

Getting Started with R

Getting R To work on the assignments you will need a computer with R installed. During the computer sessions, campus computers are available on which R is installed by default (available in My Work Space). But we highly recommend that you also install it to your own laptop or desktop computer, so that you can continue working with R where and whenever you please. R and RStudio are free and available for Windows, Mac OS, and Linux computers.

To install R, visit the “Comprehensive R Archive Network” (CRAN) website¹. At the top of the main CRAN page there are links to download instructions for each of the above operating systems .

Windows users should select the base package and then download a file with a name that looks something like this: “R-2.x.1-win32.exe”. After downloading is complete, running this file should install R. Mac users want a file named “R-2.x.1.dmg” or similar. Linux users will have to find a similar file under the appropriate flavor of Linux, or install the system through a package manager of their choice. Please note, the instructions below are written for a Windows system. While you are absolutely allowed use your Mac or Linux PC, there may be some small differences that you will be responsible for handling.

Getting RStudio We recommend that you use RStudio as your integrated development environment (IDE). Again, campus computers come with RStudio pre-installed, but you can easily install it to any other computer (Windows, Mac OS X, Linux). To do so, visit <https://rstudio.org/>. Click *Download RStudio*, click *Download RStudio Desktop*, click *Recommended For Your System*, download the appropriate file and run it (choose default answers for all questions).

Once RStudio is installed, open it from the desktop icon or the Start-Programs menu; on campus computers, you may have to look under “Physics” in the Programs menu or search the Start Menu.

Console versus scripts R works by typing *commands* into a *console*. The computer executes your commands when you press *enter*; you’ll practice with this below. A particular statistical analysis typically requires the execution of a long list of commands. It is important to store all these commands in a *text file*, so that you can easily redo or correct your analysis. A text file that contains all commands required for an analysis is called a *script*. You can open a new, empty text file in RStudio by clicking *file, new file, R Script*.

During each session, you will create a script (or perhaps multiple scripts) that contains all analyses and calculations of that session. Save your scripts for use in future weeks. Spend some time carefully annotating them with the help of the hash sign (*#*) as explained below. This is your way to learn and to remember *what* you did and *why* you did it. Your scripts will become your personal “cheat sheets”.

¹<http://cran.us.r-project.org/>

Tip 1 Write your name and the date of creation at the top of the script.

Tip 2 In RStudio, you can send a line or multiple lines of your script to the console by selecting the lines and subsequently pressing [Ctrl] + [Enter].

Getting help

There are many ways you can get help.

- You can access the documentation (help files) for any function by typing a question mark followed by the name of the function, or by using the function `help()`. For example:

```
> ?library
```

or

```
> help(library)
```

will both open the documentation for the `library()` function.

- You can also get online help from the CRAN website², in particular by clicking on the *Manuals* link on the left-hand side of the web page. There, you can download the manual named “An Introduction to R” by Venables and Smith; this manual is also available on Blackboard, under *Course Content, Documentation R*.
- Check out Appendix ?? of this guide, which is a Reference Card listing commonly used R functions.
- Honors student Wendy Lichtenauer wrote a manual with very clear explanations in Dutch of most R functions used in this course. This “Handleiding voor R” is included as Appendix ?? in this course guide. We highly recommend that you use it throughout the course.
- Google it. There is a huge community of R users and most of the problems that you may encounter have been discussed and solved in online forums—so use Google to find them. For example: if you want to compute the square root of two, but you do not know the command for that, type “square root function R” in Google.
- Ask questions during the computer session. We’re there to help.
- Carefully study the lecture slides.

Typesetting conventions

In this course guide, we use certain typesetting conventions to distinguish various text elements.

Normal text Normal text (*e.g.*, descriptions and notes) is displayed in a Roman font.

R code Code is set in typewriter font, with a gray background. Lines of code are preceded by a > (“greater than”) sign, like this:

²<https://cran.r-project.org/>

```
> install.packages("binom", dependencies = TRUE) # install package  
      binom
```

Note that function names and other keywords are set in bolt face (as in `install.packages`) and character strings are dark gray (as in "binom").

Annotations On any line of code, any text appearing *after* a hash sign (#) is ignored by R and can therefore be used to annotate the code. In this book, annotations are set in a *gray, Roman, italics* font.

R output Output produced by R is set in a slightly smaller typewriter font with a normal white background.