

R: Nano Guide

Pekka Pere

Department of Mathematics and Systems Analysis
Aalto University

10.10.2023


1. R

- ▶ R is a programming language which is especially suitable for statistical analyses and drawing tailored graphs.
- ▶ Ross Ihaka and Robert Gentleman (University of Auckland, New Zealand) composed R as an open-source equivalent to the commercial S programming language. The name R acknowledges S and refers to the initials of the authors of R. John Chambers is the creator of the S language and also a key developer of the R language. Numerous other researchers have contributed to R. R foundation (Wien, Austria) steers the development of R.
- ▶ Version 1.0 was published in February 2000.
- ▶ R is free! You can use it also after graduation.
- ▶ R is so widely used that it can be regarded as the *lingua franca* of statistics.

More information is on The Comprehensive R Archive Network (CRAN) web page <https://www.r-project.org/>. CRAN is a huge pool of everything related to R.

2. Installation of R

Windows:


- ▶ Go to web page <https://ftp.acc.umu.se/mirror/CRAN/>.¹
- ▶ Click the chain of links *Download R for Windows* → *install R for the first time* → *Download R <version number such as 4.3.1> for Windows* and the icon *Save file*. Installation software for R is now downloaded to your computer (takes a short while).
- ▶ Open the downloaded file R-“version number”-win.exe (menu at the upper-right-hand-corner of your web browser or Downloads from File Management). Accept installation, the default values of the emerging options, and finally tick the choices for the creation of icons to the desktop and shortcut. R software will then be installed.
- ▶ The welcome screen of your computer has now the -logo.

¹Links for Mac and Linux operating systems are at the web page as well.

3. Using R

Notation:

- ▶ command prompt `>`
- ▶ assignment operator `<-`
- ▶ *n*th item returned by R `[n]`
- ▶ continuation prompt of R (example below) `+`
- ▶ comment (example below) `#`
- ▶ response of R (notation of this guide) `##`
- ▶ text deleted from response of R (notation of this guide) `- - .`

Double click the  logo to launch R. The Console Pane opens. R is used by writing a command on the line starting with the command prompt `>` and executing the command by pressing the Enter key ↵.

At its simplest, R is a (symbolic) calculator:

```
> 1+2
## [1] 3
> a <- 1
> b <- 2
> a
## [1] 1
> b
## [1] 2
> a+b
## [1] 3
```

First, $1 + 2 = 3$ was calculated. R's response was `[1] 3`. The 1st item of R's response is indicated by `[1]` — here “3”. Two hashtags `##` are the convention of this guide to indicate response of R. R does not return double hashtags.

Next, numerical values 1 and 2 were assigned as values of `a` and `b`, respectively, with the assignment operator `<-`. Commands `a` and `b` returned the values of `a` and `b`. Command `a+b` returned the sum or 3.

Many commands return multiple items. R uses square brackets to designate the first item of each line. The sequence command `seq(1,30,1)` produces figures 1 to 30 (- - indicates a deleted section of the response of R):

```
> seq(1,30,1)
## [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 - -
## [26] 26 27 28 29 30
```

The response is composed of figures 1, ..., 30. The first item on the second line is the 26th item of the response. It is the message of the line label [26] in the response of R. The R convention of labeling lines helps in deciphering the output. Typically the item number is not perceived as easily as in the present example.

Note! In the following, command prompt `>` is not anymore written in front of the commands.

Data can be imported to R in many ways. Few observations of a single variable are conveniently imported with the concatenate command `c(·)`:

```
x <- c(49,35,32,39,45)
x
## [1] 49 35 32 39 45
y <- c(45,37,30,
## +
32,40)
y
## [1] 45 37 30 32 40
```

The first set of observations was assigned to `x`. Next the contents of `x` was checked or looked at. Next a second set of observations was assigned to `y`. Above `+` is the continuation prompt which asks the user to complete the command. The command was completed on the following line.

Note! Continuation prompt `+` does not sum anything!

Large data sets are more easily imported to R by other commands such as `read.table`. It is not exemplified here.

R is case sensitive:

```
X
## Error: object 'X' not found
x
## [1] 49 35 32 39 45
```

R does not recognise X because it was not defined above. Only the lower case x was.

Note1! Try the commands above! One learns computing by experimenting.

Note2! Commands are called functions in R. They are not functions in the mathematical sense. The term command is used in this guide.

R includes numerous commands for analysing data. Sample mean, sample standard deviation, and other statistics are obtained with the commands `mean(.)`, `sd(.)`, and `summary(.)`:

```
mean(x)
## [1] 40
sd(x)
## [1] 7
summary(x)
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    32     35     39     40     45     49
```


The sample mean and standard deviation are 40 and 7, respectively. The smallest and largest observations are 32 and 49, respectively. The median is 39, and the 1st and 3rd quartiles are 35 and 45, respectively.

Command `cor(x, y)` returns the sample correlation of the variables `x` and `y`:

```
cor(x,y)
## [1] 0.8725105
```

Detailed information on the calculations above can be obtained with the commands `help(mean)`, `help(sd)`, `help(summary)`, and `help(cor)`.

R handles smoothly a legion of distributions such as the Normal, χ^2 , `t`, and `F`. Commands `pnorm(x)` and `qnorm(x)` return the value of the cumulative distribution function and the corresponding quantile at `x`, respectively ($p \leftrightarrow$ *probability*; $q \leftrightarrow$ *quantile*). The corresponding commands for the Normal distribution with mean `m` and standard deviation `s` are `pnorm(x, m, s)` and `qnorm(x, m, s)`, respectively.

Examples:

```
pnorm(-1.964)
## [1] 0.02476505
qnorm(0.02476505)
## [1] -1.964
pnorm(-1.964, -2, 2)
## [1] 0.5071806
qnorm(0.5071806, -2, 2)
## [1] -1.964
```

Note! Usually the measure of dispersion used to characterise the Normal distribution is variance (σ^2) instead of the standard deviation (σ): $N(\mu, \sigma^2)$.

The corresponding commands for the distributions χ^2 , t, and F are `pchisq(x,df)`, `pt(x,df)`, `pf(x,df1, df2)`, and `qchisq(x,df)`, `qt(x,df)`, `qf(x,df1, df2)`, respectively. Here `df`, `df1`, and `df2` refer to the degrees of freedom of the distribution.

Many commands, separated by character “;”, can be executed from a single line:

```
pt(-1.964,20); qt(0.03179091,20)
## [1] 0.03179091
## [1] -1.964
```

4. Good conventions

It is often a good idea to execute the commands from a file and save the file: Choose File (upper-left corner of Console Pane) and New script. The text editor of R is invoked. Compose the commands you need on subsequent lines, and execute them by pressing Ctrl-R on each line. Save the file from File and Save as.

The commands can be run again from File and Open script. This way it is easy to repeat and check earlier analyses or tune them for new tasks. Or you can share the file so that others can replicate and check your analyses.

R ignores the lines starting with #. Explanations of the code can be added after the hashtag:

```
x <- c(49,35,32,39,45)
# y is calculated in "Meaning of life" by Albert Nero:
y <- c(45,37,30,32,40)
```

It is essential that you comment your code!

5. Correcting and canceling

An erroneous command can be fixed by pressing the ↑ key:

```
x <- c(49 35 32 39 45)
## Error: unexpected numeric constant in "x <- c(49 35"
x <- c(49,35,32,39,45)
x
## [1] 49 35 32 39 45
```

The arrow key brings the erroneous command `x <- c(49 35 32 39 45)` back on the command line. It can then be fixed by adding commas in between the figures. (Done on the 3rd row above.)

It is especially easy to correct an erroneous command in the text editor of R because the command prevails displayed after executing it or even a number of other commands.

If R asks you to complete a command but you do not want to or do not know how to do it, cancel the command with the Escape key Esc:

```
y <- c(45,37,33,
## +
```

If you now realise that you have typed the third figure incorrectly (33 instead of 30) then cancel the command by pressing Esc.

6. On programming and packages

Elementary use of R does not require programming skills. Much can be accomplished with simple commands and scripts of R code.

Packages are available for more advanced tasks. About 20 000 packages exist (<https://cran.r-project.org/web/packages/>). E.g. statistical methods tailored for psychologists are collected into package `psych`. It can be installed and activated with the commands `install.packages("psych")` and `library(psych)`. At the time of downloading the first package R asks which server should be used for downloading the package. A suitable choice is e.g. the CRAN mirror in Sweden.

Ready-made scripts for complicated endeavors can be found from R guides or the Internet. You can copy and tune the scripts for your purposes. Remember to cite the author of the code. An example is drawing an enhanced graph. For this purpose installing a package (e.g. `ggplot2`) is an option as well.

7. Quitting and citing

To quit your R session, run `quit()`, in short `q()`:

```
q()
```

Answer No to the appearing question(s). Your session is over.

If you utilise R in your thesis or other aspirations then tell it.

Executing `citation()` assists citing:

```
citation()
```

```
## To cite R in publications use:
```

```
##
```

```
## R Core Team (2023). _R: A language and environment for statistical  
## computing_. R Foundation for Statistical Computing, Vienna, Austria.  
## <https://www.R-project.org/>.
```

```
- -
```

```
## We have invested a lot of time and effort in creating R, please cite it  
## when using it for data analysis. See also 'citation("pkgname")' for  
## citing R packages.
```

8. Guides and advice

Your command of R can be upgraded with R guides such as

- ▶ W. J. Braun and D. J. Murdoch (2021): A First Course in Statistical Programming with R, 3rd edition. CUP.
- ▶ M. J. Crawley (2013): The R Book, 2nd edition. Wiley.
- ▶ R. I. Kabacoff (2022): R in Action, 3rd edition. Manning.
- ▶ Appendix of C. Heumann, M. Schomaker, and Shalabh (2016): Introduction to Statistics and Data Analysis. With Exercises, Solutions and Applications in R. Springer.

Note1! Read guides selectively. It is usually a good idea to read the beginning chapters, and from then on to consult the guide as the need arises.

Note2! Do not just read the guides! You have to experiment with code to learn R.

Internet is a great source to find advice on R.