

Statistical Methods And Data Analysis In Developmental Psychology

Course Preparation

Filippo Gambarota

Updated on 2023-04-23

R

The course materials are created using R (4.3.0). To organize the materials I used the **R Projects** available with R Studio. I highly suggest you to use R Projects that significantly improves your workflow. A brief tutorial is available here <https://r4ds.had.co.nz/workflow-projects.html>.

R Materials

All the slides and extra materials are available on Moodle. During the course and for the exercises we will use some **custom functions** that I wrote for the course. To download and use the function you can download the `utils.R` file from Moodle or from this link(just **right-click > save**)

R Packages

We will use, directly or indirectly (for custom functions), several R packages. You can easily install all packages using:

```
pkgs <- c("effects", "tidyverse", "MASS",  
          "broom", "reshape2", "ggeffects",  
          "performance", "see", "car", "devtools")  
install.packages(pkgs)
```

Pipes

Sometimes in my code you will see a symbol like this `|>`, this is called **pipe**. Sometimes it is also written as `%>%` that is a different pipe coming from the **magrittr** package. The pipe is a very simple way to write R code when you need to apply multiple functions in succession. Practically, the pipe apply a function to an element, for example:

```
# these two are the same  
mean(x)  
  
# to x apply the mean function  
x |> mean()
```

Beyond this silly example, when we need to use multiple nested functions the pipe makes the code more readable:

```
x <- runif(10)

# without pipe
exp(min(round(x, 2)))
```

```
## [1] 1.349859
```

```
# with pipe
x |>
  round(2) |>
  min() |>
  exp()
```

```
## [1] 1.349859
```

Essentially, with the pipe we concatenate multiple functions where implicitly the first argument of the function is assigned to the object before the pipe.