

Harjoitus 4: Tutustuminen R:n ja regressioanalyysi (R)

MS-C2107 Sovelletun matematiikan tietokonetyöt 2022



4. Harjoituskerta

Aiheet:

- R:n käyttö ja syntaksi vs. Matlab
- Datan käsittely R:llä
- Regressiomallien muodostaminen R:llä

Osaamistavoitteet:

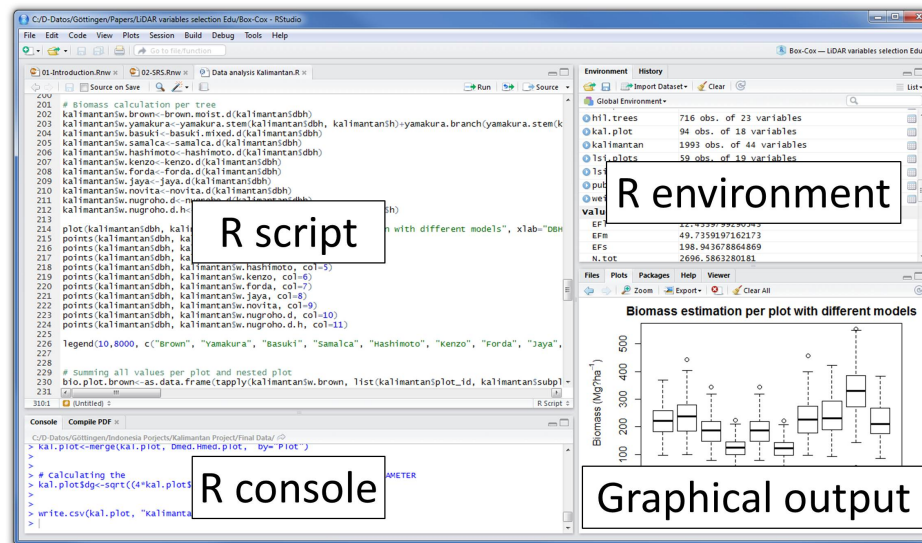
- Osaat R:n perussyntaksin ja erityispiirteet datan käsittelyssä
- Hallitset regressioanalyysin ja visualisoinnin R:llä

R ohjelmointiympäristönä

- Ilmainen ohjelmointiympäristö data-analytiikkaan, mallintamiseen sekä visualisointiin
- Pohjautuu vuonna 1967 julkaistuun S-kieleen
- Pitkälti samat toiminnallisuudet kuin matlabissa
- R on ns. ”skriptauskieli”, eli ajetaan rivi / osio kerrallaan
- Lukematon määrä paketteja (vrt. Matlabin toolboxit) eri toimintoihin ladattavissa
- R:ää käytetään enenevässä määrin esim. yritysmaailmassa

RStudio

- RStudio on graafinen ympäristö R:n ajamiseen. Kierroksen tehtävät suoritetaan käyttämällä RStudiota, joka on valmiiksi asennettu Aallon koneille.



Kuva 1: Lähde: <https://analyticsbuddhu.files.wordpress.com/2016/06/rstudio.jpg>

- Konsoli: Käytä kuten Matlabin komentoikkunaa
 - > x=1
 - > x
 - [1] 1
- Skriptit (.r-tiedostot): Valitse haluamasi pätkä ja aja ctrl+enterillä
 - > polku=getwd()
 - > polku
 - [1] "C:/Users/StudentID/Documents"
 - > setwd(polku)
- Environment: Muuttujat ja funktiot (vrt. Matlabin workspace)
- Graafiset tulokset: Kuvaajat, Help jne.
 - > help("install.packages")

R:n syntaksi - muuttujat

Muuttujaan sijoittaminen tapahtuu = tai <- operaattoreilla

- Vektorit:

- Vektorit voivat sisältää erityyppisiä muuttujia

- > #Kokonaislukuvektori

- > v1=c(1,2,3)

- > #Logiikkavektori

- > v2=c(TRUE,FALSE,TRUE)

- > #Tekstivektorin pituus

- > length(c("aa","bb","cc"))

- [1] 3

- Vektorien muokkaaminen

- > v1[1]

- [1] 1

- > #Arvon muuttaminen

- > v1[1]=2

- > v1

```
[1] 2 2 3
```

```
> #Vektorien yhdistely:
```

```
> c(c(1,2,3),c("aa","bb"))
```

```
> #Huom! Numeroista tulee tekstimuuttujia
```

```
[1] "1" "2" "3" "aa" "bb"
```

- Matemaattiset operaatiot ovat alkioittaisia vektoreille

```
> #Vektorien +,-,* ja /-laskut (alkioittaisia):
```

```
> c(1,2,3)+c(4,5,6)
```

```
[1] 5 7 9
```

```
> c(1,2,3)*c(4,5,6)
```

```
[1] 4 10 18
```

- R:n vektori ei lähtökohtaisesti ole matemaattinen vektori

- Matriisit:

- Matriisien luominen

```
> #Luodaan matriisi, jossa on 2 riviä ja 3 saraketta:  
> A <- matrix(c(1,2,3,4,5,6),nrow=2,ncol=3,byrow=TRUE)  
> A  
      [,1] [,2] [,3]  
[1,]    1    2    3  
[2,]    4    5    6
```

- Huom! `as.matrix` muuttaa muita taulukoita matriisimuotoisiksi
- Alkioihin viittaaminen

```
> #Alkioihin viittaaminen  
> A[2,3] #yksittäinen alkio  
[1] 6  
> A[2,] #koko rivi  
[1] 4 5 6  
> A[,c(1,3)] #kaikki rivit, sarakkeet 1 ja 3  
      [,1] [,2]  
[1,]    1    3
```



```
[2,]    4    6
```

- Kertolasku:

```
> #Kertolasku:
```

```
> A*A #Alkioittainen
```

```
      [,1] [,2] [,3]
```

```
[1,]    1    4    9
```

```
[2,]   16   25   36
```

```
> t(A) %*% A #Matriisien kertolasku (t ottaa transpoosin)
```

```
      [,1] [,2] [,3]
```

```
[1,]   17   22   27
```

```
[2,]   22   29   36
```

```
[3,]   27   36   45
```

- Listat:

- Tietorakenne, jonka sisällä muita objekteja

```
> l1=list(v1,v2)
```

```
> l1
```

```
[[1]]
```

```
[1] 2 2 3
```

```
[[2]]
```

```
[1] TRUE FALSE TRUE
```

- Listaan viittaaminen

```
> #Useampaan alkioon viittaus:
```

```
> l1[c(1,2)]
```

```
[[1]]
```

```
[1] 2 2 3
```

```
[[2]]
```

```
[1] TRUE FALSE TRUE
```

```
> #Lista ilman ensimmäistä alkiota:
> l1[-1]
[[1]]
[1] TRUE FALSE TRUE

> #Yksittäiseen elementtiin viittaus kaksilla hakasuluilla:
> l1[[2]]
[1] TRUE FALSE TRUE
> #Muokkaus:1
> l1[[1]][1]=100

- Nimetyt listan jäsenet:
> em=list(pituudet=c(175,169,181),painot=c(80,65,85))
> em
$pituuudet
[1] 175 169 181

$painot
[1] 80 65 85
```

```
> em$pituudet  
[1] 175 169 181  
> em["painot"]  
$painot  
[1] 80 65 85
```

- Listoja ei käytetä säilömään vektoreja tai matriiseja matemaattisessa mielessä
- Useat funktioiden palautukset tulevat lista-tietorakenteina

- Data framet

- Yleinen datan säilömuoto: Sarakkeissa muuttujat, riveillä havainnot

- Käytännössä lista saman pituisia vektoreita

```
> #Luo kolmen muuttujan havaintovektorit:
```

```
> weight=c(160,175,159)
```

```
> bloodtype=c("A+","A-","B+")
```

```
> female=c(TRUE,FALSE,TRUE)
```

```
> #Tallenna data frameen:
```

```
> df=data.frame(weight,bloodtype,female)
```

```
> df
```

```
  weight bloodtype female
1    160         A+   TRUE
2    175         A-  FALSE
3    159         B+   TRUE
```

- Viittaus taulukon alkioihin (huom! Rivit ja sarakkeet voivat olla nimettyjä)

```
> df[1,2] #Viittaus indeksien avulla
```

```
[1] A+
```

```
Levels: A- A+ B+
```

```
> df[1,"bloodtype"] # Viittaus nimen avulla
```

```
[1] A+
```

```
Levels: A- A+ B+
```

- Esikatselu ja taulukon tarkastelu

```
> head(df) # Taulukon alku
```

```
  weight bloodtype female
1    160         A+   TRUE
2    175         A-  FALSE
3    159         B+   TRUE
```

```
> #Avaa koko taulukko:
```

```
> View(df)
```

- Data framen käsittelyyn on olemassa monia tehokkaita paketteja (esim. reshape2, dplyr, data.table), jotka mahdollistavat datan hakemisen, ryhmittelyn ja muokkaamisen tietokannan tapaan

R:n syntaksi - Funktiot ja paketit

- **Funktiot** toimivat samoin kuin Matlabissa:
 - Voi tehdä omia funktioita samaan tapaan kuin pythonissa:

```
> oma_funktio=function(arg1,arg2)
+ {
+   palautus1=c(arg1,arg2)
+   palautus2=arg1*arg2
+   return(list(palautus1,palautus2))
+ }
> palautuslista=oma_funktio(1,100)
> palautuslista
[[1]]
[1] 1 100

[[2]]
[1] 100
```

- **Paketit** ovat ilmaisia lisäosia erilaisiin spesifeihin tarkoituksiin (tilastollinen analyysi, regressio, neuroverkot...)
 - Pakettien lataaminen ja käyttöönotto
 - > `install.packages("stats")` # Kun lataat paketin ekan kerran
 - > `library(stats)` # Kun otat jo ladatun paketin käyttöön
 - RStudio hoitaa pakettien lataamisen edellä mainittujen komentojen avulla
 - Paketteihin löytyy yleensä hyvät dokumentaatiot googlaamalla
 - Myös `help("paketin nimi")` toimii

Datan tuonti ja visualisointi

- Lähes kaikkien tiedostotyyppien **tuonti** R:n ja **tallennus** R:llä onnistuu (.xls,.txt,.csv,.json,.mat...)
- Komentoja esim. `read.table`, `read.csv`, `read.xlsx`... (osa vaatii paketin latauksen)
- Esimerkki: ladataan simuloitua dataa kuvitteellisen kurssin tenttipisteistä opiskeluun käytetyn ajan suhteen:

```
> opiskeludata=read.table("opiskeludata.txt",header=TRUE)
```

```
> head(opiskeludata) # Tutkitaan dataa
```

```
  opiskelutunnit tenttipisteet
1          2.477782          0.00000
2         21.587104         66.70898
3         29.884683         95.12357
4         10.636029         69.11182
5         29.137764        100.00000
6         10.393463         40.32418
```

```
> summary(opiskeludata) #Yhteenveto datasta
```

opiskelutunnit	tenttipisteet
Min. : 0.2941	Min. : 0.00
1st Qu.: 6.3613	1st Qu.: 32.74
Median :13.9828	Median : 53.95
Mean :14.5321	Mean : 55.65
3rd Qu.:22.7271	3rd Qu.: 82.30
Max. :29.8847	Max. :100.00

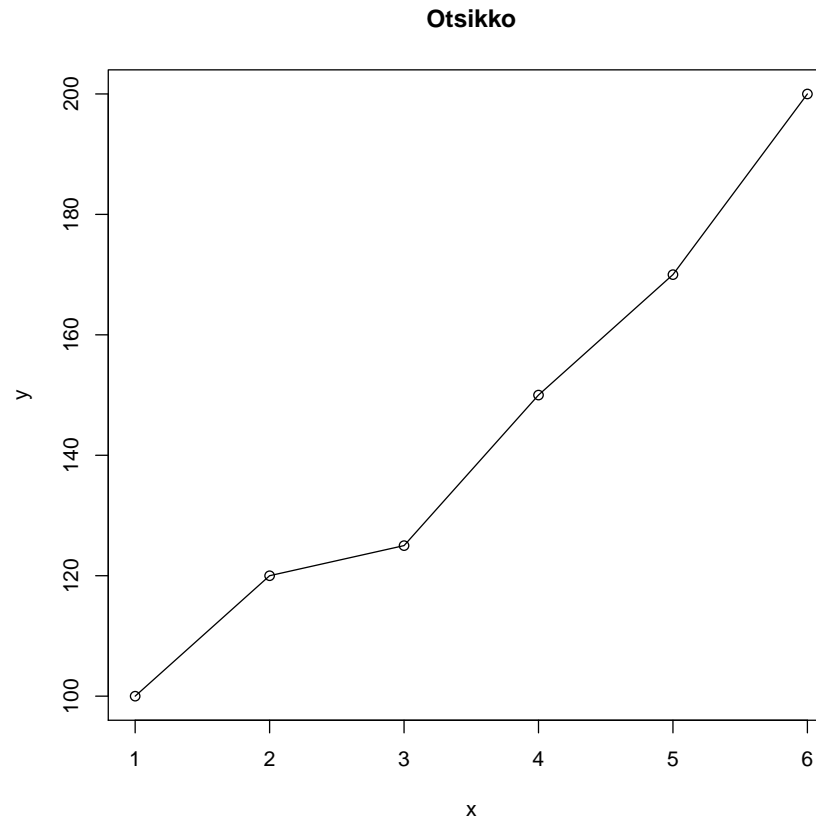
- Taulukoita voi tallentaa esim. `write.table()`-komennolla
- R:n oma tiedostomuoto datan tallentamiseen `.RData`
- `.RData`-tiedostoja voi tallentaa ja ladata `save()` ja `load()`-komennoilla

- Visualisointiin on monia komentoja, esim. `plot()` toimii:

```
> x=c(1,2,3,4,5,6)
```

```
> y=c(100,120,125,150,170,200)
```

```
> plot(x,y,'o',xlab="x",ylab="y",main="Otsikko")
```



Demo : Lineaarinen regressiomalli

- Sovitetaan lineaarinen regressiomalli, jossa opiskelutunnit (x) on selittävä muuttuja ja tenttipisteet y on selitettävä muuttuja siten, että

$$y = \alpha x + \beta + \epsilon \quad (1)$$

jossa α on regressiosuoran kulmakerroin, β vakiotermin ja ϵ mallin virhetermi.

```
> malli=lm(tenttipisteet~opiskelutunnit,data=opiskeludata)
> summary(malli)
```

Call:

```
lm(formula = tenttipisteet ~ opiskelutunnit, data = opiskeludata)
```

Residuals:

Min	1Q	Median	3Q	Max
-32.479	-8.709	1.016	7.187	25.148

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	12.0602	3.3225	3.63	0.000687	***
opiskelutunnit	2.9995	0.1944	15.43	< 2e-16	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.37 on 48 degrees of freedom

Multiple R-squared: 0.8322, Adjusted R-squared: 0.8287

F-statistic: 238.1 on 1 and 48 DF, p-value: < 2.2e-16

```
> tunnit=data.frame(opiskelutunnit=c(0,30))
```

```
> ennustetut_pisteet=predict(malli,tunnit) # Ennustetaan uusia arvoja
```

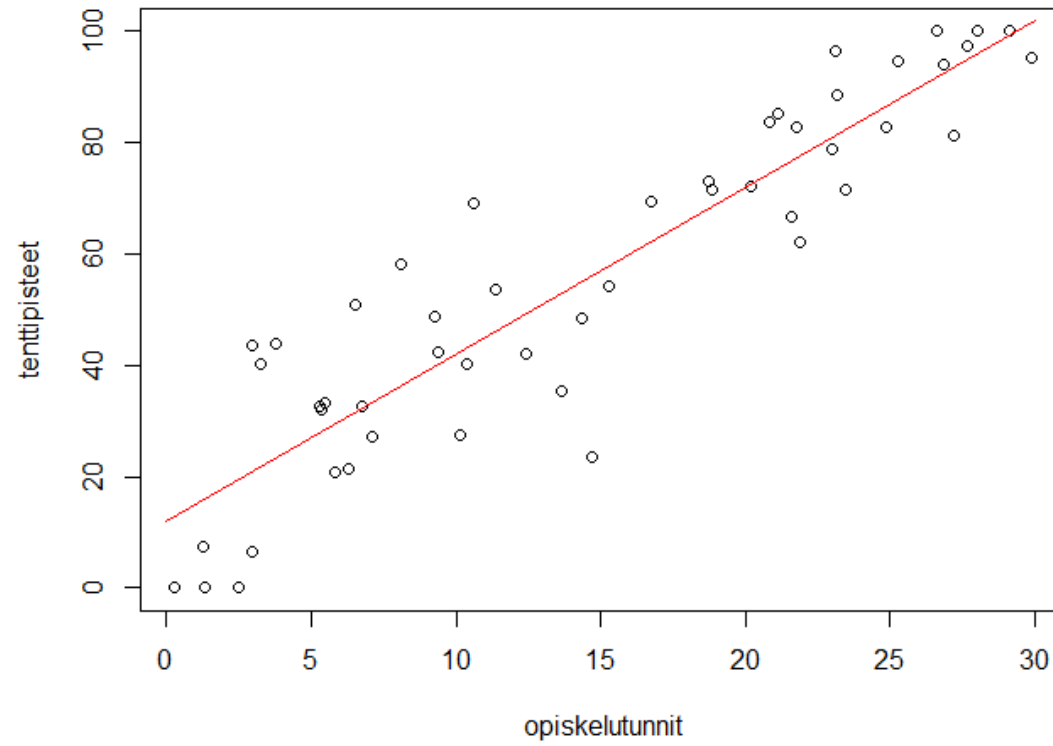
```
> ennustetut_pisteet
```

```
      1      2  
12.06023 102.04662
```

- Visualisoidaan datapisteet ja tulokset:

```
> plot(opiskeludata)
```

```
> lines(tunnit$opiskelutunnit,ennustetut_pisteet,col="red")
```



Tehtävä A: tutustuminen R:n



- Tehtävässä harjoitellaan muuttujien luomista, eri datatyyppien käyttöä sekä R:n perusfunktioita. Saat apua funktioiden käyttöön kirjoittamalla konsoliin `help("funktion nimi")`.

1. Luo vektorit $v=[1,2,3,4]$ ja $s=["1","2","3","4"]$ sekä matriisit

$$A = \begin{bmatrix} 1 & 2 \\ 9 & 10 \end{bmatrix} \text{ ja } B = \begin{bmatrix} 4 & 5 \\ 7 & 8 \end{bmatrix} \text{ ja kokeile matriisien kertolaskua ja}$$

yhteenlaskua. Luo myös lista l_1 , joka koostuu vektorista v sekä loogisesta vektorista `[TRUE,FALSE,FALSE,TRUE]`.

- ✎ Mitä tapahtuu, kun ajat komennon `as.numeric(s)`?
- ✎ Mikä on matriisien A ja B matriisitulo, ja miten saat sen laskettua?
- ✎ Millä syntaksilla voit muuttaa listan l_1 jälkimmäisen alilistan viimeisen alkion arvoksi `FALSE`?
- ✎ Millä kahdella tavalla voit luoda vektorin, jossa on luvut `1, 3, ..., 99` (vinkki: `seq()`-funktio)?

2. Muuta luomasi lista l_1 data frameksi. Kokeile data framelle funktioita `head`, `names` ja `summary`.
-  Mitä eroa on listalla ja data framella? Miten voit viitata listan ensimmäisen alilistan alkoihin? Entä data framen ensimmäiseen muuttujaan (sarakkeeseen)?
3. Luo satunnaisluvusta koostuva matriisi komennolla `dat <- matrix(rnorm(n = 15, mean = 100, sd = 30), nrow = 3, ncol = 5)`. Laske matriisin rivien keskiarvot ensin käyttämällä `for`-silmukkaa ja sen jälkeen `apply`-funktion avulla.
 Liitä molempien toteutusten R-koodi sekä tulostukset vastaukseesi.

Tehtävä B: Useamman selittäjän lineaarinen regressio



- Tehtävänä on ennustaa talojen mediaanihintaa valittujen muuttujien perusteella käyttämällä lineaarista regressiota.
- Usean selittäjän lineaarinen regressiomalli on muotoa

$$y = w_0 + w_1x_1 + w_2x_2 + \dots + w_nx_n + \epsilon \quad (2)$$

jossa y on mallin selitettävä muuttuja, x_i ovat selittäviä muuttujia, w_i regressiokertoimet ja ϵ virhetermi.

- Käytössä on ns. Boston Housing-aineisto, jonka saa ladattua ottamalla käyttöön MASS-paketin. Tämän jälkeen datasetti löytyy Boston-muuttujasta.
- Jaetaan alkuperäinen aineisto opetus- ja testiaineistoihin (training & test set). Opetusaineistoa käytetään mallin rakentamiseen ja testiaineistoa mallin ennustuskyvyn arviointiin.

Tehtävä B: Useamman selittäjän lineaarinen regressio

1. Tutustu ensin aineiston muuttujiin selityksineen täällä:
<https://www.kaggle.com/c/boston-housing>.
 Minkä muuttujien uskot parhaiten selittävän talojen mediaanihintoja?
2. Jaa aineisto opetus- ja testiaineistoihin siten, että opetusaineiston koko on n. 80% koko aineiston koosta. Voit käyttää apuna esim. `sample()`-komentoa.
3. Tutki aineistoa esimerkiksi `names-` ja `summary-`muuttujien avulla. Piirrä hajontamatriisi opetusaineistostasi (`plot(opetussetti)`), ja tutki, mitkä tekijät näyttävät sen perusteella korreloivan parhaiten talojen mediaanihinnan kanssa. Piirrä myös korrelaatiomatriisi opetusaineistosta `cor()`- ja `heatmap()`-komentojen avulla.
 Liitä piirtämäsi korrelaatiokuva (`heatmap`) vastauksiisi. Mitkä muuttujat korreloivat talon mediaanihinnan kanssa parhaiten?

4. Valitse mallisi selittäviksi muuttujiksi `rm`, `ptratio` ja `lstat`. Luo näiden muuttujien avulla lineaarinen regressiomalli selittämään talojen mediaanihintoja käyttäen ainoastaan opetusaineistoa.

👉 Mitkä ovat luomasi mallin regressiokertoimet?




5. Tutkitaan mallin ennustuskykyä testiaineiston avulla. Laske mallin ennusteet y_{malli} testiaineistolle, ja vertaile mallin ennustamia hintoja testiaineiston todellisiin hintoihin $y_{\text{todellinen}}$ laskemalla keskimääräinen absoluuttinen virhe (*mean absolute error* (MAE)) seuraavasti:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_{\text{todellinen}}^i - y_{\text{malli}}^i| \quad (3)$$

👉 Mitä saat virheen arvoksi?

👉 Miten arvioisit mallin ennustuskykyä? Miten mallin tarkkuutta voitaisiin parantaa? Mainitse ainakin kaksi asiaa.

Kotitehtävä: Lineaarinen energiankulutusmalli R:llä

- Toteuta 3. kierroksen lineaarinen energiankulutusmalli (A-tehtävä) R:llä.
 -  Plotatessa voit piirtää datapisteet palloina ja luottamusvälit jatkuvana viivana. Muista liittää piirtämäsi kuva vastaukseen.
 - Katso demo 1:stä apua lineaarisen mallin luomiseen ja ennusteiden laskemiseen. Muista, että `predict`-komennolle annettavassa data framessa on oltava samanniminen sarake, kuin mallisi selittävä muuttuja on nimeltään.
 - Käytä luottamusvälien laskemiseen `predict`:in argumentteja `interval="prediction"` ja `level`. Googlaa tarvittaessa luottamusvälin laskeminen `predict`-komennolla R:llä.
 -  Kirjoita lähdekoodin kommentteihin huomioita koodin eroista ja samankaltaisuuksista Matlabin kanssa.
 -  Liitä vastauksiin kommentoitu lähdekoodi.