 __00007G
__00007R __000005 __0000JJ __0000JK __0000JL __0000JM __0000JN __0000JO
__0000JP __0000JQ __0000JR __0000E2 __0000E4 __0000E6 __0000EB __0000ED
__0000EF __0000EK __0000EM __0000EO __0000EU __0000EW __0000EY __0000F3
__0000F5 __0000F7 __0000FC __0000FE __0000FG __0000FM __0000FO __0000FQ
__0000FV __0000FX __0000FZ __0000G4 __0000G6 __0000G8 __0000GE __0000GG
__0000GI __0000GN __0000GP __0000GR __0000GW __0000GY __0000H0 __0000H6
__0000H8 __0000HA __0000HF __0000HH __0000HJ __0000HO __0000HQ __0000HS
__0000HY __0000I0 __0000I2 __0000I7 __0000I9 __0000IB __0000IG __0000II
__0000IK __0000IQ __0000IS __0000IU __0000IZ __0000J1 __0000J3 __0000J8
__0000JA __0000JC __0000DZ __0000E8 __0000EH __0000ER __0000F0 __0000F9
__0000FJ __0000FS __0000G1 __0000GB __0000GK __0000GT __0000H3 __0000HC
__0000HL __0000HV __0000I4 __0000ID __0000IN __0000IW __0000J5 _cons
Instruments for equation **eq2_FE:** **__000004 __00000H __00000S __000013 __00001L**
__00001W __000027 __00002P __000030 __00003B __00003T __000044 __00004F
__00004X __000058 __00005J __000061 __00006C __00006N __000075 __00007G
__00007R __000005 __0000JJ __0000JK __0000JL __0000JM __0000JN __0000JO
__0000JP __0000JQ __0000JR __0000E2 __0000E4 __0000E6 __0000EB __0000ED
__0000EF __0000EK __0000EM __0000EO __0000EU __0000EW __0000EY __0000F3
__0000F5 __0000F7 __0000FC __0000FE __0000FG __0000FM __0000FO __0000FQ
__0000FV __0000FX __0000FZ __0000G4 __0000G6 __0000G8 __0000GE __0000GG
__0000GI __0000GN __0000GP __0000GR __0000GW __0000GY __0000H0 __0000H6
__0000H8 __0000HA __0000HF __0000HH __0000HJ __0000HO __0000HQ __0000HS
__0000HY __0000I0 __0000I2 __0000I7 __0000I9 __0000IB __0000IG __0000II
__0000IK __0000IQ __0000IS __0000IU __0000IZ __0000J1 __0000J3 __0000J8
__0000JA __0000JC __0000DZ __0000E8 __0000EH __0000ER __0000F0 __0000F9
__0000FJ __0000FS __0000G1 __0000GB __0000GK __0000GT __0000H3 __0000HC
__0000HL __0000HV __0000I4 __0000ID __0000IN __0000IW __0000J5 _cons

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
```
(file **diag_3good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactIns > t_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex** not found)
(output written to diag_3good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex)
```
1242
1243        estat overid
```

Test of overidentifying restriction:

Hansen's J chi2(**208**) = **4781.02** (p = **0.0000**)

```
1244        di "$S_DATE $S_TIME"
```
9 Dec 2023 17:33:44

```
1245        unique hhidi
  Number of unique values of hhidi is  24757
  Number of records is  128640
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  24757
1248        unique group_round
  Number of unique values of group_round is  4599
  Number of records is  128640
1249        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4599
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  5.3831268
1251        estimates save "$filename_FE", replace
  (note: file diag_3good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_ex
  > actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster not found)
  file diag_3good_norm_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst
  > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}

1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
```

```
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_table7_24.smcl
   log type:  smcl
  closed on:   9 Dec 2023, 17:33:44
```

A.8 Table 8

A.8.1 Columns 1 and 4

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\rep_table8_col1.
> smcl
  log type:  smcl
 opened on:   8 Dec 2023, 14:38:20

1 .
2 . // wrapper for analysis
3 .
4 . *global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"
5 . *global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"
6 . global ROOT "C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication"

7 .
8 . global CODE "$ROOT/code"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=0

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
  >                                                    change these according to what c
  > olumn we are doing (uncomment below)
50. *local group_def "dist"
51. local group_def "fsuXseg"

52. local group_def "fsuXsegXreligXsched"

53.
54. // comparision group
55. local groupComp_def ""

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.          syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
70.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,120   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\mdelma\OneDrive -
      TUNI.fi\Tiedostot\replication/data/laspeyres_state_reshape.dta saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication/data/laspeyres_state_2cats_reshape.dta saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

    Result                      Number of obs
    -----------------------------------------
    Not matched                             0
    Matched                           234,590  (_merge==3)
    -----------------------------------------

114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "

    Result                      Number of obs
    -----------------------------------------
    Not matched                             0
    Matched                           234,590  (_merge==3)
    -----------------------------------------

116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"
```

```
117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)

127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964**	**29,621**	**124,591**	**39,414**	**234,590**

```
            |    100.00     100.00     100.00     100.00 |    100.00

129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,252 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,011 observations deleted)

159 drop if hhsize>12
  (684 observations deleted)
```

```
160 gen married=marstat==2

161 drop if age<20
  (1,192 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d
```

weight

Percentiles		Smallest		
1%	.004815	.0000482		
5%	.0180562	.0000482		
10%	.0337049	.0000482	Obs	87,172
25%	.1023184	.0001445	Sum of wgt.	87,172
50%	.3490865		Mean	.6966955
		Largest	Std. dev.	1.042372
75%	.909351	26.1213		
90%	1.719447	26.1213	Variance	1.08654
95%	2.45045	29.15594	Skewness	5.872976
99%	4.554976	29.52562	Kurtosis	81.60083

```
166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,335 real changes made)

168
169 g lux=ilux+vlux

170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	14,474	16.76	16.76
2	8,384	9.71	26.46
3	15,490	17.93	44.40
4	11,821	13.69	58.08
5	14,247	16.49	74.58
6	21,957	25.42	100.00
Total	86,373	100.00	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
```

```
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,372 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,625 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)

195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,625 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb
```

```
213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,372 missing values generated)
224         g land=exp(demog23)
  (12,746 missing values generated)
225         replace z4=0 if land<=0.005
  (29,505 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,121 real changes made)
227         g z5=demog24
  (30 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
```

```
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }

297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }

316
317 if short_zlist==6 {
```

```
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }

333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,746 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
```

```
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]
```

Variable	Obs	Weight	Mean	Std. dev.	Min	Max
expenditure	**73,598**	**45470.3285**	**4589.343**	**2838.635**	**308.7667**	**28976.34**

```
375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }

384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }
```

```
397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }

424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
```

```
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,598 real changes made)

460 replace x2=x*x
  (73,598 real changes made)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }

464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)
```

```
470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,598 real changes made)

478 replace x2=x2_temp
  (73,598 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (48,790 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)

488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.         rename q`j' qi`j'
    3.         capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
```

```
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\mdelma\AppData\Local\Temp\ST_2c78_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                  qui su `var'
    3.                  //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.          }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.          rename qi`j' qk`j'
    3.          capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk
```

```
530
531 forval t=1/$T {
    2.          rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,808 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
549 forval j=1/4 {
    2.          g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.          capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
```

```
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                          qui su `var'
    3.                          //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                          file write sumstat "`sumstats_row_`var''" _n
    5.                  }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1
```

```
601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1

625 scalar estimate_FE=1

626
627 // estimation options
628 global w_initial "winit(unadjusted)"

629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

630 global wmatrix "wmatrix(robust)"

631 global trace_level ""

632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

633 global technique ""

634 global maxiter ""

635 global technique_RE ""

636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

637 global derivatives "quickderivatives"

638 scalar init_FE_from_RE=1
```

```
639 global clust_var "state_district_round"

640 local clus_name "_clus_Dist"

641
642
643
644
645
646
647
648
649 // xxzx to keep names same
650 local wmatrix_desc ""

651 local sectorName "_urbOnly"

652 local dropZdesc "_dropPre"

653 local desc_struc_v0 ""

654 local inst_desc "_exactInst"

655 local inst_rootp "_noPPXdInst"

656 local weight_desc ""

657 local RE_sq_inst_desc ""

658 local cnstname ""

659 local actual_pqhat_desc ""

660 local expUpname "_updateExp"

661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
```

```
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //              ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T

694 local Tm1=`T'-1

695 local Tindiv : word count $zlist_indiv

696
697 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
706
707 if `J'==2 {
708         replace p1=P1
  (128,974 real changes made)
```

```
709         replace p2=P2
  (128,974 real changes made)
710         replace qi1=Qi1
  (128,974 real changes made)
711         replace qk1=Qk1
  (128,974 real changes made)
712         replace qikbar1=Qikbar1
  (128,974 real changes made)
713         replace qi2=Qi2
  (128,974 real changes made)
714         replace qk2=Qk2
  (128,974 real changes made)
715         replace qikbar2=Qikbar2
  (128,974 real changes made)
716
717         foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                  qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721         forv j = 1(1)3 {
    2.                  replace p`j'=P`j'
    3.                  replace qi`j'=Qi`j'
    4.                  replace qk`j'=Qk`j'
    5.                  replace qikbar`j'=Qikbar`j'
    6.          }
722
723         foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                  qui replace `var'=.
    3.          }
724 }

725
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
```

```
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }

751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"
```

```
759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
773         local Drp`j' "`Drp`j'' )"
   15.          di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1      {
779         local pC "0"
780         local pC1 "0"
781         local pCi "0"
```

```
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }

798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"
```

```
805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.          local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.          if `jplus1'<=`J' {
    5.                  forval k=`jplus1'/`J' {
    6.                          local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                  }
    8.          }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                               local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                  if noMeasError==1                       local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'              - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                              l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                ))"
    3.          if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                        )
  > )"
    4. }
```

```
828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2
  > ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) )*{b1} + ({A11:}*p1*qikba
  > r1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d
  > 1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))*({b1} - 2*(( ({A11:
  > }*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         ))

836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                 local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                 local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.         }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                 forval l=1/`J' {
    3.                         local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                         local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                 }
    6.         }
```

```
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                                "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                    "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                    "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0}))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                          "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) +
  >  ({A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C
  > 2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikb
  > ar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:
  > }) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C
  > 2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qi
  > kbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )
  > *{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) )
```

```
873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > one}*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}
  > *ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi
  > 4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p
  > 1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         )
  > )

874
875 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi

882 g double `ximxk'=xi-xk

883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.         tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.         g double `xizi`t''=xi*zi`t'
    4.         g double `xkzk`t''=xk*zk`t'
    5.         g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.         g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.         g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.         g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.         g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.         forval j=1/`J' {
   11.                 tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                 g double `zi`t'p`j''= zi`t'*p`j'
   13.                 g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                 g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                 g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                 forv s=1/`T' {
   17.                         tempvar zi`t'zi`s'p`j'
   18.                         g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                 }
   20.         }
   21. }
```

```
891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }

903
904 local zrg ""

905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var'"
    6.          }
    7. }

906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var'"
    5.
911         forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var'"
   10.                  local xprg "`xprg' `xp`j'`var'"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"
```

```
917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""

925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }

929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.          forval k=`j'/`J' {
    3.                  tempvar p`j'p`k'
    4.                  g `p`j'p`k''=p`j'*p`k'
    5.                  local roundinsts "`roundinsts' `p`j'p`k''"
    6.          }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"
```

```
938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.          if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.          if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
951          tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.          predict `qikhat`j''
    6.          summ `qikhat`j''
    7.
952          // use all prices times all qikhats as instruments, interacted with xi
953          forval s=1/`J' {
    8.                  tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                  g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                  g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                  g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                  local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                  local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.          }
   16. }
```

Linear regression

Number of obs	= **128,974**
F(129, 4606)	= **26.14**
Prob > F	= **0.0000**
R-squared	= **0.3152**
Root MSE	= **.2269**

(Std. err. adjusted for **4,607** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**13.69334**	**13.55249**	**1.01**	**0.312**	**-12.87603**	**40.26271**
__00009L	**-1.110131**	**5.877016**	**-0.19**	**0.850**	**-12.6319**	**10.41164**
__00009O	**-4.733269**	**10.60998**	**-0.45**	**0.656**	**-25.53392**	**16.06738**
xi	**.1542528**	**.0129684**	**11.89**	**0.000**	**.1288285**	**.179677**
__000002	**-.0112918**	**.004482**	**-2.52**	**0.012**	**-.0200787**	**-.0025048**
__00000E	**-.194703**	**.1469556**	**-1.32**	**0.185**	**-.4828064**	**.0934004**
__00000P	**-.2144092**	**.1437722**	**-1.49**	**0.136**	**-.4962716**	**.0674532**
__000017	**-.3329652**	**.3024964**	**-1.10**	**0.271**	**-.926003**	**.2600726**
__00001I	**.8297193**	**.3175878**	**2.61**	**0.009**	**.207095**	**1.452344**
__000020	**.0387541**	**.069878**	**0.55**	**0.579**	**-.0982402**	**.1757484**
__00002B	**.0940218**	**.0729821**	**1.29**	**0.198**	**-.0490581**	**.2371017**
__00002T	**.3858779**	**.1029798**	**3.75**	**0.000**	**.1839882**	**.5877676**
__000034	**-.0583253**	**.0942387**	**-0.62**	**0.536**	**-.2430783**	**.1264276**
__00003M	**-.0332164**	**.0891579**	**-0.37**	**0.709**	**-.2080087**	**.1415758**

__00003X	.0170335	.0856899	0.20	0.842	-.1509598	.1850268
__00004F	-.178486	.0702955	-2.54	0.011	-.3162989	-.0406731
__00004Q	.1342376	.0690241	1.94	0.052	-.0010827	.2695578
__000058	-.2802252	.103477	-2.71	0.007	-.4830898	-.0773606
__00005J	.2540516	.1083275	2.35	0.019	.0416778	.4664255
__00005U	-1.676831	.6485308	-2.59	0.010	-2.948262	-.4053997
__00005V	-.0863972	.1131713	-0.76	0.445	-.3082672	.1354728
__00005W	.5447908	.2574777	2.12	0.034	.0400112	1.04957
__00005X	1.402448	.6639915	2.11	0.035	.100706	2.704189
__00005Y	-.0301473	.1387529	-0.22	0.828	-.3021695	.2418749
__00005Z	.5699259	.2445907	2.33	0.020	.0904109	1.049441
__000060	-.1600202	.0966932	-1.65	0.098	-.3495851	.0295448
__000061	.0977049	.0575434	1.70	0.090	-.0151078	.2105176
__000062	.0428928	.0429226	1.00	0.318	-.041256	.1270416
__000063	.0701739	.0706444	0.99	0.321	-.0683231	.2086708
__000064	.030134	.0833975	0.36	0.718	-.1333651	.1936331
__000065	3.106176	1.600722	1.94	0.052	-.0320064	6.244358
__000066	.6258157	.2514537	2.49	0.013	.132846	1.118785
__000067	-1.750687	.6275414	-2.79	0.005	-2.980969	-.5204052
__000068	-1.70795	1.514265	-1.13	0.259	-4.676635	1.260735
__000069	-.6392621	.2880699	-2.22	0.027	-1.204017	-.0745071
__00006A	-1.447494	.5331064	-2.72	0.007	-2.492638	-.4023496
__00006B	.1143157	.1861472	0.61	0.539	-.2506221	.4792534
__00006C	-.6737508	.1248459	-5.40	0.000	-.9185085	-.428993
__00006D	.0086179	.0904776	0.10	0.924	-.1687615	.1859973
__00006E	-.1547435	.1558768	-0.99	0.321	-.4603367	.1508497
__00006F	.0079178	.1887856	0.04	0.967	-.3621924	.3780279
__00006G	.1841894	.3353829	0.55	0.583	-.4733217	.8417006
__00006H	.0818198	.0556612	1.47	0.142	-.0273029	.1909424
__00006I	-.1449273	.1360559	-1.07	0.287	-.4116621	.1218075
__00006J	-.3319474	.3416688	-0.97	0.331	-1.001782	.337887
__00006K	.115213	.0734683	1.57	0.117	-.02882	.259246
__00006L	-.1232845	.1459414	-0.84	0.398	-.4093994	.1628305
__00006M	.0164437	.0526674	0.31	0.755	-.0868095	.1196969
__00006N	.0152068	.030842	0.49	0.622	-.0452583	.075672
__00006O	.0227044	.0225757	1.01	0.315	-.0215547	.0669635
__00006P	-.03007	.0389464	-0.77	0.440	-.1064237	.0462836
__00006Q	.0502646	.045697	1.10	0.271	-.0393233	.1398526
__00006R	-.3284892	.536709	-0.61	0.541	-1.380696	.7237175
__00006S	.0072765	.0674667	0.11	0.914	-.1249907	.1395436
__00006T	.2145689	.1812298	1.18	0.236	-.1407283	.5698661
__00006U	-.3585753	.5151685	-0.70	0.486	-1.368552	.6514019
__00006V	.1153582	.0976867	1.18	0.238	-.0761546	.306871
__00006W	.1530382	.1846788	0.83	0.407	-.2090208	.5150973
__00006X	.0084402	.0695142	0.12	0.903	-.127841	.1447214
__00006Y	.0378441	.0427916	0.88	0.377	-.046048	.1217362
__00006Z	-.0854709	.0295355	-2.89	0.004	-.1433746	-.0275673
__000070	-.0236411	.051811	-0.46	0.648	-.1252155	.0779333
__000071	-.0082357	.0619905	-0.13	0.894	-.1297667	.1132953
__000072	-.5957069	.4286418	-1.39	0.165	-1.43605	.2446365
__000073	-.1199411	.0650262	-1.84	0.065	-.2474236	.0075414
__000074	.3563352	.1760313	2.02	0.043	.0112296	.7014408
__000075	.3234496	.418276	0.77	0.439	-.4965718	1.143471
__000076	.1783071	.0847964	2.10	0.036	.0120656	.3445486
__000077	.1804002	.1579553	1.14	0.253	-.1292679	.4900683
__000078	-.1174549	.0606993	-1.94	0.053	-.2364547	.0015448
__000079	.0307615	.0359028	0.86	0.392	-.0396252	.1011481
__00007A	.0546076	.0275187	1.98	0.047	.0006579	.1085574
__00007B	.0309967	.0445767	0.70	0.487	-.0563949	.1183884
__00007C	.0129862	.0536016	0.24	0.809	-.0920985	.118071
__00007D	.0407402	.3212664	0.13	0.899	-.5890959	.6705762
__00007E	.019936	.0495537	0.40	0.687	-.077213	.117085
__00007F	-.0389346	.1292091	-0.30	0.763	-.2922464	.2143772
__00007G	.1274936	.3220552	0.40	0.692	-.5038889	.7588762
__00007H	-.0984334	.064691	-1.52	0.128	-.2252587	.0283919
__00007I	-.0914847	.1220888	-0.75	0.454	-.3308373	.1478679
__00007J	.0234645	.0429728	0.55	0.585	-.0607829	.1077119
__00007K	-.1039152	.0309415	-3.36	0.001	-.1645753	-.0432552
__00007L	-.0160361	.0218201	-0.73	0.462	-.058814	.0267417
__00007M	-.0229297	.0359033	-0.64	0.523	-.0933175	.047458
__00007N	.0036027	.0419515	0.09	0.932	-.0786424	.0858478
__00007O	-.3739519	.5188624	-0.72	0.471	-1.391171	.643267

__00007P	.1133294	.0696332	1.63	0.104	-.0231851	.2498439
__00007Q	-.0568902	.1923662	-0.30	0.767	-.4340202	.3202398
__00007R	.6919887	.5226084	1.32	0.186	-.3325741	1.716552
__00007S	-.2403262	.1006609	-2.39	0.017	-.4376698	-.0429826
__00007T	-.6408032	.1863361	-3.44	0.001	-1.006111	-.2754952
__00007U	.098694	.0664232	1.49	0.137	-.0315273	.2289154
__00007V	-.278446	.0425803	-6.54	0.000	-.3619237	-.1949682
__00007W	-.0891032	.0337654	-2.64	0.008	-.1552996	-.0229069
__00007X	-.0370522	.0549492	-0.67	0.500	-.144779	.0706746
__00007Y	-.0619248	.0654583	-0.95	0.344	-.1902545	.0664049
__000080	25.625	31.95096	0.80	0.423	-37.01418	88.26419
__000082	-5.487622	24.61323	-0.22	0.824	-53.74134	42.7661
__000085	-.7120201	.7512535	-0.95	0.343	-2.184837	.7607967
__000087	1.062719	.7348901	1.45	0.148	-.3780172	2.503456
__00008A	-2.431463	6.483566	-0.38	0.708	-15.14236	10.27943
__00008C	-2.000707	4.911279	-0.41	0.684	-11.62917	7.627752
__00008F	-38.58905	38.79451	-0.99	0.320	-114.6449	37.46679
__00008H	13.79728	30.14331	0.46	0.647	-45.29805	72.89262
__00008K	1.878918	.8833347	2.13	0.033	.1471584	3.610677
__00008M	-4.469693	1.107196	-4.04	0.000	-6.640327	-2.299058
__00008P	3.740991	1.752229	2.13	0.033	.3057824	7.176199
__00008R	-1.04097	1.838004	-0.57	0.571	-4.644338	2.562398
__00008U	.3363507	.6456738	0.52	0.602	-.9294793	1.602181
__00008W	.2672875	.648201	0.41	0.680	-1.003497	1.538072
__00008Z	1.136905	.388594	2.93	0.003	.3750744	1.898735
__000091	-.6608855	.2932888	-2.25	0.024	-1.235872	-.0858989
__000094	-.2911185	.2573144	-1.13	0.258	-.795578	.2133411
__000096	-1.281662	.2694973	-4.76	0.000	-1.810006	-.7533184
__000099	.587976	.4582331	1.28	0.200	-.3103803	1.486332
__00009B	.1132766	.5157609	0.22	0.826	-.8978619	1.124415
__00009E	.6410662	.6060613	1.06	0.290	-.5471044	1.829237
__00009G	-1.030265	.6295528	-1.64	0.102	-2.26449	.20396
xbarlag	-18.20947	25.29013	-0.72	0.472	-67.79024	31.3713
x2barlag	-.8002884	.6917998	-1.16	0.247	-2.156547	.5559707
xbar2lag	5.035549	4.963205	1.01	0.310	-4.69471	14.76581
rootxbarlag	21.43046	31.33555	0.68	0.494	-40.00223	82.86315
z1barlag	2.904258	1.001333	2.90	0.004	.9411651	4.867351
z2barlag	-2.214254	1.610643	-1.37	0.169	-5.371886	.9433787
z3barlag	-.6676996	.5272239	-1.27	0.205	-1.701311	.3659118
z4barlag	-.1618506	.2961068	-0.55	0.585	-.7423617	.4186605
z5barlag	1.6909	.2935116	5.76	0.000	1.115477	2.266324
z6barlag	-.6622385	.4784714	-1.38	0.166	-1.600272	.2757948
z7barlag	.3017138	.5132942	0.59	0.557	-.7045888	1.308016
_cons	-6.518304	10.8535	-0.60	0.548	-27.79637	14.75976

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	128,974	.3624343	.153863	.0124187	1.178719

Linear regression

Number of obs	=	128,974
F(129, 4606)	=	25.46
Prob > F	=	0.0000
R-squared	=	0.2757
Root MSE	=	.15342

(Std. err. adjusted for **4,607** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	11.04298	9.882496	1.12	0.264	-8.331449	30.41741
__00009L	-11.08456	3.819238	-2.90	0.004	-18.57209	-3.597022
__00009O	-12.25048	8.495736	-1.44	0.149	-28.9062	4.405231
xi	.1261209	.0081986	15.38	0.000	.1100478	.1421941
__000002	-.0183007	.0025358	-7.22	0.000	-.0232721	-.0133293
__00000E	-.1716009	.1124381	-1.53	0.127	-.3920335	.0488317
__00000P	-.0550169	.1116381	-0.49	0.622	-.2738811	.1638472
__000017	.0164436	.1995102	0.08	0.934	-.374692	.4075792
__00001I	.2583078	.2393828	1.08	0.281	-.2109971	.7276127
__000020	.000019	.0513655	0.00	1.000	-.1006819	.10072

__00002B	.0240214	.0536677	0.45	0.654	-.0811931	.1292358
__00002T	.1256433	.089026	1.41	0.158	-.0488902	.3001769
__000034	-.0738162	.0809484	-0.91	0.362	-.232514	.0848815
__00003M	-.0777664	.0685109	-1.14	0.256	-.2120806	.0565479
__00003X	.0386064	.0736734	0.52	0.600	-.1058289	.1830416
__00004F	-.1021838	.0483561	-2.11	0.035	-.196985	-.0073826
__00004Q	.0141293	.0525069	0.27	0.788	-.0888094	.117068
__000058	-.1444148	.06621	-2.18	0.029	-.2742181	-.0146116
__00005J	.0880692	.072095	1.22	0.222	-.0532714	.2294098
__00005U	-.2820775	.4738934	-0.60	0.552	-1.211136	.6469805
__00005V	-.08005	.0820701	-0.98	0.329	-.2409468	.0808468
__00005W	.0706916	.1961874	0.36	0.719	-.3139297	.4553129
__00005X	.7923297	.4957958	1.60	0.110	-.1796677	1.764327
__00005Y	-.0629549	.1139345	-0.55	0.581	-.2863212	.1604114
__00005Z	-.2370382	.1983777	-1.19	0.232	-.6259536	.1518772
__000060	-.1554869	.0922199	-1.69	0.092	-.3362822	.0253083
__000061	.0284558	.058965	0.48	0.629	-.0871439	.1440556
__000062	.0292613	.0355138	0.82	0.410	-.0403627	.0988853
__000063	-.01966	.0556613	-0.35	0.724	-.1287829	.0894629
__000064	-.1020435	.0671704	-1.52	0.129	-.2337296	.0296426
__000065	.2949826	1.045546	0.28	0.778	-1.754789	2.344754
__000066	.291117	.1715777	1.70	0.090	-.0452575	.6274914
__000067	-.5608972	.417833	-1.34	0.180	-1.38005	.2582557
__000068	.3062338	1.060624	0.29	0.773	-1.773098	2.385566
__000069	-.3091697	.2090238	-1.48	0.139	-.7189564	.1006171
__00006A	-.7119156	.4063879	-1.75	0.080	-1.508631	.0847994
__00006B	-.1469194	.150338	-0.98	0.328	-.4416539	.147815
__00006C	-.2425035	.1219733	-1.99	0.047	-.4816297	-.0033773
__00006D	-.0395717	.0598117	-0.66	0.508	-.1568313	.077688
__00006E	-.2197359	.1221078	-1.80	0.072	-.4591256	.0196538
__00006F	-.0841948	.1438272	-0.59	0.558	-.3661649	.1977754
__00006G	-.2447193	.2350442	-1.04	0.298	-.7055185	.2160798
__00006H	.0620903	.0347608	1.79	0.074	-.0060575	.1302382
__00006I	-.0274007	.0882244	-0.31	0.756	-.2003629	.1455614
__00006J	.2019808	.2557185	0.79	0.430	-.29935	.7033115
__00006K	.0690481	.0561443	1.23	0.219	-.0410215	.1791178
__00006L	-.0836018	.1006057	-0.83	0.406	-.2808373	.1136336
__00006M	-.0448285	.0517783	-0.87	0.387	-.1463388	.0566817
__00006N	.0114031	.0279966	0.41	0.684	-.0434836	.0662898
__00006O	-.0213436	.0152649	-1.40	0.162	-.0512701	.0085829
__00006P	-.02569	.0290026	-0.89	0.376	-.082549	.0311691
__00006Q	.0092866	.0354489	0.26	0.793	-.0602101	.0787834
__00006R	-.3521608	.5453736	-0.65	0.518	-1.421354	.7170329
__00006S	-.0771044	.0560862	-1.37	0.169	-.1870603	.0328515
__00006T	.2398658	.1915329	1.25	0.211	-.1356304	.6153621
__00006U	.2286432	.5083884	0.45	0.653	-.7680417	1.225328
__00006V	-.1014478	.0788211	-1.29	0.198	-.255975	.0530794
__00006W	-.1165861	.1699335	-0.69	0.493	-.4497372	.2165649
__00006X	.0125101	.0671144	0.19	0.852	-.1190663	.1440866
__00006Y	.0692888	.0419454	1.65	0.099	-.0129443	.1515219
__00006Z	-.0687698	.0274601	-2.50	0.012	-.1226047	-.0149349
__000070	.0221093	.049956	0.44	0.658	-.0758284	.120047
__000071	-.0382494	.0601072	-0.64	0.525	-.1560884	.0795896
__000072	-.5115537	.3361516	-1.52	0.128	-1.170572	.1474645
__000073	-.0962537	.0491302	-1.96	0.050	-.1925724	.000065
__000074	.3133667	.1316176	2.38	0.017	.0553332	.5714002
__000075	.3168034	.329926	0.96	0.337	-.3300095	.9636164
__000076	-.001279	.0708961	-0.02	0.986	-.1402693	.1377113
__000077	.1030017	.1278116	0.81	0.420	-.1475703	.3535738
__000078	.0031095	.0501568	0.06	0.951	-.0952218	.1014408
__000079	.0395355	.0392165	1.01	0.313	-.0373476	.1164186
__00007A	.0356331	.0222577	1.60	0.109	-.0080027	.0792689
__00007B	.0044644	.035547	0.13	0.900	-.0652247	.0741535
__00007C	-.0792193	.0439844	-1.80	0.072	-.1654499	.0070113
__00007D	-.2069261	.2290524	-0.90	0.366	-.6559785	.2421264
__00007E	-.0355108	.0390832	-0.91	0.364	-.1121326	.0411109
__00007F	.0630469	.0946748	0.67	0.505	-.1225611	.2486549
__00007G	.3391181	.2416332	1.40	0.161	-.1345988	.812835
__00007H	-.0455141	.0543385	-0.84	0.402	-.1520437	.0610155
__00007I	-.0883664	.1032688	-0.86	0.392	-.2908228	.1140899
__00007J	.005779	.0362662	0.16	0.873	-.0653201	.0768781
__00007K	-.0267187	.0308722	-0.87	0.387	-.0872429	.0338056

__00007L	.0080626	.0164195	0.49	0.623	-.0241275	.0402528
__00007M	-.0116627	.0308219	-0.38	0.705	-.0720884	.0487631
__00007N	.0047715	.0361935	0.13	0.895	-.0661852	.0757282
__00007O	-.2264556	.3645559	-0.62	0.535	-.9411598	.4882486
__00007P	.0542315	.0483483	1.12	0.262	-.0405544	.1490174
__00007Q	-.0519681	.1373552	-0.38	0.705	-.3212501	.217314
__00007R	.412765	.362169	1.14	0.254	-.2972598	1.12279
__00007S	-.0692891	.0727322	-0.95	0.341	-.211879	.0733009
__00007T	-.315609	.1417789	-2.23	0.026	-.5935636	-.0376544
__00007U	.0378757	.049846	0.76	0.447	-.0598464	.1355977
__00007V	-.0884664	.037675	-2.35	0.019	-.1623274	-.0146054
__00007W	-.0316875	.0214115	-1.48	0.139	-.0736644	.0102894
__00007X	-.0301939	.040703	-0.74	0.458	-.1099911	.0496034
__00007Y	-.0117595	.0485999	-0.24	0.809	-.1070385	.0835196
__000080	11.89749	22.44002	0.53	0.596	-32.0957	55.89069
__000082	-46.67611	19.92086	-2.34	0.019	-85.73054	-7.621676
__000085	-1.426598	.5315055	-2.68	0.007	-2.468604	-.384593
__000087	.6423641	.5649073	1.14	0.256	-.4651248	1.749853
__00008A	-.1264639	4.53491	-0.03	0.978	-9.017061	8.764133
__00008C	8.227376	3.975914	2.07	0.039	.4326801	16.02207
__00008F	-16.15495	27.79975	-0.58	0.561	-70.65578	38.34588
__00008H	58.60442	24.52452	2.39	0.017	10.52461	106.6842
__00008K	-.9120177	.7389375	-1.23	0.217	-2.360689	.5366539
__00008M	-.800309	.9095271	-0.88	0.379	-2.583418	.9828
__00008P	2.32448	1.476553	1.57	0.115	-.5702716	5.219231
__00008R	-7.829535	1.492813	-5.24	0.000	-10.75616	-4.902905
__00008U	.4275766	.5444134	0.79	0.432	-.6397345	1.494888
__00008W	-.39655	.582238	-0.68	0.496	-1.538015	.7449154
__00008Z	1.100479	.3587513	3.07	0.002	.3971544	1.803803
__000091	.2442068	.3268234	0.75	0.455	-.3965236	.8849373
__000094	.8859246	.2010314	4.41	0.000	.4918068	1.280042
__000096	.1079339	.2347101	0.46	0.646	-.3522103	.5680782
__000099	.0901705	.3725852	0.24	0.809	-.640275	.820616
__00009B	.2050926	.4097797	0.50	0.617	-.598272	1.008457
__00009E	.1544237	.4927222	0.31	0.754	-.811548	1.120395
__00009G	-.5467409	.495705	-1.10	0.270	-1.51856	.4250784
xbarlag	42.513	19.05995	2.23	0.026	5.146363	79.87964
x2barlag	.5490172	.5000276	1.10	0.272	-.4312766	1.529311
xbar2lag	-9.282651	3.734373	-2.49	0.013	-16.60381	-1.96149
rootxbarlag	-51.85661	23.70854	-2.19	0.029	-98.3367	-5.376518
z1barlag	1.961006	.9031768	2.17	0.030	.1903468	3.731665
z2barlag	6.476994	1.536294	4.22	0.000	3.465122	9.488866
z3barlag	.0634618	.5429013	0.12	0.907	-1.000885	1.127808
z4barlag	-1.258378	.3340826	-3.77	0.000	-1.913339	-.6034156
z5barlag	-1.070814	.243176	-4.40	0.000	-1.547555	-.5940721
z6barlag	-.264468	.4428088	-0.60	0.550	-1.132585	.6036494
z7barlag	.4378998	.5011147	0.87	0.382	-.5445251	1.420325
_cons	15.36738	8.150964	1.89	0.059	-.6124176	31.34717

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A2	128,974	.5445803	.094604	.2734714	1.004489

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"

957
958 * interact qhat_hats with Alist
```

```
959 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9

960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
964 }

965
966 local r_RE      "`r_RE' `qhat_hats'"

967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }

978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS		
Model	**16809.4556**	**13**	**1293.03505**	Number of obs	= **128,974**
Residual	**1434.97936**	**128,960**	**.011127321**	F(13, 128960)	> **99999.00**
				Prob > F	= **0.0000**
				R-squared	= **0.9213**
				Adj R-squared	= **0.9213**
Total	**18244.435**	**128,973**	**.141459336**	Root MSE	= **.10549**

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	-.0165424	.0009775	-16.92	0.000	-.0184583	-.0146265
ybarx	-.0234522	.0010772	-21.77	0.000	-.0255636	-.0213409
xi2	.0556701	.000559	99.59	0.000	.0545744	.0567657
ybar	-.1895462	.0024781	-76.49	0.000	-.1944032	-.1846891
xi	.4878409	.0017809	273.92	0.000	.4843503	.4913315
ybar_q1	.4304893	.0029597	145.45	0.000	.4246883	.4362903
pzi1	-.3421396	.0016197	-211.24	0.000	-.3453142	-.338965
pzi2	-.0616591	.0027766	-22.21	0.000	-.0671012	-.056217
pzi3	-.0039179	.0008384	-4.67	0.000	-.0055611	-.0022747
pzi4	-.0210364	.0008325	-25.27	0.000	-.022668	-.0194047
pzi5	.0098744	.0009033	10.93	0.000	.0081041	.0116448
pzi6	-.0046044	.0007453	-6.18	0.000	-.0060652	-.0031436
pzi7	-.0016567	.0009535	-1.74	0.082	-.0035256	.0002122
_cons	.0531716	.0018763	28.34	0.000	.0494941	.0568491

```
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = _b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                local coef = _b[pzi`j']
    3.                local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1025}

1026
1027if `J'==3 & simple_model==0
  >                           local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"
```

```
1029if `J'==4 & simple_model==0
  >                           local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  (setting technique to dfp)
  Iteration 0:  GMM criterion Q(b) =   .0965146
  Iteration 1:  GMM criterion Q(b) =  .06342943
  Iteration 2:  GMM criterion Q(b) =  .02670546
  Iteration 3:  GMM criterion Q(b) =  .00445943
  Iteration 4:  GMM criterion Q(b) =  .00190604
  Iteration 5:  GMM criterion Q(b) =  .00075043
  Iteration 6:  GMM criterion Q(b) =  .00056255
  Iteration 7:  GMM criterion Q(b) =  .00038892
  Iteration 8:  GMM criterion Q(b) =  .00031527
  Iteration 9:  GMM criterion Q(b) =  .00028759

  Step 2
  (setting technique to dfp)
  Iteration 0:  GMM criterion Q(b) =  .07617522
  Iteration 1:  GMM criterion Q(b) =  .07613958
  Iteration 2:  GMM criterion Q(b) =  .06063142
  Iteration 3:  GMM criterion Q(b) =  .05635555
  Iteration 4:  GMM criterion Q(b) =   .0486192
  Iteration 5:  GMM criterion Q(b) =   .0338748
  Iteration 6:  GMM criterion Q(b) =  .03053555
  Iteration 7:  GMM criterion Q(b) =  .02464775
  Iteration 8:  GMM criterion Q(b) =  .02028452
  Iteration 9:  GMM criterion Q(b) =   .0180642
  (switching technique to nr)
  Iteration 10: GMM criterion Q(b) =  .01691984  (not concave)
  Iteration 11: GMM criterion Q(b) =  .01670006  (not concave)
  (switching technique to dfp)
  Iteration 12: GMM criterion Q(b) =  .01661212
  Iteration 13: GMM criterion Q(b) =  .01657646
  Iteration 14: GMM criterion Q(b) =  .01640743
  Iteration 15: GMM criterion Q(b) =  .01640205
  Iteration 16: GMM criterion Q(b) =  .01639863
  Iteration 17: GMM criterion Q(b) =  .01639028
  Iteration 18: GMM criterion Q(b) =  .01638641
  Iteration 19: GMM criterion Q(b) =  .01638441
  Iteration 20: GMM criterion Q(b) =   .0163807
  Iteration 21: GMM criterion Q(b) =  .01637836

  GMM estimation

  Number of parameters =  20
  Number of moments    =  28
  Initial weight matrix: Identity                  Number of obs   =    128,974
  GMM weight matrix:     Robust
```

(Std. err. adjusted for **564** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
A11						
one	**.5453506**	**.0840333**	**6.49**	**0.000**	**.3806484**	**.7100529**
C1						
zi1	**-.4056248**	**.2907586**	**-1.40**	**0.163**	**-.9755012**	**.1642517**
zi2	**.0543032**	**.3854934**	**0.14**	**0.888**	**-.7012501**	**.8098565**
zi3	**.148706**	**.0973851**	**1.53**	**0.127**	**-.0421653**	**.3395773**
zi4	**.0253351**	**.1446463**	**0.18**	**0.861**	**-.2581664**	**.3088366**
zi5	**.0304173**	**.1054496**	**0.29**	**0.773**	**-.17626**	**.2370947**
zi6	**.0748633**	**.076044**	**0.98**	**0.325**	**-.0741803**	**.2239069**
zi7	**.0504256**	**.1051159**	**0.48**	**0.631**	**-.1555978**	**.256449**
C2						
zi1	**.2353342**	**.174162**	**1.35**	**0.177**	**-.106017**	**.5766855**
zi2	**-.021696**	**.2445194**	**-0.09**	**0.929**	**-.5009451**	**.4575532**
zi3	**.0779403**	**.061264**	**1.27**	**0.203**	**-.0421349**	**.1980155**
zi4	**.0177379**	**.0841644**	**0.21**	**0.833**	**-.1472212**	**.182697**
zi5	**-.0111676**	**.06864**	**-0.16**	**0.871**	**-.1456995**	**.1233643**
zi6	**.0477505**	**.0486162**	**0.98**	**0.326**	**-.0475354**	**.1430365**
zi7	**.0236685**	**.0663797**	**0.36**	**0.721**	**-.1064332**	**.1537703**
/d1	**.018551**	**.0123512**	**1.50**	**0.133**	**-.0056569**	**.0427589**
/b1	**.5843175**	**.019919**	**29.33**	**0.000**	**.545277**	**.6233581**
/AVA11	**.0213824**	**10.04791**	**0.00**	**0.998**	**-19.67217**	**19.71493**
/AVA12	**-.0598241**	**9.472448**	**-0.01**	**0.995**	**-18.62548**	**18.50583**
/AVA22	**.1255027**	**9.453796**	**0.01**	**0.989**	**-18.4036**	**18.6546**

Instruments for equation **eq1_RE: __00009J __00009L __00009O xi __000002 __00000E __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5 __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons**

```
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                // calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
```

```
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst_noP
      > PXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not found)
  (output written to diag_2good_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_
  > exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24808
  Number of records is  128974

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
                 e(N_hh) =  24808

1077unique group_round
  Number of unique values of group_round is  4607
  Number of records is  128974

1078estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  4607
```

```
1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
         e(avg_grp_size) =  5.3848491

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactIn
  > st_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster not found)
  file diag_2good_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst_noPP
  > XdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
```

```
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*`p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
1119                                                    gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                  local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                          }
   30.                                  }
   31.                          }
   32.                  }
   33.          }
1120         if `Tindivp1'<=`T' {
1121                 forval t=`Tindivp1'/`T' {
    2.                           tempvar zi`t'xdiff
    3.                           g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                           local zg "`zg' zi`t'"
    5.                           forval j=1/`J' {
    6.                                   tempvar zi`t'p`j'xdiff
    7.                                   g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                   local pzg "`pzg' `zi`t'p`j''"
    9.                                   local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                           }
   11.                  }
1122         }
1123
1124         // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1125         // denoting zg as group-zs
1126         //  zg,pzgx,rprp,rprpx
1127         local rootprootp ""
1128         local rootprootpxdiff ""
1129
1130         forval j=1/`J' {
    2.                   forval s=`j'/`J' {
    3.                           tempvar xdiffrp`j'rp`s'
    4.                           g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1131                         if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1132                         local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                   }
    8.           }
1133
1134         local pqhatxdiff ""
1135         forval j=1/`J' {
    2.                   forval s=1/`J' {
    3.                           foreach name of global Alist {
    4.                           tempvar   p`s'qikhat`j'xdiff`name'
    5.                           g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
```

```
1136                                local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
1137
1138        // set instruments
1139        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1140}

1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                          local `++t'
    3.                          local p_nice = subinstr("`p'",":_cons","",.)
    4.                          local est = coefs[1,`t']
    5.                          if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                  }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
```

```
1173                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                  }
   16.          }
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1    `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
```

```
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00417214
  Iteration 1:  GMM criterion Q(b) =  .00183614
  Iteration 2:  GMM criterion Q(b) =  .00138584
  Iteration 3:  GMM criterion Q(b) =   .0004324
  Iteration 4:  GMM criterion Q(b) =  .00015064
  Iteration 5:  GMM criterion Q(b) =  2.379e-06

  Step 2
  Iteration 0:  GMM criterion Q(b) =   .0150805
  Iteration 1:  GMM criterion Q(b) =   .0147228
  Iteration 2:  GMM criterion Q(b) =  .01180045
  Iteration 3:  GMM criterion Q(b) =  .01177673

  GMM estimation

  Number of parameters =  17
  Number of moments    =  63
  Initial weight matrix: Identity                  Number of obs   =    128,974
  GMM weight matrix:     Robust
```

(Std. err. adjusted for **564** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11						
_cons	**.5409649**	**.0687737**	**7.87**	**0.000**	**.4061709**	**.6757589**
C12						
_cons	**-.0169266**	**.0905086**	**-0.19**	**0.852**	**-.1943201**	**.1604669**
C13						
_cons	**-.0202678**	**.0246512**	**-0.82**	**0.411**	**-.0685832**	**.0280476**
C14						
_cons	**.0317942**	**.0224419**	**1.42**	**0.157**	**-.0121911**	**.0757795**
C15						
_cons	**.0747022**	**.0420237**	**1.78**	**0.075**	**-.0076628**	**.1570672**
C16						
_cons	**-.0467624**	**.031507**	**-1.48**	**0.138**	**-.108515**	**.0149901**
C17						
_cons	**-.0072835**	**.0317732**	**-0.23**	**0.819**	**-.0695578**	**.0549908**
C21						
_cons	**.9328782**	**.0578272**	**16.13**	**0.000**	**.8195388**	**1.046217**
C22						
_cons	**.0848308**	**.0720739**	**1.18**	**0.239**	**-.0564313**	**.226093**
C23						
_cons	**-.0030681**	**.020267**	**-0.15**	**0.880**	**-.0427906**	**.0366545**
C24						
_cons	**.0248742**	**.01484**	**1.68**	**0.094**	**-.0042116**	**.05396**
C25						
_cons	**.0517416**	**.0411343**	**1.26**	**0.208**	**-.0288802**	**.1323633**
C26						
_cons	**-.0397478**	**.026385**	**-1.51**	**0.132**	**-.0914613**	**.0119658**
C27						
_cons	**-.0106307**	**.0257818**	**-0.41**	**0.680**	**-.0611621**	**.0399007**
d1						
_cons	**.1026875**	**.0089873**	**11.43**	**0.000**	**.0850727**	**.1203024**
b1						
_cons	**.6020308**	**.0402554**	**14.96**	**0.000**	**.5231316**	**.68093**
A11						
one	**.6129621**	**.1820549**	**3.37**	**0.001**	**.256141**	**.9697833**

Instruments for equation **eq1_FE**: **__000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7 __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1 __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY __0000D5 _cons**

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
  (file
      diag_2good_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst_noP
      > PXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex not found)
  (output written to diag_2good_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_
  > exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex)
1242
1243        estat overid

    Test of overidentifying restriction:

    Hansen's J chi2(46) = 1518.89 (p = 0.0000)
1244        di "$S_DATE $S_TIME"
   8 Dec 2023 14:59:29
1245        unique hhidi
  Number of unique values of hhidi is  24808
  Number of records is  128974
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  24808
1248        unique group_round
  Number of unique values of group_round is  4607
  Number of records is  128974
1249        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4607
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  5.3848491
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactIn
  > st_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster not found)
  file diag_2good_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst_noPP
  > XdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
```

```
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}

1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\rep_table8_col1.
  > smcl
   log type:  smcl
  closed on:   8 Dec 2023, 14:59:29
-------------------------------------------------------------------------------
```

A.8.2 Columns 2 and 5

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\rep_table8_col2.
> smcl
  log type:  smcl
 opened on:   8 Dec 2023, 15:23:54

1 .
2 . // wrapper for analysis
3 .
4 . *global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"
5 . *global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"
6 . global ROOT "C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication"

7 .
8 . global CODE "$ROOT/code"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=1

25. global use_vislux=0

26. global use_norm=0

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
  >                                                    change these according to what c
  > olumn we are doing (uncomment below)
50. *local group_def "dist"
51. local group_def "fsuXseg"

52. local group_def "fsuXsegXreligXsched"

53.
54. // comparision group
55. local groupComp_def ""

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.          syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
70.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,120   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\mdelma\OneDrive -
      TUNI.fi\Tiedostot\replication/data/laspeyres_state_reshape.dta saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication/data/laspeyres_state_2cats_reshape.dta saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

    Result                      Number of obs
    -----------------------------------------
    Not matched                             0
    Matched                           234,590  (_merge==3)
    -----------------------------------------

114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "

    Result                      Number of obs
    -----------------------------------------
    Not matched                             0
    Matched                           234,590  (_merge==3)
    -----------------------------------------

116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"
```

```
117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)

127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964**	**29,621**	**124,591**	**39,414**	**234,590**

```
            |    100.00     100.00     100.00     100.00 |    100.00

. g scheduled = socgroup==1 | socgroup==2

. g hindu = religion==1

. g muslim = religion==2

. egen hinduXscheduled = group(religion hindu)

. g schednh = scheduled==1 & hindu==0

. g schedh  = scheduled==1 & hindu==1

. g nschedh = scheduled==0 & hindu==1

. g nschednh = scheduled==0 & hindu==0

. g nonHindu=religion>1

. 
. // simplify education categories
. recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

. 
. * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
. local cutoff_group_size=10

. scalar cutoff_expenditure_percentile=99

. scalar scale_factor=1

. 
. // generate variables
. g low_educ = z9==0 & z10==0

. 
. // drop top and bottom percentiles of expenditure in each round/state
. bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

. drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

. drop expenditure_cutoff

. bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

. drop if expenditure<expenditure_cutoff
  (2,252 observations deleted)

. 
. // keep urban houselds smaller than 13 and with head older than 20
. keep if urban==1
  (141,011 observations deleted)

. drop if hhsize>12
  (684 observations deleted)
```

```
160 gen married=marstat==2

161 drop if age<20
  (1,192 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d
```

weight

Percentiles		Smallest		
1%	.004815	.0000482		
5%	.0180562	.0000482		
10%	.0337049	.0000482	Obs	87,172
25%	.1023184	.0001445	Sum of wgt.	87,172
50%	.3490865		Mean	.6966955
		Largest	Std. dev.	1.042372
75%	.909351	26.1213		
90%	1.719447	26.1213	Variance	1.08654
95%	2.45045	29.15594	Skewness	5.872976
99%	4.554976	29.52562	Kurtosis	81.60083

```
166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,335 real changes made)

168
169 g lux=ilux+vlux

170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	14,474	16.76	16.76
2	8,384	9.71	26.46
3	15,490	17.93	44.40
4	11,821	13.69	58.08
5	14,247	16.49	74.58
6	21,957	25.42	100.00
Total	86,373	100.00	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
```

```
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,372 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,625 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)

195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,625 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb
```

```
213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,372 missing values generated)
224         g land=exp(demog23)
  (12,746 missing values generated)
225         replace z4=0 if land<=0.005
  (29,505 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,121 real changes made)
227         g z5=demog24
  (30 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
```

```
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }

297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }

316
317 if short_zlist==6 {
```

```
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }

333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,746 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
```

```
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]
```

Variable	Obs	Weight	Mean	Std. dev.	Min	Max
expenditure	**73,598**	**45470.3285**	**4589.343**	**2838.635**	**308.7667**	**28976.34**

```
375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }

384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }
```

```
397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }

424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
```

```
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,598 real changes made)

460 replace x2=x*x
  (73,598 real changes made)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }

464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)
```

```
470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,598 real changes made)

478 replace x2=x2_temp
  (73,598 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (48,790 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)

488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.         rename q`j' qi`j'
    3.         capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
```

```
499 forval t=1/$T {
    2.         rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\mdelma\AppData\Local\Temp\ST_2c78_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk
```

```
530
531 forval t=1/$T {
    2.          rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,808 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
549 forval j=1/4 {
    2.          g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.          capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
```

```
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                          qui su `var'
    3.                          //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                          file write sumstat "`sumstats_row_`var''" _n
    5.                  }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1
```

```
601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1

625 scalar estimate_FE=1

626
627 // estimation options
628 global w_initial "winit(unadjusted)"

629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

630 global wmatrix "wmatrix(robust)"

631 global trace_level ""

632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

633 global technique ""

634 global maxiter ""

635 global technique_RE ""

636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

637 global derivatives "quickderivatives"

638 scalar init_FE_from_RE=1
```

```
639 global clust_var "state_district_round"

640 local clus_name "_clus_Dist"

641
642
643
644
645
646
647
648
649 // xxzx to keep names same
650 local wmatrix_desc ""

651 local sectorName "_urbOnly"

652 local dropZdesc "_dropPre"

653 local desc_struc_v0 ""

654 local inst_desc "_exactInst"

655 local inst_rootp "_noPPXdInst"

656 local weight_desc ""

657 local RE_sq_inst_desc ""

658 local cnstname ""

659 local actual_pqhat_desc ""

660 local expUpname "_updateExp"

661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
```

```
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //              ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T

694 local Tm1=`T'-1

695 local Tindiv : word count $zlist_indiv

696
697 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
706
707 if `J'==2 {
708         replace p1=P1
  (128,974 real changes made)
```

```
709         replace p2=P2
  (128,974 real changes made)
710         replace qi1=Qi1
  (128,956 real changes made)
711         replace qk1=Qk1
  (128,956 real changes made)
712         replace qikbar1=Qikbar1
  (128,974 real changes made)
713         replace qi2=Qi2
  (128,974 real changes made)
714         replace qk2=Qk2
  (128,974 real changes made)
715         replace qikbar2=Qikbar2
  (128,974 real changes made)
716
717         foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                  qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721         forv j = 1(1)3 {
    2.                  replace p`j'=P`j'
    3.                  replace qi`j'=Qi`j'
    4.                  replace qk`j'=Qk`j'
    5.                  replace qikbar`j'=Qikbar`j'
    6.          }
722
723         foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                  qui replace `var'=.
    3.          }
724 }

725
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
```

```
. local b_p "exp( 0"

. forval j=1/`Jm1' {
  2.         local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
  3. }

. local b_p "`b_p' + ln(p`J') )"

. if simple_model==1      local b_p "1"

. di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

.
. // pCj is the sum of these two, for use in the RE model
. // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
. // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
. forval j=1/`J' {
  2.         local pC`j' "(p`j'*{C`j':$zilist})"
  3. }

. local Tindivp1=`Tindiv'+1

. forval j=1/`J' {
  2.         local pCi`j' "( 0 "
  3.         local pCk`j' "( 0 "
  4.         local pCg`j' "( 0 "
  5.         forval t=1/`Tindiv' {
  6.                 local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
  7.                 local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
  8.         }
  9.         forval t=`Tindivp1'/`T' {
 10.                 local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
 11.         }
 12.         local pCi`j' "`pCi`j'' )"
 13.         local pCk`j' "`pCk`j'' )"
 14.         local pCg`j' "`pCg`j'' )"
 15. }

.
. // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
. local pC "( 0"

. local pCi "( 0"

. local pCk "( 0"

. local pCg "( 0"

. forval j=1/`J' {
  2.         local pC "`pC' + `pC`j'"
  3.         local pCi "`pCi' + `pCi`j'"
  4.         local pCk "`pCk' + `pCk`j'"
  5.         local pCg "`pCg' + `pCg`j'"
  6. }

. local pC "`pC' )"
```

```
local pCi "`pCi' )"

local pCk "`pCk' )"

local pCg "`pCg' )"

// the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
> e GMM statement
local pC_2nd "( 0"

forval j=1/`J' {
        local pC`j' "(p`j'*{C`j':})"
        local pC_2nd "`pC_2nd' + `pC`j''"
}

local pC_2nd "`pC_2nd' )"

// make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
//                                        expenditure shifter is p*diag(D) + (1/2) Sum
>   s~j rp'Drp
// price terms do NOT include their diagonal elements, because these are picked up b
> y the constant term in z (zT)
local rpDrp "( 0"

forval j=1/`J' {
        local r=`j'+1
        local m=`j'-1
        forval s=`r'/`J' {
                if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
> )*2"
        }
        local Drp`j' "( 0"
        forval s=1/`m' {
                if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
> j')"
        }
        forval s=`r'/`J' {
                if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
> s')"
        }

        local Drp`j' "`Drp`j'' )"
        di "`Drp`j''"
}
( 0 )
( 0 )

local rpDrp "`rpDrp' )"

di "`rpDrp'"
( 0 )

if simple_model==1      {
        local pC "0"
        local pC1 "0"
        local pCi "0"
```

```
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.          if Afull==1 {
    3.                  forval j=1/`J' {
    4.                          local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                  if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                  if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                  if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                          }
   11.                          local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.          forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }

798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"
```

```
805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.          local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.          if `jplus1'<=`J' {
    5.                  forval k=`jplus1'/`J' {
    6.                          local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                  }
    8.          }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                               local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                  if noMeasError==1                       local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'              - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                              l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                ))"
    3.          if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                        )
  > )"
    4. }
```

```
828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2
  > ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) )*{b1} + ({A11:}*p1*qikba
  > r1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d
  > 1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))*({b1} - 2*(( ({A11:
  > }*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         ))

836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                 local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                 local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.         }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                 forval l=1/`J' {
    3.                         local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                         local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                 }
    6.         }
```

```
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                                "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                    "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                    "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0}))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                          "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) +
  >  ({A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C
  > 2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikb
  > ar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:
  > }) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C
  > 2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qi
  > kbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )
  > *{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) )
```

```
873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > one}*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}
  > *ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi
  > 4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p
  > 1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         )
  > )

874
875 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi

882 g double `ximxk'=xi-xk

883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.         tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.         g double `xizi`t''=xi*zi`t'
    4.         g double `xkzk`t''=xk*zk`t'
    5.         g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.         g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.         g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.         g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.         g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.         forval j=1/`J' {
   11.                 tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                 g double `zi`t'p`j''= zi`t'*p`j'
   13.                 g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                 g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                 g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                 forv s=1/`T' {
   17.                         tempvar zi`t'zi`s'p`j'
   18.                         g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                 }
   20.         }
   21. }
```

```
891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"
894 if noMeasError==1       local rg "xbar"
895
896 local rg "`rg' x2barlag"
897 local rg "`rg' xbar2lag"
898 local rg "`rg' rootxbarlag"
899
900 local Tindivm1=`Tindiv'-1
901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }
903
904 local zrg ""
905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var'"
    6.          }
    7. }
906
907 local xrg ""
908 local prg ""
909 local xprg ""
910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var'"
    5.
911         forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var'"
   10.                  local xprg "`xprg' `xp`j'`var'"
   11.          }
   12. }
912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"
```

```
917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""

925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }

929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.          forval k=`j'/`J' {
    3.                  tempvar p`j'p`k'
    4.                  g `p`j'p`k''=p`j'*p`k'
    5.                  local roundinsts "`roundinsts' `p`j'p`k''"
    6.          }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"
```

```
938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.          if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.          if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
951          tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.          predict `qikhat`j''
    6.          summ `qikhat`j''
    7.
952          // use all prices times all qikhats as instruments, interacted with xi
953          forval s=1/`J' {
    8.                  tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                  g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                  g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                  g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                  local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                  local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.          }
   16. }
```

Linear regression

Number of obs	= **128,974**
F(128, 4606)	= .
Prob > F	= .
R-squared	= **0.2696**
Root MSE	= **.15968**

(Std. err. adjusted for **4,607** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**-.621922**	**2.828687**	**-0.22**	**0.826**	**-6.167504**	**4.92366**
__00009L	**2.132501**	**1.578459**	**1.35**	**0.177**	**-.962035**	**5.227037**
__00009O	**.9740743**	**8.236484**	**0.12**	**0.906**	**-15.17338**	**17.12153**
xi	**.1097298**	**.0089591**	**12.25**	**0.000**	**.0921656**	**.1272939**
__000002	**-.0092384**	**.003017**	**-3.06**	**0.002**	**-.0151531**	**-.0033236**
__00000E	**-.0134039**	**.0491733**	**-0.27**	**0.785**	**-.1098071**	**.0829993**
__00000P	**-.1628184**	**.1217377**	**-1.34**	**0.181**	**-.4014826**	**.0758458**
__000017	**-.1092464**	**.1049223**	**-1.04**	**0.298**	**-.3149443**	**.0964515**
__00001I	**.7242587**	**.2577761**	**2.81**	**0.005**	**.218894**	**1.229623**
__000020	**-.011978**	**.0240414**	**-0.50**	**0.618**	**-.0591106**	**.0351546**
__00002B	**.1197424**	**.0587746**	**2.04**	**0.042**	**.0045161**	**.2349687**
__00002T	**-.0174585**	**.0393565**	**-0.44**	**0.657**	**-.094616**	**.059699**
__000034	**.1353784**	**.090178**	**1.50**	**0.133**	**-.0414136**	**.3121704**
__00003M	**.0121189**	**.0293995**	**0.41**	**0.680**	**-.0455181**	**.0697559**
__						

00003X	-.0364655	.0615171	-0.59	0.553	-.1570685	.0841376
00004F	-.0268502	.02378	-1.13	0.259	-.0734703	.01977
00004Q	.0198515	.0538881	0.37	0.713	-.0857951	.1254981
000058	-.1151595	.037585	-3.06	0.002	-.1888441	-.0414749
00005J	.2597711	.0957793	2.71	0.007	.0719977	.4475444
00005U	-.9036346	.4542468	-1.99	0.047	-1.794176	-.0130932
00005V	.0091007	.0808822	0.11	0.910	-.1494672	.1676686
00005W	.2372932	.1689897	1.40	0.160	-.0940075	.5685939
00005X	.6758329	.4453855	1.52	0.129	-.197336	1.549002
00005Y	.024238	.089797	0.27	0.787	-.1518071	.2002831
00005Z	.2583701	.1592294	1.62	0.105	-.0537959	.5705361
000060	-.0697087	.0606238	-1.15	0.250	-.1885604	.049143
000061	.0332992	.0426542	0.78	0.435	-.0503235	.1169219
000062	.0246027	.0313427	0.78	0.433	-.0368441	.0860494
000063	.0462894	.0481604	0.96	0.337	-.0481281	.140707
000064	.0021425	.0606117	0.04	0.972	-.1166855	.1209705
000065	1.842614	1.106716	1.66	0.096	-.3270803	4.012307
000066	.5219972	.1784311	2.93	0.003	.1721866	.8718077
000067	-1.142812	.4317973	-2.65	0.008	-1.989341	-.2962819
000068	-1.238818	1.024953	-1.21	0.227	-3.248218	.7705817
000069	-.2489501	.19126	-1.30	0.193	-.6239113	.126011
00006A	-1.044447	.3710979	-2.81	0.005	-1.771976	-.3169169
00006B	-.1050427	.1359402	-0.77	0.440	-.3715505	.1614652
00006C	-.2672949	.0946434	-2.82	0.005	-.4528413	-.0817485
00006D	-.0206005	.0666272	-0.31	0.757	-.1512217	.1100207
00006E	-.1361801	.1097699	-1.24	0.215	-.3513817	.0790215
00006F	-.0971818	.136051	-0.71	0.475	-.363907	.1695433
00006G	.2498027	.2597007	0.96	0.336	-.2593351	.7589404
00006H	.0302825	.03864	0.78	0.433	-.0454704	.1060354
00006I	-.1249745	.1001201	-1.25	0.212	-.3212579	.0713089
00006J	-.2462232	.2442976	-1.01	0.314	-.7251635	.2327172
00006K	.0442331	.0493901	0.90	0.371	-.0525952	.1410614
00006L	-.0812363	.0966985	-0.84	0.401	-.2708117	.1083392
00006M	-.0531166	.0349086	-1.52	0.128	-.1215542	.0153211
00006N	.0202081	.0228913	0.88	0.377	-.0246699	.0650861
00006O	.0057033	.0168003	0.34	0.734	-.0272334	.0386399
00006P	-.0106641	.0274586	-0.39	0.698	-.0644962	.0431681
00006Q	.0320368	.0326882	0.98	0.327	-.0320476	.0961213
00006R	-.3191862	.4110702	-0.78	0.438	-1.125081	.4867084
00006S	-.0615515	.049428	-1.25	0.213	-.158454	.0353509
00006T	.2199327	.1326095	1.66	0.097	-.0400454	.4799107
00006U	.0497131	.3746165	0.13	0.894	-.6847148	.784141
00006V	-.0223665	.0670003	-0.33	0.739	-.1537191	.1089861
00006W	-.0668583	.1321237	-0.51	0.613	-.3258841	.1921675
00006X	.0104791	.0556209	0.19	0.851	-.0985644	.1195227
00006Y	.0339043	.0340443	1.00	0.319	-.0328387	.1006474
00006Z	-.0396079	.0222223	-1.78	0.075	-.0831742	.0039584
000070	-.0069299	.0388595	-0.18	0.858	-.0831131	.0692532
000071	.0151677	.0462141	0.33	0.743	-.0754341	.1057696
000072	-.3313206	.2777205	-1.19	0.233	-.8757859	.2131447
000073	-.052371	.0431292	-1.21	0.225	-.1369248	.0321828
000074	.18864	.1167971	1.62	0.106	-.0403384	.4176184
000075	.1357502	.2653578	0.51	0.609	-.3844781	.6559786
000076	.1256551	.0576023	2.18	0.029	.012727	.2385832
000077	.1725672	.1082115	1.59	0.111	-.0395792	.3847136
000078	-.0652976	.0417261	-1.56	0.118	-.1471007	.0165055
000079	.0214854	.0282178	0.76	0.446	-.033835	.0768058
00007A	.0379368	.0201683	1.88	0.060	-.0016029	.0774764
00007B	.0367438	.030554	1.20	0.229	-.0231568	.0966443
00007C	.0204296	.036784	0.56	0.579	-.0516845	.0925438
00007D	.0111748	.2377548	0.05	0.963	-.4549387	.4772882
00007E	.0271805	.0374696	0.73	0.468	-.0462778	.1006387
00007F	-.011187	.0958599	-0.12	0.907	-.1991183	.1767443
00007G	-.0073506	.2234895	-0.03	0.974	-.4454971	.430796
00007H	-.0530381	.0459868	-1.15	0.249	-.1431942	.0371181
00007I	.0310391	.0848714	0.37	0.715	-.1353496	.1974278
00007J	.0277542	.0314538	0.88	0.378	-.0339103	.0894187
00007K	-.0411019	.02326	-1.77	0.077	-.0867026	.0044987
00007L	-.0231711	.0157654	-1.47	0.142	-.0540787	.0077366
00007M	-.0024671	.0258727	-0.10	0.924	-.05319	.0482558
00007N	.0074499	.0311094	0.24	0.811	-.0535393	.0684392
00007O	-.1947362	.3978999	-0.49	0.625	-.9748107	.5853383

__00007P	.1192206	.0504151	2.36	0.018	.0203828	.2180584
__00007Q	-.0467862	.1447935	-0.32	0.747	-.3306508	.2370784
__00007R	.1621708	.3858765	0.42	0.674	-.594332	.9186736
__00007S	-.0869645	.0687978	-1.26	0.206	-.2218411	.0479122
__00007T	-.3552513	.1335794	-2.66	0.008	-.6171309	-.0933717
__00007U	.0120313	.0513124	0.23	0.815	-.0885657	.1126282
__00007V	-.1069118	.0328228	-3.26	0.001	-.1712603	-.0425633
__00007W	-.0513844	.02479	-2.07	0.038	-.0999848	-.002784
__00007X	-.0064018	.0427747	-0.15	0.881	-.0902607	.0774571
__00007Y	-.025039	.0493972	-0.51	0.612	-.1218812	.0718032
__000080	.2063745	5.852724	0.04	0.972	-11.26777	11.68052
__000082	12.19666	19.40683	0.63	0.530	-25.85001	50.24334
__000085	-.6821914	.2657123	-2.57	0.010	-1.203115	-.161268
__000087	.6788013	.5388478	1.26	0.208	-.3775986	1.735201
__00008A	.7938271	1.095944	0.72	0.469	-1.354748	2.942402
__00008C	-4.449394	3.883239	-1.15	0.252	-12.0624	3.163615
__00008F	-.2443399	7.50892	-0.03	0.974	-14.96542	14.47674
__00008H	-10.30683	23.57898	-0.44	0.662	-56.53294	35.91927
__00008K	-.3489577	.3355763	-1.04	0.298	-1.006848	.3089327
__00008M	-1.810872	.9160784	-1.98	0.048	-3.606825	-.0149195
__00008P	-.2691425	.5105692	-0.53	0.598	-1.270103	.7318179
__00008R	-.8546085	1.251505	-0.68	0.495	-3.308159	1.598942
__00008U	-.0376672	.209408	-0.18	0.857	-.4482071	.3728728
__00008W	.0159905	.4517019	0.04	0.972	-.8695617	.9015427
__00008Z	-.1033549	.1198689	-0.86	0.389	-.3383553	.1316456
__000091	.8413224	.412636	2.04	0.042	.0323581	1.650287
__000094	.013251	.0984609	0.13	0.893	-.1797796	.2062816
__000096	-.4053394	.3374291	-1.20	0.230	-1.066862	.2561833
__000099	-.1593085	.1387733	-1.15	0.251	-.4313706	.1127536
__00009B	.9280677	.4007297	2.32	0.021	.1424454	1.71369
__00009E	-.0082879	.1755156	-0.05	0.962	-.3523825	.3358068
__00009G	-.0648203	.4418331	-0.15	0.883	-.9310248	.8013842
xbarlag	-14.5907	19.56854	-0.75	0.456	-52.95443	23.77302
x2barlag	-.292977	.5456595	-0.54	0.591	-1.362731	.7767769
xbar2lag	4.703308	3.968123	1.19	0.236	-3.076116	12.48273
rootxbarlag	12.67282	23.61285	0.54	0.592	-33.61968	58.96531
z1barlag	2.277384	.8508699	2.68	0.007	.6092715	3.945497
z2barlag	1.423004	1.22941	1.16	0.247	-.9872294	3.833237
z3barlag	.116594	.4190552	0.28	0.781	-.7049549	.9381429
z4barlag	-.7383797	.3511476	-2.10	0.036	-1.426797	-.0499622
z5barlag	.4172598	.3095804	1.35	0.178	-.1896661	1.024186
z6barlag	-.8304414	.4053514	-2.05	0.041	-1.625124	-.0357583
z7barlag	-.0028667	.4437912	-0.01	0.995	-.87291	.8671766
_cons	-3.093906	8.083257	-0.38	0.702	-18.94096	12.75315

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	128,974	.2476256	.096956	.0423699	.8494236

Linear regression

Number of obs	128,974
F(129, 4606)	21.94
Prob > F	0.0000
R-squared	0.2747
Root MSE	.19485

(Std. err. adjusted for **4,607** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	-7.405388	4.714876	-1.57	0.116	-16.6488	1.838028
__00009L	4.075189	1.859394	2.19	0.028	.4298846	7.720493
__00009O	-1.15559	7.938425	-0.15	0.884	-16.71871	14.40753
xi	.1592749	.0106586	14.94	0.000	.138379	.1801708
__000002	-.0200402	.003386	-5.92	0.000	-.0266785	-.0134019
__00000E	-.0027395	.0708979	-0.04	0.969	-.1417334	.1362544
__00000P	-.3318819	.1421529	-2.33	0.020	-.6105697	-.0531941
__000017	.0454038	.1401932	0.32	0.746	-.229442	.3202497
__00001I	.2748524	.3411305	0.81	0.420	-.3939269	.9436318
__000020	-.004117	.0310402	-0.13	0.894	-.0649706	.0567367

__00002B	.0423174	.0696238	0.61	0.543	-.0941786	.1788134
__00002T	-.1333624	.0499725	-2.67	0.008	-.2313325	-.0353924
__000034	.3769556	.1103624	3.42	0.001	.1605925	.5933188
__00003M	.0257394	.0443616	0.58	0.562	-.0612306	.1127094
__00003X	-.0018136	.0878889	-0.02	0.984	-.1741179	.1704907
__00004F	-.0257017	.0339722	-0.76	0.449	-.0923035	.0409001
__00004Q	-.0004343	.0677624	-0.01	0.995	-.1332811	.1324124
__000058	-.0598173	.0458945	-1.30	0.193	-.1497925	.0301579
__00005J	.0828248	.1050174	0.79	0.430	-.1230597	.2887093
__00005U	-.7448436	.5707631	-1.30	0.192	-1.863813	.3741255
__00005V	-.0974636	.0960597	-1.01	0.310	-.2857867	.0908594
__00005W	.182599	.2302329	0.79	0.428	-.2687677	.6339657
__00005X	1.18919	.5999635	1.98	0.048	.0129743	2.365406
__00005Y	-.1240521	.1334331	-0.93	0.353	-.3856449	.1375407
__00005Z	.0302432	.2343133	0.13	0.897	-.4291231	.4896095
__000060	-.1826532	.1058745	-1.73	0.085	-.3902179	.0249115
__000061	.0778316	.0741596	1.05	0.294	-.0675567	.2232199
__000062	.0436765	.0476096	0.92	0.359	-.0496611	.1370142
__000063	-.0084432	.0631852	-0.13	0.894	-.1323164	.11543
__000064	-.0959356	.0774497	-1.24	0.216	-.2477741	.0559028
__000065	1.68715	1.339927	1.26	0.208	-.9397481	4.314049
__000066	.4900151	.2152946	2.28	0.023	.0679344	.9120957
__000067	-1.263578	.533116	-2.37	0.018	-2.30874	-.218415
__000068	-.6099217	1.325175	-0.46	0.645	-3.207899	1.988056
__000069	-.3904678	.2646424	-1.48	0.140	-.9092937	.128358
__00006A	-.918621	.5062809	-1.81	0.070	-1.911174	.0739321
__00006B	-.0864807	.1869215	-0.46	0.644	-.4529364	.279975
__00006C	-.5471017	.154432	-3.54	0.000	-.8498624	-.244341
__00006D	-.0046837	.0888651	-0.05	0.958	-.178902	.1695345
__00006E	-.1120237	.1546218	-0.72	0.469	-.4151565	.1911091
__00006F	.0541519	.1835511	0.30	0.768	-.3056963	.414
__00006G	-.3829122	.3040509	-1.26	0.208	-.9789976	.2131732
__00006H	.0623753	.0452963	1.38	0.169	-.0264272	.1511778
__00006I	.0375851	.1154658	0.33	0.745	-.1887831	.2639534
__00006J	.2380835	.3162368	0.75	0.452	-.3818922	.8580591
__00006K	.0937777	.0672646	1.39	0.163	-.0380931	.2256485
__00006L	-.1211702	.1226245	-0.99	0.323	-.361573	.1192327
__00006M	-.0104105	.0597735	-0.17	0.862	-.1275951	.1067741
__00006N	.0227119	.0340125	0.67	0.504	-.043969	.0893927
__00006O	-.0000361	.0215686	-0.00	0.999	-.0423209	.0422487
__00006P	-.0550559	.0337724	-1.63	0.103	-.121266	.0111542
__00006Q	.0092361	.0436308	0.21	0.832	-.0763013	.0947734
__00006R	-.1682424	.5255501	-0.32	0.749	-1.198572	.8620876
__00006S	-.1222291	.0622511	-1.96	0.050	-.2442711	-.000187
__00006T	.3205738	.1794223	1.79	0.074	-.0311799	.6723276
__00006U	-.2079034	.4948908	-0.42	0.674	-1.178126	.7623196
__00006V	-.1326525	.0869573	-1.53	0.127	-.3031304	.0378255
__00006W	-.0595375	.1823022	-0.33	0.744	-.4169371	.2978621
__00006X	.0099024	.0755847	0.13	0.896	-.1382797	.1580846
__00006Y	.1465168	.0461353	3.18	0.002	.0560695	.236964
__00006Z	-.0190001	.0323448	-0.59	0.557	-.0824114	.0444112
__000070	-.0586846	.0520745	-1.13	0.260	-.1607756	.0434064
__000071	-.0358189	.0652057	-0.55	0.583	-.1636534	.0920156
__000072	-.6271521	.4071429	-1.54	0.124	-1.425347	.171043
__000073	-.1069206	.0603518	-1.77	0.077	-.225239	.0113977
__000074	.3896957	.162337	2.40	0.016	.0714374	.707954
__000075	.2648984	.3868335	0.68	0.494	-.4934805	1.023277
__000076	.0864019	.0839326	1.03	0.303	-.0781462	.2509501
__000077	.1754688	.1506472	1.16	0.244	-.1198719	.4708094
__000078	-.0451532	.0628631	-0.72	0.473	-.1683949	.0780886
__000079	.0330742	.0443248	0.75	0.456	-.0538236	.1199719
__00007A	.0329937	.0298542	1.11	0.269	-.0255349	.0915222
__00007B	.0284434	.0431711	0.66	0.510	-.0561926	.1130794
__00007C	-.0548047	.0520868	-1.05	0.293	-.1569198	.0473104
__00007D	.0407504	.3065026	0.13	0.894	-.5601416	.6416424
__00007E	.0084819	.046048	0.18	0.854	-.0817942	.098758
__00007F	-.0675653	.1186789	-0.57	0.569	-.3002329	.1651023
__00007G	.1295939	.2965988	0.44	0.662	-.451882	.7110697
__00007H	-.0615415	.0651713	-0.94	0.345	-.1893085	.0662255
__00007I	-.0892649	.1177003	-0.76	0.448	-.3200139	.1414842
__00007J	-.0228747	.0452562	-0.51	0.613	-.1115985	.0658492
__00007K	-.0232042	.037308	-0.62	0.534	-.0963457	.0499373

__00007L	-.0011067	.0220231	-0.05	0.960	-.0442825	.0420692
__00007M	.0205484	.0349368	0.59	0.556	-.0479445	.0890412
__00007N	.0405511	.0421779	0.96	0.336	-.0421378	.12324
__00007O	.022848	.4493909	0.05	0.959	-.8581735	.9038695
__00007P	.0859071	.0580983	1.48	0.139	-.0279933	.1998076
__00007Q	-.167686	.1654545	-1.01	0.311	-.492056	.156684
__00007R	.2185426	.4320171	0.51	0.613	-.6284179	1.065503
__00007S	-.1190555	.0884008	-1.35	0.178	-.2923634	.0542523
__00007T	-.4128379	.1625973	-2.54	0.011	-.7316066	-.0940693
__00007U	.0279133	.0605671	0.46	0.645	-.0908272	.1466539
__00007V	-.1244522	.0475107	-2.62	0.009	-.217596	-.0313084
__00007W	-.0367091	.0309605	-1.19	0.236	-.0974065	.0239883
__00007X	.0345978	.0480146	0.72	0.471	-.0595339	.1287294
__00007Y	.0247227	.0587834	0.42	0.674	-.090521	.1399664
__000080	-13.53973	10.32598	-1.31	0.190	-33.78359	6.704138
__000082	14.29388	19.11882	0.75	0.455	-23.18817	51.77593
__000085	-1.05366	.3580541	-2.94	0.003	-1.755617	-.3517021
__000087	.5219502	.5733458	0.91	0.363	-.6020823	1.645983
__00008A	3.896407	1.935557	2.01	0.044	.1017891	7.691026
__00008C	-6.05272	4.030816	-1.50	0.133	-13.95505	1.84961
__00008F	16.99359	13.16178	1.29	0.197	-8.809812	42.79698
__00008H	-8.233611	23.03943	-0.36	0.721	-53.40194	36.93472
__00008K	-.6643529	.4854342	-1.37	0.171	-1.616037	.2873308
__00008M	-2.964778	1.267853	-2.34	0.019	-5.450377	-.4791795
__00008P	-.7542989	.7454165	-1.01	0.312	-2.215672	.7070747
__00008R	-2.438387	1.710166	-1.43	0.154	-5.791131	.9143579
__00008U	.2126371	.3292505	0.65	0.518	-.4328517	.8581258
__00008W	.3306526	.6584439	0.50	0.616	-.9602129	1.621518
__00008Z	-.2516613	.1982794	-1.27	0.204	-.6403839	.1370613
__000091	1.190545	.580771	2.05	0.040	.0519553	2.329134
__000094	.4410552	.1443237	3.06	0.002	.1581115	.7239989
__000096	-1.112376	.503671	-2.21	0.027	-2.099812	-.124939
__000099	-.2672785	.1955724	-1.37	0.172	-.650694	.1161371
__00009B	.7876566	.5670523	1.39	0.165	-.3240376	1.899351
__00009E	-.300011	.2679631	-1.12	0.263	-.8253471	.2253251
__00009G	-.8942213	.6111622	-1.46	0.143	-2.092392	.3039495
xbarlag	-2.525939	21.97408	-0.11	0.908	-45.60566	40.55378
x2barlag	.2513144	.5843091	0.43	0.667	-.8942114	1.39684
xbar2lag	3.349466	4.516566	0.74	0.458	-5.505167	12.2041
rootxbarlag	-7.746505	26.48737	-0.29	0.770	-59.67444	44.18143
z1barlag	4.009081	1.204697	3.33	0.001	1.647298	6.370864
z2barlag	3.75006	1.722058	2.18	0.029	.3740003	7.12612
z3barlag	-.5456679	.6323427	-0.86	0.388	-1.785363	.6940268
z4barlag	-.863721	.4916498	-1.76	0.079	-1.82759	.1001482
z5barlag	.6540187	.4609402	1.42	0.156	-.2496449	1.557682
z6barlag	-.5601901	.563603	-0.99	0.320	-1.665122	.5447418
z7barlag	1.146835	.5993802	1.91	0.056	-.0282372	2.321908
_cons	4.591675	8.970658	0.51	0.609	-12.99511	22.17846

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A2	128,974	.6301469	.1198491	.30498	1.073063

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"

957
958 * interact qhat_hats with Alist
```

```
959 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9

960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
964 }

965
966 local r_RE      "`r_RE' `qhat_hats'"

967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }

978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS		
Model	**9454.18093**	**13**	**727.244687**	Number of obs	= **128,974**
Residual	**1730.5616**	**128,960**	**.013419367**	F(13, 128960)	= **54193.66**
				Prob > F	= **0.0000**
				R-squared	= **0.8453**
				Adj R-squared	= **0.8453**
Total	**11184.7425**	**128,973**	**.086721581**	Root MSE	= **.11584**

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	-.0247608	.0010698	-23.14	0.000	-.0268577	-.0226639
ybarx	-.0012534	.0011819	-1.06	0.289	-.0035699	.0010631
xi2	.0663044	.0006105	108.61	0.000	.0651079	.0675009
ybar	-.0390938	.0026831	-14.57	0.000	-.0443526	-.0338351
xi	.2573844	.0019153	134.38	0.000	.2536303	.2611384
ybar_q1	.2488814	.0039969	62.27	0.000	.2410475	.2567153
pzi1	-.1309208	.001543	-84.85	0.000	-.1339451	-.1278965
pzi2	-.0718	.0023967	-29.96	0.000	-.0764976	-.0671024
pzi3	.0079983	.0007896	10.13	0.000	.0064507	.009546
pzi4	.0052521	.0007944	6.61	0.000	.0036951	.0068091
pzi5	.0079407	.0008685	9.14	0.000	.0062384	.0096429
pzi6	.0051724	.0006956	7.44	0.000	.0038091	.0065357
pzi7	.0214219	.0008917	24.02	0.000	.0196742	.0231697
_cons	.0094794	.0018754	5.05	0.000	.0058037	.0131551

```
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = _b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                local coef = _b[pzi`j']
    3.                local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1025}

1026
1027if `J'==3 & simple_model==0
  >                           local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"
```

```
1029if `J'==4 & simple_model==0
  >                           local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  (setting technique to dfp)
  Iteration 0:  GMM criterion Q(b) =  .06956051
  Iteration 1:  GMM criterion Q(b) =  .02511392
  Iteration 2:  GMM criterion Q(b) =  .01124309
  Iteration 3:  GMM criterion Q(b) =  .00511007
  Iteration 4:  GMM criterion Q(b) =   .0003746
  Iteration 5:  GMM criterion Q(b) =  .00008873
  Iteration 6:  GMM criterion Q(b) =  .00005258
  Iteration 7:  GMM criterion Q(b) =  .00004134
  Iteration 8:  GMM criterion Q(b) =  .00003787
  Iteration 9:  GMM criterion Q(b) =  .00003533

  Step 2
  (setting technique to dfp)
  Iteration 0:  GMM criterion Q(b) =  .02707487
  Iteration 1:  GMM criterion Q(b) =  .02627153  (backed up)
  Iteration 2:  GMM criterion Q(b) =  .02287976
  Iteration 3:  GMM criterion Q(b) =  .02029555
  Iteration 4:  GMM criterion Q(b) =  .01856722
  Iteration 5:  GMM criterion Q(b) =  .01777148
  Iteration 6:  GMM criterion Q(b) =  .01567021
  Iteration 7:  GMM criterion Q(b) =  .01403787
  Iteration 8:  GMM criterion Q(b) =  .01246118
  Iteration 9:  GMM criterion Q(b) =  .01056441
  (switching technique to nr)
  Iteration 10: GMM criterion Q(b) =  .00827588  (not concave)
  Iteration 11: GMM criterion Q(b) =  .00820832  (not concave)
  (switching technique to dfp)
  Iteration 12: GMM criterion Q(b) =  .00784238
  Iteration 13: GMM criterion Q(b) =  .00781153
  Iteration 14: GMM criterion Q(b) =  .00760487
  Iteration 15: GMM criterion Q(b) =  .00758461
  Iteration 16: GMM criterion Q(b) =  .00756065
  Iteration 17: GMM criterion Q(b) =  .00755339
  Iteration 18: GMM criterion Q(b) =  .00751886
  Iteration 19: GMM criterion Q(b) =  .00751252
  Iteration 20: GMM criterion Q(b) =  .00749417
  Iteration 21: GMM criterion Q(b) =  .00748831

  GMM estimation

  Number of parameters =  20
  Number of moments    =  28
  Initial weight matrix: Identity                  Number of obs   =    128,974
  GMM weight matrix:     Robust
```

(Std. err. adjusted for **564** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
A11						
one	**.3897215**	**.0861653**	**4.52**	**0.000**	**.2208405**	**.5586024**
C1						
zi1	**-.1277641**	**.0700123**	**-1.82**	**0.068**	**-.2649857**	**.0094574**
zi2	**-.0837634**	**.0999788**	**-0.84**	**0.402**	**-.2797182**	**.1121915**
zi3	**.0560712**	**.0301904**	**1.86**	**0.063**	**-.0031008**	**.1152432**
zi4	**.0252734**	**.0418956**	**0.60**	**0.546**	**-.0568405**	**.1073872**
zi5	**.0082133**	**.0204937**	**0.40**	**0.689**	**-.0319535**	**.0483802**
zi6	**.0346431**	**.0267036**	**1.30**	**0.195**	**-.0176949**	**.0869811**
zi7	**.0206725**	**.0371508**	**0.56**	**0.578**	**-.0521417**	**.0934867**
C2						
zi1	**.1153085**	**.0997742**	**1.16**	**0.248**	**-.0802453**	**.3108624**
zi2	**.0862351**	**.1657513**	**0.52**	**0.603**	**-.2386315**	**.4111018**
zi3	**.0467801**	**.0476211**	**0.98**	**0.326**	**-.0465556**	**.1401157**
zi4	**.017681**	**.0565878**	**0.31**	**0.755**	**-.0932291**	**.1285911**
zi5	**-.0099237**	**.0350534**	**-0.28**	**0.777**	**-.0786271**	**.0587796**
zi6	**.0743297**	**.0445707**	**1.67**	**0.095**	**-.0130271**	**.1616866**
zi7	**-.0076096**	**.0524821**	**-0.14**	**0.885**	**-.1104726**	**.0952534**
/d1	**.0759509**	**.0081626**	**9.30**	**0.000**	**.0599525**	**.0919493**
/b1	**.3299171**	**.0220885**	**14.94**	**0.000**	**.2866244**	**.3732097**
/AVA11	**.0453016**	**.4816374**	**0.09**	**0.925**	**-.8986904**	**.9892937**
/AVA12	**-.1125307**	**.4829282**	**-0.23**	**0.816**	**-1.059053**	**.8339912**
/AVA22	**.1373173**	**.5542073**	**0.25**	**0.804**	**-.948909**	**1.223544**

Instruments for equation **eq1_RE: __00009J __00009L __00009O xi __000002 __00000E __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5 __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons**

```
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                //calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
```

```
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_vis_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst
      > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not found)
  (output written to diag_2good_vis_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_
  > all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24808
  Number of records is  128974

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
                 e(N_hh) =  24808

1077unique group_round
  Number of unique values of group_round is  4607
  Number of records is  128974

1078estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  4607
```

```
1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
         e(avg_grp_size) =  5.3848491

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_vis_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exa
  > ctInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster not found)
  file diag_2good_vis_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst_
  > noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1082        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
```

```
 17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
> '
 18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
> `l'"
 19.                                  foreach name of global Alist {
 20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
 21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
> `t'mzk`t'p`j''*`p`l'qikhat`l''*`name'
 22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
> ' `zdiff`t'p`j'p`l'q`l'`name'"
 23.                                  }
 24.                                  if `Tindivp1'<=`T' {
 25.                                          forval s=`Tindivp1'/`T' {
 26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
 27.                                                  * don't generate these to save me
> mory!
1119                                                gen double `zdiff`t'p`j'zi`s'p`l''=
>     `zdiff`t'p`j'p`l''*zi`s'
 28.                                                 local pzindivdiffpzg " `pzindivdi
> ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
 29.                                          }
 30.                                  }
 31.                          }
 32.                  }
 33.          }
1120         if `Tindivp1'<=`T' {
1121                 forval t=`Tindivp1'/`T' {
  2.                         tempvar zi`t'xdiff
  3.                         g double `zi`t'xdiff'=zi`t'*`ximxk'
  4.                         local zg "`zg' zi`t'"
  5.                         forval j=1/`J' {
  6.                                 tempvar zi`t'p`j'xdiff
  7.                                 g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
  8.                                 local pzg "`pzg' `zi`t'p`j''"
  9.                                 local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
 10.                         }
 11.                 }
1122         }
1123
1124         // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1125         // denoting zg as group-zs
1126         //  zg,pzgx,rprp,rprpx
1127         local rootprootp ""
1128         local rootprootpxdiff ""
1129
1130         forval j=1/`J' {
  2.                 forval s=`j'/`J' {
  3.                         tempvar xdiffrp`j'rp`s'
  4.                         g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
  5.
1131                       if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
> 'rp`s'' "
  6.
1132                       local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
>
  7.                 }
  8.          }
1133
1134         local pqhatxdiff ""
1135         forval j=1/`J' {
  2.                 forval s=1/`J' {
  3.                         foreach name of global Alist {
  4.                         tempvar   p`s'qikhat`j'xdiff`name'
  5.                         g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
> `qikhat`j''*`name'
  6.
```

```
1136                                local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
1137
1138        // set instruments
1139        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1140}

1141
1142
1143if simple_model==1 {
1144        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1145
1146        /******************
  >         Initial values
  >         ******************/
1147
1148        if init_FE_from_RE==1 {
1149                estimates use "$filename_RE"
1150
1151                // clear init values
1152                local initial_values_FE ""
1153
1154                // extract vector of coefs and paramter names from
1155                matrix coefs = e(b)
1156                local paramlist = e(params)
1157
1158                // iterate through paramter name list, taking
1159                local t=0
1160                foreach p of local paramlist {
    2.                          local `++t'
    3.                          local p_nice = subinstr("`p'",":_cons","",.)
    4.                          local est = coefs[1,`t']
    5.                          if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                  }
1161        }
1162
1163        // make z diff instruments
1164        local pzindivdiff ""
1165        local pz2indivdiff ""
1166        local pzXzindivdiff ""
1167        local pxzindivdiff ""
1168
1169        forv i=1/`T' {
    2.
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
```

```
1173                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                  }
   16.          }
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1     `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
```

```
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00083025
  Iteration 1:  GMM criterion Q(b) =  .00069508
  Iteration 2:  GMM criterion Q(b) =  .00005015
  Iteration 3:  GMM criterion Q(b) =  6.870e-07

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01004067
  Iteration 1:  GMM criterion Q(b) =  .00700949
  Iteration 2:  GMM criterion Q(b) =  .00585758
  Iteration 3:  GMM criterion Q(b) =  .00583185

  GMM estimation

  Number of parameters =  17
  Number of moments    =  63
  Initial weight matrix: Identity                   Number of obs   =    128,974
  GMM weight matrix:     Robust
```

(Std. err. adjusted for **564** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11						
_cons	**.2549473**	**.0270194**	**9.44**	**0.000**	**.2019902**	**.3079044**
C12						
_cons	**.095896**	**.0412829**	**2.32**	**0.020**	**.014983**	**.1768091**
C13						
_cons	**-.0103995**	**.0122156**	**-0.85**	**0.395**	**-.0343417**	**.0135427**
C14						
_cons	**-.0102775**	**.01317**	**-0.78**	**0.435**	**-.0360902**	**.0155351**
C15						
_cons	**.0296628**	**.0170423**	**1.74**	**0.082**	**-.0037395**	**.0630652**
C16						
_cons	**-.0270323**	**.0136298**	**-1.98**	**0.047**	**-.0537461**	**-.0003184**
C17						
_cons	**-.0432435**	**.0191352**	**-2.26**	**0.024**	**-.0807478**	**-.0057391**
C21						
_cons	**.7959706**	**.0608492**	**13.08**	**0.000**	**.6767083**	**.9152329**
C22						
_cons	**.4842597**	**.0927067**	**5.22**	**0.000**	**.3025579**	**.6659615**
C23						
_cons	**-.0243417**	**.0296405**	**-0.82**	**0.412**	**-.0824361**	**.0337528**
C24						
_cons	**-.0293517**	**.018954**	**-1.55**	**0.121**	**-.0665008**	**.0077973**
C25						
_cons	**.0689326**	**.0478564**	**1.44**	**0.150**	**-.0248642**	**.1627295**
C26						
_cons	**-.0770134**	**.0336713**	**-2.29**	**0.022**	**-.143008**	**-.0110188**
C27						
_cons	**-.1241062**	**.039465**	**-3.14**	**0.002**	**-.2014562**	**-.0467562**
d1						
_cons	**.1255081**	**.007888**	**15.91**	**0.000**	**.1100479**	**.1409682**
b1						
_cons	**.3650959**	**.0281223**	**12.98**	**0.000**	**.3099772**	**.4202145**
A11						
one	**.4175672**	**.1152741**	**3.62**	**0.000**	**.191634**	**.6435004**

Instruments for equation **eq1_FE**: **__000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7 __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1 __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY __0000D5 _cons**

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
  (file
      diag_2good_vis_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst
      > _noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex not found)
  (output written to diag_2good_vis_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_
  > all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex)
1242
1243        estat overid

    Test of overidentifying restriction:

    Hansen's J chi2(46) = 752.157 (p = 0.0000)
1244        di "$S_DATE $S_TIME"
   8 Dec 2023 15:45:03
1245        unique hhidi
  Number of unique values of hhidi is  24808
  Number of records is  128974
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
               e(N_hh) =  24808
1248        unique group_round
  Number of unique values of group_round is  4607
  Number of records is  128974
1249        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4607
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  5.3848491
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_vis_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exa
  > ctInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster not found)
  file diag_2good_vis_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactInst_
  > noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
1265                estimates store gmm_est
```

```
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}

1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\rep_table8_col2.
  > smcl
  log type:  smcl
 closed on:   8 Dec 2023, 15:45:03
-------------------------------------------------------------------------------
```

A.8.3 Columns 3 and 6

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\rep_table8_col3.
> smcl
  log type:  smcl
 opened on:   8 Dec 2023, 17:07:01

1 .
2 . // wrapper for analysis
3 .
4 . *global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"
5 . *global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"
6 . global ROOT "C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication"

7 .
8 . global CODE "$ROOT/code"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=1

26. global use_norm=0

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
  >                                                    change these according to what c
  > olumn we are doing (uncomment below)
50. *local group_def "dist"
51. local group_def "fsuXseg"

52. local group_def "fsuXsegXreligXsched"

53.
54. // comparision group
55. local groupComp_def ""

56. local desc_compGrp ""

57. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

58.
59. // load data
60. include "$CODE/prep demand system data.do"

61. //clear all
62. pause on

63. set more off

64. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

65.
66. scalar short_zlist=$short_zlist_val

67.
68. cap prog drop bysmeanw

69. prog def bysmeanw
    1.          syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
70.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
71.         tempvar totweight totsum varXweight
    4.
72.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
73. end

74.
75. // load prices
76. use "$DATA/laspeyres_state_core.dta", clear

77. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

78. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,120   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

79. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

80. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\mdelma\OneDrive -
      TUNI.fi\Tiedostot\replication/data/laspeyres_state_reshape.dta saved

81.
82. use "$DATA/laspeyres_state_core_2cats.dta", clear

83. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

84. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

85. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

86. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

87. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
88. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication/data/laspeyres_state_2cats_reshape.dta saved

89.
90. if $use_norm==1 {
91.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
92.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
93.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
94.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
95. }

96.
97. // read in consumption data
98. use "$DATA/years1999_2004.dta", clear

99. *use "$DATA/years1999_2004_sm.dta", clear
100 g hhid = _n

101
102 // update expenditure if normal goods
103 if $use_norm==1 {
104         egen update_exp = rowtotal(food fuel clothing other_nondur)
105         replace expenditure = update_exp
106 }

107
108 // possible groups
109 egen group_district=group(state district)

110 egen group_round_district=group(state district round)

111 egen group_round_districtUrb=group(state district round urban)

112
113 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

    Result                      Number of obs
    -----------------------------------------
    Not matched                             0
    Matched                           234,590  (_merge==3)
    -----------------------------------------

114 drop _merge

115 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "

    Result                      Number of obs
    -----------------------------------------
    Not matched                             0
    Matched                           234,590  (_merge==3)
    -----------------------------------------

116 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"
```

```
117 drop _merge

118
119 //make weights comparable across rounds
120 bys round: egen meanweight=mean(weight)

121 bys round: replace weight=weight/meanweight
  (234590 real changes made)

122 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
123
124 // clean hh chars
125 replace religion=1 if religion==.
  (20 real changes made)

126 replace socgroup=9 if socgroup==.
  (68 real changes made)

127 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

128 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964**	**29,621**	**124,591**	**39,414**	**234,590**

```
            |    100.00     100.00     100.00     100.00 |    100.00

129 g scheduled = socgroup==1 | socgroup==2

130 g hindu = religion==1

131 g muslim = religion==2

132 egen hinduXscheduled = group(religion hindu)

133 g schednh = scheduled==1 & hindu==0

134 g schedh  = scheduled==1 & hindu==1

135 g nschedh = scheduled==0 & hindu==1

136 g nschednh = scheduled==0 & hindu==0

137 g nonHindu=religion>1

138
139 // simplify education categories
140 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

141
142 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
143 local cutoff_group_size=10

144 scalar cutoff_expenditure_percentile=99

145 scalar scale_factor=1

146
147 // generate variables
148 g low_educ = z9==0 & z10==0

149
150 // drop top and bottom percentiles of expenditure in each round/state
151 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

152 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

153 drop expenditure_cutoff

154 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

155 drop if expenditure<expenditure_cutoff
  (2,252 observations deleted)

156
157 // keep urban houselds smaller than 13 and with head older than 20
158 keep if urban==1
  (141,011 observations deleted)

159 drop if hhsize>12
  (684 observations deleted)
```

```
160 gen married=marstat==2

161 drop if age<20
  (1,192 observations deleted)

162 g hhsizem1=hhsize-1

163
164 // winsorize weights
165 su weight,d
```

weight

Percentiles		Smallest		
1%	.004815	.0000482		
5%	.0180562	.0000482		
10%	.0337049	.0000482	Obs	87,172
25%	.1023184	.0001445	Sum of wgt.	87,172
50%	.3490865		Mean	.6966955
		Largest	Std. dev.	1.042372
75%	.909351	26.1213		
90%	1.719447	26.1213	Variance	1.08654
95%	2.45045	29.15594	Skewness	5.872976
99%	4.554976	29.52562	Kurtosis	81.60083

```
166 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

167 replace weight = r(p5) if weight<r(p5)
  (4,335 real changes made)

168
169 g lux=ilux+vlux

170 g necc=inec+vnec

171
172 // education variables
173 drop if zone==.
  (799 observations deleted)

174 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	14,474	16.76	16.76
2	8,384	9.71	26.46
3	15,490	17.93	44.40
4	11,821	13.69	58.08
5	14,247	16.49	74.58
6	21,957	25.42	100.00
Total	86,373	100.00	

```
175 g educmed=educsimp==2

176 g educhigh=educsimp==3

177
```

```
178 // scale age (helps numerical performance)
179 drop if age==.
  (1 observation deleted)

180 replace age=age/40
  (86,372 real changes made)

181
182 // rename z's so that they can be used in each file
183 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

184
185 // landowner dummy
186 g owns_land = landowned>.005 & landowned<.

187
188 egen group_inst=group(state district urban)

189
190 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

191 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

192 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)
  (43,625 missing values generated)

193 if "`group_def'"=="fsuXsegXreligXschedXownsland"        egen group = group(geogroup_
  > seg religion scheduled owns_land)

194 if "`group_def'"=="fsuXsegXreligXschedXehigh"                   egen group = group(g
  > eogroup_seg religion scheduled educhigh)

195
196 if $rand_grp==1 {
197         sort round
198         sort round state district
199         g shuffle = _n
200         g rand = runiform()
201         sort round rand
202         sort round state district rand
203         g group_new = group[shuffle]
204         replace group = group_new
205         drop group_new shuffle rand
206 }

207
208 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,625 missing values generated)

209 egen district_round=group(state district round urban)

210
211 // make comparision group, so can restrict to sample in other regs
212 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb
```

```
213 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

214 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)

215 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

216
217 if short_zlist==1 {
218         global T=7
219
220         g z1=hhsizem1/10
221         g z2=age/3
222         g z3=married
223         g z4=.
  (86,372 missing values generated)
224         g land=exp(demog23)
  (12,746 missing values generated)
225         replace z4=0 if land<=0.005
  (29,505 real changes made)
226         replace z4=ln(land+1) if land>0.005
  (44,121 real changes made)
227         g z5=demog24
  (30 missing values generated)
228         g z6=educmed
229         g z7=educhigh
230         order land, after(z7)
231
232         global zlist "z1 z2 z3 z4 z5 z6 z7"
233         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
234         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
235 }

236
237 if short_zlist==2 {
238         global T=10
239
240         g z1=hhsizem1/10
241         g z2=age/3
242         g z3=married
243         g z4=.
244         g land=exp(demog23)
245         replace z4=0 if land<=0.005
246         replace z4=ln(land+1) if land>0.005
247         g z5=demog24
248         g z6=educmed
249         g z7=educhigh
250         g z8 = scheduled==1 & hindu==0
251         g z9 = scheduled==0 & hindu==0
252         g z10 = scheduled==0 & hindu==1
253
254         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
255         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
257 }

258
259 if short_zlist==3 {
260         global T=8
```

```
261
262         g z1=hhsizem1/10
263         g z2=age/3
264         g z3=married
265         g z4=.
266         g land=exp(demog23)
267         replace z4=0 if land<=0.005
268         replace z4=ln(land+1) if land>0.005
269         g z5=demog24
270         g z6=educmed
271         g z7=educhigh
272         g z8 = owns_land
273
274         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
275         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
276         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
277 }

278
279 if short_zlist==4 {
280         global T=7
281
282         g z1=hhsizem1/10
283         g z2=age/3
284         g z3=married
285         g z4=.
286         g land=exp(demog23)
287         replace z4=0 if land<=0.005
288         replace z4=ln(land+1) if land>0.005
289         g z5=demog24
290         g z6=educmed
291         g z7=educhigh
292
293         global zlist "z1 z2 z3 z4 z5 z6 z7"
294         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
295         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
296 }

297
298 if short_zlist==5 {
299         global T=6
300
301         g z1=age/3
302         g z2=married
303         g z3=.
304         g land=exp(demog23)
305         replace z3=0 if land<=0.005
306         replace z3=ln(land+1) if land>0.005
307         g z4=demog24
308         g z5=educmed
309         g z6=educhigh
310         order land, after(z6)
311
312         global zlist "z1 z2 z3 z4 z5 z6"
313         global zlist_indiv "z1 z2 z3 z4 z5 z6"
314         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
315 }

316
317 if short_zlist==6 {
```

```
318         global T=5
319
320         g z1=hhsizem1/10
321         g z2=p1
322         g z3=p2
323         g z4=p3
324         g z5=p4
325         g land=exp(demog23)
326
327         global zlist "z1 z2 z3 z4 z5"
328         global zlist_indiv "z1"
329         global xzlist_indiv "xz1"
330
331         drop if land==. | demog24==.
332 }

333
334 if short_zlist==7 {
335         global T=11
336
337         g z1=hhsizem1/10
338         g z2=age/3
339         g z3=married
340         g z4=.
341         g land=exp(demog23)
342         replace z4=0 if land<=0.005
343         replace z4=ln(land+1) if land>0.005
344         g z5=demog24
345         g z6=educmed
346         g z7=educhigh
347         g z8=p1
348         g z9=p2
349         g z10=p3
350         g z11=p4
351
352         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
353         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
354         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
355 }

356
357
358
359 // drop missings and calculate size of grp
360 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,746 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

361 bys group_round: egen size_group_round=count(group_round)

362
363 // size of comparision group
364 if "`groupComp_def'"!="" {
```

```
365         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
366 }

367
368 // turn things int oper capita terms if ption flagged
369 if $meas_pc==1 {
370         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
371 }

372
373 // normalize expenditure by overall mean
374 summ expenditure [aweight=weight]
```

Variable	Obs	Weight	Mean	Std. dev.	Min	Max
expenditure	**73,598**	**45470.3285**	**4589.343**	**2838.635**	**308.7667**	**28976.34**

```
375 scalar expenditure_mean=r(mean)

376 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

377
378 if $use_norm==0 {
379         g pq1=vlux_norm
380         g pq2=ilux_norm
381         g pq3=vnec_norm
382         g pq4=inec_norm
383 }

384 else {
385         g pq1=food_norm
386         g pq2=fuel_norm
387         g pq3=clothing_norm
388         g pq4=other_nondur_norm
389 }

390 g x=expenditure_norm

391 g x2=x*x

392
393 // make q averages
394 qui bysort group_round: egen weight_group_round=sum(weight)

395
396 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }
```

```
397
398 // 2 category consumption
399 g PQ1=pq1+pq2

400 g PQ2=pq3+pq4

401 if $use_vis==1 {
402         drop PQ1 PQ2
403         g PQ1=pq1+pq3
404         g PQ2=pq2+pq4
405 }

406 if $use_vislux==1 {
407         drop PQ1 PQ2
408         g PQ1=pq1
409         g PQ2=pq2+pq3+pq4
410 }

411 if $use_norm==1 {
412         drop PQ1 PQ2
413         g PQ1=pq1
414         g PQ2=pq2+pq3+pq4
415         if `J'==3 {
416                 drop PQ1 PQ2
417                 g PQ1=pq1
418                 g PQ2=pq2
419                 g PQ3=pq3+pq4
420         }
421 }

422
423 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }

424 if `J'==3 {
425         qui g Q`J'=PQ`J'/P`J'
426         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
427 }

428
429 // only scheduled groups (with some of other group)
430 bys geogroup_seg religion: egen share_sched = mean(scheduled)

431 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

432
433 // update group epend wiht other group
434 if $only_scheduled==2 {
435         preserve
436                 keep if scheduled==0
437                 keep geogroup_seg religion Qbar* qbar*
438                 duplicates drop
439                 tempfile update_exp
440                 save `update_exp'
441         restore
442         keep if scheduled==1 & share_sched>0 & share_sched<1
443         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
```

```
444 }

445 drop share_sched

446
447 // laysperes_cpi varies by district and round
448 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4

449
450 // make instruments
451 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
452 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
453 // measure all the x-stuff in reals, using aggregate laysperes index.
454 g weight_temp=weight

455 global instlist ""

456 g xreal=x/local_cpi

457 g x_temp=x

458 g x2_temp=x2

459 replace x=xreal
  (73,598 real changes made)

460 replace x2=x*x
  (73,598 real changes made)

461
462 // make z x interactions
463 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }

464
465 levelsof round, local(roundlist)
  59 60 61 62

466 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

467
468 g xbar2lag=xbarlag^2
  (4 missing values generated)

469 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)
```

```
470
471 bysmeanw x, weight(weight) by(group) rename(xbar)

472 bysmeanw x2, weight(weight) by(group) rename(x2bar)

473 g xbar2=xbar^2

474 g rootxbar = sqrt(xbar)

475 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.          bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.          global instlist "$instlist `var'bar"
    4. }

476
477 replace x=x_temp
  (73,598 real changes made)

478 replace x2=x2_temp
  (73,598 real changes made)

479
480 // add x terms to inst lsit
481 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

482
483 // only groups of size 3 or more
484 drop if size_group_round<3
  (48,790 observations deleted)

485 if "`groupComp_def'"!="" drop if size_groupComp_round<3

486 drop size_group_round

487 bys group_round: egen size_group_round=count(group_round)

488
489 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
490 if `J'==3 local extrakeep "Q3 P3 Qbar3"

491 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

492 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

493
494 g obs_numi=_n

495
496 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

497 rename x xi

498
```

```
499 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

500 rename weight weighti

501 rename land landi

502 rename owns_land owns_landi

503 rename schednh schednhi

504 rename schedh schedhi

505 rename nschedh nschedhi

506 rename nschednh nschednhi

507 g urbani = urban

508 rename hhid hhidi

509 rename scheduled scheduledi

510 rename hhsize hhsizei

511
512 tempfile obs_i

513 save `obs_i'
  file C:\Users\mdelma\AppData\Local\Temp\ST_2c78_000001.tmp saved as .dta format

514
515 // calculate sumstats at hh level
516 if `make_sumstats'==1 {
517         count
518         local sumstats_n_obs = r(N)
519
520         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                  qui su `var'
    3.                  //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
521                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.          }
522         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
523         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
524 }

525
526 rename obs_numi obs_numk

527
528 forval j=1/4 {
    2.          rename qi`j' qk`j'
    3.          capture rename Qi`j' Qk`j'
    4. }

529 rename xi xk
```

```
530
531 forval t=1/$T {
    2.          rename zi`t' zk`t'
    3. }

532 rename weighti weightk

533 rename landi landk

534 rename owns_landi owns_landk

535 rename urbani urbank

536 rename schednhi schednhik

537 rename schedhi schedhk

538 rename nschedhi nschedhk

539 rename nschednhi nschednhk

540 rename scheduledi scheduledk

541 rename hhsizei hhsizek

542 drop geogroup

543
544 // combine into pairs within groups
545 joinby group_round using "`obs_i'"

546 drop if obs_numi==obs_numk
  (24,808 observations deleted)

547
548 // generate exclusive averages qikbarj, of quantities (not spending)
549 forval j=1/4 {
    2.          g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.          capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

550
551 // summary stats tables
552 if `make_sumstats'==1 {
553         count
554         local sumstats_n_obs_pair = r(N)
555
556         cap file close sumstat
557                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
558                 file write sumstat "\begin{table}[htbp]\centering" _n
559                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
560                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
561                 file write sumstat "\begin{tabular}{lcccccccc}" _n
562                 file write sumstat "\toprule" _n
563                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
```

```
564                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
565                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
566                 file write sumstat "\midrule" _n
567
568                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                          qui su `var'
    3.                          //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
569                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                          file write sumstat "`sumstats_row_`var''" _n
    5.                  }
570
571                 file write sumstat "\bottomrule" _n
572                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
573                 file write sumstat "\end{tabular}" _n
574                 file write sumstat "\end{table}" _n
575                 file close sumstat
576                 BREAK
577 }

578
579 // constant for estimation
580 g one=1

581 egen state_district_round=group(state district urban round)

582 egen state_district=group(state district)

583
584 // keep only thick round
585 keep if round==61
  (0 observations deleted)

586
587
588
589
590
591
592 // options
593 local output_number="14"

594
595 // dimension of A matrix
596 scalar Afull=0

597 local descA "full"

598 scalar Adiag=1

599 if Adiag==1 local descA "diag"

600 scalar same_spillover=1
```

```
601 if same_spillover==1 local descA "same"

602
603 // minimum group size (must be 3 or more)
604 local min_group_size = 3

605 scalar min_group_size=`min_group_size'

606
607 // flag for generic model
608 scalar simple_model=0

609 local simp_name ""

610 if simple_model==1 local simp_name "_simple"

611
612 // interactions with peer effects
613 global Alist "one"

614 //global Alist "one schednhi schedhi nschednhi"
615 //global Alist "one   zi8"
616 //global Alist "one   zi7"
617
618 // turn on or off measurement error correction
619 scalar noMeasError=0

620 local measError_desc ""

621 if noMeasError==1 local measError_desc "_NoMeasError"

622
623 // estimate RE and or FE
624 scalar estimate_RE=1

625 scalar estimate_FE=1

626
627 // estimation options
628 global w_initial "winit(unadjusted)"

629 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

630 global wmatrix "wmatrix(robust)"

631 global trace_level ""

632 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

633 global technique ""

634 global maxiter ""

635 global technique_RE ""

636 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

637 global derivatives "quickderivatives"

638 scalar init_FE_from_RE=1
```

```
639 global clust_var "state_district_round"

640 local clus_name "_clus_Dist"

641
642
643
644
645
646
647
648
649 // xxzx to keep names same
650 local wmatrix_desc ""

651 local sectorName "_urbOnly"

652 local dropZdesc "_dropPre"

653 local desc_struc_v0 ""

654 local inst_desc "_exactInst"

655 local inst_rootp "_noPPXdInst"

656 local weight_desc ""

657 local RE_sq_inst_desc ""

658 local cnstname ""

659 local actual_pqhat_desc ""

660 local expUpname "_updateExp"

661
662
663 if $rand_grp==0 {
664         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
665         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
666
667         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
668
669         drop if size_group_round<min_group_size
  (0 observations deleted)
670         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
671
672         //do "$CODE/estim program.do"
673         do "$CODE/estim program.do"

674 // This code estimates a QES demand system
675 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
```

```
676 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
677 // i indexes observations, k indexes other observations in the group
678 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
679 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
680 //              ensure that all zt are z-scored or otherwise have similar scales
681 // qikbarj are leave-two-out group averages
682 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
683 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
684 // note that qi and qk are actually quantities, and not spending
685 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
686
687 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
688
689 set seed 339487731

690
691 local J=J

692 local Jm1=`J'-1

693 local T=$T

694 local Tm1=`T'-1

695 local Tindiv : word count $zlist_indiv

696
697 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
698
699 tempvar clust_weight num_group_round

700 bys group_round: egen `num_group_round'=count(group_round)

701 g double `clust_weight'=size_group_round/`num_group_round'

702 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

703 global GMM_weight "[aweight=`clust_weight']"

704
705 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
706
707 if `J'==2 {
708         replace p1=P1
  (0 real changes made)
```

```
709         replace p2=P2
  (128,974 real changes made)
710         replace qi1=Qi1
  (0 real changes made)
711         replace qk1=Qk1
  (0 real changes made)
712         replace qikbar1=Qikbar1
  (0 real changes made)
713         replace qi2=Qi2
  (128,974 real changes made)
714         replace qk2=Qk2
  (128,974 real changes made)
715         replace qikbar2=Qikbar2
  (128,974 real changes made)
716
717         foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                  qui replace `var'=.
    3.          }
718 }

719
720 if `J'==3 {
721         forv j = 1(1)3 {
    2.                  replace p`j'=P`j'
    3.                  replace qi`j'=Qi`j'
    4.                  replace qk`j'=Qk`j'
    5.                  replace qikbar`j'=Qikbar`j'
    6.          }
722
723         foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                  qui replace `var'=.
    3.          }
724 }

725
726
727 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
728
729 global zilist ""

730 global zklist ""

731 forval t=1/`Tm1' {
    2.          global zilist "$zilist zi`t'"
    3.          global zklist "$zklist zk`t'"
    4. }

732 global zilistTm1 "$zilist"

733 global zklistTm1 "$zklist"

734 if `T'>0 global zilist "$zilist zi`T'"

735 if `T'>0 global zklist "$zklist zk`T'"

736
737 // create b_p and pC
738 // remember b_p doesn't vary within group
```

```
739 local b_p "exp( 0"

740 forval j=1/`Jm1' {
    2.          local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

741 local b_p "`b_p' + ln(p`J') )"

742 if simple_model==1      local b_p "1"

743 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

744
745 // pCj is the sum of these two, for use in the RE model
746 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
747 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
748 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

749 local Tindivp1=`Tindiv'+1

750 forval j=1/`J' {
    2.          local pCi`j' "( 0 "
    3.          local pCk`j' "( 0 "
    4.          local pCg`j' "( 0 "
    5.          forval t=1/`Tindiv' {
    6.                  local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                  local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.          }
    9.          forval t=`Tindivp1'/`T' {
   10.                  local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.          }
   12.          local pCi`j' "`pCi`j'' )"
   13.          local pCk`j' "`pCk`j'' )"
   14.          local pCg`j' "`pCg`j'' )"
   15. }

751
752 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
753 local pC "( 0"

754 local pCi "( 0"

755 local pCk "( 0"

756 local pCg "( 0"

757 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

758 local pC "`pC' )"
```

```
759 local pCi "`pCi' )"

760 local pCk "`pCk' )"

761 local pCg "`pCg' )"

762
763 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
764 local pC_2nd "( 0"

765 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

766 local pC_2nd "`pC_2nd' )"

767
768 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
769 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
770 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
771 local rpDrp "( 0"

772 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
773          local Drp`j' "`Drp`j'' )"
   15.          di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

774
775 local rpDrp "`rpDrp' )"

776 di "`rpDrp'"
  ( 0 )

777
778 if simple_model==1       {
779          local pC "0"
780          local pC1 "0"
781          local pCi "0"
```

```
782         local pCk "0"
783         local pC_2nd "0"
784         local rpDrp "0"
785         local Drp1 "0"
786 }

787
788 // create Aq
789 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
790 // k means use qk as the group quantity (for use in RE only)
791 local qtypelist "ikbar k bar"

792 foreach qtype of local qtypelist {
    2.          if Afull==1 {
    3.                  forval j=1/`J' {
    4.                          local A`qtype'`j' "( 0"
    5.
793                         forval l=1/`J' {
    6.                                  if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                  if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                  if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
794                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                          }
   11.                          local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
795         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
796                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
797         local A`qtype' "( 0"
   22.          forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }

798
799 // Aikbar is A'qbar-minus-ik; Ak is A'qk
800 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
801 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
802 local xi_hat            "(xi - `pCi')"

803 local xk_hat            "(xk - `pCk')"

804 local x_hat                     "(xi - `pC' - `rpDrp' )"
```

```
805 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

806
807 // construct structural vo
808 * let AVA be A'VA, and ensure its diagonals are positive.
809 local v0 "(0 "

810 forval j=1/`J' {
    2.          local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
811         local jplus1=`j'+1
    4.          if `jplus1'<=`J' {
    5.                  forval k=`jplus1'/`J' {
    6.                          local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                  }
    8.          }
    9. }

812
813 local v0 "`v0' )"

814
815 // make all equations, then put pieces together
816
817 * RE equations; first equation has first call to C, other equations use {Cj: } form
818 * note "-vj:", this is because we subtract the structural v0 term from E[q]
819 local j=1

820
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

821 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

822 if `J'>2 {
823         forval j=2/`Jm1' {
    2.
  >                               local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                  if noMeasError==1                       local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'              - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
824 }

825
826 *       FE equations
827 forval j=1/`Jm1' {
    2.                                                                              l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                ))"
    3.          if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                        )
  > )"
    4. }
```

```
828
829 // clean up equations, and combine
830 local eqs_RE ""

831 local eqs_FE ""

832 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

833
834 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2
  > ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) )*{b1} + ({A11:}*p1*qikba
  > r1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d
  > 1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) )

835 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))*({b1} - 2*(( ({A11:
  > }*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         ))

836
837 * replace the first occurrence of Ajj with the Alist
838 if Afull==0 {
839         forval j=1/`J' {
    2.                 local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                 local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.         }
840 }

841 if Afull==1 {
842         forval j=1/`J' {
    2.                 forval l=1/`J' {
    3.                         local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                         local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                 }
    6.         }
```

```
843 }

844
845 if simple_model==1 {
846         // construct z sums and interactions
847         if `T'>0 {
848
849                 local RE_cz_2            "+ {d}*(0"
850                 local RE_acz     ""
851                 local RE_acz1     ""
852                 local RE_acz2     ""
853                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
854                 local RE_czcz    ""
855                 local FE_czi     ""
856                 local FE_czk     ""
857                 local FE_czg     ""
858                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                                "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                    "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                    "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
859                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
860                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
861                 local RE_cz_2 "`RE_cz_2')^2"
862         }
863
864         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0}))))"
865
866         local xi_hat "({b}*p1*xi `FE_czi')"
867         local xk_hat "({b}*p1*xk `FE_czk')"
868
869         local eqs_FE
  >                          "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
870 }

871
872 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one}*p1*qk1) + ({A11:}*p2*qk2) )*( ({A11:}*p1*qikbar1) +
  >  ({A11:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C
  > 2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikb
  > ar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:
  > }) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C
  > 2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A11:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qi
  > kbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )
  > *{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) )
```

```
873 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > one}*p1*qikbar1) + ({A11:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}
  > *ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi
  > 4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p
  > 1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))         )
  > )

874
875 /*****************************************************************************
  >  construct instruments
  > *****************************************************************************/
876
877 //make squares and differences of x
878 tempvar x2i x2k ximxk x2imx2k xiP

879 g double `xiP'=xi*local_cpi

880 g double `x2i'=xi*xi

881 g double `x2k'=xi*xi

882 g double `ximxk'=xi-xk

883 g double `x2imx2k'=xi*xi-xk*xk

884
885 g xbarlag_x2barlag = xbarlag*x2barlag

886 g xbarlag2  = xbarlag^2

887 g x2barlag2 = x2barlag^2

888
889 //make interactions with z and zp
890 forval t=1/`T' {
    2.         tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.         g double `xizi`t''=xi*zi`t'
    4.         g double `xkzk`t''=xk*zk`t'
    5.         g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.         g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.         g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.         g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.         g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.         forval j=1/`J' {
   11.                 tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                 g double `zi`t'p`j''= zi`t'*p`j'
   13.                 g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                 g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                 g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                 forv s=1/`T' {
   17.                         tempvar zi`t'zi`s'p`j'
   18.                         g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                 }
   20.         }
   21. }
```

```
891
892 // make group-level instruments, and the instrument lists rg, xrg
893 if noMeasError==0       local rg "xbarlag"

894 if noMeasError==1       local rg "xbar"

895
896 local rg "`rg' x2barlag"

897 local rg "`rg' xbar2lag"

898 local rg "`rg' rootxbarlag"

899
900 local Tindivm1=`Tindiv'-1

901 // lag instruments
902 forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }

903
904 local zrg ""

905 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var'"
    6.          }
    7. }

906
907 local xrg ""

908 local prg ""

909 local xprg ""

910 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var'"
    5.
911         forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var'"
   10.                  local xprg "`xprg' `xp`j'`var'"
   11.          }
   12. }

912
913 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
914 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
915 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
916 local budget "xi `x2i'"
```

```
917
918 local rootprootp ""

919 local xrootprootp ""

920 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

921
922 // demographics multiplied by prices and budget
923 local zinsts ""

924 local xzinsts ""

925 local xPzinsts ""

926 local pzinsts ""

927 local xpzinsts ""

928 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }

929
930 // price insts
931 local roundinsts " "

932 forval j=1/`J' {
    2.          forval k=`j'/`J' {
    3.                  tempvar p`j'p`k'
    4.                  g `p`j'p`k''=p`j'*p`k'
    5.                  local roundinsts "`roundinsts' `p`j'p`k''"
    6.          }
    7. }

933
934 // instruments for all equations are ri:  x,x2,p,z,zx,
935 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
936 local qhat_insts ""

937 local qhat_insts "`qhat_insts' `zrg'"
```

```
938 local qhat_insts "`qhat_insts' `prg'"

939 local qhat_insts "`qhat_insts' `rg' "

940
941 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

942
943 local count: word count `r_RE'

944 di "total instruments: `count'"
  total instruments: 19

945
946 local qhat_hats ""

947 local pqhat ""

948 local xpqhat ""

949 local pqhat_sq ""

950 forval j=1/`J' {
    2.          if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.          if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
951          tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.          predict `qikhat`j''
    6.          summ `qikhat`j''
    7.
952          // use all prices times all qikhats as instruments, interacted with xi
953          forval s=1/`J' {
    8.                  tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                  g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                  g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                  g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
954                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                  local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                  local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.          }
   16. }
```

Linear regression

Number of obs	=	**128,974**
F(128, 4606)	=	.
Prob > F	=	.
R-squared	=	**0.2592**
Root MSE	=	**.15343**

(Std. err. adjusted for **4,607** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	**2.979372**	**7.45331**	**0.40**	**0.689**	**-11.63269**	**17.59143**
__00009L	**-.9886716**	**.8395248**	**-1.18**	**0.239**	**-2.634542**	**.6571992**
__00009O	**-2.721458**	**9.157242**	**-0.30**	**0.766**	**-20.67404**	**15.23112**
xi	**.0833792**	**.0086328**	**9.66**	**0.000**	**.0664549**	**.1003036**
__000002	**-.0042213**	**.0030031**	**-1.41**	**0.160**	**-.0101087**	**.0016662**
__00000E	**-.1099173**	**.051969**	**-2.12**	**0.034**	**-.2118016**	**-.0080331**
__00000P	**-.071667**	**.0983975**	**-0.73**	**0.466**	**-.2645732**	**.1212391**
__000017	**.1060454**	**.1028905**	**1.03**	**0.303**	**-.0956692**	**.3077601**
__00001I	**.4272924**	**.2106282**	**2.03**	**0.043**	**.0143601**	**.8402246**
__000020	**.0177653**	**.0230846**	**0.77**	**0.442**	**-.0274915**	**.0630221**
__00002B	**.0528902**	**.0449131**	**1.18**	**0.239**	**-.0351609**	**.1409413**
__00002T	**.0720533**	**.0363597**	**1.98**	**0.048**	**.0007707**	**.1433358**
__000034	**.0590161**	**.0664567**	**0.89**	**0.375**	**-.0712709**	**.1893031**
__00003M	**-.0917154**	**.0300858**	**-3.05**	**0.002**	**-.1506979**	**-.0327329**

00003X	-.0086549	.0549541	-0.16	0.875	-.1163913	.0990815
00004F	-.0629459	.0226701	-2.78	0.006	-.1073901	-.0185016
00004Q	.0453659	.044091	1.03	0.304	-.0410737	.1318055
000058	.0079158	.0351293	0.23	0.822	-.0609545	.0767861
00005J	.1326707	.0734313	1.81	0.071	-.0112898	.2766313
00005U	-.9901976	.4346528	-2.28	0.023	-1.842325	-.1380699
00005V	-.0112143	.0834708	-0.13	0.893	-.174857	.1524285
00005W	.3099414	.1619142	1.91	0.056	-.0074881	.6273708
00005X	.6219402	.4163901	1.49	0.135	-.1943839	1.438264
00005Y	.0164275	.0833975	0.20	0.844	-.1470716	.1799265
00005Z	.3657678	.1582991	2.31	0.021	.0554258	.6761098
000060	-.0314501	.0509975	-0.62	0.537	-.1314296	.0685294
000061	.0155911	.0428095	0.36	0.716	-.068336	.0995183
000062	.0155828	.0275206	0.57	0.571	-.0383707	.0695363
000063	.068611	.0442921	1.55	0.121	-.0182228	.1554447
000064	.0467539	.0550125	0.85	0.395	-.061097	.1546049
000065	1.458826	1.128117	1.29	0.196	-.7528249	3.670476
000066	.513404	.1689757	3.04	0.002	.1821306	.8446773
000067	-.9585227	.432597	-2.22	0.027	-1.80662	-.1104253
000068	-1.070469	1.001573	-1.07	0.285	-3.034032	.8930939
000069	-.3074658	.1825258	-1.68	0.092	-.6653038	.0503722
00006A	-.9424666	.3639356	-2.59	0.010	-1.655955	-.2289784
00006B	.0677508	.1201035	0.56	0.573	-.1677096	.3032112
00006C	-.2560935	.0865315	-2.96	0.003	-.4257367	-.0864503
00006D	-.0120327	.060352	-0.20	0.842	-.1303517	.1062862
00006E	-.1851938	.0998948	-1.85	0.064	-.3810354	.0106479
00006F	-.1015949	.1245059	-0.82	0.415	-.3456861	.1424964
00006G	.1823662	.2381098	0.77	0.444	-.2844431	.6491756
00006H	.0423144	.0356861	1.19	0.236	-.0276475	.1122762
00006I	-.1229967	.0917579	-1.34	0.180	-.3028861	.0568926
00006J	-.1755059	.2299437	-0.76	0.445	-.6263057	.2752939
00006K	.0494597	.0455503	1.09	0.278	-.0398407	.1387602
00006L	-.0724701	.0948508	-0.76	0.445	-.2584232	.1134829
00006M	-.0327463	.0289038	-1.13	0.257	-.0894116	.0239191
00006N	.0199515	.0224503	0.89	0.374	-.0240619	.0639649
00006O	-.0040899	.0152754	-0.27	0.789	-.034037	.0258573
00006P	-.0001706	.0246354	-0.01	0.994	-.0484678	.0481266
00006Q	.0256466	.0301093	0.85	0.394	-.0333821	.0846753
00006R	-.3941339	.3706112	-1.06	0.288	-1.12071	.3324416
00006S	-.0285867	.0470167	-0.61	0.543	-.120762	.0635885
00006T	.204689	.1243497	1.65	0.100	-.0390959	.448474
00006U	.0842296	.3565223	0.24	0.813	-.614725	.7831842
00006V	.034354	.061036	0.56	0.574	-.0853059	.1540138
00006W	-.061472	.1252907	-0.49	0.624	-.3071018	.1841578
00006X	-.0188466	.0452709	-0.42	0.677	-.1075992	.0699059
00006Y	.0220045	.0295871	0.74	0.457	-.0360004	.0800093
00006Z	-.0463906	.0197397	-2.35	0.019	-.0850899	-.0076913
000070	.0173314	.0354554	0.49	0.625	-.0521781	.086841
000071	.028886	.041029	0.70	0.481	-.0515505	.1093225
000072	-.4552934	.3016983	-1.51	0.131	-1.046767	.1361798
000073	-.0334572	.0405535	-0.83	0.409	-.1129614	.046047
000074	.1733355	.1238974	1.40	0.162	-.0695627	.4162337
000075	.3727561	.2737545	1.36	0.173	-.1639338	.909446
000076	.0966825	.054459	1.78	0.076	-.0100833	.2034483
000077	.1091672	.1067835	1.02	0.307	-.1001796	.318514
000078	-.0630403	.0386546	-1.63	0.103	-.1388218	.0127412
000079	-.0223333	.0264006	-0.85	0.398	-.0740912	.0294246
00007A	.0453396	.0184252	2.46	0.014	.0092173	.0814619
00007B	.0064519	.0263153	0.25	0.806	-.0451387	.0580424
00007C	.030462	.0338783	0.90	0.369	-.0359557	.0968798
00007D	-.039083	.2340085	-0.17	0.867	-.4978518	.4196858
00007E	.0190491	.0363229	0.52	0.600	-.0521613	.0902595
00007F	.0263194	.0905908	0.29	0.771	-.151282	.2039208
00007G	.001971	.2139992	0.01	0.993	-.4175701	.421512
00007H	-.0728438	.041428	-1.76	0.079	-.1540625	.0083749
00007I	.0648513	.0785651	0.83	0.409	-.0891739	.2188766
00007J	.0515329	.0282866	1.82	0.069	-.0039223	.1069881
00007K	-.0695986	.0231687	-3.00	0.003	-.1150204	-.0241769
00007L	-.0184278	.014701	-1.25	0.210	-.0472488	.0103931
00007M	-.0251688	.0233329	-1.08	0.281	-.0709125	.0205749
00007N	.0142408	.027684	0.51	0.607	-.0400331	.0685147
00007O	-.1186234	.3600004	-0.33	0.742	-.8243968	.5871499

__00007P	.1298598	.0470178	2.76	0.006	.0376824	.2220372
__00007Q	-.0622334	.1290311	-0.48	0.630	-.3151962	.1907294
__00007R	.021699	.3586739	0.06	0.952	-.6814736	.7248716
__00007S	-.0737968	.0635956	-1.16	0.246	-.1984747	.0508811
__00007T	-.2105944	.1275867	-1.65	0.099	-.4607255	.0395367
__00007U	.0554463	.0446784	1.24	0.215	-.0321448	.1430373
__00007V	-.1556103	.0293365	-5.30	0.000	-.2131239	-.0980967
__00007W	-.0725393	.0218396	-3.32	0.001	-.1153553	-.0297233
__00007X	-.0408146	.0369052	-1.11	0.269	-.1131664	.0315372
__00007Y	-.0404828	.04325	-0.94	0.349	-.1252735	.0443079
__000080	1.072419	18.42053	0.06	0.954	-35.04064	37.18548
__000082	-3.256359	22.43176	-0.15	0.885	-47.23337	40.72065
__000085	-.4472108	.3607717	-1.24	0.215	-1.154496	.2600746
__000087	.4362645	.4966707	0.88	0.380	-.5374481	1.409977
__00008A	1.130937	3.731001	0.30	0.762	-6.183612	8.445486
__00008C	-.4320223	4.350042	-0.10	0.921	-8.960189	8.096144
__00008F	-4.090078	22.12553	-0.18	0.853	-47.46672	39.28656
__00008H	6.727916	27.17326	0.25	0.804	-46.54469	60.00052
__00008K	.1421603	.4265366	0.33	0.739	-.6940558	.9783764
__00008M	-1.847259	.7529839	-2.45	0.014	-3.323468	-.3710493
__00008P	-.8526516	.7773582	-1.10	0.273	-2.376646	.671343
__00008R	.8627594	1.061507	0.81	0.416	-1.218302	2.943821
__00008U	-.0800333	.3273014	-0.24	0.807	-.7217008	.5616342
__00008W	-.1440019	.3976395	-0.36	0.717	-.9235658	.6355621
__00008Z	.1995054	.137483	1.45	0.147	-.0700271	.4690379
__000091	.1683496	.1797778	0.94	0.349	-.1841011	.5208003
__000094	.4329432	.155764	2.78	0.005	.127571	.7383154
__000096	-.72168	.1912884	-3.77	0.000	-1.096697	-.3466631
__000099	.2432526	.2226517	1.09	0.275	-.1932514	.6797567
__00009B	.4934628	.3394401	1.45	0.146	-.1720024	1.158928
__00009E	.0847154	.2772304	0.31	0.760	-.4587891	.6282199
__00009G	-.1533188	.3688567	-0.42	0.678	-.8764548	.5698172
xbarlag	3.322903	32.99062	0.10	0.920	-61.35452	68.00032
x2barlag	-.3357615	.5273963	-0.64	0.524	-1.369711	.698188
xbar2lag	-.3416178	6.84115	-0.05	0.960	-13.75355	13.07031
rootxbarlag	-4.340677	38.29273	-0.11	0.910	-79.41279	70.73143
z1barlag	1.758852	.7125821	2.47	0.014	.3618498	3.155854
z2barlag	.0163978	1.087793	0.02	0.988	-2.116197	2.148993
z3barlag	.2202202	.374484	0.59	0.557	-.5139479	.9543882
z4barlag	-.2669857	.239919	-1.11	0.266	-.7373419	.2033704
z5barlag	.3975819	.2876805	1.38	0.167	-.1664096	.9615735
z6barlag	-.7146541	.3491339	-2.05	0.041	-1.399124	-.0301844
z7barlag	.0101657	.3623516	0.03	0.978	-.700217	.7205484
_cons	1.427599	12.4183	0.11	0.908	-22.91821	25.77341

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	128,974	.1816939	.0907032	-.0281513	.7580224

Linear regression

Number of obs	=	128,974
F(129, 4606)	=	44.28
Prob > F	=	0.0000
R-squared	=	0.2807
Root MSE	=	.22615

(Std. err. adjusted for **4,607** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	26.23494	10.34519	2.54	0.011	5.953406	46.51647
__00009L	.6679273	1.343151	0.50	0.619	-1.965293	3.301148
__00009O	-20.65081	11.82638	-1.75	0.081	-43.83618	2.534556
xi	.1973293	.0126518	15.60	0.000	.1725256	.222133
__000002	-.0257701	.0040136	-6.42	0.000	-.0336388	-.0179014
__00000E	-.1063606	.0760621	-1.40	0.162	-.2554786	.0427575
__00000P	-.296707	.1474701	-2.01	0.044	-.585819	-.007595
__000017	-.0050972	.1563657	-0.03	0.974	-.3116489	.3014545
__00001I	.4507032	.3488215	1.29	0.196	-.2331541	1.134561
__000020	.0823164	.0385222	2.14	0.033	.0067944	.1578384

__00002B	.0499389	.0729354	0.68	0.494	-.0930494	.1929272
__00002T	.1880693	.0489667	3.84	0.000	.0920711	.2840675
__000034	.0018216	.1047815	0.02	0.986	-.2036004	.2072437
__00003M	-.1875329	.0504122	-3.72	0.000	-.2863651	-.0887008
__00003X	-.0173964	.0942178	-0.18	0.854	-.2021085	.1673156
__00004F	-.0800788	.0368503	-2.17	0.030	-.152323	-.0078345
__00004Q	-.0515979	.0697747	-0.74	0.460	-.1883898	.0851939
__000058	-.0208472	.0497924	-0.42	0.675	-.1184642	.0767697
__00005J	-.0189302	.1034297	-0.18	0.855	-.2217018	.1838415
__00005U	-.9931682	.6999366	-1.42	0.156	-2.365379	.3790429
__00005V	-.1268582	.1092693	-1.16	0.246	-.3410783	.0873619
__00005W	.2984629	.2810515	1.06	0.288	-.2525327	.8494585
__00005X	1.37195	.7153081	1.92	0.055	-.0303966	2.774296
__00005Y	-.1569406	.1583403	-0.99	0.322	-.4673635	.1534822
__00005Z	.1694876	.2854424	0.59	0.553	-.3901163	.7290914
__000060	-.2129694	.1296258	-1.64	0.100	-.467098	.0411592
__000061	.0581305	.0816781	0.71	0.477	-.1019977	.2182588
__000062	.0990001	.0508634	1.95	0.052	-.0007166	.1987167
__000063	.0078983	.0758611	0.10	0.917	-.1408257	.1566223
__000064	-.1001349	.0922199	-1.09	0.278	-.2809301	.0806603
__000065	2.051675	1.609888	1.27	0.203	-1.104477	5.207826
__000066	.5250256	.2491697	2.11	0.035	.0365335	1.013518
__000067	-1.452071	.6315357	-2.30	0.022	-2.690183	-.2139584
__000068	-.6608813	1.63587	-0.40	0.686	-3.86797	2.546208
__000069	-.3950176	.3041198	-1.30	0.194	-.9912381	.2012029
__00006A	-1.222355	.599131	-2.04	0.041	-2.396939	-.0477714
__00006B	-.2042577	.2098576	-0.97	0.330	-.6156792	.2071638
__00006C	-.6417481	.163204	-3.93	0.000	-.9617061	-.3217901
__00006D	-.0701515	.0925181	-0.76	0.448	-.2515314	.1112284
__00006E	-.2192129	.1810261	-1.21	0.226	-.5741108	.1356851
__00006F	.0244251	.2149091	0.11	0.910	-.3968997	.44575
__00006G	-.1967716	.3562519	-0.55	0.581	-.8951959	.5016527
__00006H	.0809925	.0519868	1.56	0.119	-.0209266	.1829115
__00006I	-.0246623	.1337809	-0.18	0.854	-.2869369	.2376123
__00006J	-.0039543	.3832711	-0.01	0.992	-.7553493	.7474407
__00006K	.1404968	.0818337	1.72	0.086	-.0199366	.3009301
__00006L	-.131261	.1460585	-0.90	0.369	-.4176056	.1550835
__00006M	-.0347271	.0737322	-0.47	0.638	-.1792776	.1098234
__00006N	.0539909	.0396267	1.36	0.173	-.0236964	.1316782
__00006O	-.0065582	.0224959	-0.29	0.771	-.050661	.0375446
__00006P	-.0532548	.040127	-1.33	0.185	-.131923	.0254133
__00006Q	.0175896	.0510928	0.34	0.731	-.0825767	.1177559
__00006R	-.4243	.6007972	-0.71	0.480	-1.60215	.7535504
__00006S	-.1260343	.0729877	-1.73	0.084	-.2691253	.0170566
__00006T	.4356765	.2049842	2.13	0.034	.0338093	.8375436
__00006U	-.1595926	.5828871	-0.27	0.784	-1.302331	.9831454
__00006V	-.0632109	.1041327	-0.61	0.544	-.267361	.1409391
__00006W	.1982546	.2136824	0.93	0.354	-.2206653	.6171744
__00006X	.0520521	.0860225	0.61	0.545	-.1165933	.2206974
__00006Y	.127879	.0520305	2.46	0.014	.0258743	.2298836
__00006Z	-.0923105	.0329829	-2.80	0.005	-.1569727	-.0276482
__000070	.0104379	.0667868	0.16	0.876	-.1204963	.1413721
__000071	-.0485109	.0762244	-0.64	0.525	-.1979474	.1009255
__000072	-1.320057	.4576733	-2.88	0.004	-2.217316	-.4227978
__000073	-.1140537	.0678266	-1.68	0.093	-.2470263	.0189188
__000074	.5486355	.1826736	3.00	0.003	.1905078	.9067632
__000075	1.077761	.4408978	2.44	0.015	.21339	1.942132
__000076	.0319683	.0967265	0.33	0.741	-.157662	.2215985
__000077	.0273584	.1808505	0.15	0.880	-.3271951	.381912
__000078	-.016201	.0716693	-0.23	0.821	-.1567071	.1243051
__000079	.0197655	.0498462	0.40	0.692	-.077957	.117488
__00007A	.0568007	.0321576	1.77	0.077	-.0062435	.119845
__00007B	.0402374	.0496819	0.81	0.418	-.0571629	.1376377
__00007C	-.0721507	.0597628	-1.21	0.227	-.1893144	.045013
__00007D	-.3053026	.3383505	-0.90	0.367	-.9686317	.3580266
__00007E	.0114237	.0531463	0.21	0.830	-.0927685	.1156158
__00007F	-.0048652	.1317636	-0.04	0.971	-.2631849	.2534545
__00007G	.5933743	.3483429	1.70	0.089	-.0895447	1.276293
__00007H	-.057275	.0755099	-0.76	0.448	-.2053106	.0907606
__00007I	-.2258422	.1433534	-1.58	0.115	-.5068836	.0551991
__00007J	-.0248771	.0507651	-0.49	0.624	-.124401	.0746467
__00007K	-.0901039	.0431792	-2.09	0.037	-.1747559	-.0054519

__00007L	-.0113051	.0244041	-0.46	0.643	-.0591489	.0365387
__00007M	.00426	.0427257	0.10	0.921	-.0795028	.0880228
__00007N	.0149664	.0508013	0.29	0.768	-.0846285	.1145614
__00007O	-.2848897	.5317691	-0.54	0.592	-1.327412	.7576325
__00007P	.1338711	.0668932	2.00	0.045	.0027283	.2650138
__00007Q	-.1549299	.1936691	-0.80	0.424	-.534614	.2247542
__00007R	.575967	.5294719	1.09	0.277	-.4620517	1.613986
__00007S	-.0717946	.1061562	-0.68	0.499	-.2799116	.1363224
__00007T	-.515286	.1987471	-2.59	0.010	-.9049255	-.1256465
__00007U	.0262462	.0704658	0.37	0.710	-.1119005	.164393
__00007V	-.1965602	.0530988	-3.70	0.000	-.3006594	-.0924611
__00007W	-.0720273	.0319973	-2.25	0.024	-.1347574	-.0092973
__00007X	.0190355	.0581594	0.33	0.743	-.0949847	.1330558
__00007Y	-.0039945	.0694262	-0.06	0.954	-.1401031	.132114
__000080	61.15522	23.26622	2.63	0.009	15.54227	106.7682
__000082	-40.38156	27.76197	-1.45	0.146	-94.80832	14.04521
__000085	.1146794	.60418	0.19	0.849	-1.069803	1.299162
__000087	-.5507379	.7185158	-0.77	0.443	-1.959373	.8578974
__00008A	-11.19689	4.408837	-2.54	0.011	-19.84032	-2.553455
__00008C	5.992849	5.423082	1.11	0.269	-4.63899	16.62469
__00008F	-78.43516	29.32413	-2.67	0.008	-135.9245	-20.94583
__00008H	59.24116	34.54042	1.72	0.086	-8.474611	126.9569
__00008K	.2609283	.7024421	0.37	0.710	-1.116195	1.638051
__00008M	-3.385202	1.24382	-2.72	0.007	-5.823686	-.9467179
__00008P	3.223343	1.224293	2.63	0.008	.8231429	5.623544
__00008R	-6.018684	1.809332	-3.33	0.001	-9.565842	-2.471526
__00008U	.3991878	.4492206	0.89	0.374	-.4814998	1.279875
__00008W	-.2186134	.6404313	-0.34	0.733	-1.474166	1.036939
__00008Z	.6790193	.2234303	3.04	0.002	.2409888	1.11705
__000091	1.028246	.3517317	2.92	0.003	.3386835	1.717809
__000094	.5208106	.2598659	2.00	0.045	.011349	1.030272
__000096	.0545987	.3462775	0.16	0.875	-.6242711	.7334684
__000099	-.1393278	.3683424	-0.38	0.705	-.8614554	.5827997
__00009B	.4642109	.5650746	0.82	0.411	-.643606	1.572028
__00009E	-.2080261	.4548206	-0.46	0.647	-1.099692	.6836402
__00009G	-1.107903	.6159409	-1.80	0.072	-2.315443	.0996358
xbarlag	-8.519398	34.36763	-0.25	0.804	-75.89642	58.85763
x2barlag	.1418215	.7165313	0.20	0.843	-1.262923	1.546566
xbar2lag	3.704508	7.211501	0.51	0.607	-10.43349	17.8425
rootxbarlag	3.002098	40.86115	0.07	0.941	-77.10533	83.10953
z1barlag	3.509216	1.262494	2.78	0.005	1.034124	5.984309
z2barlag	3.700408	2.01426	1.84	0.066	-.2485068	7.649323
z3barlag	-.0294668	.7721187	-0.04	0.970	-1.543189	1.484256
z4barlag	-1.495666	.4677677	-3.20	0.001	-2.412715	-.5786168
z5barlag	-.5892242	.5337253	-1.10	0.270	-1.635582	.4571332
z6barlag	-.3575106	.6426539	-0.56	0.578	-1.61742	.902399
z7barlag	1.26085	.6824868	1.85	0.065	-.0771513	2.598851
_cons	-.4657024	13.58308	-0.03	0.973	-27.09506	26.16365

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A2	128,974	.7244558	.1411872	.2962362	1.399592

```
955
956 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"

957
958 * interact qhat_hats with Alist
```

```
959 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9

960 if "$Alist"~="one" {
961         local qhat_hats_backup "`qhat_hats'"
962         local qhat_hats ""
963         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
964 }

965
966 local r_RE      "`r_RE' `qhat_hats'"

967
968 /*****************************************************************************
  > Set instruments and start values
  > *****************************************************************************/
969
970 if simple_model==1      {
971         local r_RE "xbarlag `xixbarlag' `budget'"
972         local r_RE1 ""
973
974         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
975
976         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
977 }

978
979 if `J'==2 & simple_model==0     {
980         g y = p1*qi1
981         g ybar_q1 = p1*qikbar1
982         g ybar_q2 = p2*qikbar2
983         g ybar = p1*qikbar1 + p2*qikbar2
984         g ybar2 = ybar^2
985         g ybar_q1_2 = ybar_q1^2
986         g ybar_q2_2 = ybar_q2^2
987         g ybar_q1_q2 = ybar_q1*ybar_q2
988         g ybarx = ybar*xi
989         g ybarx_1 = ybar_q1*xi
990         g ybarx_2 = ybar_q2*xi
991         g xi2 = xi^2
992
993         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
994
995         if same_spillover==1 {
996                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
```

Source	SS	df	MS		
Model	**5765.95234**	**13**	**443.534795**	Number of obs	= **128,974**
Residual	**1765.10174**	**128,960**	**.013687203**	F(13, 128960)	= **32405.07**
				Prob > F	= **0.0000**
				R-squared	= **0.7656**
				Adj R-squared	= **0.7656**
Total	**7531.05408**	**128,973**	**.058392486**	Root MSE	= **.11699**

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
ybar2	-.0178878	.0010921	-16.38	0.000	-.0200282	-.0157474
ybarx	-.0120246	.0011937	-10.07	0.000	-.0143643	-.0096849
xi2	.0785586	.0006142	127.91	0.000	.0773548	.0797624
ybar	-.0506715	.0025234	-20.08	0.000	-.0556174	-.0457257
xi	.1301748	.001941	67.07	0.000	.1263706	.1339791
ybar_q1	.3232343	.003811	84.82	0.000	.3157648	.3307038
pzi1	-.1792225	.0018793	-95.37	0.000	-.1829058	-.1755392
pzi2	-.0480145	.0029846	-16.09	0.000	-.0538642	-.0421648
pzi3	.018729	.0009691	19.33	0.000	.0168297	.0206283
pzi4	-.0037911	.0009778	-3.88	0.000	-.0057076	-.0018746
pzi5	.0082938	.001059	7.83	0.000	.0062182	.0103694
pzi6	.005031	.0008754	5.75	0.000	.0033152	.0067467
pzi7	.0255774	.0011118	23.00	0.000	.0233982	.0277566
_cons	.024937	.0020081	12.42	0.000	.0210012	.0288727

```
997                 local b = _b[xi]
998                 local a = _b[ybar_q1]
999                 if same_spillover==1 {
1000                        local a = logit(_b[ybar_q1]/2 + .5)
1001                        if `a'==. local a = .5
1002                }
1003                local d = _b[xi2]
1004                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1005        }
1006        else {
1007                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
1008                local b = _b[xi]
1009                local a1 = _b[ybar_q1]/(1-`b')
1010                local a2 = _b[ybar_q2]/`b'
1011                if same_spillover==1 {
1012                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1013                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1014
1015                        if `a1'==. local a1 = .5
1016                        if `a2'==. local a2 = .5
1017                }
1018                local d = _b[xi2]
1019
1020                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1021        }
1022
1023        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1024        forv j = 1(1)`T' {
    2.                local coef = _b[pzi`j']
    3.                local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1025}

1026
1027if `J'==3 & simple_model==0
  >                         local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1028if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"
```

```
1029if `J'==4 & simple_model==0
  >                           local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1030if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031
1032if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1033
1034if `J'==2 & estimate_RE==1 {
1035
1036        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  (setting technique to dfp)
  Iteration 0:  GMM criterion Q(b) =  .08933385
  Iteration 1:  GMM criterion Q(b) =  .00164068
  Iteration 2:  GMM criterion Q(b) =  .00103798
  Iteration 3:  GMM criterion Q(b) =  .00017432
  Iteration 4:  GMM criterion Q(b) =  .00006875
  Iteration 5:  GMM criterion Q(b) =  .00003805
  Iteration 6:  GMM criterion Q(b) =  .00003374
  Iteration 7:  GMM criterion Q(b) =  .00003243
  Iteration 8:  GMM criterion Q(b) =  .00002748
  Iteration 9:  GMM criterion Q(b) =  .00002601

  Step 2
  (setting technique to dfp)
  Iteration 0:  GMM criterion Q(b) =  .02619762
  Iteration 1:  GMM criterion Q(b) =  .02491492  (backed up)
  Iteration 2:  GMM criterion Q(b) =  .02276518
  Iteration 3:  GMM criterion Q(b) =  .02123489
  Iteration 4:  GMM criterion Q(b) =  .02040667
  Iteration 5:  GMM criterion Q(b) =  .01858504
  Iteration 6:  GMM criterion Q(b) =  .01679462
  Iteration 7:  GMM criterion Q(b) =   .0155726
  Iteration 8:  GMM criterion Q(b) =  .01106683
  Iteration 9:  GMM criterion Q(b) =  .01000991
  (switching technique to nr)
  Iteration 10: GMM criterion Q(b) =   .0087407  (not concave)
  Iteration 11: GMM criterion Q(b) =  .00854409  (not concave)
  (switching technique to dfp)
  Iteration 12: GMM criterion Q(b) =  .00840873
  Iteration 13: GMM criterion Q(b) =  .00827439
  Iteration 14: GMM criterion Q(b) =  .00821565
  Iteration 15: GMM criterion Q(b) =   .0081532
  Iteration 16: GMM criterion Q(b) =  .00810859
  Iteration 17: GMM criterion Q(b) =  .00806296
  Iteration 18: GMM criterion Q(b) =  .00800125
  Iteration 19: GMM criterion Q(b) =  .00796119
  Iteration 20: GMM criterion Q(b) =   .0078964
  Iteration 21: GMM criterion Q(b) =  .00786466

  GMM estimation

  Number of parameters =  20
  Number of moments    =  28
  Initial weight matrix: Identity                  Number of obs   =    128,974
  GMM weight matrix:     Robust
```

(Std. err. adjusted for **564** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
A11						
one	**.6157247**	**.1223972**	**5.03**	**0.000**	**.3758306**	**.8556188**
C1						
zi1	**-.1751619**	**.0807036**	**-2.17**	**0.030**	**-.333338**	**-.0169857**
zi2	**-.1424765**	**.0765273**	**-1.86**	**0.063**	**-.2924674**	**.0075143**
zi3	**.0360639**	**.0216767**	**1.66**	**0.096**	**-.0064217**	**.0785494**
zi4	**.0173521**	**.0233657**	**0.74**	**0.458**	**-.0284439**	**.0631481**
zi5	**.0009433**	**.0180407**	**0.05**	**0.958**	**-.0344158**	**.0363023**
zi6	**-.0036731**	**.0190739**	**-0.19**	**0.847**	**-.0410573**	**.0337111**
zi7	**.0063847**	**.0263525**	**0.24**	**0.809**	**-.0452652**	**.0580347**
C2						
zi1	**.1299343**	**.1390323**	**0.93**	**0.350**	**-.1425641**	**.4024326**
zi2	**.0705579**	**.1796664**	**0.39**	**0.695**	**-.2815818**	**.4226977**
zi3	**.0684155**	**.0475735**	**1.44**	**0.150**	**-.0248268**	**.1616578**
zi4	**.0036197**	**.0430979**	**0.08**	**0.933**	**-.0808506**	**.0880901**
zi5	**-.0299486**	**.0493959**	**-0.61**	**0.544**	**-.1267628**	**.0668657**
zi6	**.0101268**	**.0450589**	**0.22**	**0.822**	**-.078187**	**.0984405**
zi7	**-.0096589**	**.0533916**	**-0.18**	**0.856**	**-.1143045**	**.0949866**
/d1	**.1054099**	**.011185**	**9.42**	**0.000**	**.0834877**	**.127332**
/b1	**.217559**	**.0275209**	**7.91**	**0.000**	**.1636191**	**.2714989**
/AVA11	**.0007672**	**.5628461**	**0.00**	**0.999**	**-1.102391**	**1.103925**
/AVA12	**-.2587713**	**.3749941**	**-0.69**	**0.490**	**-.9937462**	**.4762036**
/AVA22	**.0217102**	**.262313**	**0.08**	**0.934**	**-.4924139**	**.5358343**

Instruments for equation **eq1_RE: __00009J __00009L __00009O xi __000002 __00000E __00000P __000017 __00001I __000020 __00002B __00002T __000034 __00003M __00003X __00004F __00004Q __000058 __00005J __00009W __00009Z __0000A5 __0000A8 __00009X __0000A0 __0000A6 __0000A9 _cons**

```
1037
1038        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1039        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1040
1041        if simple_model==1 {
1042                //calculate derivative and save
1043                local RE_acz_drv ""
1044                tempvar junk
1045                g `junk'=.
1046                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
1047                replace `junk'=p1*qikbar1
1048                su `junk' $GMM_weight
1049                local mean_y = r(mean)
1050                su xi $GMM_weight
1051                local mean_x = r(mean)
1052
```

```
1053                local c_term ""
1054
1055                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1056                estimates store gmm_est
1057                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1058                estimates save "${filename_RE}_dF", replace
1059                estimates restore gmm_est
1060        }
1061}

1062
1063if `J'==3 & estimate_RE==1 {
1064        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1065}

1066if `J'==4 & estimate_RE==1 {
1067        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1068}

1069esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_vislux_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactI
      > nst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not found)
  (output written to diag_2good_vislux_main_nopcross_state_Asame_fsuXsegXreligXsched_dro
  > p3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex)

1070
1071
1072// save coefs
1073matrix est_b=e(b)

1074unique hhidi
  Number of unique values of hhidi is  24808
  Number of records is  128974

1075local N_hh = r(sum)

1076estadd scalar N_hh = r(sum)

  added scalar:
                 e(N_hh) =  24808

1077unique group_round
  Number of unique values of group_round is  4607
  Number of records is  128974

1078estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  4607
```

```
1079estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
         e(avg_grp_size) =  5.3848491

1080if estimate_RE==1 {
1081        estimates save "$filename_RE", replace
  (note: file diag_2good_vislux_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_
  > exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster not found
  > )
  file diag_2good_vislux_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactIn
  > st_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1082        estimates save temp1_RE, replace
  (note: file temp1_RE.ster not found)
  file temp1_RE.ster saved
1083}

1084
1085estimates store temp1_FE

1086
1087/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1088
1089if simple_model==0 {
1090
1091        // make FE instruments
1092
1093        // denoting ximxk as x and zimzk of individual zs as zi:
1094        //   x,x2,p,zi,px,zi,pzi
1095        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1096        local xdiff "`ximxk' "
1097        local xdiffxdiff ""
1098        local zindivdiff ""
1099        local pzindivdiff ""
1100        local pzindivdiffpqhat ""
1101        local pxzindivdiff ""
1102        local p2zindiv2diff ""
1103        local zg ""
1104        local pzg ""
1105        local zindivdiffxdiff ""
1106        local pzindivdiffxdiff ""
1107        local pzindivdiffp ""
1108        local pzindivdiffpzg ""
1109        local zgxdiff ""
1110        local pzgxdiff ""
1111        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1112        g double `ximxkxdiff'=`ximxk'*`ximxk'
1113        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1114        g double `xixkxdiff'=xi*xk*`ximxk'
1115
1116        local xdiffxdiff "`x2imx2k'"
1117
1118        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
```

```
   15.                         forval l=1/`J' {
   16.                                 tempvar zdiff`t'p`j'p`l'
   17.                                 g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                 local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                 foreach name of global Alist {
   20.                                         tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                         g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*`p`l'qikhat`l''*`name'
   22.                                         local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                 }
   24.                                 if `Tindivp1'<=`T' {
   25.                                         forval s=`Tindivp1'/`T' {
   26.                                                 tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                 * don't generate these to save me
  > mory!
1119                                                      gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                    local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                         }
   30.                                 }
   31.                         }
   32.                 }
   33.         }
1120        if `Tindivp1'<=`T' {
1121                forval t=`Tindivp1'/`T' {
    2.                          tempvar zi`t'xdiff
    3.                          g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                          local zg "`zg' zi`t'"
    5.                          forval j=1/`J' {
    6.                                  tempvar zi`t'p`j'xdiff
    7.                                  g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                  local pzg "`pzg' `zi`t'p`j''"
    9.                                  local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                          }
   11.                 }
1122        }
1123
1124        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1125        // denoting zg as group-zs
1126        //  zg,pzgx,rprp,rprpx
1127        local rootprootp ""
1128        local rootprootpxdiff ""
1129
1130        forval j=1/`J' {
    2.                  forval s=`j'/`J' {
    3.                          tempvar xdiffrp`j'rp`s'
    4.                          g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1131                        if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1132                        local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                  }
    8.          }
1133
```

```
        local pqhatxdiff ""
        forval j=1/`J' {
                forval s=1/`J' {
                        foreach name of global Alist {
                                tempvar   p`s'qikhat`j'xdiff`name'
                                g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*`qikhat`j''*`name'

                                local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`name'' "
                        }
                }
        }

        // set instruments
        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzindivdiffpqhat' `pxzindivdiff'"
}


if simple_model==1 {
        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
                tempvar ximxk`var'
                g `ximxk`var''=`ximxk'*`var'
        }

        /******************
        Initial values
        ******************/

        if init_FE_from_RE==1 {
                estimates use "$filename_RE"

                // clear init values
                local initial_values_FE ""

                // extract vector of coefs and paramter names from
                matrix coefs = e(b)
                local paramlist = e(params)

                // iterate through paramter name list, taking
                local t=0
                foreach p of local paramlist {
                        local `++t'
                        local p_nice = subinstr("`p'",":_cons","",.)
                        local est = coefs[1,`t']
                        if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_values_FE "`initial_values_FE' `p_nice' `est'"
                }
        }

        // make z diff instruments
        local pzindivdiff ""
        local pz2indivdiff ""
        local pzXzindivdiff ""
        local pxzindivdiff ""

        forv i=1/`T' {
```

```
1170                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1171                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1172                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1173                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                  }
   16.         }
1174
1175        // p z interactions
1176        local pzg ""
1177        local pzgxdiff ""
1178        if `Tindivp1'<=`T' {
1179                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1180        }
1181
1182        tempvar p1ximxk  p1x2imx2k
1183        g `p1ximxk' = p1*`ximxk'
1184        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1185
1186        // define instruments
1187        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
1188
1189        // add prediction of quantity as extra instrument
1190        local pqhat ""
1191        local qhat ""
1192        local pqhatxdiff ""
1193        local pqhatxdiff2 ""
1194        local pzindivdiffpqhat ""
1195        local pzindivdiffpxbar_lag ""
1196
1197        tempvar qik_hat1
1198         reg qikbar1    `r_RE' `qhat_insts', vce(cluster group_round)
1199        predict `qik_hat1'
1200        local qhat "`qhat' `qik_hat1'"
1201
1202        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1203
1204        g double `p1_qikbar1' = p1*qikbar1
1205        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1206        predict `p1_qikhat1'
```

```
1207        local pqhat "`pqhat' `p1_qikhat1'"
1208
1209        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1210        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1211
1212        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1213
1214        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1215        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1216        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1217
1218        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1219                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1220
1221        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1222}

1223
1224// starting values
1225if init_FE_from_RE==1 & "$init_FE_file"!="" {
1226        estimates use "$init_FE_file"
1227        local initial_values_FE ""
1228
1229        // extract vector of coefs and paramter names from
1230        matrix coefs = e(b)
1231        local paramlist = e(params)
1232
1233        // iterate through paramter name list, taking
1234        local t=0
1235        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
1236}

1237
1238if estimate_FE==1 {
1239        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00075718
  Iteration 1:  GMM criterion Q(b) =  .00019183
  Iteration 2:  GMM criterion Q(b) =  4.674e-06

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01375553
  Iteration 1:  GMM criterion Q(b) =  .01175914
  Iteration 2:  GMM criterion Q(b) =  .01005464
  Iteration 3:  GMM criterion Q(b) =  .00995863
  Iteration 4:  GMM criterion Q(b) =  .00994309

  GMM estimation
```

Number of parameters = **17**
Number of moments = **63**
Initial weight matrix: **Identity** Number of obs = **128,974**
GMM weight matrix: **Robust**

(Std. err. adjusted for **564** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11						
_cons	**.1684565**	**.0229423**	**7.34**	**0.000**	**.1234904**	**.2134225**
C12						
_cons	**.0153625**	**.036708**	**0.42**	**0.676**	**-.0565839**	**.0873089**
C13						
_cons	**-.0092508**	**.0100427**	**-0.92**	**0.357**	**-.0289341**	**.0104326**
C14						
_cons	**.0260259**	**.0106844**	**2.44**	**0.015**	**.0050848**	**.046967**
C15						
_cons	**.0469943**	**.017083**	**2.75**	**0.006**	**.0135122**	**.0804764**
C16						
_cons	**-.0085841**	**.0114959**	**-0.75**	**0.455**	**-.0311157**	**.0139475**
C17						
_cons	**-.0157161**	**.0144701**	**-1.09**	**0.277**	**-.0440769**	**.0126448**
C21						
_cons	**.9917399**	**.0769206**	**12.89**	**0.000**	**.8409782**	**1.142502**
C22						
_cons	**.4396155**	**.1133862**	**3.88**	**0.000**	**.2173827**	**.6618483**
C23						
_cons	**-.0185839**	**.0323018**	**-0.58**	**0.565**	**-.0818941**	**.0447264**
C24						
_cons	**.0250839**	**.024852**	**1.01**	**0.313**	**-.0236251**	**.0737929**
C25						
_cons	**.1839788**	**.0889622**	**2.07**	**0.039**	**.0096161**	**.3583416**
C26						
_cons	**-.0260587**	**.0400674**	**-0.65**	**0.515**	**-.1045893**	**.0524719**
C27						
_cons	**-.0559261**	**.0381522**	**-1.47**	**0.143**	**-.130703**	**.0188508**
d1						
_cons	**.133057**	**.0082145**	**16.20**	**0.000**	**.1169569**	**.1491571**
b1						
_cons	**.3451176**	**.0368199**	**9.37**	**0.000**	**.2729519**	**.4172833**
A11						
one	**.654441**	**.1317421**	**4.97**	**0.000**	**.3962311**	**.9126508**

Instruments for equation **eq1_FE**: **__000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T __00005B __00005M __000005 __0000DE __0000DF __0000DG __0000DH __0000AJ __0000AL __0000AQ __0000AS __0000AY __0000B0 __0000B5 __0000B7 __0000BD __0000BF __0000BK __0000BM __0000BS __0000BU __0000BZ __0000C1 __0000C7 __0000C9 __0000CE __0000CG __0000CM __0000CO __0000CT __0000CV __0000D1 __0000D3 __0000D8 __0000DA __0000AG __0000AN __0000AV __0000B2 __0000BA __0000BH __0000BP __0000BW __0000C4 __0000CB __0000CJ __0000CQ __0000CY __0000D5 _cons**

```
1240
1241        esttab using "${filename_FE}.tex", se replace //added this
  (file
      diag_2good_vislux_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactI
      > nst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex not found)
  (output written to diag_2good_vislux_main_nopcross_state_Asame_fsuXsegXreligXsched_dro
  > p3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex)
1242
1243        estat overid

    Test of overidentifying restriction:

    Hansen's J chi2(46) =  1282.4 (p = 0.0000)
1244        di "$S_DATE $S_TIME"
   8 Dec 2023 17:31:18
1245        unique hhidi
  Number of unique values of hhidi is  24808
  Number of records is  128974
1246        local N_hh = r(sum)
1247        estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  24808
1248        unique group_round
  Number of unique values of group_round is  4607
  Number of records is  128974
1249        estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  4607
1250        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
        e(avg_grp_size) =  5.3848491
1251        estimates save "$filename_FE", replace
  (note: file diag_2good_vislux_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_
  > exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster not found
  > )
  file diag_2good_vislux_main_nopcross_state_Asame_fsuXsegXreligXsched_drop3_all_exactIn
  > st_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1252
1253        if simple_model==1 {
1254                local RE_acz_drv ""
1255                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1256                noi di "`RE_acz_drv'"
1257
1258                cap g junk=.
1259                replace junk=p1*qikbar1
1260                su junk $GMM_weight
1261                local mean_y = r(mean)
1262                su xi $GMM_weight
1263                local mean_x = r(mean)
1264
```

```
1265                estimates store gmm_est
1266                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1267                estimates save "${filename_FE}_dF", replace
1268                estimates restore gmm_est
1269        }
1270}

1271
1272
  end of do-file
1273}

1274else {
1275        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1276        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1277
1278        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
1279
1280        drop if size_group_round<min_group_size
1281        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1282
1283        do "$CODE/estim program.do"
1284}

1285
1286
1287log close
       name:  <unnamed>
        log:  C:\Users\mdelma\OneDrive - TUNI.fi\Tiedostot\replication\rep_table8_col3.
  > smcl
  log type:  smcl
 closed on:   8 Dec 2023, 17:31:18
```

B Appendix: Modification log files

B.1 Columns 1 and 2

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_modification.smcl
  log type:  smcl
 opened on:   8 Dec 2023, 13:50:15

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
  C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
50. //local group_def "dist"
51. //local group_def "fsuXseg"
52. //local group_def "fsuXsegXreligXsched"
53. local group_def "fsuXsegXreligXschedXehigh"

54.
55. // comparision group
56. local groupComp_def ""

57. local desc_compGrp ""

58. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

59.
60. // load data
61. include "$CODE/prep demand system data.do"

62. //clear all
63. pause on

64. set more off

65. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

66.
67. scalar short_zlist=$short_zlist_val

68.
69. cap prog drop bysmeanw

70. prog def bysmeanw
    1.         syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
71.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
72.         tempvar totweight totsum varXweight
    4.
73.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
74. end

75.
76. // load prices
77. use "$DATA/laspeyres_state_core.dta", clear

78. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

79. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)        expend_type   ->   (dropped)
  xij variables:
                         laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

80. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

81. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

82.
83. use "$DATA/laspeyres_state_core_2cats.dta", clear

84. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

85. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

86. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

87. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)        expend_type   ->   (dropped)
  xij variables:
                         laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

88. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
89. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

90.
91. if $use_norm==1 {
92.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
93.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                  Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations                 840   ->   280
  Number of variables                      5   ->   6
  j variable (3 values)          expend_type   ->   (dropped)
  xij variables:
                                laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

94.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
95.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
96. }

97.
98. // read in consumption data
99. use "$DATA/years1999_2004.dta", clear

100 *use "$DATA/years1999_2004_sm.dta", clear
101 g hhid = _n

102
103 // update expenditure if normal goods
104 if $use_norm==1 {
105         egen update_exp = rowtotal(food fuel clothing other_nondur)
106         replace expenditure = update_exp
  (197,188 real changes made)
107 }

108
109 // possible groups
110 egen group_district=group(state district)

111 egen group_round_district=group(state district round)

112 egen group_round_districtUrb=group(state district round urban)

113
114 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
115 drop _merge

116 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
117 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

118 drop _merge

119
120 //make weights comparable across rounds
121 bys round: egen meanweight=mean(weight)

122 bys round: replace weight=weight/meanweight
  (234590 real changes made)

123 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
124
125 // clean hh chars
126 replace religion=1 if religion==.
  (20 real changes made)

127 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
128 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

129 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
130 g scheduled = socgroup==1 | socgroup==2

131 g hindu = religion==1

132 g muslim = religion==2

133 egen hinduXscheduled = group(religion hindu)

134 g schednh = scheduled==1 & hindu==0

135 g schedh  = scheduled==1 & hindu==1

136 g nschedh = scheduled==0 & hindu==1

137 g nschednh = scheduled==0 & hindu==0

138 g nonHindu=religion>1

139
140 // simplify education categories
141 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

142
143 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
144 local cutoff_group_size=10

145 scalar cutoff_expenditure_percentile=99

146 scalar scale_factor=1

147
```

```
148 // generate variables
149 g low_educ = z9==0 & z10==0

150
151 // drop top and bottom percentiles of expenditure in each round/state
152 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

153 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

154 drop expenditure_cutoff

155 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

156 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

157
158 // keep urban houselds smaller than 13 and with head older than 20
159 keep if urban==1
  (141,042 observations deleted)

160 drop if hhsize>12
  (656 observations deleted)

161 gen married=marstat==2

162 drop if age<20
  (1,182 observations deleted)

163 g hhsizem1=hhsize-1

164
165 // winsorize weights
166 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

167 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

168 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

169
170 g lux=ilux+vlux
```

```
171 g necc=inec+vnec

172
173 // education variables
174 drop if zone==.
  (799 observations deleted)

175 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
176 g educmed=educsimp==2

177 g educhigh=educsimp==3

178
179 // scale age (helps numerical performance)
180 drop if age==.
  (1 observation deleted)

181 replace age=age/40
  (86,380 real changes made)

182
183 // rename z's so that they can be used in each file
184 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

185
186 // landowner dummy
187 g owns_land = landowned>.005 & landowned<.

188
189 egen group_inst=group(state district urban)

190
191 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

192 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

193 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)

194 if "`group_def'"=="fsuXsegXreligXschedXownsland"       egen group = group(geogroup_
  > seg religion scheduled owns_land)

195 if "`group_def'"=="fsuXsegXreligXschedXehigh"                  egen group = group(g
  > eogroup_seg religion scheduled educhigh)
  (43,637 missing values generated)
```

```
196
197 if $rand_grp==1 {
198         sort round
199         sort round state district
200         g shuffle = _n
201         g rand = runiform()
202         sort round rand
203         sort round state district rand
204         g group_new = group[shuffle]
205         replace group = group_new
206         drop group_new shuffle rand
207 }

208
209 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

210 egen district_round=group(state district round urban)

211
212 // make comparision group, so can restrict to sample in other regs
213 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

214 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

215 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)

216 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

217
218 if short_zlist==1 {
219         global T=7
220
221         g z1=hhsizem1/10
222         g z2=age/3
223         g z3=married
224         g z4=.
  (86,380 missing values generated)
225         g land=exp(demog23)
  (12,776 missing values generated)
226         replace z4=0 if land<=0.005
  (29,502 real changes made)
227         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
228         g z5=demog24
  (29 missing values generated)
229         g z6=educmed
230         g z7=educhigh
231         order land, after(z7)
232
233         global zlist "z1 z2 z3 z4 z5 z6 z7"
234         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
235         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
236 }

237
```

```
238 if short_zlist==2 {
239         global T=10
240
241         g z1=hhsizem1/10
242         g z2=age/3
243         g z3=married
244         g z4=.
245         g land=exp(demog23)
246         replace z4=0 if land<=0.005
247         replace z4=ln(land+1) if land>0.005
248         g z5=demog24
249         g z6=educmed
250         g z7=educhigh
251         g z8 = scheduled==1 & hindu==0
252         g z9 = scheduled==0 & hindu==0
253         g z10 = scheduled==0 & hindu==1
254
255         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
257         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
258 }

259
260 if short_zlist==3 {
261         global T=8
262
263         g z1=hhsizem1/10
264         g z2=age/3
265         g z3=married
266         g z4=.
267         g land=exp(demog23)
268         replace z4=0 if land<=0.005
269         replace z4=ln(land+1) if land>0.005
270         g z5=demog24
271         g z6=educmed
272         g z7=educhigh
273         g z8 = owns_land
274
275         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
276         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
277         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
278 }

279
280 if short_zlist==4 {
281         global T=7
282
283         g z1=hhsizem1/10
284         g z2=age/3
285         g z3=married
286         g z4=.
287         g land=exp(demog23)
288         replace z4=0 if land<=0.005
289         replace z4=ln(land+1) if land>0.005
290         g z5=demog24
291         g z6=educmed
292         g z7=educhigh
293
294         global zlist "z1 z2 z3 z4 z5 z6 z7"
295         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
296         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
297 }
298
299 if short_zlist==5 {
300         global T=6
301
302         g z1=age/3
303         g z2=married
304         g z3=.
305         g land=exp(demog23)
306         replace z3=0 if land<=0.005
307         replace z3=ln(land+1) if land>0.005
308         g z4=demog24
309         g z5=educmed
310         g z6=educhigh
311         order land, after(z6)
312
313         global zlist "z1 z2 z3 z4 z5 z6"
314         global zlist_indiv "z1 z2 z3 z4 z5 z6"
315         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
316 }
317
318 if short_zlist==6 {
319         global T=5
320
321         g z1=hhsizem1/10
322         g z2=p1
323         g z3=p2
324         g z4=p3
325         g z5=p4
326         g land=exp(demog23)
327
328         global zlist "z1 z2 z3 z4 z5"
329         global zlist_indiv "z1"
330         global xzlist_indiv "xz1"
331
332         drop if land==. | demog24==.
333 }
334
335 if short_zlist==7 {
336         global T=11
337
338         g z1=hhsizem1/10
339         g z2=age/3
340         g z3=married
341         g z4=.
342         g land=exp(demog23)
343         replace z4=0 if land<=0.005
344         replace z4=ln(land+1) if land>0.005
345         g z5=demog24
346         g z6=educmed
347         g z7=educhigh
348         g z8=p1
349         g z9=p2
350         g z10=p3
351         g z11=p4
352
```

```
353         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
354         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
355         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
356 }

357
358
359
360 // drop missings and calculate size of grp
361 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

362 bys group_round: egen size_group_round=count(group_round)

363
364 // size of comparision group
365 if "`groupComp_def'"!="" {
366         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
367 }

368
369 // turn things int oper capita terms if ption flagged
370 if $meas_pc==1 {
371         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
372 }

373
374 // normalize expenditure by overall mean
375 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |   73,576   45441.646    4270.163   2537.534   308.7667   27910.14

376 scalar expenditure_mean=r(mean)

377 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

378
379 if $use_norm==0 {
380         g pq1=vlux_norm
381         g pq2=ilux_norm
382         g pq3=vnec_norm
383         g pq4=inec_norm
384 }
```

```
385 else {
386         g pq1=food_norm
387         g pq2=fuel_norm
388         g pq3=clothing_norm
389         g pq4=other_nondur_norm
390 }

391 g x=expenditure_norm

392 g x2=x*x

393
394 // make q averages
395 qui bysort group_round: egen weight_group_round=sum(weight)

396
397 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

398
399 // 2 category consumption
400 g PQ1=pq1+pq2

401 g PQ2=pq3+pq4

402 if $use_vis==1 {
403         drop PQ1 PQ2
404         g PQ1=pq1+pq3
405         g PQ2=pq2+pq4
406 }

407 if $use_vislux==1 {
408         drop PQ1 PQ2
409         g PQ1=pq1
410         g PQ2=pq2+pq3+pq4
411 }

412 if $use_norm==1 {
413         drop PQ1 PQ2
414         g PQ1=pq1
415         g PQ2=pq2+pq3+pq4
416         if `J'==3 {
417                 drop PQ1 PQ2
418                 g PQ1=pq1
419                 g PQ2=pq2
420                 g PQ3=pq3+pq4
421         }
422 }

423
424 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
425 if `J'==3 {
426         qui g Q`J'=PQ`J'/P`J'
427         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
428 }

429
430 // only scheduled groups (with some of other group)
431 bys geogroup_seg religion: egen share_sched = mean(scheduled)

432 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

433
434 // update group epend wiht other group
435 if $only_scheduled==2 {
436         preserve
437                 keep if scheduled==0
438                 keep geogroup_seg religion Qbar* qbar*
439                 duplicates drop
440                 tempfile update_exp
441                 save `update_exp'
442         restore
443         keep if scheduled==1 & share_sched>0 & share_sched<1
444         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
445 }

446 drop share_sched

447
448 // laysperes_cpi varies by district and round
449 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

450
451 // make instruments
452 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
453 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
454 // measure all the x-stuff in reals, using aggregate laysperes index.
455 g weight_temp=weight

456 global instlist ""

457 g xreal=x/local_cpi
  (82 missing values generated)

458 g x_temp=x

459 g x2_temp=x2

460 replace x=xreal
  (73,576 real changes made, 82 to missing)

461 replace x2=x*x
  (73,576 real changes made, 82 to missing)

462
463 // make z x interactions
464 foreach var of varlist $zlist {
    2.         g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
465
466 levelsof round, local(roundlist)
  59 60 61 62

467 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

468
469 g xbar2lag=xbarlag^2
  (4 missing values generated)

470 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

471
472 bysmeanw x, weight(weight) by(group) rename(xbar)

473 bysmeanw x2, weight(weight) by(group) rename(x2bar)

474 g xbar2=xbar^2

475 g rootxbar = sqrt(xbar)

476 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

477
478 replace x=x_temp
  (73,576 real changes made)

479 replace x2=x2_temp
  (73,576 real changes made)

480
481 // add x terms to inst lsit
482 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

483
484 // only groups of size 3 or more
485 drop if size_group_round<3
  (53,343 observations deleted)

486 if "`groupComp_def'"!="" drop if size_groupComp_round<3

487 drop size_group_round

488 bys group_round: egen size_group_round=count(group_round)
```

```
489
490 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
491 if `J'==3 local extrakeep "Q3 P3 Qbar3"

492 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

493 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

494
495 g obs_numi=_n

496
497 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

498 rename x xi

499
500 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

501 rename weight weighti

502 rename land landi

503 rename owns_land owns_landi

504 rename schednh schednhi

505 rename schedh schedhi

506 rename nschedh nschedhi

507 rename nschednh nschednhi

508 g urbani = urban

509 rename hhid hhidi

510 rename scheduled scheduledi

511 rename hhsize hhsizei

512
513 tempfile obs_i

514 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_4630_000001.tmp saved as .dta format

515
516 // calculate sumstats at hh level
517 if `make_sumstats'==1 {
518         count
```

```
519         local sumstats_n_obs = r(N)
520
521         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
522                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
523         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
524         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
525 }

526
527 rename obs_numi obs_numk

528
529 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

530 rename xi xk

531
532 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

533 rename weighti weightk

534 rename landi landk

535 rename owns_landi owns_landk

536 rename urbani urbank

537 rename schednhi schednhik

538 rename schedhi schedhk

539 rename nschedhi nschedhk

540 rename nschednhi nschednhk

541 rename scheduledi scheduledk

542 rename hhsizei hhsizek

543 drop geogroup

544
545 // combine into pairs within groups
546 joinby group_round using "`obs_i'"

547 drop if obs_numi==obs_numk
  (20,233 observations deleted)

548
549 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
550 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

551
552 // summary stats tables
553 if `make_sumstats'==1 {
554         count
555         local sumstats_n_obs_pair = r(N)
556
557         cap file close sumstat
558                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
559                 file write sumstat "\begin{table}[htbp]\centering" _n
560                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
561                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
562                 file write sumstat "\begin{tabular}{lcccccccc}" _n
563                 file write sumstat "\toprule" _n
564                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
565                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
566                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
567                 file write sumstat "\midrule" _n
568
569                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
570                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var'" _n
    5.                 }
571
572                 file write sumstat "\bottomrule" _n
573                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
574                 file write sumstat "\end{tabular}" _n
575                 file write sumstat "\end{table}" _n
576                 file close sumstat
577                 BREAK
578 }

579
580 // constant for estimation
581 g one=1

582 egen state_district_round=group(state district urban round)

583 egen state_district=group(state district)

584
```

```
585 // keep only thick round
586 keep if round==61
  (0 observations deleted)

587
588
589
590
591
592
593 // options
594 local output_number="14"

595
596 // dimension of A matrix
597 scalar Afull=0

598 local descA "full"

599 scalar Adiag=1

600 if Adiag==1 local descA "diag"

601 scalar same_spillover=0

602 if same_spillover==1 local descA "same"

603
604 // minimum group size (must be 3 or more)
605 local min_group_size = 3

606 scalar min_group_size=`min_group_size'

607
608 // flag for generic model
609 scalar simple_model=0

610 local simp_name ""

611 if simple_model==1 local simp_name "_simple"

612
613 // interactions with peer effects
614 //global Alist "one"
615 //global Alist "one schednhi schedhi nschednhi"
616 //global Alist "one   zi8"
617 global Alist "one   zi7"

618
619 // turn on or off measurement error correction
620 scalar noMeasError=0

621 local measError_desc ""

622 if noMeasError==1 local measError_desc "_NoMeasError"

623
624 // estimate RE and or FE
625 scalar estimate_RE=1
```

```
626 scalar estimate_FE=1

627
628 // estimation options
629 global w_initial "winit(unadjusted)"

630 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

631 global wmatrix "wmatrix(robust)"

632 global trace_level ""

633 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

634 global technique ""

635 global maxiter ""

636 global technique_RE ""

637 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

638 global derivatives "quickderivatives"

639 scalar init_FE_from_RE=1

640 global clust_var "state_district_round"

641 local clus_name "_clus_Dist"

642
643
644
645
646
647
648
649
650 // xxzx to keep names same
651 local wmatrix_desc ""

652 local sectorName "_urbOnly"

653 local dropZdesc "_dropPre"

654 local desc_struc_v0 ""

655 local inst_desc "_exactInst"

656 local inst_rootp "_noPPXdInst"

657 local weight_desc ""

658 local RE_sq_inst_desc ""

659 local cnstname ""

660 local actual_pqhat_desc ""

661 local expUpname "_updateExp"
```

```
662
663
664 if $rand_grp==0 {
665         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
666         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
667
668         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
669
670         drop if size_group_round<min_group_size
  (0 observations deleted)
671         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
672
673         //do "$CODE/estim program.do"
674         do "$CODE/estim program.do"

675 // This code estimates a QES demand system
676 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
677 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
678 // i indexes observations, k indexes other observations in the group
679 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
680 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
681 //              ensure that all zt are z-scored or otherwise have similar scales
682 // qikbarj are leave-two-out group averages
683 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
684 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
685 // note that qi and qk are actually quantities, and not spending
686 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
687
688 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
689
690 set seed 339487731

691
692 local J=J

693 local Jm1=`J'-1

694 local T=$T

695 local Tm1=`T'-1
```

```
696 local Tindiv : word count $zlist_indiv

697
698 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
699
700 tempvar clust_weight num_group_round

701 bys group_round: egen `num_group_round'=count(group_round)

702 g double `clust_weight'=size_group_round/`num_group_round'

703 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

704 global GMM_weight "[aweight=`clust_weight']"

705
706 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
707
708 if `J'==2 {
709          replace p1=P1
  (84,052 real changes made)
710          replace p2=P2
  (84,052 real changes made)
711          replace qi1=Qi1
  (84,049 real changes made)
712          replace qk1=Qk1
  (84,049 real changes made)
713          replace qikbar1=Qikbar1
  (84,052 real changes made)
714          replace qi2=Qi2
  (84,052 real changes made)
715          replace qk2=Qk2
  (84,052 real changes made)
716          replace qikbar2=Qikbar2
  (84,052 real changes made)
717
718          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
719 }

720
721 if `J'==3 {
722          forv j = 1(1)3 {
    2.                 replace p`j'=P`j'
    3.                 replace qi`j'=Qi`j'
    4.                 replace qk`j'=Qk`j'
    5.                 replace qikbar`j'=Qikbar`j'
    6.          }
723
724          foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                 qui replace `var'=.
    3.          }
725 }

726
```

```
727
728 /*****************************************************************************
  >  Make moment equations
  > *****************************************************************************/
729
730 global zilist ""

731 global zklist ""

732 forval t=1/`Tm1' {
    2.         global zilist "$zilist zi`t'"
    3.         global zklist "$zklist zk`t'"
    4. }

733 global zilistTm1 "$zilist"

734 global zklistTm1 "$zklist"

735 if `T'>0 global zilist "$zilist zi`T'"

736 if `T'>0 global zklist "$zklist zk`T'"

737
738 // create b_p and pC
739 // remember b_p doesn't vary within group
740 local b_p "exp( 0"

741 forval j=1/`Jm1' {
    2.         local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

742 local b_p "`b_p' + ln(p`J') )"

743 if simple_model==1      local b_p "1"

744 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

745
746 // pCj is the sum of these two, for use in the RE model
747 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
748 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
749 forval j=1/`J' {
    2.         local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

750 local Tindivp1=`Tindiv'+1

751 forval j=1/`J' {
    2.         local pCi`j' "( 0 "
    3.         local pCk`j' "( 0 "
    4.         local pCg`j' "( 0 "
    5.         forval t=1/`Tindiv' {
    6.                 local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                 local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.         }
    9.         forval t=`Tindivp1'/`T' {
   10.                 local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.         }
   12.         local pCi`j' "`pCi`j'' )"
   13.         local pCk`j' "`pCk`j'' )"
   14.         local pCg`j' "`pCg`j'' )"
   15. }
```

```
752
753 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
754 local pC "( 0"

755 local pCi "( 0"

756 local pCk "( 0"

757 local pCg "( 0"

758 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j'"
    3.          local pCi "`pCi' + `pCi`j'"
    4.          local pCk "`pCk' + `pCk`j'"
    5.          local pCg "`pCg' + `pCg`j'"
    6. }

759 local pC "`pC' )"

760 local pCi "`pCi' )"

761 local pCk "`pCk' )"

762 local pCg "`pCg' )"

763
764 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
765 local pC_2nd "( 0"

766 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j'"
    4. }

767 local pC_2nd "`pC_2nd' )"

768
769 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
770 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
771 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
772 local rpDrp "( 0"

773 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
774         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

775
776 local rpDrp "`rpDrp' )"

777 di "`rpDrp'"
  ( 0 )

778
779 if simple_model==1      {
780         local pC "0"
781         local pC1 "0"
782         local pCi "0"
783         local pCk "0"
784         local pC_2nd "0"
785         local rpDrp "0"
786         local Drp1 "0"
787 }

788
789 // create Aq
790 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
791 // k means use qk as the group quantity (for use in RE only)
792 local qtypelist "ikbar k bar"

793 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
794                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
795                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
796         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
797                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
798         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
799
800 // Aikbar is A'qbar-minus-ik; Ak is A'qk
801 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
802 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
803 local xi_hat            "(xi - `pCi')"

804 local xk_hat            "(xk - `pCk')"

805 local x_hat                    "(xi - `pC' - `rpDrp' )"

806 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

807
808 // construct structural vo
809 * let AVA be A'VA, and ensure its diagonals are positive.
810 local v0 "(0 "

811 forval j=1/`J' {
    2.          local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
812         local jplus1=`j'+1
    4.          if `jplus1'<=`J' {
    5.                  forval k=`jplus1'/`J' {
    6.                          local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                  }
    8.          }
    9. }

813
814 local v0 "`v0' )"

815
816 // make all equations, then put pieces together
817
818 * RE equations; first equation has first call to C, other equations use {Cj: } form
819 * note "-vj:", this is because we subtract the structural v0 term from E[q]
820 local j=1

821
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

822 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

823 if `J'>2 {
824         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                  if noMeasError==1                                   local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
```

```
825 }

826
827 *       FE equations
828 forval j=1/`Jm1' {
    2.                                                                              l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                    ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

829
830 // clean up equations, and combine
831 local eqs_RE ""

832 local eqs_FE ""

833 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

834
835 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A22:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A22:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2
  > ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qikba
  > r1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d
  > 1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

836 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > }*p1*qikbar1) + ({A22:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))        ))
```

```
837
838 * replace the first occurrence of Ajj with the Alist
839 if Afull==0 {
840         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
841 }

842 if Afull==1 {
843         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
844 }

845
846 if simple_model==1 {
847         // construct z sums and interactions
848         if `T'>0 {
849
850                 local RE_cz_2            "+ {d}*(0"
851                 local RE_acz     ""
852                 local RE_acz1     ""
853                 local RE_acz2     ""
854                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
855                 local RE_czcz    ""
856                 local FE_czi     ""
857                 local FE_czk     ""
858                 local FE_czg     ""
859                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
860                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
861                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
862                 local RE_cz_2 "`RE_cz_2')^2"
863         }
864
865         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
866
867         local xi_hat "({b}*p1*xi `FE_czi')"
868         local xk_hat "({b}*p1*xk `FE_czk')"
869
870         local eqs_FE
  >                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
871 }

872
873 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one   zi7}*p1*qk1) + ({A22:one   zi7}*p2*qk2) )*( ({A11:
  > }*p1*qikbar1) + ({A22:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6
  >  zi7}) + (p2*{C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) +
  > ({A22:}*p2*qikbar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C
  > 1:}) + (p2*{C2:}) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*
  > {C1:}) + (p2*{C2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) ))*{b1}
  > + ({A11:}*p1*qikbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 +
  > p2*{AVA22}*p2 )*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

874 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > one   zi7}*p1*qikbar1) + ({A22:one   zi7}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))
  > *{d1}/exp( {b1}*ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi
  > 3 + p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p
  > 1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*z
  > k7 ))         ))

875
876 /*******************************************************************************
  >  construct instruments
  > *******************************************************************************/
877
878 //make squares and differences of x
879 tempvar x2i x2k ximxk x2imx2k xiP

880 g double `xiP'=xi*local_cpi

881 g double `x2i'=xi*xi

882 g double `x2k'=xi*xi

883 g double `ximxk'=xi-xk
```

```
884 g double `x2imx2k'=xi*xi-xk*xk

885
886 g xbarlag_x2barlag = xbarlag*x2barlag

887 g xbarlag2  = xbarlag^2

888 g x2barlag2 = x2barlag^2

889
890 //make interactions with z and zp
891 forval t=1/`T' {
    2.          tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.          g double `xizi`t''=xi*zi`t'
    4.          g double `xkzk`t''=xk*zk`t'
    5.          g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.          g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.          g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.          g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.          g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.          forval j=1/`J' {
   11.                  tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                  g double `zi`t'p`j''= zi`t'*p`j'
   13.                  g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                  g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                  g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                  forv s=1/`T' {
   17.                          tempvar zi`t'zi`s'p`j'
   18.                          g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                  }
   20.          }
   21. }

892
893 // make group-level instruments, and the instrument lists rg, xrg
894 if noMeasError==0       local rg "xbarlag"

895 if noMeasError==1       local rg "xbar"

896
897 local rg "`rg' x2barlag"

898 local rg "`rg' xbar2lag"

899 local rg "`rg' rootxbarlag"

900
901 local Tindivm1=`Tindiv'-1

902 // lag instruments
903 forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }

904
905 local zrg ""

906 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var''"
    6.          }
    7. }
```

```
907
908 local xrg ""

909 local prg ""

910 local xprg ""

911 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var''"
    5.
912          forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var''"
   10.                  local xprg "`xprg' `xp`j'`var''"
   11.          }
   12. }

913
914 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
915 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
916 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
917 local budget "xi `x2i'"

918
919 local rootprootp ""

920 local xrootprootp ""

921 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

922
923 // demographics multiplied by prices and budget
924 local zinsts ""

925 local xzinsts ""

926 local xPzinsts ""

927 local pzinsts ""

928 local xpzinsts ""

929 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }
```

```
930
931 // price insts
932 local roundinsts " "

933 forval j=1/`J' {
    2.         forval k=`j'/`J' {
    3.                 tempvar p`j'p`k'
    4.                 g `p`j'p`k''=p`j'*p`k'
    5.                 local roundinsts "`roundinsts' `p`j'p`k''"
    6.         }
    7. }

934
935 // instruments for all equations are ri:  x,x2,p,z,zx,
936 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
937 local qhat_insts ""

938 local qhat_insts "`qhat_insts' `zrg'"

939 local qhat_insts "`qhat_insts' `prg'"

940 local qhat_insts "`qhat_insts' `rg' "

941
942 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

943
944 local count: word count `r_RE'

945 di "total instruments: `count'"
  total instruments: 19

946
947 local qhat_hats ""

948 local pqhat ""

949 local xpqhat ""

950 local pqhat_sq ""

951 forval j=1/`J' {
    2.         if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.         if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
952         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
953         // use all prices times all qikhats as instruments, interacted with xi
954         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
955                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }

  Linear regression                               Number of obs     =     84,052
                                                  F(128, 4445)      =          .
                                                  Prob > F          =          .
                                                  R-squared         =     0.2501
                                                  Root MSE          =     .13615
```

(Std. err. adjusted for **4,446** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	-12.73252	7.369069	-1.73	0.084	-27.17957	1.714519
__00009L	-3.908656	1.317828	-2.97	0.003	-6.492254	-1.325057
__00009O	1.952542	3.984012	0.49	0.624	-5.858105	9.763189
xi	.1079365	.0095	11.36	0.000	.0893118	.1265612
__000002	-.0151526	.0031675	-4.78	0.000	-.0213625	-.0089427
__00000E	.0894046	.1010424	0.88	0.376	-.1086888	.287498
__00000P	.0062898	.0602955	0.10	0.917	-.1119193	.1244989
__000017	.1500829	.1818022	0.83	0.409	-.2063399	.5065058
__00001I	-.06866	.1119726	-0.61	0.540	-.288182	.150862
__000020	.027154	.0541844	0.50	0.616	-.0790744	.1333823
__00002B	-.0515801	.0294315	-1.75	0.080	-.1092805	.0061203
__00002T	-.0157976	.0754667	-0.21	0.834	-.1637499	.1321547
__000034	.0108097	.0573323	0.19	0.850	-.1015902	.1232096
__00003M	.0238553	.0605589	0.39	0.694	-.0948703	.1425809
__00003X	.0122294	.0384987	0.32	0.751	-.0632473	.087706
__00004F	-.066814	.0482764	-1.38	0.166	-.1614597	.0278317
__00004Q	.0320565	.0266222	1.20	0.229	-.0201363	.0842494
__000058	-.0519014	.095193	-0.55	0.586	-.2385271	.1347243
__00005J	-.0450822	.0577163	-0.78	0.435	-.1582348	.0680705
__00005U	.4811364	.4238445	1.14	0.256	-.3498097	1.312083
__00005V	.014548	.0768475	0.19	0.850	-.1361113	.1652074
__00005W	-.1497377	.1714757	-0.87	0.383	-.4859155	.1864401
__00005X	-.4583032	.4598629	-1.00	0.319	-1.359863	.443257
__00005Y	.1026161	.1150777	0.89	0.373	-.1229934	.3282257
__00005Z	.0137161	.1871368	0.07	0.942	-.3531653	.3805974
__000060	-.0301891	.0822054	-0.37	0.713	-.1913527	.1309745
__000061	-.0101523	.0540427	-0.19	0.851	-.1161029	.0957983
__000062	.0228172	.0375025	0.61	0.543	-.0507064	.0963408
__000063	-.0645826	.0512432	-1.26	0.208	-.1650449	.0358797
__000064	-.049856	.0648913	-0.77	0.442	-.1770753	.0773633
__000065	-1.05826	1.006148	-1.05	0.293	-3.030811	.9142914
__000066	.0057092	.1329888	0.04	0.966	-.2550151	.2664334
__000067	.2675357	.3748926	0.71	0.475	-.4674403	1.002512
__000068	1.295167	.9652226	1.34	0.180	-.5971494	3.187484
__000069	-.3305621	.194341	-1.70	0.089	-.7115671	.050443
__00006A	-.4518249	.349679	-1.29	0.196	-1.13737	.2337199
__00006B	-.1281442	.1342526	-0.95	0.340	-.3913462	.1350578
__00006C	-.0705728	.1186977	-0.59	0.552	-.3032793	.1621337
__00006D	-.0221978	.0638702	-0.35	0.728	-.1474151	.1030196
__00006E	-.268217	.1015146	-2.64	0.008	-.4672361	-.0691978
__00006F	-.1645999	.1263539	-1.30	0.193	-.4123166	.0831167
__00006G	-.2515089	.2243853	-1.12	0.262	-.6914159	.188398
__00006H	.0398296	.0368432	1.08	0.280	-.0324014	.1120606
__00006I	.0138952	.0863635	0.16	0.872	-.1554202	.1832105
__00006J	.2012072	.236487	0.85	0.395	-.262425	.6648394
__00006K	.075745	.0614647	1.23	0.218	-.0447564	.1962465
__00006L	.001442	.090585	0.02	0.987	-.1761497	.1790337
__00006M	-.0204062	.045365	-0.45	0.653	-.1093442	.0685317
__00006N	.0462515	.0261846	1.77	0.077	-.0050833	.0975863
__00006O	-.0278014	.0167481	-1.66	0.097	-.060636	.0050332
__00006P	-.017407	.0262634	-0.66	0.508	-.0688963	.0340824
__00006Q	.0035924	.0317898	0.11	0.910	-.0587315	.0659164
__00006R	-.558085	.5825991	-0.96	0.338	-1.700269	.5840992
__00006S	-.0336344	.0624044	-0.54	0.590	-.1559781	.0887093
__00006T	.2740001	.2277791	1.20	0.229	-.1725604	.7205606
__00006U	.3940912	.5036755	0.78	0.434	-.5933635	1.381546
__00006V	-.1727422	.0829799	-2.08	0.037	-.335424	-.0100603
__00006W	-.0266908	.1516779	-0.18	0.860	-.3240551	.2706734
__00006X	.0434122	.0569473	0.76	0.446	-.0682329	.1550573
__00006Y	.0353139	.0337511	1.05	0.295	-.0308551	.1014829
__00006Z	-.0519732	.0256215	-2.03	0.043	-.1022041	-.0017423
__000070	-.0032762	.0411726	-0.08	0.937	-.083995	.0774427
__000071	-.0910445	.0501467	-1.82	0.070	-.189357	.0072679
__000072	-.2795439	.3363418	-0.83	0.406	-.9389413	.3798535
__000073	-.0613711	.0465087	-1.32	0.187	-.1525513	.0298091
__000074	.1964182	.1271467	1.54	0.122	-.0528527	.4456892
__000075	.1755066	.3213495	0.55	0.585	-.4544984	.8055116

__000076	-.10287	.068151	-1.51	0.131	-.2364799	.0307398
__000077	-.0212849	.1109386	-0.19	0.848	-.2387797	.1962099
__000078	.0145733	.0464277	0.31	0.754	-.076448	.1055947
__000079	.0193689	.0385896	0.50	0.616	-.0562859	.0950237
__00007A	.023343	.0215788	1.08	0.279	-.0189622	.0656483
__00007B	-.0185474	.0333235	-0.56	0.578	-.083878	.0467831
__00007C	-.0816465	.0385531	-2.12	0.034	-.1572298	-.0060633
__00007D	.1379219	.2183944	0.63	0.528	-.2902398	.5660837
__00007E	-.0090376	.0343054	-0.26	0.792	-.0762933	.0582182
__00007F	-.0779297	.0820547	-0.95	0.342	-.2387977	.0829384
__00007G	.0601434	.2347476	0.26	0.798	-.4000787	.5203656
__00007H	-.0655412	.0545923	-1.20	0.230	-.1725693	.0414869
__00007I	-.064272	.0917696	-0.70	0.484	-.244186	.1156421
__00007J	-.0275497	.0354447	-0.78	0.437	-.097039	.0419395
__00007K	-.0355562	.0295291	-1.20	0.229	-.093448	.0223356
__00007L	.0057291	.0170182	0.34	0.736	-.0276349	.0390932
__00007M	-.0103075	.0283361	-0.36	0.716	-.0658604	.0452455
__00007N	.0406568	.031951	1.27	0.203	-.021983	.1032965
__00007O	-.3553033	.4323883	-0.82	0.411	-1.203	.4923931
__00007P	.1045004	.0715063	1.46	0.144	-.0356876	.2446883
__00007Q	-.0804392	.1705848	-0.47	0.637	-.4148703	.2539919
__00007R	.4957288	.4610228	1.08	0.282	-.4081053	1.399563
__00007S	-.0405426	.1102724	-0.37	0.713	-.2567314	.1756462
__00007T	-.2696218	.1965574	-1.37	0.170	-.6549722	.1157286
__00007U	.1165778	.0754072	1.55	0.122	-.0312578	.2644135
__00007V	.0006939	.0543475	0.01	0.990	-.1058542	.107242
__00007W	.0437502	.0326678	1.34	0.181	-.0202949	.1077953
__00007X	-.0277642	.0580464	-0.48	0.632	-.141564	.0860356
__00007Y	.0669801	.0687706	0.97	0.330	-.0678445	.2018047
__000080	-34.90927	17.15202	-2.04	0.042	-68.53576	-1.282783
__000082	2.77102	8.913254	0.31	0.756	-14.70339	20.24543
__000085	-.356454	.3873145	-0.92	0.357	-1.115783	.4028752
__000087	.4267385	.2658285	1.61	0.108	-.0944176	.9478946
__00008A	7.124866	3.344714	2.13	0.033	.5675621	13.68217
__00008C	-1.771513	1.695097	-1.05	0.296	-5.094748	1.551721
__00008F	42.89889	21.2218	2.02	0.043	1.293591	84.50418
__00008H	-.7682622	11.37816	-0.07	0.946	-23.07512	21.5386
__00008K	-.4212837	.5179607	-0.81	0.416	-1.436744	.5941772
__00008M	-1.071909	.4960119	-2.16	0.031	-2.044339	-.0994784
__00008P	.358066	.9117491	0.39	0.695	-1.429416	2.145548
__00008R	-1.327817	.6602769	-2.01	0.044	-2.622289	-.0333457
__00008U	-.081808	.3440669	-0.24	0.812	-.7563504	.5927344
__00008W	.4068209	.2689278	1.51	0.130	-.1204114	.9340532
__00008Z	.7600942	.2284302	3.33	0.001	.3122573	1.207931
__000091	-.4406069	.1607908	-2.74	0.006	-.7558369	-.125377
__000094	.1844437	.177546	1.04	0.299	-.1636348	.5325222
__000096	.0574517	.1500922	0.38	0.702	-.2368037	.3517072
__000099	-.2482066	.2524644	-0.98	0.326	-.7431626	.2467494
__00009B	.0618988	.2010635	0.31	0.758	-.3322857	.4560833
__00009E	-.1201186	.3108826	-0.39	0.699	-.7296032	.489366
__00009G	-.4216688	.2265725	-1.86	0.063	-.8658637	.0225262
xbarlag	32.89004	22.04866	1.49	0.136	-10.33632	76.11639
x2barlag	-.3384075	.4240148	-0.80	0.425	-1.169688	.4928726
xbar2lag	-4.997865	4.347636	-1.15	0.250	-13.5214	3.525666
rootxbarlag	-43.33674	27.18811	-1.59	0.111	-96.63897	9.96549
z1barlag	1.728336	.8419091	2.05	0.040	.0777746	3.378896
z2barlag	1.142782	1.170059	0.98	0.329	-1.151115	3.43668
z3barlag	-.3725164	.4869302	-0.77	0.444	-1.327142	.5821092
z4barlag	-.2461954	.2698192	-0.91	0.362	-.7751753	.2827845
z5barlag	-.2633075	.1880512	-1.40	0.162	-.6319814	.1053664
z6barlag	.2532203	.348117	0.73	0.467	-.4292622	.9357029
z7barlag	.5552983	.3880471	1.43	0.152	-.2054672	1.316064
_cons	15.31784	9.30933	1.65	0.100	-2.933083	33.56876

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	84,052	.4253486	.0785636	.199297	.8855615

Linear regression

Number of obs	=	**84,052**
F(128, 4445)	=	.
Prob > F	=	.
R-squared	=	**0.4130**
Root MSE	=	**.18969**

(Std. err. adjusted for **4,446** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	9.629519	8.08328	1.19	0.234	-6.217733	25.47677
__00009L	1.106562	1.793862	0.62	0.537	-2.4103	4.623423
__00009O	2.819742	4.609513	0.61	0.541	-6.217197	11.85668
xi	.1251284	.0157269	7.96	0.000	.0942959	.155961
__000002	-.0020993	.0062796	-0.33	0.738	-.0144104	.0102119
__00000E	.0212689	.1233318	0.17	0.863	-.2205228	.2630607
__00000P	-.0639746	.0779059	-0.82	0.412	-.2167089	.0887597
__000017	.2896356	.2595834	1.12	0.265	-.2192771	.7985483
__00001I	.1762503	.166318	1.06	0.289	-.1498157	.5023164
__000020	.0708586	.0653964	1.08	0.279	-.057351	.1990682
__00002B	.0402053	.0371738	1.08	0.280	-.0326739	.1130845
__00002T	.2554372	.0792663	3.22	0.001	.1000358	.4108387
__000034	-.053384	.0654696	-0.82	0.415	-.181737	.0749691
__00003M	.029735	.0723042	0.41	0.681	-.1120172	.1714872
__00003X	-.0018951	.0472524	-0.04	0.968	-.0945333	.0907431
__00004F	-.0634327	.0566019	-1.12	0.262	-.1744006	.0475351
__00004Q	.027829	.0358356	0.78	0.437	-.0424267	.0980847
__000058	.0150677	.1604314	0.09	0.925	-.2994577	.3295931
__00005J	.0177828	.0974184	0.18	0.855	-.1732059	.2087714
__00005U	-.326043	.5650175	-0.58	0.564	-1.433759	.7816726
__00005V	.0570096	.0880207	0.65	0.517	-.1155548	.229574
__00005W	-.0213147	.2226448	-0.10	0.924	-.4578093	.41518
__00005X	.2829804	.5728053	0.49	0.621	-.8400031	1.405964
__00005Y	.1505949	.1440063	1.05	0.296	-.1317291	.432919
__00005Z	.1165222	.2342244	0.50	0.619	-.3426741	.5757186
__000060	-.1906844	.1186647	-1.61	0.108	-.4233262	.0419574
__000061	.057797	.0579556	1.00	0.319	-.0558248	.1714188
__000062	-.0355028	.046257	-0.77	0.443	-.1261895	.055184
__000063	.0126584	.0588251	0.22	0.830	-.1026681	.127985
__000064	-.0006398	.0727721	-0.01	0.993	-.1433092	.1420297
__000065	2.08159	1.269419	1.64	0.101	-.407103	4.570282
__000066	.1521409	.1994681	0.76	0.446	-.238916	.5431977
__000067	-.8048261	.5069429	-1.59	0.112	-1.798687	.1890343
__000068	-1.313909	1.259324	-1.04	0.297	-3.782811	1.154993
__000069	-.5768386	.2729932	-2.11	0.035	-1.112041	-.0416359
__00006A	-.1026947	.5271538	-0.19	0.846	-1.136179	.9307892
__00006B	-.1269372	.1864405	-0.68	0.496	-.4924534	.2385791
__00006C	-.4209147	.1299051	-3.24	0.001	-.6755934	-.166236
__00006D	-.0107406	.0876206	-0.12	0.902	-.1825207	.1610394
__00006E	-.2439648	.1385943	-1.76	0.078	-.5156787	.0277491
__00006F	-.2456211	.168179	-1.46	0.144	-.5753356	.0840933
__00006G	.1892016	.3713697	0.51	0.610	-.5388678	.917271
__00006H	.0602881	.0488747	1.23	0.217	-.0355306	.1561068
__00006I	-.1195145	.1419753	-0.84	0.400	-.3978568	.1588279
__00006J	-.3124916	.3363266	-0.93	0.353	-.971859	.3468759
__00006K	.0562282	.0772067	0.73	0.466	-.0951355	.2075918
__00006L	.0001789	.1217172	0.00	0.999	-.2384475	.2388052
__00006M	.0226298	.06748	0.34	0.737	-.1096645	.1549242
__00006N	.0202324	.0294364	0.69	0.492	-.0374775	.0779424
__00006O	.0013999	.0227663	0.06	0.951	-.0432333	.0460331
__00006P	-.0435669	.0319756	-1.36	0.173	-.1062549	.0191212
__00006Q	.0023696	.0397697	0.06	0.952	-.0755988	.0803379
__00006R	-.028743	.4379357	-0.07	0.948	-.887315	.8298291
__00006S	.0648557	.0669205	0.97	0.333	-.0663418	.1960531
__00006T	.043945	.1615795	0.27	0.786	-.2728312	.3607212
__00006U	-.4790627	.4205855	-1.14	0.255	-1.30362	.3454943
__00006V	.0551929	.0915788	0.60	0.547	-.1243471	.2347329
__00006W	.2464766	.1838457	1.34	0.180	-.1139525	.6069056
__00006X	.097147	.065464	1.48	0.138	-.031195	.2254891
__00006Y	.1130417	.0411917	2.74	0.006	.0322854	.1937981
__00006Z	-.1040381	.0313626	-3.32	0.001	-.1655243	-.0425519

__000070	-.0975131	.0491546	-1.98	0.047	-.1938805	-.0011457
__000071	-.0342095	.062284	-0.55	0.583	-.1563171	.0878981
__000072	-.5269194	.3543749	-1.49	0.137	-1.221671	.1678319
__000073	-.0468236	.0544254	-0.86	0.390	-.1535244	.0598773
__000074	.2469635	.1436979	1.72	0.086	-.034756	.5286829
__000075	.2392732	.3375853	0.71	0.478	-.422562	.9011085
__000076	.1281287	.0813462	1.58	0.115	-.0313503	.2876076
__000077	.0200856	.1290184	0.16	0.876	-.2328546	.2730258
__000078	-.0140711	.0556515	-0.25	0.800	-.1231758	.0950335
__000079	.0814369	.0374041	2.18	0.030	.0081063	.1547676
__00007A	.0421845	.0252955	1.67	0.095	-.0074073	.0917763
__00007B	-.0024611	.0375964	-0.07	0.948	-.0761689	.0712467
__00007C	.0120392	.0431916	0.28	0.780	-.0726377	.0967162
__00007D	.3125724	.2440836	1.28	0.200	-.165953	.7910978
__00007E	.025896	.0403614	0.64	0.521	-.0532324	.1050243
__00007F	-.1596487	.0964736	-1.65	0.098	-.348785	.0294876
__00007G	-.051805	.2549994	-0.20	0.839	-.5517309	.4481208
__00007H	-.0953725	.0592071	-1.61	0.107	-.211448	.0207029
__00007I	-.1163743	.1102801	-1.06	0.291	-.3325782	.0998296
__00007J	.018921	.038441	0.49	0.623	-.0564426	.0942845
__00007K	-.0869756	.0300097	-2.90	0.004	-.1458095	-.0281417
__00007L	-.0127916	.0204976	-0.62	0.533	-.0529771	.0273939
__00007M	.010521	.0282985	0.37	0.710	-.0449581	.0660001
__00007N	-.0180405	.0350262	-0.52	0.607	-.0867092	.0506282
__00007O	-.0081516	.8382152	-0.01	0.992	-1.651471	1.635167
__00007P	.3107059	.1183335	2.63	0.009	.0787133	.5426985
__00007Q	-.4109411	.3260791	-1.26	0.208	-1.050218	.2283363
__00007R	.3407095	.7987095	0.43	0.670	-1.225159	1.906578
__00007S	-.5534748	.1682856	-3.29	0.001	-.8833984	-.2235513
__00007T	-.2196563	.2969799	-0.74	0.460	-.8018847	.3625721
__00007U	.2872022	.1201019	2.39	0.017	.0517427	.5226617
__00007V	-.2009352	.0679227	-2.96	0.003	-.3340976	-.0677728
__00007W	.0140617	.0573991	0.24	0.806	-.098469	.1265924
__00007X	-.056224	.0826135	-0.68	0.496	-.2181876	.1057397
__00007Y	-.0080098	.094626	-0.08	0.933	-.1935238	.1775043
__000080	23.04426	17.9944	1.28	0.200	-12.23372	58.32224
__000082	6.739301	10.28004	0.66	0.512	-13.4147	26.8933
__000085	.0366167	.4613836	0.08	0.937	-.8679248	.9411582
__000087	.1273868	.3305412	0.39	0.700	-.5206385	.775412
__00008A	-4.520475	3.435896	-1.32	0.188	-11.25654	2.215591
__00008C	-1.253674	1.946983	-0.64	0.520	-5.070729	2.563382
__00008F	-28.44369	22.68573	-1.25	0.210	-72.91903	16.03164
__00008H	-8.673547	13.05254	-0.66	0.506	-34.26303	16.91593
__00008K	-.3171968	.5838269	-0.54	0.587	-1.461788	.8273946
__00008M	-1.378233	.5476595	-2.52	0.012	-2.451918	-.304548
__00008P	1.346432	1.090681	1.23	0.217	-.791847	3.48471
__00008R	-1.460585	.8627891	-1.69	0.091	-3.152082	.2309109
__00008U	-1.032953	.4043974	-2.55	0.011	-1.825773	-.2401328
__00008W	.5845627	.3269034	1.79	0.074	-.0563307	1.225456
__00008Z	.3738756	.2324088	1.61	0.108	-.0817613	.8295126
__000091	.0382166	.1811049	0.21	0.833	-.316839	.3932723
__000094	-.6116784	.2279377	-2.68	0.007	-1.05855	-.1648071
__000096	-.1050342	.1884987	-0.56	0.577	-.4745854	.2645171
__000099	.6643731	.2628324	2.53	0.012	.1490908	1.179656
__00009B	.2677219	.2431047	1.10	0.271	-.2088842	.7443281
__00009E	-.1939407	.3366949	-0.58	0.565	-.8540303	.4661489
__00009G	-.2679983	.2820832	-0.95	0.342	-.8210218	.2850252
xbarlag	-31.50109	20.33455	-1.55	0.121	-71.36692	8.364739
x2barlag	-.4339314	.5076501	-0.85	0.393	-1.429178	.5613155
xbar2lag	6.658686	3.97354	1.68	0.094	-1.13143	14.4488
rootxbarlag	38.90188	25.41688	1.53	0.126	-10.92786	88.73163
z1barlag	1.921348	.8663586	2.22	0.027	.2228537	3.619842
z2barlag	.04958	1.394448	0.04	0.972	-2.684232	2.783392
z3barlag	.4241796	.5396868	0.79	0.432	-.6338751	1.482234
z4barlag	-.2806967	.2681117	-1.05	0.295	-.8063291	.2449357
z5barlag	.8190618	.2280605	3.59	0.000	.3719497	1.266174
z6barlag	-.8933056	.411293	-2.17	0.030	-1.699645	-.0869666
z7barlag	.4867033	.4527264	1.08	0.282	-.4008658	1.374273
_cons	-13.94648	8.863302	-1.57	0.116	-31.32297	3.430002

(option **xb** assumed; fitted values)

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000A2 |     84,052    .4027835    .1589901    .1248021   1.461195
```

```
956
957 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"
958
959 * interact qhat_hats with Alist
960 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9
961 if "$Alist"~="one" {
962         local qhat_hats_backup "`qhat_hats'"
963         local qhat_hats ""
964         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __00009W |     84,052    .4615108    .1003007    .1982114   1.064132
         one |     84,052           1           0           1          1
__00009W_one |     84,052    .4615108    .1003007    .1982114   1.064132
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __00009W |     84,052    .4615108    .1003007    .1982114   1.064132
         zi7 |     84,052    .1309427    .3373397           0          1
__00009W_zi7 |     84,052    .0770198    .2018732           0   1.064132
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __00009Z |     84,052    .5102133    .1199581    .1972201   1.195608
         one |     84,052           1           0           1          1
__00009Z_one |     84,052    .5102133    .1199581    .1972201   1.195608
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __00009Z |     84,052    .5102133    .1199581    .1972201   1.195608
         zi7 |     84,052    .1309427    .3373397           0          1
__00009Z_zi7 |     84,052    .0857241     .225065           0   1.195608
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000A5 |     84,052    .4407914     .189912    .1215886   1.546164
         one |     84,052           1           0           1          1
__0000A5_one |     84,052    .4407914     .189912    .1215886   1.546164
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000A5 |     84,052    .4407914     .189912    .1215886   1.546164
         zi7 |     84,052    .1309427    .3373397           0          1
__0000A5_zi7 |     84,052    .1008931    .2711148           0   1.546164
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000A8 |     84,052    .4829239    .2085856    .136642   1.972777
         one |     84,052           1           0           1          1
__0000A8_one |     84,052    .4829239    .2085856    .136642   1.972777
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
      __0000A8 |     84,052    .4829239    .2085856    .136642   1.972777
         zi7 |     84,052    .1309427    .3373397           0          1
__0000A8_zi7 |     84,052    .1117997    .3010185           0   1.972777
```

Variable	Obs	Mean	Std. dev.	Min	Max
__00009X	84,052	.4702234	.3851815	.023333	3.618202
one	84,052	1	0	1	1
__00009X_one	84,052	.4702234	.3851815	.023333	3.618201

Variable	Obs	Mean	Std. dev.	Min	Max
__00009X	84,052	.4702234	.3851815	.023333	3.618202
zi7	84,052	.1309427	.3373397	0	1
__00009X_zi7	84,052	.1171405	.362284	0	3.618201

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A0	84,052	.5196248	.4319204	.0242968	4.40807
one	84,052	1	0	1	1
__0000A0_one	84,052	.5196248	.4319204	.0242968	4.408071

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A0	84,052	.5196248	.4319204	.0242968	4.40807
zi7	84,052	.1309427	.3373397	0	1
__0000A0_zi7	84,052	.1304855	.4063337	0	4.408071

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A6	84,052	.48585	.5328118	.0206382	6.614207
one	84,052	1	0	1	1
__0000A6_one	84,052	.48585	.5328118	.0206382	6.614207

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A6	84,052	.48585	.5328118	.0206382	6.614207
zi7	84,052	.1309427	.3373397	0	1
__0000A6_zi7	84,052	.1587206	.5187135	0	6.614207

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A9	84,052	.5337806	.599019	.0214907	8.280542
one	84,052	1	0	1	1
__0000A9_one	84,052	.5337806	.599019	.0214907	8.280542

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A9	84,052	.5337806	.599019	.0214907	8.280542
zi7	84,052	.1309427	.3373397	0	1
__0000A9_zi7	84,052	.1765088	.585031	0	8.280542

```
965 }

966
967 local r_RE      "`r_RE' `qhat_hats'"

968
969 /*******************************************************************************
  > Set instruments and start values
  > *******************************************************************************/
970
971 if simple_model==1      {
972         local r_RE "xbarlag `xixbarlag' `budget'"
973         local r_RE1 ""
```

```
974
975         forv i=1/`T' {
    2.                 local r_RE "`r_RE' zi`i'"
    3.         }
976
977         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
978 }

979
980 if `J'==2 & simple_model==0     {
981         g y = p1*qi1
982         g ybar_q1 = p1*qikbar1
983         g ybar_q2 = p2*qikbar2
984         g ybar = p1*qikbar1 + p2*qikbar2
985         g ybar2 = ybar^2
986         g ybar_q1_2 = ybar_q1^2
987         g ybar_q2_2 = ybar_q2^2
988         g ybar_q1_q2 = ybar_q1*ybar_q2
989         g ybarx = ybar*xi
990         g ybarx_1 = ybar_q1*xi
991         g ybarx_2 = ybar_q2*xi
992         g xi2 = xi^2
993
994         forv j = 1(1)`T' {
    2.                 g pzi`j' = p1*zi`j'
    3.         }
995
996         if same_spillover==1 {
997                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
998                 local b = _b[xi]
999                 local a = _b[ybar_q1]
1000                if same_spillover==1 {
1001                        local a = logit(_b[ybar_q1]/2 + .5)
1002                        if `a'==. local a = .5
1003                }
1004                local d = _b[xi2]
1005                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1006        }
1007        else {
1008                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
```

Source	SS	df	MS		
Model	**3745.8161**	**16**	**234.113506**	Number of obs	= **84,052**
Residual	**577.498027**	**84,035**	**.006872113**	F(16, 84035)	= **34067.18**
				Prob > F	= **0.0000**
				R-squared	= **0.8664**
				Adj R-squared	= **0.8664**
Total	**4323.31413**	**84,051**	**.051436796**	Root MSE	= **.0829**

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
xi2	**-.0380531**	**.0007269**	**-52.35**	**0.000**	**-.0394778**	**-.0366285**
ybar_q1_2	**-.1266132**	**.0099958**	**-12.67**	**0.000**	**-.146205**	**-.1070215**
ybar_q2_2	**.106099**	**.0025665**	**41.34**	**0.000**	**.1010687**	**.1111294**
ybarx_1	**.2170387**	**.0040107**	**54.11**	**0.000**	**.2091776**	**.2248997**
ybarx_2	**-.1088644**	**.0021948**	**-49.60**	**0.000**	**-.1131661**	**-.1045626**
ybar_q1_q2	**-.0625185**	**.0089455**	**-6.99**	**0.000**	**-.0800516**	**-.0449854**
xi	**.3874886**	**.0021597**	**179.42**	**0.000**	**.3832556**	**.3917216**
ybar_q1	**.2079331**	**.008093**	**25.69**	**0.000**	**.1920709**	**.2237954**
ybar_q2	**-.1471019**	**.0044131**	**-33.33**	**0.000**	**-.1557516**	**-.1384522**
pzi1	**.2032607**	**.0015097**	**134.64**	**0.000**	**.2003017**	**.2062197**
pzi2	**.0139818**	**.0024327**	**5.75**	**0.000**	**.0092138**	**.0187498**
pzi3	**-.0028606**	**.0007467**	**-3.83**	**0.000**	**-.0043241**	**-.001397**
pzi4	**.0093582**	**.0008016**	**11.67**	**0.000**	**.0077871**	**.0109293**
pzi5	**-.0115325**	**.0007586**	**-15.20**	**0.000**	**-.0130194**	**-.0100457**
pzi6	**-.0001209**	**.00062**	**-0.20**	**0.845**	**-.0013361**	**.0010942**
pzi7	**-.0036092**	**.0010691**	**-3.38**	**0.001**	**-.0057045**	**-.0015138**
_cons	**-.0048422**	**.0021592**	**-2.24**	**0.025**	**-.0090742**	**-.0006103**

```
1009                local b = _b[xi]
1010                local a1 = _b[ybar_q1]/(1-`b')
1011                local a2 = _b[ybar_q2]/`b'
1012                if same_spillover==1 {
1013                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1014                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1015
1016                        if `a1'==. local a1 = .5
1017                        if `a2'==. local a2 = .5
1018                }
1019                local d = _b[xi2]
1020
1021                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1022        }
1023
1024        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1025        forv j = 1(1)`T' {
    2.                  local coef = _b[pzi`j']
    3.                  local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1026}

1027
1028if `J'==3 & simple_model==0
  >                             local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1030if `J'==4 & simple_model==0
  >                             local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1032
1033if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1034
1035if `J'==2 & estimate_RE==1 {
1036
1037        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00117105
  Iteration 1:  GMM criterion Q(b) =  .00088095
  Iteration 2:  GMM criterion Q(b) =   .0005399
  Iteration 3:  GMM criterion Q(b) =  .00014971
  Iteration 4:  GMM criterion Q(b) =   .0001264

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .00615758
  Iteration 1:  GMM criterion Q(b) =  .00526018
  Iteration 2:  GMM criterion Q(b) =   .0050956
  Iteration 3:  GMM criterion Q(b) =   .0047819

  GMM estimation

  Number of parameters =  23
  Number of moments    =  36
  Initial weight matrix: Unadjusted                 Number of obs   =     84,052
  GMM weight matrix:     Robust
```

(Std. err. adjusted for **561** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
A11						
one	.7048788	.0397658	17.73	0.000	.6269392	.7828184
zi7	-.1723597	.0516795	-3.34	0.001	-.2736496	-.0710699
A22						
one	.6496691	.0435246	14.93	0.000	.5643624	.7349758
zi7	-.1692322	.0768955	-2.20	0.028	-.3199446	-.0185198
C1						
zi1	.459543	.0488091	9.42	0.000	.3638789	.5552072
zi2	-.298735	.0848856	-3.52	0.000	-.4651076	-.1323624
zi3	.0135866	.0228002	0.60	0.551	-.031101	.0582741
zi4	.0986633	.0260686	3.78	0.000	.0475698	.1497569
zi5	.0442149	.024436	1.81	0.070	-.0036788	.0921087
zi6	-.0092271	.0170754	-0.54	0.589	-.0426943	.0242401
zi7	-.0064892	.0308463	-0.21	0.833	-.0669467	.0539684
C2						
zi1	.2392336	.0755102	3.17	0.002	.0912363	.3872309
zi2	-.5083536	.1334985	-3.81	0.000	-.7700058	-.2467013
zi3	.0258167	.0359118	0.72	0.472	-.0445691	.0962026
zi4	.1586987	.0409454	3.88	0.000	.0784472	.2389503
zi5	.0797242	.0309841	2.57	0.010	.0189966	.1404519
zi6	-.0179251	.0243051	-0.74	0.461	-.0655623	.0297121
zi7	-.0772149	.0402942	-1.92	0.055	-.1561901	.0017603
/d1	-.089963	.0076951	-11.69	0.000	-.105045	-.0748809
/b1	.4191021	.0239534	17.50	0.000	.3721543	.4660498
/AVA11	1.101851	.6462922	1.70	0.088	-.1648581	2.368561
/AVA12	-1.031426	.5679454	-1.82	0.069	-2.144579	.0817262
/AVA22	.4388451	.5370758	0.82	0.414	-.613804	1.491494

Instruments for equation **eq1_RE**: **__00009J __00009L __00009O xi __000002 __00000E __00000P __000017 __0000II __000020 __00002B __00002T __000034 __00003M __00003X __00004F __00004Q __000058 __00005J __00009W_one __00009W_zi7 __00009Z_one __00009Z_zi7 __0000A5_one __0000A5_zi7 __0000A8_one __0000A8_zi7 __00009X_one __00009X_zi7 __0000A0_one __0000A0_zi7 __0000A6_one __0000A6_zi7 __0000A9_one __0000A9_zi7 _cons**

```
1038
1039        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1040        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1041
1042        if simple_model==1 {
1043                /7 calculate derivative and save
1044                local RE_acz_drv ""
1045                tempvar junk
1046                g `junk'=.
1047                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
```

```
1048                replace `junk'=p1*qikbar1
1049                su `junk' $GMM_weight
1050                local mean_y = r(mean)
1051                su xi $GMM_weight
1052                local mean_x = r(mean)
1053
1054                local c_term ""
1055
1056                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1057                estimates store gmm_est
1058                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1059                estimates save "${filename_RE}_dF", replace
1060                estimates restore gmm_est
1061        }
1062}

1063
1064if `J'==3 & estimate_RE==1 {
1065        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1066}

1067if `J'==4 & estimate_RE==1 {
1068        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1069}

1070esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh_drop3_all_ex
      > actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not
      found)
  (output written to diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh
  > _drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex
  > )

1071
1072
1073// save coefs
1074matrix est_b=e(b)

1075unique hhidi
  Number of unique values of hhidi is  20233
  Number of records is  84052

1076local N_hh = r(sum)

1077estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  20233
```

```
1078unique group_round
  Number of unique values of group_round is  4446
  Number of records is  84052

1079estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  4446

1080estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
         e(avg_grp_size) =  4.5508322

1081if estimate_RE==1 {
1082        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh_drop3_
  > all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster not f
  > ound)
  file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh_drop3_all_exa
  > ctInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1083        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1084}

1085
1086estimates store temp1_FE

1087
1088/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1089
1090if simple_model==0 {
1091
1092        // make FE instruments
1093
1094        // denoting ximxk as x and zimzk of individual zs as zi:
1095        //   x,x2,p,zi,px,zi,pzi
1096        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1097        local xdiff "`ximxk' "
1098        local xdiffxdiff ""
1099        local zindivdiff ""
1100        local pzindivdiff ""
1101        local pzindivdiffpqhat ""
1102        local pxzindivdiff ""
1103        local p2zindiv2diff ""
1104        local zg ""
1105        local pzg ""
1106        local zindivdiffxdiff ""
1107        local pzindivdiffxdiff ""
1108        local pzindivdiffp ""
1109        local pzindivdiffpzg ""
1110        local zgxdiff ""
1111        local pzgxdiff ""
1112        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1113        g double `ximxkxdiff'=`ximxk'*`ximxk'
1114        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1115        g double `xixkxdiff'=xi*xk*`ximxk'
1116
```

```
1117        local xdiffxdiff "`x2imx2k'"
1118
1119        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
1120                                                gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                  local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                          }
   30.                                  }
   31.                          }
   32.                  }
   33.          }
1121        if `Tindivp1'<=`T' {
1122                forval t=`Tindivp1'/`T' {
    2.                          tempvar zi`t'xdiff
    3.                          g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                          local zg "`zg' zi`t'"
    5.                          forval j=1/`J' {
    6.                                  tempvar zi`t'p`j'xdiff
    7.                                  g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                  local pzg "`pzg' `zi`t'p`j''"
    9.                                  local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                          }
   11.                  }
1123        }
```

```
1124
1125        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1126        // denoting zg as group-zs
1127        //  zg,pzgx,rprp,rprpx
1128        local rootprootp ""
1129        local rootprootpxdiff ""
1130
1131        forval j=1/`J' {
    2.                  forval s=`j'/`J' {
    3.                          tempvar xdiffrp`j'rp`s'
    4.                          g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1132                        if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1133                        local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                  }
    8.          }
1134
1135        local pqhatxdiff ""
1136        forval j=1/`J' {
    2.                  forval s=1/`J' {
    3.                          foreach name of global Alist {
    4.                                  tempvar   p`s'qikhat`j'xdiff`name'
    5.                                  g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
1137                                    local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
1138
1139        // set instruments
1140        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1141}

1142
1143
1144if simple_model==1 {
1145        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1146
1147        /******************
  >         Initial values
  >         ******************/
1148
1149        if init_FE_from_RE==1 {
1150                estimates use "$filename_RE"
1151
1152                // clear init values
1153                local initial_values_FE ""
1154
1155                // extract vector of coefs and paramter names from
1156                matrix coefs = e(b)
1157                local paramlist = e(params)
```

```
1158
1159                // iterate through paramter name list, taking
1160                local t=0
1161                foreach p of local paramlist {
    2.                          local `++t'
    3.                          local p_nice = subinstr("`p'",":_cons","",.)
    4.                          local est = coefs[1,`t']
    5.                          if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                  }
1162        }
1163
1164        // make z diff instruments
1165        local pzindivdiff ""
1166        local pz2indivdiff ""
1167        local pzXzindivdiff ""
1168        local pxzindivdiff ""
1169
1170        forv i=1/`T' {
    2.
1171                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1172                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1173                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1174                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                 }
   16.         }
1175
1176        // p z interactions
1177        local pzg ""
1178        local pzgxdiff ""
1179        if `Tindivp1'<=`T' {
1180                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1181        }
1182
1183        tempvar p1ximxk  p1x2imx2k
1184        g `p1ximxk' = p1*`ximxk'
1185        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1186
1187        // define instruments
1188        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
```

```
1189
1190        // add prediction of quantity as extra instrument
1191        local pqhat ""
1192        local qhat ""
1193        local pqhatxdiff ""
1194        local pqhatxdiff2 ""
1195        local pzindivdiffpqhat ""
1196        local pzindivdiffpxbar_lag ""
1197
1198        tempvar qik_hat1
1199         reg qikbar1    `r_RE' `qhat_insts', vce(cluster group_round)
1200        predict `qik_hat1'
1201        local qhat "`qhat' `qik_hat1'"
1202
1203        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1204
1205        g double `p1_qikbar1' = p1*qikbar1
1206        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1207        predict `p1_qikhat1'
1208        local pqhat "`pqhat' `p1_qikhat1'"
1209
1210        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1211        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1212
1213        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1214
1215        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1216        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1217        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1218
1219        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1220                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1221
1222        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1223}

1224
1225// starting values
1226if init_FE_from_RE==1 & "$init_FE_file"!="" {
1227        estimates use "$init_FE_file"
1228        local initial_values_FE ""
1229
1230        // extract vector of coefs and paramter names from
1231        matrix coefs = e(b)
1232        local paramlist = e(params)
1233
1234        // iterate through paramter name list, taking
1235        local t=0
1236        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
```

```
1237}

1238
1239if estimate_FE==1 {
1240        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique
  note: instruments __00005B __00005M __0000D7 __0000DA __0000DG __0000DJ __0000DP
    __0000DQ __0000DS __0000DT __0000DY __0000DZ __0000E1 __0000E2 omitted because
    of collinearity.

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00064443
  Iteration 1:  GMM criterion Q(b) =  .00025878
  Iteration 2:  GMM criterion Q(b) =  .00019292

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01014222
  Iteration 1:  GMM criterion Q(b) =  .00936002
  Iteration 2:  GMM criterion Q(b) =  .00897646
  Iteration 3:  GMM criterion Q(b) =  .00895632

  GMM estimation

  Number of parameters =  20
  Number of moments    =  81
  Initial weight matrix: Unadjusted                 Number of obs   =     84,052
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 561 clusters in state_district_round)
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11						
_cons	.5739494	.0283178	20.27	0.000	.5184475	.6294513
C12						
_cons	-.0983264	.0471608	-2.08	0.037	-.1907598	-.005893
C13						
_cons	-.0232263	.0133533	-1.74	0.082	-.0493982	.0029455
C14						
_cons	-.0214373	.0140281	-1.53	0.126	-.0489319	.0060573
C15						
_cons	-.0141264	.0184913	-0.76	0.445	-.0503687	.022116
C16						
_cons	-.0241854	.0143834	-1.68	0.093	-.0523764	.0040056
C17						
_cons	-.7546413	.4205034	-1.79	0.073	-1.578813	.0695303
C21						
_cons	.385837	.0382534	10.09	0.000	.3108617	.4608123
C22						
_cons	-.1585207	.0658059	-2.41	0.016	-.2874979	-.0295434
C23						
_cons	-.0266065	.0174189	-1.53	0.127	-.060747	.007534
C24						
_cons	-.0524119	.0229507	-2.28	0.022	-.0973946	-.0074293
C25						
_cons	.0003574	.0223227	0.02	0.987	-.0433942	.0441091
C26						

	_cons	-.0296129	.0180847	-1.64	0.102	-.0650583	.0058324
C27							
	_cons	-.2415565	.3077055	-0.79	0.432	-.8446483	.3615353
d1							
	_cons	-.0787658	.0090108	-8.74	0.000	-.0964266	-.0611049
b1							
	_cons	.4326473	.0227039	19.06	0.000	.3881485	.4771461
A11							
	one	2.275154	.7979256	2.85	0.004	.711248	3.839059
	zi7	.2517687	.8487665	0.30	0.767	-1.411783	1.915321
A22							
	one	-1.487813	.7095467	-2.10	0.036	-2.878499	-.0971266
	zi7	1.244006	.5841811	2.13	0.033	.0990324	2.38898

Instruments for equation **eq1_FE**: __000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T o.__00005B o.__00005M __000005 __0000E6 __0000E7 __0000E8 __0000E9 __0000EA __0000EB __0000EC __0000ED __0000AJ __0000AK __0000AM __0000AN __0000AS __0000AT __0000AV __0000AW __0000B2 __0000B3 __0000B5 __0000B6 __0000BB __0000BC __0000BE __0000BF __0000BL __0000BM __0000BO __0000BP __0000BU __0000BV __0000BX __0000BY __0000C4 __0000C5 __0000C7 __0000C8 __0000CD __0000CE __0000CG __0000CH __0000CN __0000CO __0000CQ __0000CR __0000CW __0000CX __0000CZ __0000D0 __0000D6 o.__0000D7 __0000D9 o.__0000DA __0000DF o.__0000DG __0000DI o.__0000DJ o.__0000DP o.__0000DQ o.__0000DS o.__0000DT o.__0000DY o.__0000DZ o.__0000E1 o.__0000E2 __0000AG __0000AP __0000AZ __0000B8 __0000BI __0000BR __0000C1 __0000CA __0000CK __0000CT __0000D3 __0000DC __0000DM __0000DV _cons

```
1241
1242        esttab using "${filename_FE}.tex", se replace //added this
(file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh_drop3_all_ex
> actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex not found)
(output written to diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh
> _drop3_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex
> )
1243
1244        estat overid

    Test of overidentifying restriction:

    Hansen's J chi2(61) = 752.796 (p = 0.0000)
1245        di "$S_DATE $S_TIME"
   8 Dec 2023 13:54:17
1246        unique hhidi
  Number of unique values of hhidi is  20233
  Number of records is  84052
1247        local N_hh = r(sum)
1248        estadd scalar N_hh = r(sum)

  added scalar:
              e(N_hh) =  20233
1249        unique group_round
  Number of unique values of group_round is  4446
  Number of records is  84052
```

```
1250        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  4446
1251        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  4.5508322
1252        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh_drop3_
  > all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster not f
  > ound)
  file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh_drop3_all_exa
  > ctInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1253
1254        if simple_model==1 {
1255                local RE_acz_drv ""
1256                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1257                noi di "`RE_acz_drv'"
1258
1259                cap g junk=.
1260                replace junk=p1*qikbar1
1261                su junk $GMM_weight
1262                local mean_y = r(mean)
1263                su xi $GMM_weight
1264                local mean_x = r(mean)
1265
1266                estimates store gmm_est
1267                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1268                estimates save "${filename_FE}_dF", replace
1269                estimates restore gmm_est
1270        }
1271}

1272
1273
  end of do-file
1274}

1275else {
1276        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1277        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1278
1279        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
```

```
1280
1281        drop if size_group_round<min_group_size
1282        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1283
1284        do "$CODE/estim program.do"
1285}

1286
1287
1288log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_modification.smcl
   log type:  smcl
  closed on:   8 Dec 2023, 13:54:17
-------------------------------------------------------------------------------
```

B.2 Columns 3 and 4

Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_modification.smcl
  log type:  smcl
 opened on:   8 Dec 2023, 14:28:17

1 .
2 . // wrapper for analysis
3 .
4 . global ROOT "/Users/Sam/Dropbox/School/Krishna/Indian/replication"

5 . global ROOT "/Volumes/APPLE HDD/Dropbox/School/Krishna/Indian/replication"

6 . global ROOT "C:\Users\pousim2\Downloads\rep\replication"

7 .
8 . global CODE "$ROOT/cmod"

9 . global OUTPUT "$ROOT/output"

10. global DATA     "$ROOT/data"

11.
12. set more off

13. cd "$OUTPUT"
C:\Users\pousim2\Downloads\rep\replication\output

14.
15. global short_zlist_val=1

16.
17. // restrict sample to only scheduled castes
18. global only_scheduled=0

19. local desc_samp ""

20. if $only_scheduled==1 local desc_samp "_onlySched"

21. if $only_scheduled==2 local desc_samp "_onlySchedCrossGrp"

22.
23. // type of expenditure
24. global use_vis=0

25. global use_vislux=0

26. global use_norm=1

27. local catname ""

28. if $use_vis==1 local catname "_vis"

29. if $use_vislux==1 local catname "_vislux"

30. if $use_norm==1 local catname "_norm"

31.
32. // random groupings
33. global rand_grp=0
```

```
34. local grp_type ""

35. if $rand_grp==1 local grp_type "_Random"

36.
37. // make summary stats table
38. local make_sumstats = 0

39.
40. // measure expenditure in per capita terms
41. global meas_pc = 0

42. local typeExp ""

43. if $meas_pc==1 local typeExp "_ExpPC"

44.
45. // number of goods
46. scalar J=2

47. local J = J

48.
49. // group def options
50. //local group_def "dist"
51. //local group_def "fsuXseg"
52. //local group_def "fsuXsegXreligXsched"
53. local group_def "fsuXsegXreligXschedXehigh"

54.
55. // comparision group
56. local groupComp_def ""

57. local desc_compGrp ""

58. if "`groupComp_def'"!="" local desc_compGrp "_comp`groupComp_def'"

59.
60. // load data
61. include "$CODE/prep demand system data.do"

62. //clear all
63. pause on

64. set more off

65. set matsize 11000
  set matsize ignored.
      Matrix sizes are no longer limited by c(matsize) in modern Statas.  Matrix sizes
      are now limited by edition of Stata.  See limits for more details.

66.
67. scalar short_zlist=$short_zlist_val

68.
69. cap prog drop bysmeanw

70. prog def bysmeanw
    1.         syntax varname, weight(varname) by(varlist) [rename(name)]
    2.
```

```
71.         if "`rename'" == "" local rename = "`varlist'bar"
    3.
72.         tempvar totweight totsum varXweight
    4.
73.         bys `by': egen double `totweight' = total((`varlist'<.)*`weight')
    5.          bys `by': egen double `totsum'    = total( `varlist'*`weight')
    6.          g `rename' = `totsum'/`totweight'
    7.
74. end

75.
76. // load prices
77. use "$DATA/laspeyres_state_core.dta", clear

78. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core", clear

79. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3 4)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations             1,118   ->   280
  Number of variables                    5   ->   7
  j variable (4 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2 ... l
  > aspeyres_state4
  ------------------------------------------------------------------------------

80. forval i=1/4 {
    2.          rename laspeyres_state`i' p`i'
    3. }

81. save "$DATA/laspeyres_state_reshape.dta", replace
  file C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_reshape.dta
      saved

82.
83. use "$DATA/laspeyres_state_core_2cats.dta", clear

84. if $use_vis==1 use "$DATA/laspeyres_state_core_2catsVis", clear

85. if $use_vislux==1 use "$DATA/laspeyres_state_core_2catsVislux", clear

86. if $use_norm==1 use "$DATA/laspeyres_normgoods_state_core_2cats", clear

87. reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2)

  Data                                Long   ->   Wide
  ------------------------------------------------------------------------------
  Number of observations              560   ->   280
  Number of variables                    5   ->   5
  j variable (2 values)          expend_type   ->   (dropped)
  xij variables:
                             laspeyres_state   ->   laspeyres_state1 laspeyres_state2
  ------------------------------------------------------------------------------

88. forval i=1/2 {
    2.          rename laspeyres_state`i' P`i'
    3. }
```

```
89. save "$DATA/laspeyres_state_2cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_2cats_reshape.dt
      > a saved

90.
91. if $use_norm==1 {
92.         use "$DATA/laspeyres_normgoods_state_core_3cats.dta", clear
93.         reshape wide laspeyres_state, i(round state urban) j(expend_type)
  (j = 1 2 3)

  Data                                Long   ->   Wide
  -----------------------------------------------------------------------------
  Number of observations               840   ->   280
  Number of variables                    5   ->   6
  j variable (3 values)       expend_type   ->   (dropped)
  xij variables:
                         laspeyres_state   ->   laspeyres_state1 laspeyres_state2 laspe
  > yres_state3
  -----------------------------------------------------------------------------

94.         forval i=1/3 {
    2.                 rename laspeyres_state`i' P`i'
    3.         }
95.         save "$DATA/laspeyres_state_3cats_reshape.dta", replace
  file
      C:\Users\pousim2\Downloads\rep\replication/data/laspeyres_state_3cats_reshape.dt
      > a saved
96. }

97.
98. // read in consumption data
99. use "$DATA/years1999_2004.dta", clear

100 *use "$DATA/years1999_2004_sm.dta", clear
101 g hhid = _n

102
103 // update expenditure if normal goods
104 if $use_norm==1 {
105         egen update_exp = rowtotal(food fuel clothing other_nondur)
106         replace expenditure = update_exp
  (197,188 real changes made)
107 }

108
109 // possible groups
110 egen group_district=group(state district)

111 egen group_round_district=group(state district round)

112 egen group_round_districtUrb=group(state district round urban)

113
114 merge m:1 round state urban using "$DATA/laspeyres_state_reshape.dta"
  (variable urban was byte, now float to accommodate using data's values)

      Result                      Number of obs
      -----------------------------------------
      Not matched                             0
      Matched                           234,590  (_merge==3)
      -----------------------------------------
```

```
115 drop _merge

116 if `J'!=3 merge m:1 round state urban using "$DATA/laspeyres_state_2cats_reshape.dta
  > "
```

Result	Number of obs	
Not matched	**0**	
Matched	**234,590**	(_merge==3)

```
117 else                    merge m:1 round state urban using "$DATA/laspeyres_state_3ca
  > ts_reshape.dta"

118 drop _merge

119
120 //make weights comparable across rounds
121 bys round: egen meanweight=mean(weight)

122 bys round: replace weight=weight/meanweight
  (234590 real changes made)

123 bys round: summ weight
```

-> round = 59

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**40,964**	**1**	**1.404501**	**.0004533**	**108.1072**

-> round = 60

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**29,621**	**1**	**1.068088**	**.0007265**	**18.32417**

-> round = 61

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**124,591**	**1**	**1.193275**	**.0008336**	**32.6936**

-> round = 62

Variable	Obs	Mean	Std. dev.	Min	Max
weight	**39,414**	**1**	**2.16061**	**.0000482**	**66.4409**

```
124
125 // clean hh chars
126 replace religion=1 if religion==.
  (20 real changes made)

127 replace socgroup=9 if socgroup==.
  (68 real changes made)
```

```
128 g religcaste=(religion==1)*(socgroup<3)+(religion==1)*(socgroup==9)*2+(religion>1)*3

129 tab religcaste round, col
```

Key
frequency *column percentage*

religcaste	round 59	60	61	62	Total
1	**7,679** **18.75**	**6,046** **20.41**	**26,561** **21.32**	**7,027** **17.83**	**47,313** **20.17**
2	**24,549** **59.93**	**17,492** **59.05**	**68,656** **55.11**	**23,152** **58.74**	**133,849** **57.06**
3	**8,736** **21.33**	**6,083** **20.54**	**29,374** **23.58**	**9,235** **23.43**	**53,428** **22.78**
Total	**40,964** **100.00**	**29,621** **100.00**	**124,591** **100.00**	**39,414** **100.00**	**234,590** **100.00**

```
130 g scheduled = socgroup==1 | socgroup==2

131 g hindu = religion==1

132 g muslim = religion==2

133 egen hinduXscheduled = group(religion hindu)

134 g schednh = scheduled==1 & hindu==0

135 g schedh  = scheduled==1 & hindu==1

136 g nschedh = scheduled==0 & hindu==1

137 g nschednh = scheduled==0 & hindu==0

138 g nonHindu=religion>1

139
140 // simplify education categories
141 recode educ (1 2=1 "Illiterate and Barely Literate") (3 4 5 =2 "Less than HS") (6 7
  > 8=3 "HS or more"), gen(educsimp)
  (165,087 differences between educ and educsimp)

142
143 * cutoff_group_size=10, no expenditure trim, cut pairwise comparisons at 0.10 yields
  >  passable first stage; scale_factor does not affect GMM
144 local cutoff_group_size=10

145 scalar cutoff_expenditure_percentile=99

146 scalar scale_factor=1

147
```

```
148 // generate variables
149 g low_educ = z9==0 & z10==0

150
151 // drop top and bottom percentiles of expenditure in each round/state
152 bys round state: egen expenditure_cutoff=pctile(expenditure),p(99)

153 drop if expenditure>expenditure_cutoff
  (2,279 observations deleted)

154 drop expenditure_cutoff

155 bys round state: egen expenditure_cutoff=pctile(expenditure),p(1)

156 drop if expenditure<expenditure_cutoff
  (2,251 observations deleted)

157
158 // keep urban houselds smaller than 13 and with head older than 20
159 keep if urban==1
  (141,042 observations deleted)

160 drop if hhsize>12
  (656 observations deleted)

161 gen married=marstat==2

162 drop if age<20
  (1,182 observations deleted)

163 g hhsizem1=hhsize-1

164
165 // winsorize weights
166 su weight,d

                            weight
-------------------------------------------------------------
      Percentiles      Smallest
 1%      .004815       .0000482
 5%     .0180562       .0000482
10%     .0337049       .0000482       Obs              87,180
25%     .1023208       .0001445       Sum of wgt.      87,180

50%     .3493874                      Mean            .6966798
                       Largest        Std. dev.        1.03883
75%     .9093118       25.2859
90%     1.718852       26.1213        Variance        1.079168
95%     2.451768       26.1213        Skewness        5.702664
99%      4.56912       29.15594       Kurtosis        76.01125

167 replace weight = r(p95) if weight>r(p95)
  (4,358 real changes made)

168 replace weight = r(p5) if weight<r(p5)
  (4,326 real changes made)

169
170 g lux=ilux+vlux
```

```
171 g necc=inec+vnec

172
173 // education variables
174 drop if zone==.
  (799 observations deleted)

175 tab zone, gen(zone_)
```

Zone	Freq.	Percent	Cum.
1	**14,469**	**16.75**	**16.75**
2	**8,372**	**9.69**	**26.44**
3	**15,493**	**17.94**	**44.38**
4	**11,835**	**13.70**	**58.08**
5	**14,251**	**16.50**	**74.58**
6	**21,961**	**25.42**	**100.00**
Total	**86,381**	**100.00**	

```
176 g educmed=educsimp==2

177 g educhigh=educsimp==3

178
179 // scale age (helps numerical performance)
180 drop if age==.
  (1 observation deleted)

181 replace age=age/40
  (86,380 real changes made)

182
183 // rename z's so that they can be used in each file
184 forval i=1/24 {
    2.          rename z`i' demog`i'
    3. }

185
186 // landowner dummy
187 g owns_land = landowned>.005 & landowned<.

188
189 egen group_inst=group(state district urban)

190
191 if "`group_def'"=="dist"
  >                     g group=group_round_districtUrb

192 if "`group_def'"=="fsuXseg"
  >       g group=geogroup_seg

193 if "`group_def'"=="fsuXsegXreligXsched"                egen group = group(geogroup_
  > seg religion scheduled)

194 if "`group_def'"=="fsuXsegXreligXschedXownsland"       egen group = group(geogroup_
  > seg religion scheduled owns_land)

195 if "`group_def'"=="fsuXsegXreligXschedXehigh"                  egen group = group(g
  > eogroup_seg religion scheduled educhigh)
  (43,637 missing values generated)
```

```
196
197 if $rand_grp==1 {
198         sort round
199         sort round state district
200         g shuffle = _n
201         g rand = runiform()
202         sort round rand
203         sort round state district rand
204         g group_new = group[shuffle]
205         replace group = group_new
206         drop group_new shuffle rand
207 }

208
209 egen group_round=group(group round) // because group is different in each round, sam
  > e as group
  (43,637 missing values generated)

210 egen district_round=group(state district round urban)

211
212 // make comparision group, so can restrict to sample in other regs
213 if "`groupComp_def'"=="dist"
  >                     g groupComp=group_round_districtUrb

214 if "`groupComp_def'"=="fsuXseg"
  >       g groupComp=geogroup_seg

215 if "`groupComp_def'"=="fsuXsegXreligXsched"                     egen groupComp = gro
  > up(geogroup_seg religion scheduled)

216 if "`groupComp_def'"!="" egen groupComp_round = group(groupComp round)

217
218 if short_zlist==1 {
219         global T=7
220
221         g z1=hhsizem1/10
222         g z2=age/3
223         g z3=married
224         g z4=.
  (86,380 missing values generated)
225         g land=exp(demog23)
  (12,776 missing values generated)
226         replace z4=0 if land<=0.005
  (29,502 real changes made)
227         replace z4=ln(land+1) if land>0.005
  (44,102 real changes made)
228         g z5=demog24
  (29 missing values generated)
229         g z6=educmed
230         g z7=educhigh
231         order land, after(z7)
232
233         global zlist "z1 z2 z3 z4 z5 z6 z7"
234         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
235         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
236 }

237
```

```
238 if short_zlist==2 {
239         global T=10
240
241         g z1=hhsizem1/10
242         g z2=age/3
243         g z3=married
244         g z4=.
245         g land=exp(demog23)
246         replace z4=0 if land<=0.005
247         replace z4=ln(land+1) if land>0.005
248         g z5=demog24
249         g z6=educmed
250         g z7=educhigh
251         g z8 = scheduled==1 & hindu==0
252         g z9 = scheduled==0 & hindu==0
253         g z10 = scheduled==0 & hindu==1
254
255         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
256         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10"
257         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8 xz9 xz10"
258 }

259
260 if short_zlist==3 {
261         global T=8
262
263         g z1=hhsizem1/10
264         g z2=age/3
265         g z3=married
266         g z4=.
267         g land=exp(demog23)
268         replace z4=0 if land<=0.005
269         replace z4=ln(land+1) if land>0.005
270         g z5=demog24
271         g z6=educmed
272         g z7=educhigh
273         g z8 = owns_land
274
275         global zlist "z1 z2 z3 z4 z5 z6 z7 z8"
276         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7 z8"
277         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7 xz8"
278 }

279
280 if short_zlist==4 {
281         global T=7
282
283         g z1=hhsizem1/10
284         g z2=age/3
285         g z3=married
286         g z4=.
287         g land=exp(demog23)
288         replace z4=0 if land<=0.005
289         replace z4=ln(land+1) if land>0.005
290         g z5=demog24
291         g z6=educmed
292         g z7=educhigh
293
294         global zlist "z1 z2 z3 z4 z5 z6 z7"
295         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
```

```
296         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
297 }
298
299 if short_zlist==5 {
300         global T=6
301
302         g z1=age/3
303         g z2=married
304         g z3=.
305         g land=exp(demog23)
306         replace z3=0 if land<=0.005
307         replace z3=ln(land+1) if land>0.005
308         g z4=demog24
309         g z5=educmed
310         g z6=educhigh
311         order land, after(z6)
312
313         global zlist "z1 z2 z3 z4 z5 z6"
314         global zlist_indiv "z1 z2 z3 z4 z5 z6"
315         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6"
316 }
317
318 if short_zlist==6 {
319         global T=5
320
321         g z1=hhsizem1/10
322         g z2=p1
323         g z3=p2
324         g z4=p3
325         g z5=p4
326         g land=exp(demog23)
327
328         global zlist "z1 z2 z3 z4 z5"
329         global zlist_indiv "z1"
330         global xzlist_indiv "xz1"
331
332         drop if land==. | demog24==.
333 }
334
335 if short_zlist==7 {
336         global T=11
337
338         g z1=hhsizem1/10
339         g z2=age/3
340         g z3=married
341         g z4=.
342         g land=exp(demog23)
343         replace z4=0 if land<=0.005
344         replace z4=ln(land+1) if land>0.005
345         g z5=demog24
346         g z6=educmed
347         g z7=educhigh
348         g z8=p1
349         g z9=p2
350         g z10=p3
351         g z11=p4
352
```

```
353         global zlist "z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11"
354         global zlist_indiv "z1 z2 z3 z4 z5 z6 z7"
355         global xzlist_indiv "xz1 xz2 xz3 xz4 xz5 xz6 xz7"
356 }

357
358
359
360 // drop missings and calculate size of grp
361 forv j = 1(1)$T {
    2.          drop if z`j'==.
    3. }
  (0 observations deleted)
  (0 observations deleted)
  (0 observations deleted)
  (12,776 observations deleted)
  (28 observations deleted)
  (0 observations deleted)
  (0 observations deleted)

362 bys group_round: egen size_group_round=count(group_round)

363
364 // size of comparision group
365 if "`groupComp_def'"!="" {
366         bys groupComp_round: egen size_groupComp_round=count(groupComp_round)
367 }

368
369 // turn things int oper capita terms if ption flagged
370 if $meas_pc==1 {
371         foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing o
  > ther_nondur {
    2.                  replace `var' = `var'/hhsize
    3.          }
372 }

373
374 // normalize expenditure by overall mean
375 summ expenditure [aweight=weight]

    Variable |      Obs      Weight        Mean   Std. dev.       Min        Max
-------------+---------------------------------------------------------------------
 expenditure |  73,576   45441.646    4270.163   2537.534   308.7667   27910.14

376 scalar expenditure_mean=r(mean)

377 foreach var in expenditure vlux ilux vnec inec lux necc food fuel clothing other_non
  > dur {
    2.          g `var'_norm=`var'/expenditure_mean
    3. }

378
379 if $use_norm==0 {
380         g pq1=vlux_norm
381         g pq2=ilux_norm
382         g pq3=vnec_norm
383         g pq4=inec_norm
384 }
```

```
385 else {
386         g pq1=food_norm
387         g pq2=fuel_norm
388         g pq3=clothing_norm
389         g pq4=other_nondur_norm
390 }

391 g x=expenditure_norm

392 g x2=x*x

393
394 // make q averages
395 qui bysort group_round: egen weight_group_round=sum(weight)

396
397 forval j=1/4 {
    2.                          qui g s`j'=pq`j'/expenditure_norm
    3.                          qui g q`j'=pq`j'/p`j'
    4.                          qui bysmeanw q`j', weight(weight) by(group_round) rename(
  > qbar`j')
    5.                          qui summ q`j' [aweight=weight]
    6.                          scalar qallbar`j'=r(mean)
    7.                          qui summ s`j' [aweight=weight]
    8.                          scalar sallbar`j'=r(mean)
    9. }

398
399 // 2 category consumption
400 g PQ1=pq1+pq2

401 g PQ2=pq3+pq4

402 if $use_vis==1 {
403         drop PQ1 PQ2
404         g PQ1=pq1+pq3
405         g PQ2=pq2+pq4
406 }

407 if $use_vislux==1 {
408         drop PQ1 PQ2
409         g PQ1=pq1
410         g PQ2=pq2+pq3+pq4
411 }

412 if $use_norm==1 {
413         drop PQ1 PQ2
414         g PQ1=pq1
415         g PQ2=pq2+pq3+pq4
416         if `J'==3 {
417                 drop PQ1 PQ2
418                 g PQ1=pq1
419                 g PQ2=pq2
420                 g PQ3=pq3+pq4
421         }
422 }

423
424 forval j=1/2 {
    2.          qui g Q`j'=PQ`j'/P`j'
    3.          qui bysmeanw Q`j', weight(weight) by(group_round) rename(Qbar`j')
    4. }
```

```
425 if `J'==3 {
426         qui g Q`J'=PQ`J'/P`J'
427         qui bysmeanw Q`J', weight(weight) by(group_round) rename(Qbar`J')
428 }

429
430 // only scheduled groups (with some of other group)
431 bys geogroup_seg religion: egen share_sched = mean(scheduled)

432 if $only_scheduled==1 keep if scheduled==1 & share_sched>0 & share_sched<1

433
434 // update group epend wiht other group
435 if $only_scheduled==2 {
436         preserve
437                 keep if scheduled==0
438                 keep geogroup_seg religion Qbar* qbar*
439                 duplicates drop
440                 tempfile update_exp
441                 save `update_exp'
442         restore
443         keep if scheduled==1 & share_sched>0 & share_sched<1
444         merge m:1 geogroup_seg religion using `update_exp', update replace keep(3 4
  > 5) nogen
445 }

446 drop share_sched

447
448 // laysperes_cpi varies by district and round
449 g local_cpi=p1^sallbar1*p2^sallbar2*p3^sallbar3*p4^sallbar4
  (82 missing values generated)

450
451 // make instruments
452 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
453 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
454 // measure all the x-stuff in reals, using aggregate laysperes index.
455 g weight_temp=weight

456 global instlist ""

457 g xreal=x/local_cpi
  (82 missing values generated)

458 g x_temp=x

459 g x2_temp=x2

460 replace x=xreal
  (73,576 real changes made, 82 to missing)

461 replace x2=x*x
  (73,576 real changes made, 82 to missing)

462
463 // make z x interactions
464 foreach var of varlist $zlist {
    2.          g x`var'=x*`var'
    3. }
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
  (82 missing values generated)
```

```
465
466 levelsof round, local(roundlist)
  59 60 61 62

467 foreach var of varlist x x2 $zlist_indiv $xzlist_indiv {
    2.         qui g `var'barlag=.
    3.         foreach rnd of local roundlist {
    4.                         qui replace weight_temp=0 if round==`rnd'
    5.                         qui bysmeanw `var', weight(weight_temp) by(group_inst) re
  > name(`var'bar`rnd')
    6.                         qui replace `var'barlag=`var'bar`rnd' if round==`rnd'
    7.                         qui drop `var'bar`rnd'
    8.                         qui replace weight_temp=weight
    9.         }
   10.         global instlist "$instlist `var'barlag"
   11. }

468
469 g xbar2lag=xbarlag^2
  (4 missing values generated)

470 g rootxbarlag=sqrt(xbarlag)
  (4 missing values generated)

471
472 bysmeanw x, weight(weight) by(group) rename(xbar)

473 bysmeanw x2, weight(weight) by(group) rename(x2bar)

474 g xbar2=xbar^2

475 g rootxbar = sqrt(xbar)

476 foreach var of varlist $zlist_indiv $xzlist_indiv {
    2.         bysmeanw `var', weight(weight) by(group) rename(`var'bar)
    3.         global instlist "$instlist `var'bar"
    4. }

477
478 replace x=x_temp
  (73,576 real changes made)

479 replace x2=x2_temp
  (73,576 real changes made)

480
481 // add x terms to inst lsit
482 global instlist "$instlist xbar2lag rootxbarlag xbar xbar2 x2bar rootxbar"

483
484 // only groups of size 3 or more
485 drop if size_group_round<3
  (53,343 observations deleted)

486 if "`groupComp_def'"!="" drop if size_groupComp_round<3

487 drop size_group_round

488 bys group_round: egen size_group_round=count(group_round)
```

```
489
490 // x x2 are nominal, but xbar, x2bar, xbar2 are real, and are used to predict real q
  > uantities qbarj
491 if `J'==3 local extrakeep "Q3 P3 Qbar3"

492 if $meas_pc==1 local extrakeep "`extrakeep' hhsize"

493 keep hhid q1 q2 q3 q4 Q1 Q2 p1 p2 p3 p4 P1 P2 local_cpi x $zlist qbar1-qbar4 Qbar1 Q
  > bar2 group group_round round state district weight size_group*_round weight_group_ro
  > und $instlist urban land owns_land schednh schedh nschedh nschednh scheduled geogrou
  > p hhsize `extrakeep'

494
495 g obs_numi=_n

496
497 forval j=1/4 {
    2.          rename q`j' qi`j'
    3.          capture rename Q`j' Qi`j'
    4. }

498 rename x xi

499
500 forval t=1/$T {
    2.          rename z`t' zi`t'
    3. }

501 rename weight weighti

502 rename land landi

503 rename owns_land owns_landi

504 rename schednh schednhi

505 rename schedh schedhi

506 rename nschedh nschedhi

507 rename nschednh nschednhi

508 g urbani = urban

509 rename hhid hhidi

510 rename scheduled scheduledi

511 rename hhsize hhsizei

512
513 tempfile obs_i

514 save `obs_i'
  file C:\Users\pousim2\AppData\Local\Temp\ST_4630_000001.tmp saved as .dta format

515
516 // calculate sumstats at hh level
517 if `make_sumstats'==1 {
518         count
```

```
519         local sumstats_n_obs = r(N)
520
521         foreach var of varlist xi Qi1 Qi2 P1 P2 zi1-zi$T {
    2.                 qui su `var'
    3.                 //local sumstats_row_`var' "`var' & `=round(`=r(mean)',.01)' & `=
  > round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(max)',.01)' "
522                 local sumstats_row_`var' "`var' & `: di %13.2gc =r(mean)' & `: di %1
  > 3.2gc =r(sd)' & `: di %13.2gc =r(min)' & `: di %13.2gc =r(max)' "
    4.         }
523         local sumstats_row_Qikbar1 "Qikbar1 & & & &"
524         local sumstats_row_Qikbar2 "Qikbar2 & & & &"
525 }

526
527 rename obs_numi obs_numk

528
529 forval j=1/4 {
    2.         rename qi`j' qk`j'
    3.         capture rename Qi`j' Qk`j'
    4. }

530 rename xi xk

531
532 forval t=1/$T {
    2.         rename zi`t' zk`t'
    3. }

533 rename weighti weightk

534 rename landi landk

535 rename owns_landi owns_landk

536 rename urbani urbank

537 rename schednhi schednhik

538 rename schedhi schedhk

539 rename nschedhi nschedhk

540 rename nschednhi nschednhk

541 rename scheduledi scheduledk

542 rename hhsizei hhsizek

543 drop geogroup

544
545 // combine into pairs within groups
546 joinby group_round using "`obs_i'"

547 drop if obs_numi==obs_numk
  (20,233 observations deleted)

548
549 // generate exclusive averages qikbarj, of quantities (not spending)
```

```
550 forval j=1/4 {
    2.         g qikbar`j'=(weight_group_round*qbar`j'-weighti*qi`j'-weightk*qk`j')/(wei
  > ght_group_round-weighti-weightk)
    3.         capture g Qikbar`j'=(weight_group_round*Qbar`j'-weighti*Qi`j'-weightk*Qk`
  > j')/(weight_group_round-weighti-weightk)
    4. }

551
552 // summary stats tables
553 if `make_sumstats'==1 {
554         count
555         local sumstats_n_obs_pair = r(N)
556
557         cap file close sumstat
558                 file open sumstat using "$OUTPUT/consumption_sumstats.tex", write re
  > place
559                 file write sumstat "\begin{table}[htbp]\centering" _n
560                 file write sumstat "\def\sym#1{\ifmmode^{#1}\else\(^{#1}\)\fi}" _n
561                 file write sumstat "\caption{Summary statistics for consumption data
  > }\label{tab:sumstats}" _n
562                 file write sumstat "\begin{tabular}{lcccccccc}" _n
563                 file write sumstat "\toprule" _n
564                 file write sumstat "& \multicolumn{4}{c}{Observations } & \multicolu
  > mn{4}{c}{Pairs }\tabularnewline" _n
565                 file write sumstat "& \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats
  > _n_obs'')} & \multicolumn{4}{c}{ (N=`: di %13.0gc `sumstats_n_obs_pair'')}\tabularne
  > wline" _n
566                 file write sumstat "\cmidrule(lr){2-5}\cmidrule(lr){6-9}  & Mean  &
  > SD  & Min  & Max  & Mean  & SD  & Min  & Max\tabularnewline" _n
567                 file write sumstat "\midrule" _n
568
569                 foreach var of varlist xi Qi1 Qi2 Qikbar1 Qikbar2 P1 P2 zi1-zi$T {
    2.                         qui su `var'
    3.                         //local sumstats_row_`var' "`sumstats_row_`var'' & `=roun
  > d(`=r(mean)',.01)' & `=round(`=r(sd)',.01)' & `=round(`=r(min)',.01)' & `=round(`=r(
  > max)',.01)' "
570                         local sumstats_row_`var' "`sumstats_row_`var'' & `: di %13.2
  > gc =r(mean)' & `: di %13.2gc =r(sd)' &`: di %13.2gc =r(min)' &`: di %13.2gc =r(max
  > )' \\"
    4.                         file write sumstat "`sumstats_row_`var''" _n
    5.                 }
571
572                 file write sumstat "\bottomrule" _n
573                 file write sumstat "\multicolumn{9}{p{.5\hsize}}{\footnotesize Table
  >  reports summary statistics for estimation sample.}\\ " _n
574                 file write sumstat "\end{tabular}" _n
575                 file write sumstat "\end{table}" _n
576                 file close sumstat
577                 BREAK
578 }

579
580 // constant for estimation
581 g one=1

582 egen state_district_round=group(state district urban round)

583 egen state_district=group(state district)

584
```

```
585 // keep only thick round
586 keep if round==61
  (0 observations deleted)

587
588
589
590
591
592
593 // options
594 local output_number="14"

595
596 // dimension of A matrix
597 scalar Afull=0

598 local descA "full"

599 scalar Adiag=1

600 if Adiag==1 local descA "diag"

601 scalar same_spillover=0

602 if same_spillover==1 local descA "same"

603
604 // minimum group size (must be 3 or more)
605 local min_group_size = 5

606 scalar min_group_size=`min_group_size'

607
608 // flag for generic model
609 scalar simple_model=0

610 local simp_name ""

611 if simple_model==1 local simp_name "_simple"

612
613 // interactions with peer effects
614 //global Alist "one"
615 //global Alist "one schednhi schedhi nschednhi"
616 //global Alist "one   zi8"
617 global Alist "one   zi7"

618
619 // turn on or off measurement error correction
620 scalar noMeasError=0

621 local measError_desc ""

622 if noMeasError==1 local measError_desc "_NoMeasError"

623
624 // estimate RE and or FE
625 scalar estimate_RE=1
```

```
626 scalar estimate_FE=1

627
628 // estimation options
629 global w_initial "winit(unadjusted)"

630 if J==3 | J==4 | $use_norm==0 global w_initial "winit(identity)"

631 global wmatrix "wmatrix(robust)"

632 global trace_level ""

633 global tol_level "conv_ptol(1e-3) conv_vtol(1e-3)"

634 global technique ""

635 global maxiter ""

636 global technique_RE ""

637 if $use_norm==0 global technique_RE "technique(dfp 10 nr 2)"

638 global derivatives "quickderivatives"

639 scalar init_FE_from_RE=1

640 global clust_var "state_district_round"

641 local clus_name "_clus_Dist"

642
643
644
645
646
647
648
649
650 // xxzx to keep names same
651 local wmatrix_desc ""

652 local sectorName "_urbOnly"

653 local dropZdesc "_dropPre"

654 local desc_struc_v0 ""

655 local inst_desc "_exactInst"

656 local inst_rootp "_noPPXdInst"

657 local weight_desc ""

658 local RE_sq_inst_desc ""

659 local cnstname ""

660 local actual_pqhat_desc ""

661 local expUpname "_updateExp"
```

```
662
663
664 if $rand_grp==0 {
665         global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_RE"
666         global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_FE"
667
668         global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
669
670         drop if size_group_round<min_group_size
  (21,984 observations deleted)
671         if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
672
673         //do "$CODE/estim program.do"
674         do "$CODE/estim program.do"

675 // This code estimates a QES demand system
676 // j indexes goods j=1,..,J, t indexes demographics t=1,...,T
677 // qbarj are group-average spending on each of J goods (e.g., qbar1, qbar2, qbar3 fo
  > r 3 goods)
678 // i indexes observations, k indexes other observations in the group
679 // zt are T demographic shifters (e.g., z1,z2,z3,z4 for 4 characteristics---the firs
  > t should be a constant)
680 // qij,xi,zit and qkj,xk,zkt are pairs of observations within group
681 //              ensure that all zt are z-scored or otherwise have similar scales
682 // qikbarj are leave-two-out group averages
683 // xbarlag is the average x in the same group in other periods, and xbar2lag is its
  > square; x2barlag is the average x2;
684 //    letting t=1,...,T: ztbarlag is the average zt in other periods; xztbarlag is t
  > he average of x*zt in other periods
685 // note that qi and qk are actually quantities, and not spending
686 // note that x and x2 are nominal, but xbar etc are average real expenditures from o
  > ther periods
687
688 /*******************************************************************************
  >  Proceed in 4 parts:
  >
  >  1/ adjust data
  >  2/ construct moment equations
  >  3/ make instruments
  >  4/ estimate
  > *******************************************************************************/
689
690 set seed 339487731

691
692 local J=J

693 local Jm1=`J'-1

694 local T=$T

695 local Tm1=`T'-1
```

```
696 local Tindiv : word count $zlist_indiv

697
698 /*******************************************************************************
  > Set weights
  > *******************************************************************************/
699
700 tempvar clust_weight num_group_round

701 bys group_round: egen `num_group_round'=count(group_round)

702 g double `clust_weight'=size_group_round/`num_group_round'

703 if $meas_pc==1 replace `clust_weight' = `clust_weight'*hhsizei*hhsizek

704 global GMM_weight "[aweight=`clust_weight']"

705
706 /*******************************************************************************
  >  Update data for J=2, and J=3
  > *******************************************************************************/
707
708 if `J'==2 {
709          replace p1=P1
  (62,068 real changes made)
710          replace p2=P2
  (62,068 real changes made)
711          replace qi1=Qi1
  (62,068 real changes made)
712          replace qk1=Qk1
  (62,068 real changes made)
713          replace qikbar1=Qikbar1
  (62,068 real changes made)
714          replace qi2=Qi2
  (62,068 real changes made)
715          replace qk2=Qk2
  (62,068 real changes made)
716          replace qikbar2=Qikbar2
  (62,068 real changes made)
717
718          foreach var of varlist qi3 qi4 qk3 qk4 p3 p4 qikbar3 qikbar4 {
    2.                   qui replace `var'=.
    3.          }
719 }

720
721 if `J'==3 {
722          forv j = 1(1)3 {
    2.                   replace p`j'=P`j'
    3.                   replace qi`j'=Qi`j'
    4.                   replace qk`j'=Qk`j'
    5.                   replace qikbar`j'=Qikbar`j'
    6.          }
723
724          foreach var of varlist qi4 qk4 p4 qikbar4 {
    2.                   qui replace `var'=.
    3.          }
725 }

726
```

```
727
728 /*******************************************************************************
  >  Make moment equations
  > *******************************************************************************/
729
730 global zilist ""

731 global zklist ""

732 forval t=1/`Tm1' {
    2.         global zilist "$zilist zi`t'"
    3.         global zklist "$zklist zk`t'"
    4. }

733 global zilistTm1 "$zilist"

734 global zklistTm1 "$zklist"

735 if `T'>0 global zilist "$zilist zi`T'"

736 if `T'>0 global zklist "$zklist zk`T'"

737
738 // create b_p and pC
739 // remember b_p doesn't vary within group
740 local b_p "exp( 0"

741 forval j=1/`Jm1' {
    2.         local b_p "`b_p' + {b`j'}*ln(p`j'/p`J')"
    3. }

742 local b_p "`b_p' + ln(p`J') )"

743 if simple_model==1      local b_p "1"

744 di "`b_p'"
  exp( 0 + {b1}*ln(p1/p2) + ln(p2) )

745
746 // pCj is the sum of these two, for use in the RE model
747 // pCij, pCkj are the individually varying parts of Cz*pj, for use in the FE model
748 // pCgj is the part of Cz*pj that does not vary within group, for use in the FE mode
  > l
749 forval j=1/`J' {
    2.         local pC`j' "(p`j'*{C`j':$zilist})"
    3. }

750 local Tindivp1=`Tindiv'+1

751 forval j=1/`J' {
    2.         local pCi`j' "( 0 "
    3.         local pCk`j' "( 0 "
    4.         local pCg`j' "( 0 "
    5.         forval t=1/`Tindiv' {
    6.                 local pCi`j' "`pCi`j'' + p`j'*{C`j'`t'}*zi`t' "
    7.                 local pCk`j' "`pCk`j'' + p`j'*{C`j'`t'}*zk`t' "
    8.         }
    9.         forval t=`Tindivp1'/`T' {
   10.                 local pCg`j' "`pCg`j'' + p`j'*{C`j'`t'}*zi`t' "
   11.         }
   12.         local pCi`j' "`pCi`j'' )"
   13.         local pCk`j' "`pCk`j'' )"
   14.         local pCg`j' "`pCg`j'' )"
   15. }
```

```
752
753 // pC, pCi, pCk, pCg are the sums of the j's computed above, and including the J'th
  > element
754 local pC "( 0"

755 local pCi "( 0"

756 local pCk "( 0"

757 local pCg "( 0"

758 forval j=1/`J' {
    2.          local pC "`pC' + `pC`j''"
    3.          local pCi "`pCi' + `pCi`j''"
    4.          local pCk "`pCk' + `pCk`j''"
    5.          local pCg "`pCg' + `pCg`j''"
    6. }

759 local pC "`pC' )"

760 local pCi "`pCi' )"

761 local pCk "`pCk' )"

762 local pCg "`pCg' )"

763
764 // the RE model needs versions of pC and pCj for 2nd and subsequent occurences in th
  > e GMM statement
765 local pC_2nd "( 0"

766 forval j=1/`J' {
    2.          local pC`j' "(p`j'*{C`j':})"
    3.          local pC_2nd "`pC_2nd' + `pC`j''"
    4. }

767 local pC_2nd "`pC_2nd' )"

768
769 // make price terms:    budget shifter is rp'Drp, where rp is the root-price vector
770 //                                       expenditure shifter is p*diag(D) + (1/2) Sum
  >    s~j rp'Drp
771 // price terms do NOT include their diagonal elements, because these are picked up b
  > y the constant term in z (zT)
772 local rpDrp "( 0"

773 forval j=1/`J' {
    2.          local r=`j'+1
    3.          local m=`j'-1
    4.          forval s=`r'/`J' {
    5.                  if `j'==`s' local rpDrp "`rpDrp' + {D`j'`s'}*sqrt(p`j')*sqrt(p`s'
  > )*2"
    6.          }
    7.          local Drp`j' "( 0"
    8.          forval s=1/`m' {
    9.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`s'`j'}*sqrt(p`s')*sqrt(p`
  > j')"
   10.          }
   11.          forval s=`r'/`J' {
   12.                  if `j'==`s' local Drp`j' "`Drp`j'' + {D`j'`s'}*sqrt(p`j')*sqrt(p`
  > s')"
   13.          }
   14.
```

```
774         local Drp`j' "`Drp`j'' )"
   15.         di "`Drp`j''"
   16. }
  ( 0 )
  ( 0 )

775
776 local rpDrp "`rpDrp' )"

777 di "`rpDrp'"
  ( 0 )

778
779 if simple_model==1      {
780         local pC "0"
781         local pC1 "0"
782         local pCi "0"
783         local pCk "0"
784         local pC_2nd "0"
785         local rpDrp "0"
786         local Drp1 "0"
787 }

788
789 // create Aq
790 // note that ikbar means group-level average excluding observations i and k, for use
  >  in FE and RE
791 // k means use qk as the group quantity (for use in RE only)
792 local qtypelist "ikbar k bar"

793 foreach qtype of local qtypelist {
    2.         if Afull==1 {
    3.                 forval j=1/`J' {
    4.                         local A`qtype'`j' "( 0"
    5.
794                         forval l=1/`J' {
    6.                                 if same_spillover==0            local A_term "{A`
  > j'`l':}"
    7.                                 if same_spillover==1 & `j'==`l' local A_term "{A1
  > 1:}"
    8.                                 if same_spillover==1 & `j'!=`l' local A_term "{A1
  > 2:}"
    9.
795                                 local A`qtype'`j' "`A`qtype'`j'' + `A_term'*p`j'*q`q
  > type'`l'"
   10.                         }
   11.                         local A`qtype'`j' "`A`qtype'`j'' )"
   12.                 }
   13.         }
   14.
796         if Adiag==1 {
   15.                 forval j=1/`J' {
   16.                         if same_spillover==0 local A_term "{A`j'`j':}"
   17.                         if same_spillover==1 local A_term "{A11:}"
   18.
797                         local A`qtype'`j' "(`A_term'*p`j'*q`qtype'`j')"
   19.                 }
   20.         }
   21.
798         local A`qtype' "( 0"
   22.         forval j=1/`J' {
   23.                 local A`qtype' "`A`qtype'' + `A`qtype'`j''"
   24.         }
   25.         local A`qtype' "`A`qtype'' )"
   26. }
```

```
799
800 // Aikbar is A'qbar-minus-ik; Ak is A'qk
801 // x_hat, x_hat_2nd are  x-pCz (full z), used in RE
802 // xi_hat and xk_hat are x minus the individually varying part of p'Czi an p'Czk, us
  > ed in FE
803 local xi_hat            "(xi - `pCi')"

804 local xk_hat            "(xk - `pCk')"

805 local x_hat                    "(xi - `pC' - `rpDrp' )"

806 local x_hat_2nd         "(xi - `pC_2nd' - `rpDrp' )"

807
808 // construct structural vo
809 * let AVA be A'VA, and ensure its diagonals are positive.
810 local v0 "(0 "

811 forval j=1/`J' {
    2.          local v0 "`v0' + p`j'*{AVA`j'`j'}*p`j'"
    3.
812         local jplus1=`j'+1
    4.          if `jplus1'<=`J' {
    5.                  forval k=`jplus1'/`J' {
    6.                          local v0 "`v0' + 2*p`j'*{AVA`j'`k'}*p`k'"
    7.                  }
    8.          }
    9. }

813
814 local v0 "`v0' )"

815
816 // make all equations, then put pieces together
817
818 * RE equations; first equation has first call to C, other equations use {Cj: } form
819 * note "-vj:", this is because we subtract the structural v0 term from E[q]
820 local j=1

821
  >             local eq`j'_RE                  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aikbar
  > '     - 2*`x_hat'*`Aikbar'     + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2
  > nd'-`Aikbar')*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"

822 if noMeasError==1                       local eq`j'_RE                  "(eq`j'_RE:
  > (p`j'*qi`j' - ((`Abar'*`Abar'     - 2*`x_hat'*`Abar'       + `x_hat_2nd'*`x_hat_2nd'
  > ) * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'  )*{b`j'} + `Abar`j''   + `pC`j'' + `Drp`j''
  > - `v0'*{d`j'}/`b_p' ) ) )"

823 if `J'>2 {
824         forval j=2/`Jm1' {
    2.
  >                                 local eq`j'_RE  "(eq`j'_RE: (p`j'*qi`j' - ((`Ak'*`Aik
  > bar'               - 2*`x_hat_2nd'*`Aikbar' + `x_hat_2nd'*`x_hat_2nd') * {d`j'}/`b_p
  > ' + (`x_hat_2nd'-`Aikbar'  )*{b`j'} + `Aikbar`j'' + `pC`j'' + `Drp`j'' - `v0'*{d`j'}
  > /`b_p' ) ) )"
    3.                  if noMeasError==1                                   local eq`j'_RE  "(eq`j'_R
  > E: (p`j'*qi`j' - ((`Abar'*`Abar'                - 2*`x_hat_2nd'*`Abar'   + `x_hat_2nd
  > '*`x_hat_2nd') * {d`j'}/`b_p' + (`x_hat_2nd'-`Abar'    )*{b`j'} + `Abar`j''   + `pC`
  > j'' + `Drp`j'' - `v0'*{d`j'}/`b_p' ) ) )"
    4.          }
```

```
825 }

826
827 *       FE equations
828 forval j=1/`Jm1' {
    2.                                                                              l
  > ocal eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') - (`xi_hat'^2 - `xk_hat'^2) * {
  > d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Aikbar' + `pCg' + `rpDrp')*{d`j'}/
  > `b_p') - (`pCi`j'' - `pCk`j'')                    ))"
    3.         if noMeasError==1 local eq`j'_FE "(eq`j'_FE: ((p`j'*qi`j' - p`j'*qk`j') -
  >  (`xi_hat'^2 - `xk_hat'^2) * {d`j'}/`b_p' - (`xi_hat' - `xk_hat')*({b`j'} - 2*(`Abar
  > '   + `pCg' + `rpDrp')*{d`j'}/`b_p') - (`pCi`j'' - `pCk`j'')                      )
  > )"
    4. }

829
830 // clean up equations, and combine
831 local eqs_RE ""

832 local eqs_FE ""

833 forval j=1/`Jm1' {
    2.         local eq`j'_RE: subinstr local eq`j'_RE "( 0 +" "(", all
    3.         local eq`j'_RE: subinstr local eq`j'_RE "( 0  +" "(", all
    4.         local eq`j'_RE: subinstr local eq`j'_RE "  " " ", all
    5.         local eq`j'_FE: subinstr local eq`j'_FE "( 0 +" "(", all
    6.         local eq`j'_FE: subinstr local eq`j'_FE "( 0  +" "(", all
    7.         local eq`j'_FE: subinstr local eq`j'_FE "  " " ", all
    8.         local eqs_RE "`eqs_RE' `eq`j'_RE'"
    9.         local eqs_FE "`eqs_FE' `eq`j'_FE'"
   10. }

834
835 noi di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:}*p1*qk1) + ({A22:}*p2*qk2) )*( ({A11:}*p1*qikbar1) + ({
  > A22:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) + (p2*{C2:
  > zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2
  > ) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C1:}) + (p2*{C2:})
  > ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*{C1:}) + (p2*{C2:}
  > ) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) ))*{b1} + ({A11:}*p1*qikba
  > r1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 + p2*{AVA22}*p2 )*{d
  > 1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

836 noi di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > }*p1*qikbar1) + ({A22:}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))*{d1}/exp( {b1}*ln
  > (p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p1*{C14}*zi4 +
  >  p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{
  > C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ))        ))
```

```
837
838 * replace the first occurrence of Ajj with the Alist
839 if Afull==0 {
840         forval j=1/`J' {
    2.                  local eqs_RE=subinstr("`eqs_RE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    3.                  local eqs_FE=subinstr("`eqs_FE'","{A`j'`j':}","{A`j'`j':$Alist}",
  > 1)
    4.          }
841 }

842 if Afull==1 {
843         forval j=1/`J' {
    2.                  forval l=1/`J' {
    3.                          local eqs_RE=subinstr("`eqs_RE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    4.                          local eqs_FE=subinstr("`eqs_FE'","{A`j'`l':}","{A`j'`l':$
  > Alist}",1)
    5.                  }
    6.          }
844 }

845
846 if simple_model==1 {
847         // construct z sums and interactions
848         if `T'>0 {
849
850                 local RE_cz_2            "+ {d}*(0"
851                 local RE_acz     ""
852                 local RE_acz1     ""
853                 local RE_acz2     ""
854                 local RE_acz_drv "" /// for calculating the derivative
  >                 local RE_cz_cxz  ""
855                 local RE_czcz    ""
856                 local FE_czi     ""
857                 local FE_czk     ""
858                 local FE_czg     ""
859                 forv i=1/`T' {
    2.                          local RE_cz_2  "`RE_cz_2' + {C`i'}*zi`i'*p1"
    3.                          local RE_acz                            "`RE_acz'      +
  > 2*{d}*{a} *{C`i'}*zi`i'*p1"
    4.                          local RE_acz1                   "`RE_acz2'     + 2*{d}*{a
  > 1}*{C`i'}*zi`i'*p1"
    5.                          local RE_acz2                   "`RE_acz2'     + 2*{d}*{a
  > 2}*{C`i'}*zi`i'*p1"
    6.                          local RE_cz_cxz "`RE_cz_cxz' + {C`i'}*zi`i'*p1 + 2*{d}*{b
  > }*{C`i'}*xi*zi`i'*p1"
    7.
860                         forv j=1/`T' {
    8.                                  local RE_czcz "`RE_czcz' + {d}*{C`i'}*{C`j'}*zi`i
  > '*zi`j'"
    9.                          }
   10.
861                         if `i'<=`T' {
   11.                                  local FE_czi "`FE_czi' + {C`i'}*zi`i'*p1"
   12.                                  local FE_czk "`FE_czk' + {C`i'}*zk`i'*p1"
   13.                          }
   14.                          else {
   15.                                  local FE_czg "`FE_czg' + {C`i'}*zi`i'*p1"
   16.                          }
   17.                 }
```

```
862                 local RE_cz_2 "`RE_cz_2')^2"
863         }
864
865         local eqs_RE "(eq1_RE: (p1*qi1-(({a}*p1*qikbar1*{a}*p1*qk1*{d}+( {a}      `RE
  > _acz' +2*{a}*{b}*{d}*xi)*p1*qikbar1+(xi*{b} `RE_cz_cxz' +xi*xi*{b}*{b}*{d}) `RE_cz_2
  > ' +{v0})))))"
866
867         local xi_hat "({b}*p1*xi `FE_czi')"
868         local xk_hat "({b}*p1*xk `FE_czk')"
869
870         local eqs_FE
  >                           "(eq1_FE: (p1*(qi1-qk1) - (`xi_hat'^2 - `xk_hat'^2)*{d}
  > - (`xi_hat' - `xk_hat')*(1 + 2*{d}*({a}*p1*qikbar1 `FE_czg' ))  ))"
871 }

872
873 di "`eqs_RE'"
   (eq1_RE: (p1*qi1 - ((( ({A11:one   zi7}*p1*qk1) + ({A22:one   zi7}*p2*qk2) )*( ({A11:
  > }*p1*qikbar1) + ({A22:}*p2*qikbar2) )   - 2*(xi - ( (p1*{C1: zi1 zi2 zi3 zi4 zi5 zi6
  >  zi7}) + (p2*{C2: zi1 zi2 zi3 zi4 zi5 zi6 zi7}) ) - ( 0 ) )*( ({A11:}*p1*qikbar1) +
  > ({A22:}*p2*qikbar2) )   + (xi - ( (p1*{C1:}) + (p2*{C2:}) ) - ( 0 ) )*(xi - ( (p1*{C
  > 1:}) + (p2*{C2:}) ) - ( 0 ) )) * {d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) + ((xi - ( (p1*
  > {C1:}) + (p2*{C2:}) ) - ( 0 ) )-( ({A11:}*p1*qikbar1) + ({A22:}*p2*qikbar2) ))*{b1}
  > + ({A11:}*p1*qikbar1) + (p1*{C1:}) + ( 0 ) - (0 + p1*{AVA11}*p1 + 2*p1*{AVA12}*p2 +
  > p2*{AVA22}*p2 )*{d1}/exp( {b1}*ln(p1/p2) + ln(p2) ) ) ) )

874 di "`eqs_FE'"
   (eq1_FE: ((p1*qi1 - p1*qk1) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 +
  >  p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{
  > C22}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7
  > ) ))^2 - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C
  > 15}*zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*z
  > k3 + p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) ))^2) * {d1}/exp( {
  > b1}*ln(p1/p2) + ln(p2) ) - ((xi - ( ( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi3 + p
  > 1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) + ( p2*{C21}*zi1 + p2*{C2
  > 2}*zi2 + p2*{C23}*zi3 + p2*{C24}*zi4 + p2*{C25}*zi5 + p2*{C26}*zi6 + p2*{C27}*zi7 )
  > )) - (xk - ( ( p1*{C11}*zk1 + p1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*
  > zk5 + p1*{C16}*zk6 + p1*{C17}*zk7 ) + ( p2*{C21}*zk1 + p2*{C22}*zk2 + p2*{C23}*zk3 +
  >  p2*{C24}*zk4 + p2*{C25}*zk5 + p2*{C26}*zk6 + p2*{C27}*zk7 ) )))*({b1} - 2*(( ({A11:
  > one   zi7}*p1*qikbar1) + ({A22:one   zi7}*p2*qikbar2) ) + ( ( 0 ) + ( 0 ) ) + ( 0 ))
  > *{d1}/exp( {b1}*ln(p1/p2) + ln(p2) )) - (( p1*{C11}*zi1 + p1*{C12}*zi2 + p1*{C13}*zi
  > 3 + p1*{C14}*zi4 + p1*{C15}*zi5 + p1*{C16}*zi6 + p1*{C17}*zi7 ) - ( p1*{C11}*zk1 + p
  > 1*{C12}*zk2 + p1*{C13}*zk3 + p1*{C14}*zk4 + p1*{C15}*zk5 + p1*{C16}*zk6 + p1*{C17}*z
  > k7 ))         ))

875
876 /*******************************************************************************
  >  construct instruments
  > *******************************************************************************/
877
878 //make squares and differences of x
879 tempvar x2i x2k ximxk x2imx2k xiP

880 g double `xiP'=xi*local_cpi

881 g double `x2i'=xi*xi

882 g double `x2k'=xi*xi

883 g double `ximxk'=xi-xk
```

```
884 g double `x2imx2k'=xi*xi-xk*xk

885
886 g xbarlag_x2barlag = xbarlag*x2barlag

887 g xbarlag2  = xbarlag^2

888 g x2barlag2 = x2barlag^2

889
890 //make interactions with z and zp
891 forval t=1/`T' {
    2.          tempvar xizi`t' xkzk`t' xiPzi`t' zi`t'mzk`t' z2i`t'mz2k`t' xizi`t'mxkzk`t
  > ' zi`t'mzk`t'_xbarlag
    3.          g double `xizi`t''=xi*zi`t'
    4.          g double `xkzk`t''=xk*zk`t'
    5.          g double `xiPzi`t''=xi*local_cpi*zi`t'
    6.          g double `zi`t'mzk`t''=zi`t'-zk`t'
    7.          g double `z2i`t'mz2k`t'' = zi`t'^2-zk`t'^2
    8.          g double `zi`t'mzk`t'_xbarlag' = `zi`t'mzk`t''*xbarlag
    9.          g double `xizi`t'mxkzk`t''=`xizi`t''-`xkzk`t''
   10.          forval j=1/`J' {
   11.                  tempvar zi`t'p`j' zi`t'p`j'_sq xizi`t'p`j' zi`t'mzk`t'p`j'
   12.                  g double `zi`t'p`j''= zi`t'*p`j'
   13.                  g double `zi`t'p`j'_sq' = (zi`t'*p`j')^2
   14.                  g double `xizi`t'p`j''=xi*zi`t'*p`j'
   15.                  g double `zi`t'mzk`t'p`j''=`zi`t'mzk`t''*p`j'
   16.                  forv s=1/`T' {
   17.                          tempvar zi`t'zi`s'p`j'
   18.                          g double `zi`t'zi`s'p`j'' = zi`t'*zi`s'*p`j'*p`j'
   19.                  }
   20.          }
   21. }

892
893 // make group-level instruments, and the instrument lists rg, xrg
894 if noMeasError==0       local rg "xbarlag"

895 if noMeasError==1       local rg "xbar"

896
897 local rg "`rg' x2barlag"

898 local rg "`rg' xbar2lag"

899 local rg "`rg' rootxbarlag"

900
901 local Tindivm1=`Tindiv'-1

902 // lag instruments
903 forval t=1/`Tindiv' {
    2.                  local rg "`rg' z`t'barlag"
    3. }

904
905 local zrg ""

906 forval t=1/`Tindiv' {
    2.          foreach var of varlist `rg' {
    3.                  tempvar zi`t'`var'
    4.                  g double `zi`t'`var''=zi`t'*`var'
    5.                  local zrg "`zrg' `zi`t'`var''"
    6.          }
    7. }
```

```
907
908 local xrg ""

909 local prg ""

910 local xprg ""

911 foreach var of varlist `rg' {
    2.          tempvar xi`var'
    3.          g double `xi`var''=xi*`var'
    4.          local xrg "`xrg' `xi`var''"
    5.
912          forval j=1/`J' {
    6.                  tempvar p`j'`var' xp`j'`var'
    7.                  g double `p`j'`var''=p`j'*`var'
    8.                  g double `xp`j'`var''=xi*p`j'*`var'
    9.                  local prg "`prg' `p`j'`var''"
   10.                  local xprg "`xprg' `xp`j'`var''"
   11.          }
   12. }

913
914 // make individual-level instruments: xi, xi2, pj, xipj, zitpJ, xizitpJ, round, 1
915 // and individual-level equation specific instruments: rpjrpl, ztpj, round*pj
916 //   a longer individual-level list is: xi, xi2, pj, xipj, rpjrps, xirpjrps, zitpj,
  > xzitpj
917 local budget "xi `x2i'"

918
919 local rootprootp ""

920 local xrootprootp ""

921 forval j=1/`J' {
    2.          tempvar xip`j'
    3.          g double `xip`j''=xi*p`j'
    4.          local pinsts "`pinsts' p`j'"
    5.          local xpinsts "`xpinsts' `xip`j''"
    6.          forval s=`j'/`J' {
    7.                  tempvar rp`j'rp`s' xirp`j'rp`s'
    8.                  g double `rp`j'rp`s''=sqrt(p`j')*sqrt(p`s')
    9.                  g double `xirp`j'rp`s''=xi*sqrt(p`j')*sqrt(p`s')
   10.                  local rootprootp "`rootprootp' `rp`j'rp`s'' "
   11.                  local xrootprootp "`xrootprootp' `xirp`j'rp`s''"
   12.          }
   13. }

922
923 // demographics multiplied by prices and budget
924 local zinsts ""

925 local xzinsts ""

926 local xPzinsts ""

927 local pzinsts ""

928 local xpzinsts ""

929 forval t=1/`T' {
    2.          local zinsts "`zinsts' zi`t'"
    3.          local xzinsts "`xzinsts' `xizi`t''"
    4.          local xPzinsts "`xPzinsts' `xiPzi`t''"
    5.          forval j=1/`J' {
    6.                  local pzinsts "`pzinsts' `zi`t'p`j''"
    7.                  local xpzinsts "`xpzinsts' `xizi`t'p`j''"
    8.          }
    9. }
```

```
930
931 // price insts
932 local roundinsts " "

933 forval j=1/`J' {
    2.         forval k=`j'/`J' {
    3.                 tempvar p`j'p`k'
    4.                 g `p`j'p`k''=p`j'*p`k'
    5.                 local roundinsts "`roundinsts' `p`j'p`k''"
    6.         }
    7. }

934
935 // instruments for all equations are ri:  x,x2,p,z,zx,
936 // instruments for each equation are rij: rootpjps,zpj,xzpj,roundpj
937 local qhat_insts ""

938 local qhat_insts "`qhat_insts' `zrg'"

939 local qhat_insts "`qhat_insts' `prg'"

940 local qhat_insts "`qhat_insts' `rg' "

941
942 local r_RE
  >     "`rootprootp' `budget' `pzinsts' "

943
944 local count: word count `r_RE'

945 di "total instruments: `count'"
  total instruments: 19

946
947 local qhat_hats ""

948 local pqhat ""

949 local xpqhat ""

950 local pqhat_sq ""

951 forval j=1/`J' {
    2.         if noMeasError==0 reg qikbar`j' `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    3.         if noMeasError==1 reg qbar`j'   `r_RE' `qhat_insts', vce(cluster group_ro
  > und)
    4.
952         tempvar qikhat`j' xiPqikbar`j' xiPqikhat`j'
    5.         predict `qikhat`j''
    6.         summ `qikhat`j''
    7.
953         // use all prices times all qikhats as instruments, interacted with xi
954         forval s=1/`J' {
    8.                 tempvar  p`s'qikhat`j' xip`s'qikhat`j' p`s'qikhat`j'_sq
    9.                 g double `p`s'qikhat`j''=p`s'*`qikhat`j''
   10.                 g double `xip`s'qikhat`j''=xi*p`s'*`qikhat`j''
   11.                 g double `p`s'qikhat`j'_sq' = `p`s'qikhat`j''^2
   12.
955                 local pqhat "`pqhat' `p`s'qikhat`j''"
   13.                 local xpqhat "`xpqhat' `xip`s'qikhat`j''"
  >
   14.                 local pqhat_sq "`pqhat_sq' `p`s'qikhat`j'_sq'"
   15.         }
   16. }

  Linear regression                               Number of obs     =     62,068
                                                  F(129, 1822)      =      13.41
                                                  Prob > F          =     0.0000
                                                  R-squared         =     0.2969
                                                  Root MSE          =     .11998
```

(Std. err. adjusted for **1,823** clusters in **group_round**)

qikbar1	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	-15.62496	11.14762	-1.40	0.161	-37.48842	6.238498
__00009L	-5.694352	1.769534	-3.22	0.001	-9.164881	-2.223823
__00009O	-.0898593	5.69898	-0.02	0.987	-11.26708	11.08736
xi	.1093644	.0120696	9.06	0.000	.0856926	.1330361
__000002	-.0175332	.0040602	-4.32	0.000	-.0254964	-.0095701
__00000E	.1646033	.1218901	1.35	0.177	-.0744557	.4036622
__00000P	.0375777	.0708215	0.53	0.596	-.1013222	.1764777
__000017	.0372027	.2323608	0.16	0.873	-.4185187	.4929242
__00001I	.0122255	.1339738	0.09	0.927	-.2505329	.2749839
__000020	.01725	.0685531	0.25	0.801	-.117201	.151701
__00002B	-.086604	.0341291	-2.54	0.011	-.1535402	-.0196678
__00002T	-.037764	.0950725	-0.40	0.691	-.2242266	.1486986
__000034	.0229813	.0678959	0.34	0.735	-.1101808	.1561433
__00003M	-.0345163	.0751636	-0.46	0.646	-.1819322	.1128997
__00003X	.0643228	.0458318	1.40	0.161	-.0255657	.1542112
__00004F	-.064432	.0606871	-1.06	0.289	-.1834555	.0545916
__00004Q	.0727465	.0314796	2.31	0.021	.0110067	.1344863
__000058	.0179734	.1558395	0.12	0.908	-.2876695	.3236163
__00005J	-.0064204	.099763	-0.06	0.949	-.2020822	.1892414
__00005U	.6277288	.5194896	1.21	0.227	-.391129	1.646587
__00005V	-.0427076	.0944871	-0.45	0.651	-.2280221	.1426068
__00005W	-.1049472	.2133224	-0.49	0.623	-.5233292	.3134349
__00005X	-.6371748	.5495603	-1.16	0.246	-1.715009	.4406597
__00005Y	.1767534	.1362418	1.30	0.195	-.0904532	.4439601
__00005Z	-.0006113	.2148129	-0.00	0.998	-.4219168	.4206942
__000060	-.1070656	.0934965	-1.15	0.252	-.2904371	.076306
__000061	-.0696882	.0628823	-1.11	0.268	-.1930173	.0536408
__000062	-.0030212	.0435354	-0.07	0.945	-.0884056	.0823633
__000063	-.0589478	.0604646	-0.97	0.330	-.1775349	.0596393
__000064	-.0753106	.0757835	-0.99	0.320	-.2239424	.0733211
__000065	.0044509	1.190951	0.00	0.997	-2.331323	2.340224
__000066	.089907	.1632888	0.55	0.582	-.2303459	.4101599
__000067	-.1599277	.449129	-0.36	0.722	-1.04079	.7209342
__000068	.5222414	1.182984	0.44	0.659	-1.797906	2.842388
__000069	-.4110476	.240604	-1.71	0.088	-.8829363	.0608411
__00006A	-.193893	.4320549	-0.45	0.654	-1.041268	.6534821
__00006B	-.0501954	.1644547	-0.31	0.760	-.372735	.2723443
__00006C	-.0984359	.1427691	-0.69	0.491	-.3784442	.1815724
__00006D	.0356043	.0784334	0.45	0.650	-.1182245	.1894332
__00006E	-.3495454	.1294155	-2.70	0.007	-.6033638	-.0957271
__00006F	-.2836118	.1538794	-1.84	0.065	-.5854103	.0181867
__00006G	-.1046584	.2599282	-0.40	0.687	-.6144469	.4051301
__00006H	.0419914	.0448239	0.94	0.349	-.0459203	.1299031
__00006I	-.0324298	.1000843	-0.32	0.746	-.2287219	.1638622
__00006J	.1160089	.2867523	0.40	0.686	-.4463889	.6784067
__00006K	.0828456	.0737265	1.12	0.261	-.0617517	.2274429
__00006L	.1138808	.1083426	1.05	0.293	-.098608	.3263695
__00006M	-.0130985	.0525702	-0.25	0.803	-.1162027	.0900057
__00006N	.0423571	.0305975	1.38	0.166	-.0176528	.1023669
__00006O	-.0161864	.0198042	-0.82	0.414	-.0550277	.0226549
__00006P	-.0431565	.0321903	-1.34	0.180	-.1062903	.0199773
__00006Q	-.0071925	.0381333	-0.19	0.850	-.0819821	.0675971
__00006R	-.9883727	.6654448	-1.49	0.138	-2.293488	.3167421
__00006S	-.0120063	.0792085	-0.15	0.880	-.1673553	.1433428
__00006T	.3877656	.2504471	1.55	0.122	-.1034279	.8789592
__00006U	.7525377	.5931724	1.27	0.205	-.4108316	1.915907
__00006V	-.1929558	.1001111	-1.93	0.054	-.3893004	.0033888
__00006W	-.1209397	.1883086	-0.64	0.521	-.4902631	.2483837
__00006X	.0237223	.0751011	0.32	0.752	-.1235711	.1710156
__00006Y	.0671592	.0461109	1.46	0.145	-.0232767	.157595
__00006Z	-.0773876	.0325781	-2.38	0.018	-.1412819	-.0134932
__000070	.0220747	.0553785	0.40	0.690	-.0865373	.1306867
__000071	-.1444812	.0586287	-2.46	0.014	-.2594677	-.0294947
__000072	-.3514064	.3762777	-0.93	0.350	-1.089387	.3865745
__000073	-.0559993	.0586072	-0.96	0.339	-.1709437	.0589451
__000074	.2068742	.1450457	1.43	0.154	-.077599	.4913474
__000075	.2547829	.384638	0.66	0.508	-.4995949	1.009161

__000076	-.1604921	.0829018	-1.94	0.053	-.3230846	.0021004
__000077	-.0580174	.1319028	-0.44	0.660	-.3167141	.2006792
__000078	.0095738	.0565775	0.17	0.866	-.1013898	.1205375
__000079	.0245068	.0440584	0.56	0.578	-.0619035	.1109171
__00007A	.0193851	.0261555	0.74	0.459	-.0319128	.0706829
__00007B	.0136053	.0407108	0.33	0.738	-.0662395	.0934501
__00007C	-.0676001	.0466253	-1.45	0.147	-.1590448	.0238446
__00007D	.2544028	.2605061	0.98	0.329	-.2565192	.7653248
__00007E	-.0223966	.0413804	-0.54	0.588	-.1035546	.0587614
__00007F	-.0905673	.0970562	-0.93	0.351	-.2809205	.0997858
__00007G	-.0379067	.2888421	-0.13	0.896	-.6044032	.5285898
__00007H	-.1145857	.0677104	-1.69	0.091	-.2473839	.0182125
__00007I	-.0926646	.1099174	-0.84	0.399	-.308242	.1229129
__00007J	-.0271767	.043412	-0.63	0.531	-.1123192	.0579659
__00007K	-.0439803	.0348603	-1.26	0.207	-.1123507	.0243901
__00007L	.0072253	.0199205	0.36	0.717	-.0318442	.0462947
__00007M	-.0272536	.0355777	-0.77	0.444	-.0970311	.0425238
__00007N	.0374114	.0390538	0.96	0.338	-.0391835	.1140064
__00007O	-.7134478	.6540978	-1.09	0.276	-1.996308	.5694125
__00007P	.1115341	.1046883	1.07	0.287	-.0937877	.3168559
__00007Q	.0385847	.2439318	0.16	0.874	-.4398307	.5170002
__00007R	.7256149	.7635875	0.95	0.342	-.771984	2.223214
__00007S	-.3499064	.1922601	-1.82	0.069	-.7269798	.027167
__00007T	-.4470888	.3213183	-1.39	0.164	-1.07728	.1831022
__00007U	.280947	.1338225	2.10	0.036	.0184854	.5434086
__00007V	.0875172	.0949657	0.92	0.357	-.0987359	.2737703
__00007W	.0156971	.0508049	0.31	0.757	-.0839449	.1153391
__00007X	-.166674	.1061081	-1.57	0.116	-.3747804	.0414323
__00007Y	-.0432058	.1114424	-0.39	0.698	-.2617742	.1753625
__000080	-42.20104	26.11418	-1.62	0.106	-93.41791	9.015837
__000082	-.8073647	12.88608	-0.06	0.950	-26.08041	24.46568
__000085	-.855311	.5043269	-1.70	0.090	-1.844431	.1338086
__000087	.6769865	.3477237	1.95	0.052	-.0049924	1.358965
__00008A	8.795803	5.055058	1.74	0.082	-1.118516	18.71012
__00008C	-1.920748	2.414343	-0.80	0.426	-6.655919	2.814423
__00008F	52.96193	32.20131	1.64	0.100	-10.19344	116.1173
__00008H	5.439382	16.37594	0.33	0.740	-26.67821	37.55698
__00008K	-.5121017	.6859389	-0.75	0.455	-1.857411	.8332075
__00008M	-1.350068	.6656467	-2.03	0.043	-2.655578	-.0445569
__00008P	1.49516	1.171106	1.28	0.202	-.8016912	3.792012
__00008R	-1.752059	.8480895	-2.07	0.039	-3.415389	-.0887288
__00008U	-.2547631	.4475336	-0.57	0.569	-1.132496	.6229696
__00008W	.6693217	.3612041	1.85	0.064	-.039096	1.377739
__00008Z	.5319931	.3008837	1.77	0.077	-.0581202	1.122106
__000091	-.5783491	.1990989	-2.90	0.004	-.9688351	-.1878631
__000094	.2125919	.2274438	0.93	0.350	-.2334861	.6586698
__000096	.135296	.1978459	0.68	0.494	-.2527326	.5233246
__000099	-.5910504	.342127	-1.73	0.084	-1.262053	.0799518
__00009B	.3031933	.2693564	1.13	0.260	-.2250866	.8314731
__00009E	-.3835706	.4129312	-0.93	0.353	-1.193439	.4262977
__00009G	-.3490025	.2962347	-1.18	0.239	-.9299977	.2319927
xbarlag	43.92494	32.76699	1.34	0.180	-20.33988	108.1898
x2barlag	-.1403469	.5658573	-0.25	0.804	-1.250144	.9694503
xbar2lag	-6.29365	6.445831	-0.98	0.329	-18.93564	6.348344
rootxbarlag	-60.5426	40.11608	-1.51	0.131	-139.2209	18.13575
z1barlag	2.165194	1.120341	1.93	0.053	-.0320931	4.362482
z2barlag	.2825746	1.498763	0.19	0.850	-2.6569	3.222049
z3barlag	-.5049669	.6465696	-0.78	0.435	-1.773062	.7631286
z4barlag	.2074032	.3587745	0.58	0.563	-.4962492	.9110557
z5barlag	-.3997773	.2419945	-1.65	0.099	-.874393	.0748384
z6barlag	.3778619	.4593329	0.82	0.411	-.5230124	1.278736
z7barlag	.8311257	.5096431	1.63	0.103	-.1684205	1.830672
_cons	22.67146	13.64507	1.66	0.097	-4.090169	49.43308

(option **xb** assumed; fitted values)

Variable	Obs	Mean	Std. dev.	Min	Max
__00009T	62,068	.4208295	.0778873	.1915261	1.077513

Linear regression

Number of obs	**62,068**
F(129, 1822)	**23.88**
Prob > F	**0.0000**
R-squared	**0.4784**
Root MSE	**.15952**

(Std. err. adjusted for **1,823** clusters in **group_round**)

qikbar2	Coefficient	Robust std. err.	t	P>|t|	[95% conf.	interval]
__00009J	17.08161	11.6734	1.46	0.144	-5.813048	39.97626
__00009L	-1.55756	2.462453	-0.63	0.527	-6.387087	3.271968
__00009O	3.097797	7.122871	0.43	0.664	-10.87205	17.06765
xi	.1179883	.0194184	6.08	0.000	.0799036	.156073
__000002	-.004599	.0078923	-0.58	0.560	-.020078	.0108799
__00000E	.1029051	.1497342	0.69	0.492	-.1907636	.3965737
__00000P	-.0920231	.0958976	-0.96	0.337	-.280104	.0960577
__000017	.2816812	.3250327	0.87	0.386	-.3557947	.9191571
__00001I	.3484056	.2021772	1.72	0.085	-.0481177	.744929
__000020	.0660536	.0778177	0.85	0.396	-.0865677	.2186749
__00002B	.0233964	.0424514	0.55	0.582	-.0598621	.106655
__00002T	.1755713	.1044302	1.68	0.093	-.0292441	.3803867
__000034	-.0182285	.0770871	-0.24	0.813	-.1694168	.1329598
__00003M	-.012718	.0913777	-0.14	0.889	-.191934	.166498
__00003X	.0061537	.0575444	0.11	0.915	-.1067062	.1190136
__00004F	-.0812919	.0717964	-1.13	0.258	-.2221037	.05952
__00004Q	.0799666	.0436594	1.83	0.067	-.0056611	.1655943
__000058	-.1784887	.2691469	-0.66	0.507	-.7063575	.3493801
__00005J	.1999833	.1807384	1.11	0.269	-.1544929	.5544596
__00005U	-.7437811	.693668	-1.07	0.284	-2.104249	.6166869
__00005V	.0196502	.1069722	0.18	0.854	-.1901507	.2294512
__00005W	.1606832	.2757816	0.58	0.560	-.3801982	.7015646
__00005X	.6752112	.6875046	0.98	0.326	-.6731687	2.023591
__00005Y	.254902	.17036	1.50	0.135	-.0792195	.5890235
__00005Z	-.0432676	.2709832	-0.16	0.873	-.5747379	.4882027
__000060	-.3251089	.1343512	-2.42	0.016	-.5886074	-.0616104
__000061	.0266255	.067019	0.40	0.691	-.1048166	.1580675
__000062	-.0781643	.0554077	-1.41	0.159	-.1868337	.030505
__000063	.0191998	.07193	0.27	0.790	-.1218742	.1602738
__000064	-.0445956	.0871382	-0.51	0.609	-.2154969	.1263056
__000065	3.221297	1.506369	2.14	0.033	.2669054	6.175688
__000066	.2117384	.2438644	0.87	0.385	-.2665447	.6900214
__000067	-1.140077	.6091741	-1.87	0.061	-2.33483	.0546759
__000068	-2.549518	1.528932	-1.67	0.096	-5.548161	.4491263
__000069	-.7754704	.3302917	-2.35	0.019	-1.42326	-.1276803
__00006A	.4468124	.6404596	0.70	0.485	-.8092998	1.702925
__00006B	.0174085	.2212283	0.08	0.937	-.4164793	.4512963
__00006C	-.4489524	.1527805	-2.94	0.003	-.7485957	-.1493092
__00006D	-.00632	.1066279	-0.06	0.953	-.2154457	.2028057
__00006E	-.3599806	.1768411	-2.04	0.042	-.7068132	-.0131481
__00006F	-.370012	.2029664	-1.82	0.068	-.7680832	.0280592
__00006G	.4574074	.4135166	1.11	0.269	-.3536089	1.268424
__00006H	.0431662	.0576892	0.75	0.454	-.0699777	.1563101
__00006I	-.1872093	.1580874	-1.18	0.236	-.4972609	.1228424
__00006J	-.4760806	.389179	-1.22	0.221	-1.239364	.2872032
__00006K	.0538621	.0876052	0.61	0.539	-.1179549	.2256792
__00006L	.0636558	.1384149	0.46	0.646	-.2078126	.3351243
__00006M	.0356109	.0737305	0.48	0.629	-.1089942	.180216
__00006N	.0020755	.0324692	0.06	0.949	-.0616052	.0657563
__00006O	.0110056	.0255579	0.43	0.667	-.0391203	.0611314
__00006P	-.0661403	.0383385	-1.73	0.085	-.1413323	.0090518
__00006Q	-.0158687	.0463034	-0.34	0.732	-.106682	.0749447
__00006R	-.1323997	.5962053	-0.22	0.824	-1.301717	1.036918
__00006S	.133058	.0870083	1.53	0.126	-.0375884	.3037045
__00006T	-.0221252	.2135555	-0.10	0.917	-.4409645	.396714
__00006U	-.2830195	.588041	-0.48	0.630	-1.436325	.8702858
__00006V	-.0116308	.1081847	-0.11	0.914	-.2238098	.2005481
__00006W	.174134	.2331012	0.75	0.455	-.2830396	.6313077
__00006X	.0910853	.0899486	1.01	0.311	-.0853278	.2674984
__00006Y	.1809457	.0542171	3.34	0.001	.0746114	.2872799
__00006Z	-.1245535	.0388067	-3.21	0.001	-.2006638	-.0484432

__000070	-.1033275	.0648902	-1.59	0.111	-.2305944	.0239394
__000071	-.0997885	.0756932	-1.32	0.188	-.2482431	.048666
__000072	-.8101125	.404618	-2.00	0.045	-1.603676	-.0165485
__000073	-.0564459	.0681197	-0.83	0.407	-.1900468	.077155
__000074	.3265654	.1676586	1.95	0.052	-.0022579	.6553887
__000075	.5390301	.4045194	1.33	0.183	-.2543404	1.332401
__000076	.1258582	.1006604	1.25	0.211	-.0715637	.3232801
__000077	-.034015	.1560583	-0.22	0.827	-.3400869	.2720569
__000078	-.0467356	.0672	-0.70	0.487	-.1785329	.0850616
__000079	.0905514	.0445321	2.03	0.042	.0032121	.1778908
__00007A	.0317297	.0305393	1.04	0.299	-.028166	.0916255
__00007B	.0066415	.0479143	0.14	0.890	-.0873312	.1006142
__00007C	.0114005	.0529091	0.22	0.829	-.0923684	.1151693
__00007D	.3726315	.290408	1.28	0.200	-.1969361	.9421991
__00007E	.0029701	.0490259	0.06	0.952	-.0931829	.099123
__00007F	-.1365233	.1154237	-1.18	0.237	-.3629001	.0898534
__00007G	-.1281323	.3107908	-0.41	0.680	-.7376761	.4814114
__00007H	-.1614504	.0733547	-2.20	0.028	-.3053185	-.0175823
__00007I	-.1002651	.1312053	-0.76	0.445	-.3575937	.1570636
__00007J	.028133	.0460473	0.61	0.541	-.0621779	.118444
__00007K	-.1067392	.0358891	-2.97	0.003	-.1771272	-.0363511
__00007L	-.0213276	.0242201	-0.88	0.379	-.0688296	.0261744
__00007M	.007752	.0353977	0.22	0.827	-.0616724	.0771763
__00007N	-.0357064	.042941	-0.83	0.406	-.1199252	.0485124
__00007O	-2.156739	1.196737	-1.80	0.072	-4.503859	.1903807
__00007P	.2277103	.1757316	1.30	0.195	-.1169462	.5723667
__00007Q	.2764129	.4415428	0.63	0.531	-.5895704	1.142396
__00007R	2.285177	1.314492	1.74	0.082	-.2928922	4.863246
__00007S	-1.207186	.293171	-4.12	0.000	-1.782173	-.6322
__00007T	-.8478679	.5805262	-1.46	0.144	-1.986435	.290699
__00007U	.5694521	.2416268	2.36	0.019	.0955574	1.043347
__00007V	-.2144137	.1303942	-1.64	0.100	-.4701515	.041324
__00007W	-.0662418	.1060579	-0.62	0.532	-.2742497	.1417661
__00007X	-.2307048	.1555576	-1.48	0.138	-.5357947	.074385
__00007Y	-.1895898	.1628907	-1.16	0.245	-.5090619	.1298823
__000080	37.02817	25.36055	1.46	0.144	-12.71063	86.76698
__000082	7.674764	15.94501	0.48	0.630	-23.59765	38.94718
__000085	-.458307	.6093348	-0.75	0.452	-1.653375	.736761
__000087	.5982868	.4427787	1.35	0.177	-.2701205	1.466694
__00008A	-6.551961	4.71456	-1.39	0.165	-15.79847	2.69455
__00008C	-2.315906	2.883113	-0.80	0.422	-7.970459	3.338647
__00008F	-45.89088	32.31706	-1.42	0.156	-109.2733	17.49149
__00008H	-8.892641	20.34433	-0.44	0.662	-48.79331	31.00802
__00008K	.1880201	.7698317	0.24	0.807	-1.321825	1.697865
__00008M	-1.505334	.7554431	-1.99	0.046	-2.98696	-.0237089
__00008P	1.777641	1.422047	1.25	0.211	-1.011372	4.566655
__00008R	-.6295638	1.165092	-0.54	0.589	-2.914621	1.655493
__00008U	-1.420255	.5223803	-2.72	0.007	-2.444782	-.3957277
__00008W	.937523	.4373194	2.14	0.032	.0798229	1.795223
__00008Z	.2557078	.3033405	0.84	0.399	-.3392238	.8506394
__000091	-.1170399	.2289891	-0.51	0.609	-.5661486	.3320688
__000094	-.609952	.2957423	-2.06	0.039	-1.189982	-.0299224
__000096	.0167996	.2442819	0.07	0.945	-.4623024	.4959016
__000099	.6839201	.3522254	1.94	0.052	-.0068878	1.374728
__00009B	.423636	.3331756	1.27	0.204	-.2298103	1.077082
__00009E	-.085455	.4434001	-0.19	0.847	-.955081	.784171
__00009G	-.1037876	.3728575	-0.28	0.781	-.8350607	.6274855
xbarlag	-47.91697	28.76947	-1.67	0.096	-104.3416	8.507646
x2barlag	-.5032647	.6962915	-0.72	0.470	-1.868878	.8623487
xbar2lag	10.26275	5.468939	1.88	0.061	-.4633016	20.9888
rootxbarlag	58.02308	36.13073	1.61	0.108	-12.83893	128.8851
z1barlag	1.587867	1.155684	1.37	0.170	-.6787371	3.854471
z2barlag	-1.581201	1.770778	-0.89	0.372	-5.054169	1.891767
z3barlag	.4134219	.7125977	0.58	0.562	-.9841722	1.811016
z4barlag	.0717129	.3517959	0.20	0.838	-.6182528	.7616787
z5barlag	.6846556	.2954634	2.32	0.021	.105173	1.264138
z6barlag	-1.04491	.5437817	-1.92	0.055	-2.111411	.0215913
z7barlag	.2628658	.5960508	0.44	0.659	-.9061488	1.43188
_cons	-19.22773	12.74163	-1.51	0.131	-44.21746	5.761996

(option **xb** assumed; fitted values)

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
     __0000A2 |     62,068    .3879752    .1526093   .0863061   1.746245
```

```
956
957 local qhat_hats "`qhat_hats' `pqhat' `xpqhat'"
958
959 * interact qhat_hats with Alist
960 di "`qhat_hats'"
    __00009W __00009Z __0000A5 __0000A8  __00009X __0000A0 __0000A6 __0000A9
961 if "$Alist"~="one" {
962         local qhat_hats_backup "`qhat_hats'"
963         local qhat_hats ""
964         foreach var1 of local qhat_hats_backup {
    2.                  foreach var2 of global Alist {
    3.                          g `var1'_`var2'=`var1'*`var2'
    4.                          sum `var1' `var2' `var1'_`var2'
    5.                          local qhat_hats "`qhat_hats' `var1'_`var2'"
    6.                  }
    7.          }
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
     __00009W |     62,068    .4574398    .0992715   .1996678   1.140172
         one |     62,068           1           0          1          1
__00009W_one |     62,068    .4574398    .0992715   .1996678   1.140172
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
     __00009W |     62,068    .4574398    .0992715   .1996678   1.140172
         zi7 |     62,068    .0868402    .2816031          0          1
__00009W_zi7 |     62,068    .0517662    .1711353          0   1.140172
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
     __00009Z |     62,068    .5072062    .1194309   .1986692   1.454764
         one |     62,068           1           0          1          1
__00009Z_one |     62,068    .5072062    .1194309   .1986692   1.454764
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
     __00009Z |     62,068    .5072062    .1194309   .1986692   1.454764
         zi7 |     62,068    .0868402    .2816031          0          1
__00009Z_zi7 |     62,068    .0578694    .1917272          0   1.454764
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
     __0000A5 |     62,068    .4255233    .1831281   .0840839    1.84779
         one |     62,068           1           0          1          1
__0000A5_one |     62,068    .4255233    .1831281   .0840839    1.84779
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
     __0000A5 |     62,068    .4255233    .1831281   .0840839    1.84779
         zi7 |     62,068    .0868402    .2816031          0          1
__0000A5_zi7 |     62,068    .0710904    .2401832          0    1.84779
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
     __0000A8 |     62,068    .4674328    .2030224   .0958688   2.357627
         one |     62,068           1           0          1          1
__0000A8_one |     62,068    .4674328    .2030224   .0958688   2.357627
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
     __0000A8 |     62,068    .4674328    .2030224   .0958688   2.357627
         zi7 |     62,068    .0868402    .2816031          0          1
__0000A8_zi7 |     62,068     .079402    .2700589          0   2.357627
```

Variable	Obs	Mean	Std. dev.	Min	Max
__00009X	62,068	.4531807	.3708038	.0235361	3.887367
one	62,068	1	0	1	1
__00009X_one	62,068	.4531807	.3708038	.0235361	3.887367

Variable	Obs	Mean	Std. dev.	Min	Max
__00009X	62,068	.4531807	.3708038	.0235361	3.887367
zi7	62,068	.0868402	.2816031	0	1
__00009X_zi7	62,068	.0822467	.3176493	0	3.887367

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A0	62,068	.5022971	.4185809	.0245083	4.959957
one	62,068	1	0	1	1
__0000A0_one	62,068	.5022971	.4185809	.0245083	4.959957

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A0	62,068	.5022971	.4185809	.0245083	4.959957
zi7	62,068	.0868402	.2816031	0	1
__0000A0_zi7	62,068	.0923039	.3605183	0	4.959957

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A6	62,068	.4549896	.508909	.0207021	7.352572
one	62,068	1	0	1	1
__0000A6_one	62,068	.4549896	.508909	.0207021	7.352572

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A6	62,068	.4549896	.508909	.0207021	7.352572
zi7	62,068	.0868402	.2816031	0	1
__0000A6_zi7	62,068	.1166475	.4747296	0	7.352572

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A9	62,068	.5015765	.5800013	.0215572	9.381271
one	62,068	1	0	1	1
__0000A9_one	62,068	.5015765	.5800013	.0215572	9.38127

Variable	Obs	Mean	Std. dev.	Min	Max
__0000A9	62,068	.5015765	.5800013	.0215572	9.381271
zi7	62,068	.0868402	.2816031	0	1
__0000A9_zi7	62,068	.1311749	.5456854	0	9.38127

```
965 }

966
967 local r_RE      "`r_RE' `qhat_hats'"

968
969 /*******************************************************************************
  > Set instruments and start values
  > *******************************************************************************/
970
971 if simple_model==1      {
972         local r_RE "xbarlag `xixbarlag' `budget'"
973         local r_RE1 ""
```

```
974
975         forv i=1/`T' {
    2.                  local r_RE "`r_RE' zi`i'"
    3.          }
976
977         local initial_values_RE "a -0.6674374  b .3126426  d  0.6948198  v0   -0.006
  > 3737  "
978 }

979
980 if `J'==2 & simple_model==0     {
981         g y = p1*qi1
982         g ybar_q1 = p1*qikbar1
983         g ybar_q2 = p2*qikbar2
984         g ybar = p1*qikbar1 + p2*qikbar2
985         g ybar2 = ybar^2
986         g ybar_q1_2 = ybar_q1^2
987         g ybar_q2_2 = ybar_q2^2
988         g ybar_q1_q2 = ybar_q1*ybar_q2
989         g ybarx = ybar*xi
990         g ybarx_1 = ybar_q1*xi
991         g ybarx_2 = ybar_q2*xi
992         g xi2 = xi^2
993
994         forv j = 1(1)`T' {
    2.                  g pzi`j' = p1*zi`j'
    3.          }
995
996         if same_spillover==1 {
997                 reg y ybar2 ybarx xi2 ybar xi ybar_q1 pzi*
998                 local b = _b[xi]
999                 local a = _b[ybar_q1]
1000                if same_spillover==1 {
1001                        local a = logit(_b[ybar_q1]/2 + .5)
1002                        if `a'==. local a = .5
1003                }
1004                local d = _b[xi2]
1005                local initial_values_RE "A11:one `a' b1 `b' d1 `d'"
1006        }
1007        else {
1008                reg y xi2 ybar_q1_2 ybar_q2_2 ybarx_1 ybarx_2 ybar_q1_q2 xi ybar_q1
  > ybar_q2 pzi*
```

Source	SS	df	MS	Number of obs	=	62,068
				F(16, 62051)	=	27278.78
Model	2741.85324	16	171.365827	Prob > F	=	0.0000
Residual	389.805521	62,051	.006282018	R-squared	=	0.8755
				Adj R-squared	=	0.8755
Total	3131.65876	62,067	.0504561	Root MSE	=	.07926

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
xi2	-.0350102	.0008488	-41.25	0.000	-.0366739	-.0333466
ybar_q1_2	-.1020534	.0144398	-7.07	0.000	-.1303556	-.0737513
ybar_q2_2	.1472725	.0039692	37.10	0.000	.1394929	.1550522
ybarx_1	.2713171	.005181	52.37	0.000	.2611623	.2814719
ybarx_2	-.1382924	.0029161	-47.42	0.000	-.144008	-.1325769
ybar_q1_q2	-.1377699	.0138865	-9.92	0.000	-.1649875	-.1105523
xi	.3757109	.0025376	148.06	0.000	.3707372	.3806846
ybar_q1	.2099972	.0107077	19.61	0.000	.1890101	.2309843
ybar_q2	-.1543075	.0058364	-26.44	0.000	-.1657469	-.1428681
pzi1	.1909039	.001685	113.30	0.000	.1876013	.1942065
pzi2	.0147535	.0026857	5.49	0.000	.0094896	.0200175
pzi3	-.0031766	.0008218	-3.87	0.000	-.0047874	-.0015658
pzi4	.00991	.0009101	10.89	0.000	.0081261	.0116939
pzi5	-.0110655	.0008257	-13.40	0.000	-.012684	-.0094471
pzi6	-.0015413	.0006718	-2.29	0.022	-.002858	-.0002245
pzi7	-.0065573	.0013698	-4.79	0.000	-.0092421	-.0038725
_cons	.0018044	.0026093	0.69	0.489	-.0033099	.0069187

```
1009                local b = _b[xi]
1010                local a1 = _b[ybar_q1]/(1-`b')
1011                local a2 = _b[ybar_q2]/`b'
1012                if same_spillover==1 {
1013                        local a1 = logit(_b[ybar_q1]/)2*(1-`b')) + .5)
1014                        local a2 =  logit(-_b[ybar_q2]/(2*`b') + .5)
1015
1016                        if `a1'==. local a1 = .5
1017                        if `a2'==. local a2 = .5
1018                }
1019                local d = _b[xi2]
1020
1021                local initial_values_RE "A11:one `a1' A22:one `a2' b1 `b' d1 `d'"
1022        }
1023
1024        drop y ybar ybar2 ybarx xi2 ybar_q1 ybar_q2 pzi* ybar_q1_2 ybar_q2_2 ybar_q1
  > _q2
1025        forv j = 1(1)`T' {
    2.                  local coef = _b[pzi`j']
    3.                  local initial_values_RE "`initial_values_RE' C1:zi`j' `coef'"
    4.          }
1026}

1027
1028if `J'==3 & simple_model==0
  >                             local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1029if `J'==3 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 0.01 b2 0.26 d1 0.03 d2 0.01"

1030if `J'==4 & simple_model==0
  >                             local initial_values_RE "A11:one 0.6 A22:one 0.6 A33:one
  >  0.6 A44:one 0.6 b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1031if `J'==4 & simple_model==0 & same_spillover==1 local initial_values_RE "A11:one 0.6
  >  b1 -0.08 b2 0.24 b3 0.24 d1 0.01 d2 0.01 d3 -0.01"

1032
1033if simple_model==0 local initial_values_RE "`initial_values_RE' AVA11:_cons 0.1 AVA2
  > 2:_cons 0.2"

1034
1035if `J'==2 & estimate_RE==1 {
1036
1037        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(`r_RE') $trace_level  ///
  >                 $derivatives   ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ') $tol_level $maxiter $technique $technique_RE

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00211777
  Iteration 1:  GMM criterion Q(b) =   .0017908
  Iteration 2:  GMM criterion Q(b) =  .00019979
  Iteration 3:  GMM criterion Q(b) =  .00013994

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01052146
  Iteration 1:  GMM criterion Q(b) =  .01017858
  Iteration 2:  GMM criterion Q(b) =  .00816979
  Iteration 3:  GMM criterion Q(b) =  .00815759
  Iteration 4:  GMM criterion Q(b) =  .00813836
  Iteration 5:  GMM criterion Q(b) =  .00813162

  GMM estimation

  Number of parameters =  23
  Number of moments    =  36
  Initial weight matrix: Unadjusted                 Number of obs   =     62,068
  GMM weight matrix:     Robust
```

(Std. err. adjusted for **494** clusters in **state_district_round**)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
A11						
one	.7318883	.0432265	16.93	0.000	.6471659	.8166106
zi7	-.1812373	.0921043	-1.97	0.049	-.3617584	-.0007161
A22						
one	.7322945	.0511659	14.31	0.000	.6320112	.8325778
zi7	-.2756609	.1167394	-2.36	0.018	-.5044659	-.046856
C1						
zi1	.4130719	.0629576	6.56	0.000	.2896774	.5364665
zi2	-.3873531	.1155084	-3.35	0.001	-.6137455	-.1609607
zi3	-.0386857	.0261047	-1.48	0.138	-.0898499	.0124786
zi4	.1392316	.0375616	3.71	0.000	.0656122	.212851
zi5	.06542	.0263325	2.48	0.013	.0138092	.1170307
zi6	-.020545	.0208881	-0.98	0.325	-.061485	.020395
zi7	-.0425158	.0473028	-0.90	0.369	-.1352275	.0501959
C2						
zi1	.1988457	.0961883	2.07	0.039	.0103201	.3873712
zi2	-.6388354	.1771527	-3.61	0.000	-.9860484	-.2916224
zi3	-.0551168	.0377482	-1.46	0.144	-.1291019	.0188684
zi4	.2029281	.0538194	3.77	0.000	.0974439	.3084122
zi5	.1102927	.0326426	3.38	0.001	.0463144	.174271
zi6	-.0322669	.0291199	-1.11	0.268	-.0893409	.0248071
zi7	-.0801479	.0624595	-1.28	0.199	-.2025663	.0422706
/d1	-.0815342	.0100185	-8.14	0.000	-.1011701	-.0618983
/b1	.4486495	.0278838	16.09	0.000	.3939982	.5033009
/AVA11	1.104604	.8423605	1.31	0.190	-.5463925	2.7556
/AVA12	-.7192814	.7233229	-0.99	0.320	-2.136968	.6984054
/AVA22	-.2035701	.6769549	-0.30	0.764	-1.530377	1.123237

Instruments for equation **eq1_RE**: **__00009J __00009L __00009O xi __000002 __00000E __00000P __000017 __0000II __000020 __00002B __00002T __000034 __00003M __00003X __00004F __00004Q __000058 __00005J __00009W_one __00009W_zi7 __00009Z_one __00009Z_zi7 __0000A5_one __0000A5_zi7 __0000A8_one __0000A8_zi7 __00009X_one __00009X_zi7 __0000A0_one __0000A0_zi7 __0000A6_one __0000A6_zi7 __0000A9_one __0000A9_zi7 _cons**

```
1038
1039        if simple_model==1 capture noisily nlcom _b[/a]+_b[/c]+2*_b[/a]*_b[/d]*(0.3*
  > _b[/a]+1.19*_b[/b])
1040        if simple_model==1 capture noisily nlcom _b[/a]+2*_b[/a]*_b[/d]*(0.3*_b[/a]+
  > 1.19*_b[/b])
1041
1042        if simple_model==1 {
1043                /7 calculate derivative and save
1044                local RE_acz_drv ""
1045                tempvar junk
1046                g `junk'=.
1047                forv i=1/`Tm1' {
    2.                          replace `junk' = zi`i'*p1
    3.                          su `junk' $GMM_weight
    4.                          local mean_zi`i'_p1 = r(mean)
    5.                          local RE_acz_drv        "`RE_acz_drv' + _b[/C`i']*`mean_z
  > i`i'_p1'"
    6.                  }
```

```
1048                replace `junk'=p1*qikbar1
1049                su `junk' $GMM_weight
1050                local mean_y = r(mean)
1051                su xi $GMM_weight
1052                local mean_x = r(mean)
1053
1054                local c_term ""
1055
1056                noi di "_b[/a] `c_term' + 2*_b[/a]*_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a] `RE_acz_drv')"
1057                estimates store gmm_est
1058                nlcom _b[/a]  `c_term' + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*
  > _b[/a]  `RE_acz_drv'), post
1059                estimates save "${filename_RE}_dF", replace
1060                estimates restore gmm_est
1061        }
1062}

1063
1064if `J'==3 & estimate_RE==1 {
1065        capture noisily gmm `eqs_RE', ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2')  ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1066}

1067if `J'==4 & estimate_RE==1 {
1068        capture noisily gmm `eqs_RE' $GMM_weight, ///
  >                 instruments(1:`r_RE' `r_RE1') instruments(2:`r_RE' `r_RE2') instrume
  > nts(3:`r_RE' `r_RE3') ///
  >                 $derivatives $trace_level  ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_RE
  > ')  $tol_level $maxiter $technique
1069}

1070esttab using "${filename_RE}.tex", se replace //added this
  (file
      diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh_drop5_all_ex
      > actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex not
      found)
  (output written to diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh
  > _drop5_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.tex
  > )

1071
1072
1073// save coefs
1074matrix est_b=e(b)

1075unique hhidi
  Number of unique values of hhidi is  11323
  Number of records is  62068

1076local N_hh = r(sum)

1077estadd scalar N_hh = r(sum)

  added scalar:
                e(N_hh) =  11323
```

```
1078unique group_round
  Number of unique values of group_round is  1823
  Number of records is  62068

1079estadd scalar N_grp = r(sum)

  added scalar:
               e(N_grp) =  1823

1080estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  6.2111903

1081if estimate_RE==1 {
1082        estimates save "$filename_RE", replace
  (note: file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh_drop5_
  > all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster not f
  > ound)
  file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh_drop5_all_exa
  > ctInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_RE.ster saved
1083        estimates save temp1_RE, replace
  file temp1_RE.ster saved
1084}

1085
1086estimates store temp1_FE

1087
1088/*******************************************************************************
  > Set starting values and instruments for FE analysis
  > *******************************************************************************/
1089
1090if simple_model==0 {
1091
1092        // make FE instruments
1093
1094        // denoting ximxk as x and zimzk of individual zs as zi:
1095        //   x,x2,p,zi,px,zi,pzi
1096        // local xdiff "`ximxk' `x2imx2k' `xixk'"
1097        local xdiff "`ximxk' "
1098        local xdiffxdiff ""
1099        local zindivdiff ""
1100        local pzindivdiff ""
1101        local pzindivdiffpqhat ""
1102        local pxzindivdiff ""
1103        local p2zindiv2diff ""
1104        local zg ""
1105        local pzg ""
1106        local zindivdiffxdiff ""
1107        local pzindivdiffxdiff ""
1108        local pzindivdiffp ""
1109        local pzindivdiffpzg ""
1110        local zgxdiff ""
1111        local pzgxdiff ""
1112        tempvar ximxkxdiff x2imx2kxdiff xixkxdiff
1113        g double `ximxkxdiff'=`ximxk'*`ximxk'
1114        g double `x2imx2kxdiff'=`x2imx2k'*`ximxk'
1115        g double `xixkxdiff'=xi*xk*`ximxk'
1116
```

```
1117        local xdiffxdiff "`x2imx2k'"
1118
1119        forval t=1/`Tindiv' {
    2.                  tempvar zi`t'mzk`t'xdiff
    3.                  g double `zi`t'mzk`t'xdiff'=`zi`t'mzk`t''*`ximxk'
    4.                  local zindivdiff "`zindivdiff' `zi`t'mzk`t'' "
    5.                  local zindivdiffxdiff "`zindivdiffxdiff' `zi`t'mzk`t'xdiff' p`j'2
  > z`t'2diff "
    6.                  forval j=1/`J' {
    7.                          tempvar zi`t'mzk`t'p`j'xdiff xizi`t'mxkzk`t'p`j' p`j'2z`t
  > '2diff
    8.                          g double `zi`t'mzk`t'p`j'xdiff'=`zi`t'mzk`t'p`j''*`ximxk'
    9.                          g double `xizi`t'mxkzk`t'p`j''=(xi*zi`t'-xk*zk`t')*p`j'
   10.                          g double `p`j'2z`t'2diff'=(p`j')^2*((zi`t')^2-(zk`t')^2)
   11.                          local pzindivdiff "`pzindivdiff'  `zi`t'mzk`t'p`j'' "
   12.                          local pzindivdiffxdiff "`pzindivdiffxdiff'  `zi`t'mzk`t'p
  > `j'xdiff' "
   13.                          local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j''
  > "
   14.                          local p2zindiv2diff "`p2zindiv2diff' `p`j'2z`t'2diff'"
   15.                          forval l=1/`J' {
   16.                                  tempvar zdiff`t'p`j'p`l'
   17.                                  g double `zdiff`t'p`j'p`l''=`zi`t'mzk`t'p`j''*p`l
  > '
   18.                                  local pzindivdiffp "`pzindivdiffp' `zdiff`t'p`j'p
  > `l'"
   19.                                  foreach name of global Alist {
   20.                                          tempvar  zdiff`t'p`j'p`l'q`l'`name'
   21.                                          g double `zdiff`t'p`j'p`l'q`l'`name''=`zi
  > `t'mzk`t'p`j''*p`l'qikhat`l''*`name'
   22.                                          local pzindivdiffpqhat "`pzindivdiffpqhat
  > ' `zdiff`t'p`j'p`l'q`l'`name'"
   23.                                  }
   24.                                  if `Tindivp1'<=`T' {
   25.                                          forval s=`Tindivp1'/`T' {
   26.                                                  tempvar zdiff`t'p`j'zi`s'p`l'
   27.                                                  * don't generate these to save me
  > mory!
1120                                                 gen double `zdiff`t'p`j'zi`s'p`l''=
  >     `zdiff`t'p`j'p`l''*zi`s'
   28.                                                  local pzindivdiffpzg " `pzindivdi
  > ffpzg' `zdiff`t'p`j'zi`s'p`l'' "
   29.                                          }
   30.                                  }
   31.                          }
   32.                  }
   33.          }
1121        if `Tindivp1'<=`T' {
1122                forval t=`Tindivp1'/`T' {
    2.                          tempvar zi`t'xdiff
    3.                          g double `zi`t'xdiff'=zi`t'*`ximxk'
    4.                          local zg "`zg' zi`t'"
    5.                          forval j=1/`J' {
    6.                                  tempvar zi`t'p`j'xdiff
    7.                                  g double `zi`t'p`j'xdiff'=`zi`t'p`j''*`ximxk'
    8.                                  local pzg "`pzg' `zi`t'p`j''"
    9.                                  local pzgxdiff "`pzgxdiff' `zi`t'p`j'xdiff'"
   10.                          }
   11.                  }
1123        }
```

```
1124
1125        // pzgxdiff for group-varying parts of C, rootpxdiff for R, pqhatxdiff for A
1126        // denoting zg as group-zs
1127        //  zg,pzgx,rprp,rprpx
1128        local rootprootp ""
1129        local rootprootpxdiff ""
1130
1131        forval j=1/`J' {
    2.                  forval s=`j'/`J' {
    3.                          tempvar xdiffrp`j'rp`s'
    4.                          g double `xdiffrp`j'rp`s''=`ximxk'*sqrt(p`j')*sqrt(p`s')
    5.
1132                        if (`s'==`j')           local rootprootp "`rootprootp' `rp`j
  > 'rp`s'' "
    6.
1133                        local rootprootpxdiff "`rootprootpxdiff' `xdiffrp`j'rp`s'' "
  >
    7.                  }
    8.          }
1134
1135        local pqhatxdiff ""
1136        forval j=1/`J' {
    2.                  forval s=1/`J' {
    3.                          foreach name of global Alist {
    4.                                  tempvar   p`s'qikhat`j'xdiff`name'
    5.                                  g double `p`s'qikhat`j'xdiff`name''=`ximxk'*p`s'*
  > `qikhat`j''*`name'
    6.
1137                                    local pqhatxdiff "`pqhatxdiff' `p`s'qikhat`j'xdiff`n
  > ame'' "
    7.                          }
    8.                  }
    9.          }
1138
1139        // set instruments
1140        local r_FE "`xdiff' `pzindivdiff' `xdiffxdiff' `pzgxdiff' `pqhatxdiff' `pzin
  > divdiffpqhat' `pxzindivdiff'"
1141}

1142
1143
1144if simple_model==1 {
1145        foreach var in xbarlag x2barlag xbar2lag rootxbarlag {
    2.                  tempvar ximxk`var'
    3.                  g `ximxk`var''=`ximxk'*`var'
    4.          }
1146
1147        /******************
  >         Initial values
  >         ******************/
1148
1149        if init_FE_from_RE==1 {
1150                estimates use "$filename_RE"
1151
1152                // clear init values
1153                local initial_values_FE ""
1154
1155                // extract vector of coefs and paramter names from
1156                matrix coefs = e(b)
1157                local paramlist = e(params)
```

```
1158
1159                // iterate through paramter name list, taking
1160                local t=0
1161                foreach p of local paramlist {
    2.                          local `++t'
    3.                          local p_nice = subinstr("`p'",":_cons","",.)
    4.                          local est = coefs[1,`t']
    5.                          if "`p_nice'"!="c" & "`p_nice'"!="v0" local initial_value
  > s_FE "`initial_values_FE' `p_nice' `est'"
    6.                  }
1162        }
1163
1164        // make z diff instruments
1165        local pzindivdiff ""
1166        local pz2indivdiff ""
1167        local pzXzindivdiff ""
1168        local pxzindivdiff ""
1169
1170        forv i=1/`T' {
    2.
1171                tempvar z2i`i'mz2k`i'p1p1
    3.                  g `z2i`i'mz2k`i'p1p1' = `z2i`i'mz2k`i''*p1*p1
    4.
1172                local pzindivdiff "`pzindivdiff' `zi`i'mzk`i'p1' "
    5.                  local pz2indivdiff "`pz2indivdiff' `z2i`i'mz2k`i'p1p1'"
    6.
1173                tempvar xizi`i'mxkzk`i'p1
    7.                  g double `xizi`i'mxkzk`i'p1'=(xi*zi`i'-xk*zk`i')*p1*p1
    8.                  local pxzindivdiff "`pxzindivdiff' `xizi`t'mxkzk`t'p`j'' "
    9.
1174                forv j=1(1)`T' {
   10.                          if `j'>`i' {
   11.                                  tempvar zi`j'zi`i'mzkz`j'k`i'p1
   12.                                  g double `zi`j'zi`i'mzkz`j'k`i'p1'=(zi`j'*zi`i'-z
  > k`j'*zk`i')*p1*p1
   13.                                  local pzXzindivdiff "`pzXzindivdiff' `zi`j'zi`i'm
  > zkz`j'k`i'p1'"
   14.                          }
   15.                 }
   16.         }
1175
1176        // p z interactions
1177        local pzg ""
1178        local pzgxdiff ""
1179        if `Tindivp1'<=`T' {
1180                forval t=`Tindivp1'/`T' {
    2.                          tempvar p1xdiffz`t'
    3.                          g double `p1xdiffz`t''=p1*`ximxk'*zi`t'
    4.                          local pzgxdiff "`pzgxdiff' `p1xdiffz`t''"
    5.                          local pzg "`pzg' `zi`t'p1'"
    6.                  }
1181        }
1182
1183        tempvar p1ximxk  p1x2imx2k
1184        g `p1ximxk' = p1*`ximxk'
1185        g `p1x2imx2k' = (p1^2)*`x2imx2k'
1186
1187        // define instruments
1188        local r_FE "`p1ximxk' `pzindivdiff' `p1x2imx2k'  `pz2indivdiff' `pzgxdiff'
  >    `pxzindivdiff' `pzXzindivdiff'"
```

```
1189
1190        // add prediction of quantity as extra instrument
1191        local pqhat ""
1192        local qhat ""
1193        local pqhatxdiff ""
1194        local pqhatxdiff2 ""
1195        local pzindivdiffpqhat ""
1196        local pzindivdiffpxbar_lag ""
1197
1198        tempvar qik_hat1
1199         reg qikbar1    `r_RE' `qhat_insts', vce(cluster group_round)
1200        predict `qik_hat1'
1201        local qhat "`qhat' `qik_hat1'"
1202
1203        tempvar p1_qikbar1 p1_qikhat1 p1qikbar1xdiff p1qikhat1xdiff p1qikbar1xdiff2
  > p1qikhat1xdiff2
1204
1205        g double `p1_qikbar1' = p1*qikbar1
1206        reg `p1_qikbar1'        `r_FE' `qhat_insts', vce(cluster group_round)
1207        predict `p1_qikhat1'
1208        local pqhat "`pqhat' `p1_qikhat1'"
1209
1210        g double `p1qikbar1xdiff' = p1*qikbar1*`p1ximxk'
1211        g `p1qikhat1xdiff' = `p1_qikhat1'*`p1ximxk'
1212
1213        local pqhatxdiff "`pqhatxdiff' `p1qikhat1xdiff'"
1214
1215        g double `p1qikbar1xdiff2'=p1*qikbar1*`x2imx2k'
1216        g `p1qikhat1xdiff2' = `p1_qikhat1'*`x2imx2k'
1217        local pqhatxdiff2 "`pqhatxdiff2' `p1qikhat1xdiff2'"
1218
1219        forv t=1/`Tindiv' {
    2.                  tempvar zdiff`t'p1p1qhat zdiff`t'p1p1qbar pzdiff`t'_xbarlag
    3.                  g `zdiff`t'p1p1qbar' = (zi`t'-zk`t')*p1*p1*qikbar1
    4.                  g `zdiff`t'p1p1qhat' = (zi`t'-zk`t')*p1*`p1_qikhat1'
    5.                  local pzindivdiffpqhat "`pzindivdiffpqhat' `zdiff`t'p1p1qhat'"
    6.
1220                g `pzdiff`t'_xbarlag' = (zi`t'-zk`t')*p1*xbarlag
    7.                  local pzindivdiffpxbar_lag "`pzindivdiffpxbar_lag' `pzdiff`t'_xba
  > rlag'"
    8.          }
1221
1222        local r_FE "`r_FE' `pqhatxdiff'   `pzindivdiffpqhat' `pqhat' "
1223}

1224
1225// starting values
1226if init_FE_from_RE==1 & "$init_FE_file"!="" {
1227        estimates use "$init_FE_file"
1228        local initial_values_FE ""
1229
1230        // extract vector of coefs and paramter names from
1231        matrix coefs = e(b)
1232        local paramlist = e(params)
1233
1234        // iterate through paramter name list, taking
1235        local t=0
1236        foreach p of local paramlist {
    2.                  local `++t'
    3.                  local p_nice = subinstr("`p'",":_cons","",.)
    4.                  if substr("`p_nice'",1,1)!="A" local p_nice = subinstr("`p_nice'"
  > ,":zi","",.)
    5.                  local est = coefs[1,`t']
    6.                  if "`p_nice'"!="c" & "`p_nice'"!="v0" & regexm("`p_nice'","AVA")=
  > =0 local initial_values_FE "`initial_values_FE' `p_nice' `est'"
    7.          }
```

```
1237}

1238
1239if estimate_FE==1 {
1240        capture noisily gmm `eqs_FE' $GMM_weight , instruments(`r_FE') $derivatives
  > $trace_level $tol_level $maxiter ///
  >                 vce(clust ${clust_var}) $wmatrix  $w_initial from(`initial_values_FE
  > ')  $technique
  note: instruments __00005B __00005M __0000D7 __0000DA __0000DG __0000DJ __0000DP
    __0000DQ __0000DS __0000DT __0000DY __0000DZ __0000E1 __0000E2 omitted because
    of collinearity.

  Step 1
  Iteration 0:  GMM criterion Q(b) =  .00080439
  Iteration 1:  GMM criterion Q(b) =  .00035881
  Iteration 2:  GMM criterion Q(b) =  .00026019

  Step 2
  Iteration 0:  GMM criterion Q(b) =  .01786949
  Iteration 1:  GMM criterion Q(b) =   .0153499
  Iteration 2:  GMM criterion Q(b) =  .01514223
  Iteration 3:  GMM criterion Q(b) =   .0151158

  GMM estimation

  Number of parameters =  20
  Number of moments    =  81
  Initial weight matrix: Unadjusted                 Number of obs   =     62,068
  GMM weight matrix:     Robust

                  (Std. err. adjusted for 494 clusters in state_district_round)
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
C11						
_cons	.4959935	.035456	13.99	0.000	.4265011	.5654859
C12						
_cons	-.0916188	.0522531	-1.75	0.080	-.1940331	.0107955
C13						
_cons	-.0208345	.0156569	-1.33	0.183	-.0515215	.0098524
C14						
_cons	-.0371959	.0194675	-1.91	0.056	-.0753515	.0009598
C15						
_cons	.0116451	.0198678	0.59	0.558	-.0272951	.0505853
C16						
_cons	-.0467979	.0165855	-2.82	0.005	-.079305	-.0142908
C17						
_cons	-3.124349	.7807613	-4.00	0.000	-4.654613	-1.594085
C21						
_cons	.2957514	.0475438	6.22	0.000	.2025673	.3889356
C22						
_cons	-.153735	.0717415	-2.14	0.032	-.2943457	-.0131242
C23						
_cons	-.027324	.0206712	-1.32	0.186	-.0678388	.0131909
C24						
_cons	-.0832615	.0314301	-2.65	0.008	-.1448634	-.0216597
C25						
_cons	.0410852	.0244391	1.68	0.093	-.0068146	.088985
C26						

	_cons	-.0571932	.0212769	-2.69	0.007	-.0988952	-.0154913
C27							
	_cons	1.419211	.6219856	2.28	0.023	.2001417	2.63828
d1							
	_cons	-.0659388	.0072444	-9.10	0.000	-.0801376	-.0517401
b1							
	_cons	.4116599	.0199416	20.64	0.000	.3725751	.4507447
A11							
	one	3.360398	.7098971	4.73	0.000	1.969026	4.751771
	zi7	1.597347	.8474311	1.88	0.059	-.0635869	3.258282
A22							
	one	-2.156251	.6330155	-3.41	0.001	-3.396939	-.9155637
	zi7	1.000465	.6402758	1.56	0.118	-.2544521	2.255383

Instruments for equation **eq1_FE**: **__000004 __00000H __00000S __00001A __00001L __000023 __00002E __00002W __000037 __00003P __000040 __00004I __00004T o.__00005B o.__00005M __000005 __0000E6 __0000E7 __0000E8 __0000E9 __0000EA __0000EB __0000EC __0000ED __0000AJ __0000AK __0000AM __0000AN __0000AS __0000AT __0000AV __0000AW __0000B2 __0000B3 __0000B5 __0000B6 __0000BB __0000BC __0000BE __0000BF __0000BL __0000BM __0000BO __0000BP __0000BU __0000BV __0000BX __0000BY __0000C4 __0000C5 __0000C7 __0000C8 __0000CD __0000CE __0000CG __0000CH __0000CN __0000CO __0000CQ __0000CR __0000CW __0000CX __0000CZ __0000D0 __0000D6 o.__0000D7 __0000D9 o.__0000DA __0000DF o.__0000DG __0000DI o.__0000DJ o.__0000DP o.__0000DQ o.__0000DS o.__0000DT o.__0000DY o.__0000DZ o.__0000E1 o.__0000E2 __0000AG __0000AP __0000AZ __0000B8 __0000BI __0000BR __0000C1 __0000CA __0000CK __0000CT __0000D3 __0000DC __0000DM __0000DV _cons**

```
1241
1242        esttab using "${filename_FE}.tex", se replace //added this
(file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh_drop5_all_ex
> actInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex not found)
(output written to diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh
> _drop5_all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.tex
> )
1243
1244        estat overid

    Test of overidentifying restriction:

    Hansen's J chi2(61) = 938.207 (p = 0.0000)
1245        di "$S_DATE $S_TIME"
   8 Dec 2023 14:30:57
1246        unique hhidi
  Number of unique values of hhidi is  11323
  Number of records is  62068
1247        local N_hh = r(sum)
1248        estadd scalar N_hh = r(sum)

  added scalar:
              e(N_hh) =  11323
1249        unique group_round
  Number of unique values of group_round is  1823
  Number of records is  62068
```

```
1250        estadd scalar N_grp = r(sum)

  added scalar:
              e(N_grp) =  1823
1251        estadd scalar avg_grp_size = `N_hh'/r(sum)

  added scalar:
       e(avg_grp_size) =  6.2111903
1252        estimates save "$filename_FE", replace
  (note: file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh_drop5_
  > all_exactInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster not f
  > ound)
  file diag_2good_norm_main_nopcross_state_Adiag_fsuXsegXreligXschedXehigh_drop5_all_exa
  > ctInst_noPPXdInst_14_zlist1_updateExp_urbOnly_clus_Dist_dropPre_FE.ster saved
1253
1254        if simple_model==1 {
1255                local RE_acz_drv ""
1256                forv i=1/`Tindiv' {
    2.                          cap drop junk
    3.                          g junk = zi`i'*p1
    4.                          su junk $GMM_weight
    5.                          local mean_zi`i'_p1 = r(mean)
    6.                                  local RE_acz_drv       "`RE_acz_drv' + _b[/C`i']
  > *`mean_zi`i'_p1'"
    7.                  }
1257                noi di "`RE_acz_drv'"
1258
1259                cap g junk=.
1260                replace junk=p1*qikbar1
1261                su junk $GMM_weight
1262                local mean_y = r(mean)
1263                su xi $GMM_weight
1264                local mean_x = r(mean)
1265
1266                estimates store gmm_est
1267                nlcom _b[/a]  + 2*_b[/a] *_b[/d]*(`mean_x'*_b[/b]+`mean_y'*_b[/a]  `
  > RE_acz_drv'), post
1268                estimates save "${filename_FE}_dF", replace
1269                estimates restore gmm_est
1270        }
1271}

1272
1273
  end of do-file
1274}

1275else {
1276        global filename_RE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_RE"
1277        global filename_FE "diag_`J'good`catname'`simp_name'_main_nopcross_state_A`d
  > escA'`cnstname'_`group_def'`grp_type'`desc_compGrp'_drop`min_group_size'_all`inst_de
  > sc'`inst_rootp'_`output_number'_zlist${short_zlist_val}`expUpname'`measError_desc'`R
  > E_sq_inst_desc'`desc_struc_v0'`sectorName'`clus_name'`wmatrix_desc'`actual_pqhat_des
  > c'`dropZdesc'`weight_desc'`desc_samp'`typeExp'_randgrp${rand_grp_num}_FE"
1278
1279        global init_FE_file=subinstr("$filename_FE","_FE","_RE",.)
```

```
1280
1281        drop if size_group_round<min_group_size
1282        if "`groupComp_def'"!="" drop if size_groupComp_round<min_group_size
1283
1284        do "$CODE/estim program.do"
1285}

1286
1287
1288log close
       name:  <unnamed>
        log:  C:\Users\pousim2\Downloads\rep\replication\output\rep_modification.smcl
   log type:  smcl
  closed on:   8 Dec 2023, 14:30:58
-------------------------------------------------------------------------------
```