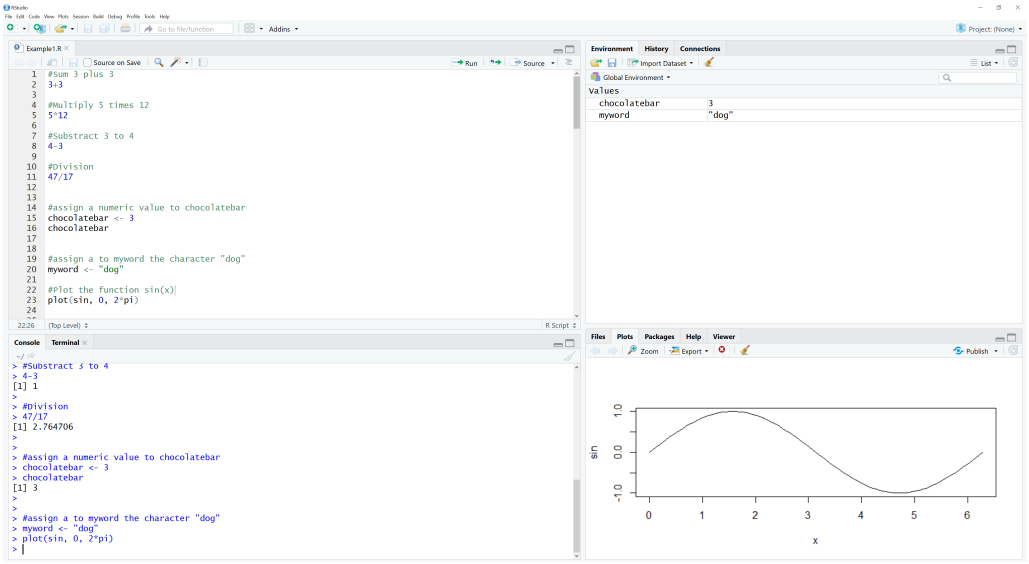
Introduction to R
(part 1)

Daniel Sánchez-Taltavull and Deborah Stroka

October-December 2020

1 Introduction to basic R concepts

- R is a programming language for statistics and is very popular in biology and bioinformatics.
- It is easy to use, easy to learn, and it is very likely that everything you need has already been coded by someone else.
- It can be used to do statistics at all levels: from simple things such as a t-test or a one-way ANOVA, to advanced statistical modelling such as generalized linear models.
- It is an excellent tool for data visualization. For example Principal Component Analysis (PCA) is widely used to represent bulk RNA-seq data, while t-stochastic neighbourhood embedding (tSNE) is used to represent single cell RNA-seq data.
- During the course we will use Rstudio to code in R.



  - The top left panel shows the R script, this is where you write your code.
  - The bottom left panel shows the console, this is where things are run.
  - The top right panel is the environment, this is were you can see your stored variables.
  - The plots and the help tab are in the lower right panel.
- To create a new R script click on file, New file and Create R script.
- You can save your R script by clicking file and save.

- You can load any saved R script by clicking file and Open file.
- Basic operations:

  The input `3+3` gives the output `6`,

  The input `5*12` gives the output `60`,

  The input `35/14` gives the output `2.5`.
- variables are used to store data. Some basic objects are
  - *numerics* are used to store numbers.
    `chocolate <- 3` stores 3 in the variable chocolate
    `soda<-7` stores 7 in the variable soda
    `chocolate+soda` gives `10`.
    WARNING: `chocolate+Soda` gives an error message. R is case sensitive.
  - Scienitific notation can be used for large numbers, `7.9.e23` is the same as `790000000000000000000000`.
  - `Inf` represents a division by 0. `1/0 = Inf`. `NaN` represents not a number, for example `0/0`.
  - A word writen in quotation `"Hello"` is readed as a *character*.
    `word <- "dog"`
  - *Logical* objects contain either `TRUE` or `FALSE`.
    `3<4` will give as an output `TRUE`.
    `7<(4+2)` will give as an output `FALSE`.
  - `TRUE` and `FALSE` can be used for numerical operations. `TRUE` has the value `1`, `FALSE` has the value 0. Therefore, `TRUE+TRUE+FALSE` is 2.
- The type of an object can be checked with the function `class()`
- A function is a set of actions to perform a task. It requires an input and it produces an output. The structure is `NameOfTheFunction(input)`.
  - `mean(c(1,3,8))` gives `4`.
  - `Mean(c(1,3,8))` gives an error message because R is case sensitive.
  - `plot(x=c(1,2,3,4), y=c(1,4,9,16))` plots points at coordinates the coordinates (1,1), (2,4), (3,9) and (4,16).
  - `plot(x=c(1,2,3,4), y=c(1,4,9,16), "l")` makes the same plot as before, but connecting the points with lines.
- Packages contain one or more functions and, after loading them, those functions are available to use.
- A package is installed with the function install.packages()

  `install.packages("ggplot2")` installs the package *ggplot2*.

- The functions of a package are loaded into your R session with the function library

  `library(ggplot2)` loads the package *ggplot2*.

- You can read the help by typing `?` in front of a function.

  `?plot` will open the help of the function `plot`.