

# User Guide

We provide a range of sample source codes in the R language in our book<sup>1</sup>. They serve to illustrate how the described methods may be used in R, and you are encouraged to reuse and adjust them for your own work.

We chose R as it is possibly the most powerful free software for statistical analysis, which is used by many researchers across research fields. We recommend using it with the software RStudio, which is a user-friendly interface to R. You can imagine this as R being a car engine, with RStudio being the rest of the car with a steering wheel, pedals dashboard, etc. – something you can easily interact with.

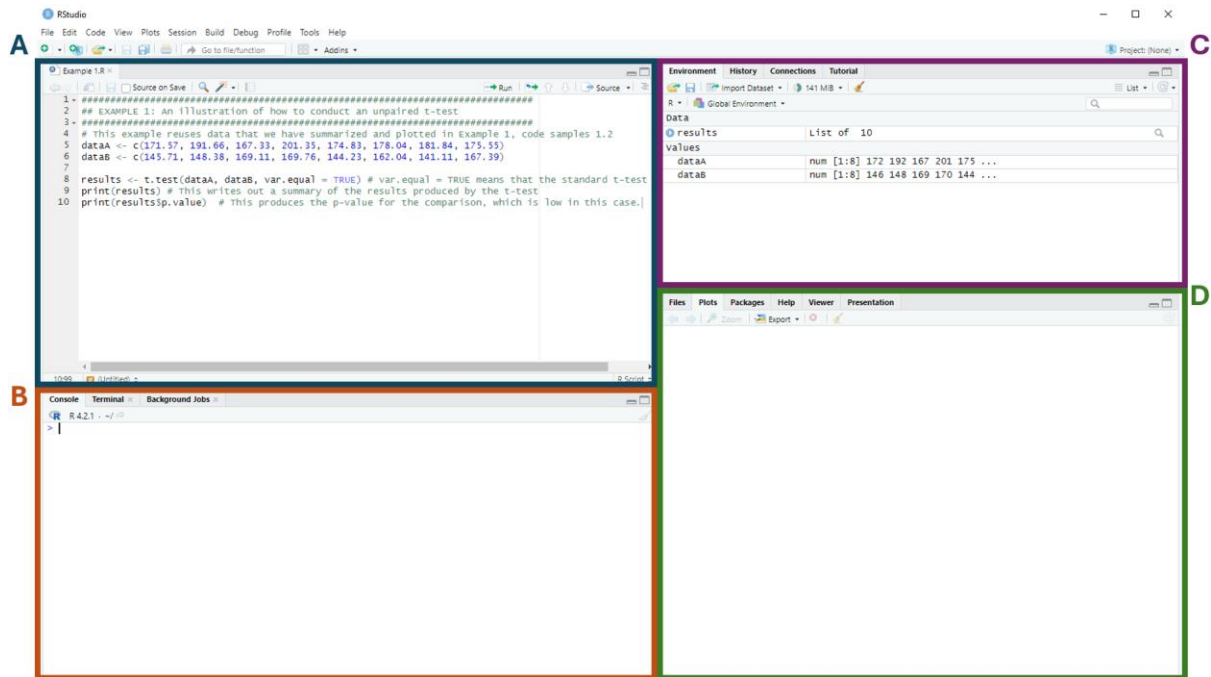
1. We recommend you first install the newest R from here: <https://cran.r-project.org/> (selecting the installer suitable for your operating system).
2. Then you can install RStudio from here: <https://posit.co/download/rstudio-desktop/> – selecting an installer from the provided list.

Opening and running RStudio for the first time may feel overwhelming if you are not used to such software. However, it is not necessary for you to understand all the functionality of RStudio, nor know the whole R language. It is absolutely fine to learn step by step. Chatbots can be a tremendous help when you are not sure how to do something, even though we emphasize that for actual data analysis, one needs to be able to understand the code fully – it is not enough to ask a Chatbot for code and trust it without checking.

Here we explain the very first steps and point you to other resources you can look at afterwards. RStudio on Windows looks as follows (it is similar in other operating systems):

---

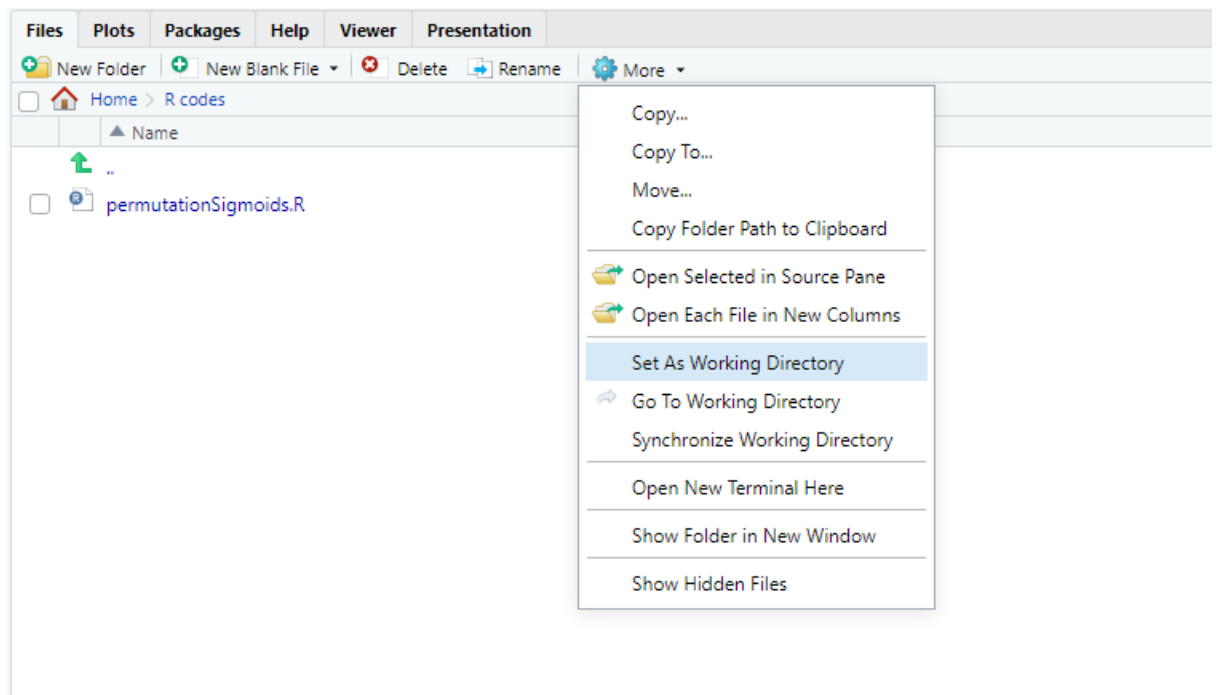
<sup>1</sup> Tomek & Eisner - *Basic Statistics for Life Scientists: A Concise Handbook of Essential Techniques*. ISBN: 978-1-394-28496-2.



It has a set of menus at the top and then four panes (which we highlighted by adding colour rectangles) underneath.

- The top left (A, blue) one shows a *script* – a sequence of commands to be ran by R, line after line. You run the script by selecting all the lines (or just using the Ctrl+A shortcut with the cursor anywhere in the script) and then clicking the Run (or Ctrl+Enter) button in the top right part of this pane. In the example above, this would perform a *t*-test on two sets of observations.
- The bottom left pane (B, orange) is the *console*. Through this, you can interactively run R commands. This is useful if you, e.g., want to look at values of variables in memory, change their values, etc. Running a script is mostly equivalent to copying every line in a script to the console and running it.
- The top right pane (C, purple) contains multiple tabs, with the typically most useful one being *Environment*. This contains an overview of the data stored in the computer memory.
- The bottom right pane (D, green) contains multiple tabs again. Often you will want to use *Plots* (showing plots produced by your code) and *Files* (where you can

navigate through your files to, e.g., find input files for your codes).

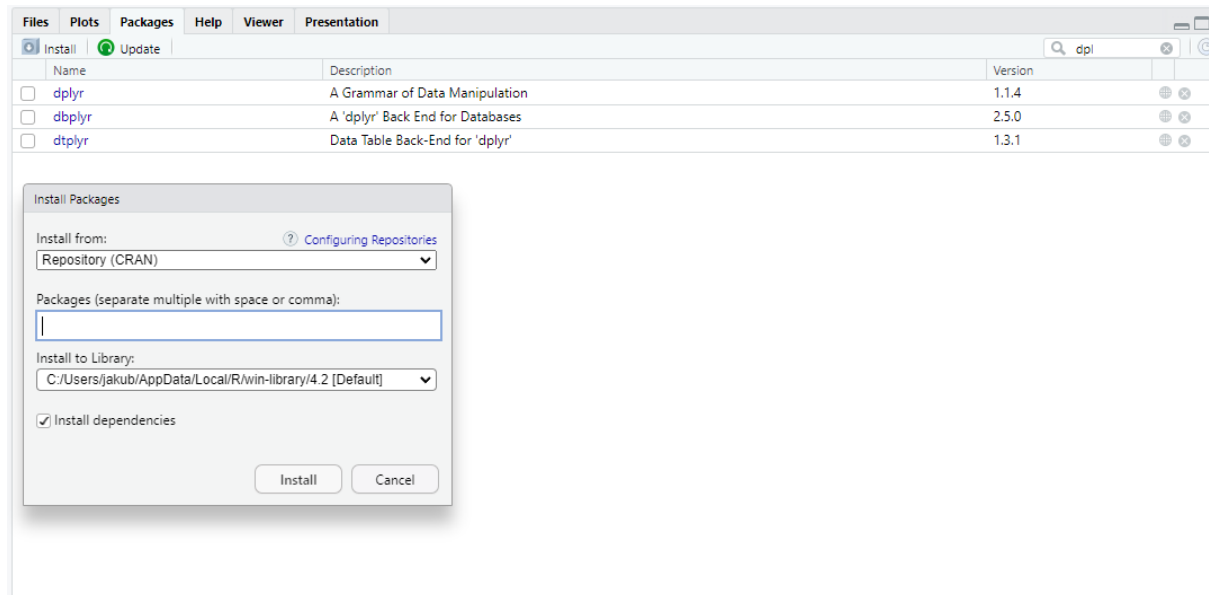


One important concept in R is that of a *working directory* – this is where R will read and store files when you do not explicitly state where it should look. Specifically, if you are saving figures produced by your code, they will be by default stored in the working directory. Often, you may want to set the working directory to the script you are running – you can do this by right-clicking the name of the script in the top left pane, and clicking ‘Set Working Directory’. This ensures any data you read or plots you save are stored in the same folder as your script, keeping your project organized. Additionally, you can navigate through your computer to the folder you want to be the working directory (in the bottom right pane, Files tab), hitting ‘More’ next to the cog icon and clicking ‘Set As Working Directory’ there.

Through the menu at the top of the RStudio window, you can open all the sample scripts we provide by going to File/Open File.

You will see that some of the scripts contain commands like `library(XYZ)` at the start. This tells R to load a library/package with a certain functionality. The basic R can carry out numerous tasks, but having all the possible functionality packaged with it would be impractical. Hence, the basic R can be extended by libraries that provide specialized functions, less common statistical tests, different ways of visualization, etc. Before loading a library X, you first need to install it – this can be done using the command `install.packages("X")` in the console, with X being the name of the library you want. Alternatively, you can go to the tab Packages in the bottom right pane and hit

the Install button, which will open a dialog box like this, where you can specify the library you want to install.



Installing a library is permanent – it stays installed unless you uninstall it. However, loading a package is not permanent – if you close RStudio and then open it again, all the libraries you need must be loaded again using the *library* command.

The sample scripts we provide show a small part of R. To get a better understanding of the language, we recommend **Hands-On Programming with R** (<https://rstudio-education.github.io/hopr/>), **R for Data Science** (<https://r4ds.hadley.nz/>), or **Swirl** (<https://swirlstats.com/>). However, there are plenty of other good resources, be it textbooks, videos, or online tutorials – choose one that works for you.