1- Introduction to the R language

Alex Sanchez, Miriam Mota, Ricardo Gonzalo and Mireia Ferrer

Statistics and Bioinformatics Unit. Vall d'Hebron Institut de Recerca

Readme

Introduction to R

Outline

- A first contact with R & Rstudio.
  - How does one work with R
- A primer of data import
  - Reading data into R
- A primer of communication
  - R Notebooks and RMarkdown

A first contact with R, Rstudio and the tidyverse

What is R?

- R is a *language and environment* for statistical computing and graphics.
- R provides a wide variety of statistical and graphical techniques, and is highly extensible.
- It compiles and runs on a wide variety of UNIX platforms and similar systems Windows and MacOS.

R PRO's (why you are here!)

- The system is
  - free (as in *free beer*)
  - It's platform independent
  - It is constantly improving (2 new versions/year)
- It is a statistical tool
  - Implements almost every statistical method that exists
  - Great graphics (Examples)
  - Simple reporting tools
  - Also state-of-the-art in Bioinformatics through the Bioconductor Project.
- Programming language
  - Easy to automate repetitive tasks (Example_1.1)
  - Possibility to create user friendly web interfaces with a moderate effort. (Examples)

R CON's

- R is mainly used issuing commands from a console
  - less user friendly than almost any other statistical tool you may know.
- Constantly having new versions may affect our projects
- Not necessarily the best language nor suitable for every existing task

How is R used

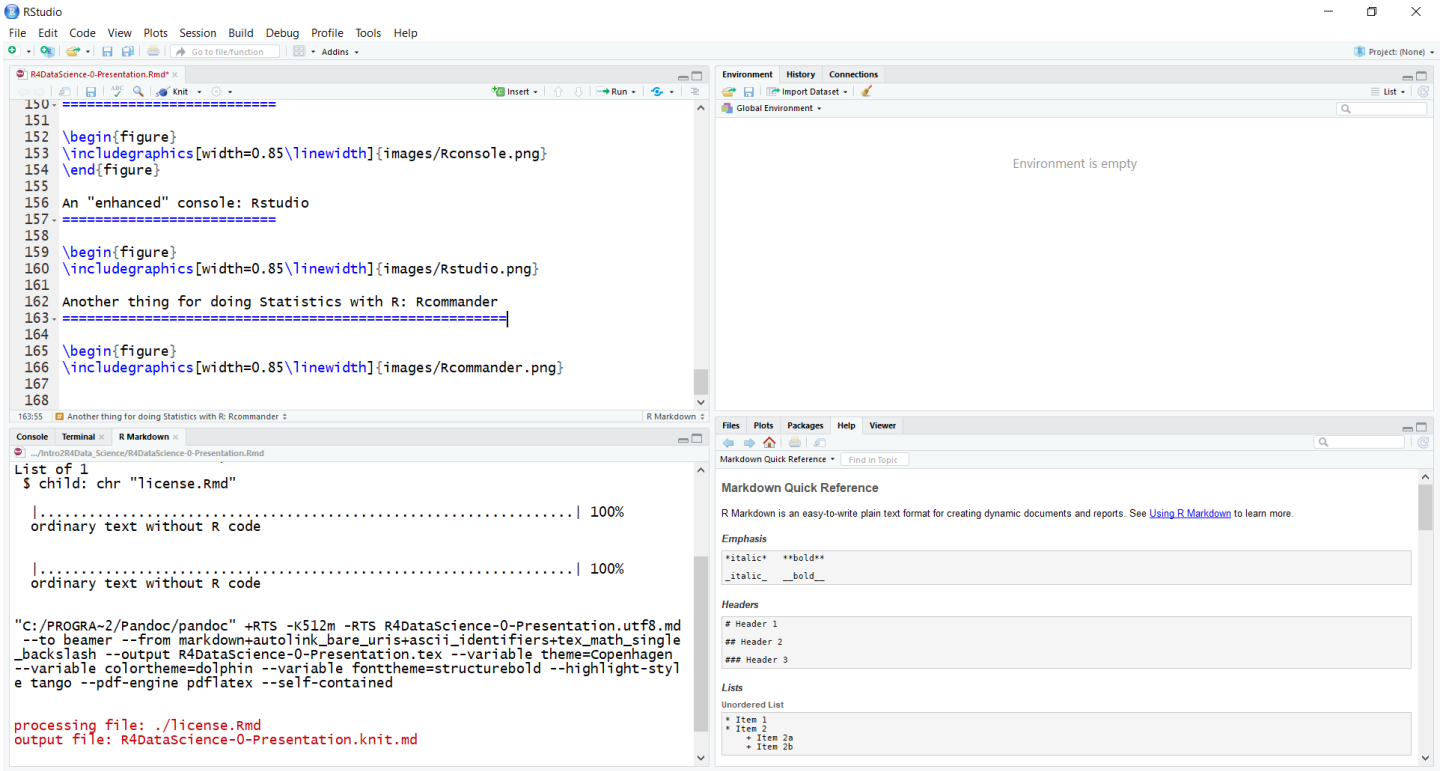
- Traditionally R was used from an Operating System console ("Terminal")
- This is an intimidating approach for many users
- A variety of options exist to decrease the learning curve.
  - Use a supportive development environment such as **Rstudio**
  - Use an interface to Statistical tools, such as **Rcommander** or ::DeduceR** allowing to concetrate an Statistics, not in commands.

A raw R console in linux

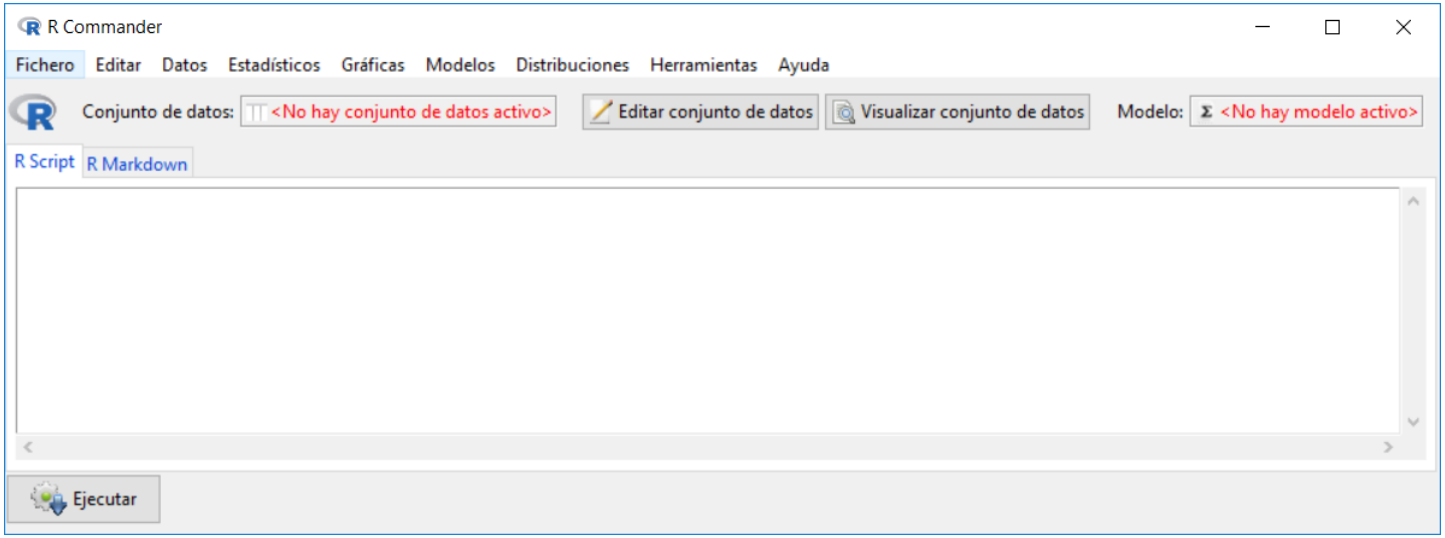
alex@DESKTOP-DH5G1PA: ~

```
> expres <- c(1.02,3.1, 0.8, 1.4,2)
> expres
[1] 1.02 3.10 0.80 1.40 2.00
> logExp <- log(expres)
> sigExp1 <- t.test(logExp)
> sigExp1

        One Sample t-test

data:  logExp
t = 1.6276, df = 4, p-value = 0.1789
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.2763764  1.0594486
sample estimates:
mean of x
0.3915361

>
```

An "enhanced" console: Rstudio



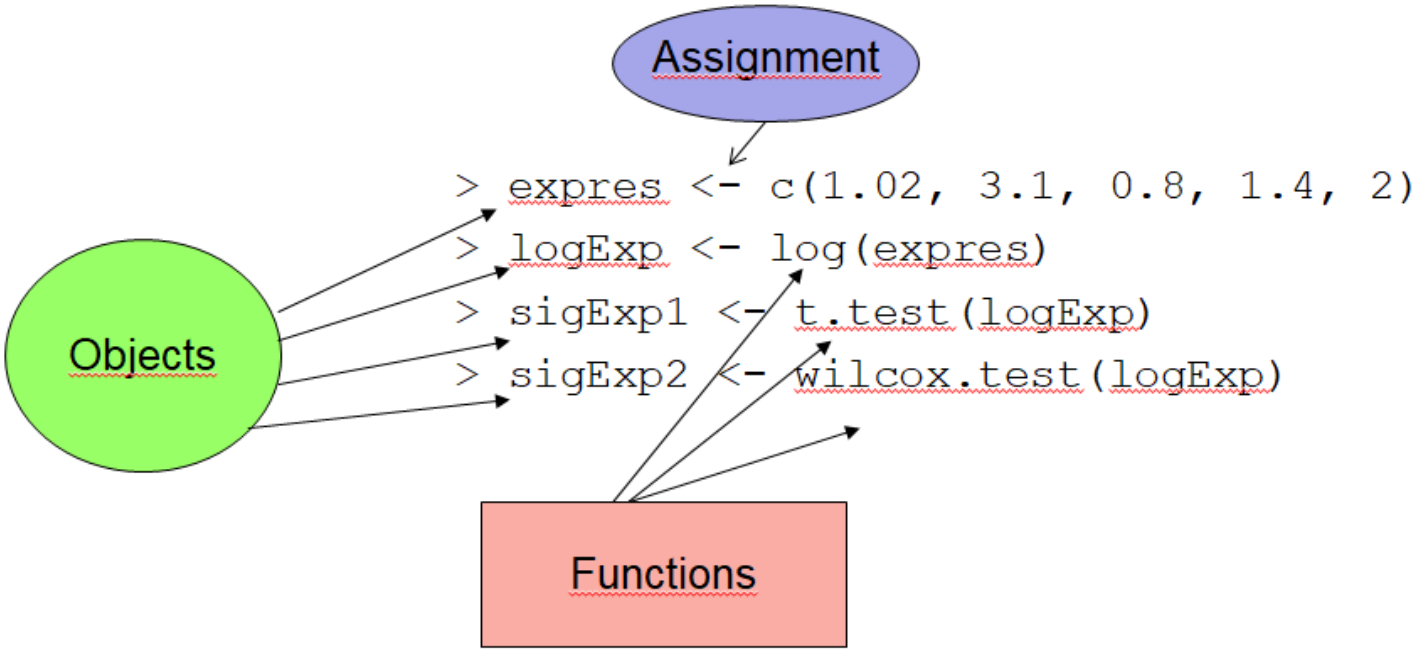
Something that is not a console: Rcommander



Using R

Commands, Objects and Functions

- Shortly, using R consists of
  - Working with *objects* using *commands* and *functions*



Variables and data types

- Data managed in R ...
  - is stored as *variables*
- Variables can be of distinct types
  - Numerical
    - numeric (13.7)
    - int (3)
  - Character
    - "R is cute"
  - Factors
    - A,B,C,D
    - WT, Mut

R packages

- R can be used for many different types of data processing and analysis from distinct fields, besides statistics such as Ecology, Omics Sciences, Psychology etc.
- All these capabilities are not present from the begining because most of them will never be used by most users.
- Instead, thay can be added when needed by
  1. installing and
  2. loading the appropriate packages.

Installing and loading packages

We want to analyze some data using cox proportional hazards model.

```
res.cox <- coxph(Surv(time, status) ~ sex, data = lung)
```

```
Error in coxph(Surv(time, status) ~ sex, data = lung)
: could not find function "coxph"
```

We need to install and load the package before we can use it.

```
install.packages("survival")
library(survival)
res.cox <- coxph(Surv(time, status) ~ sex, data = lung)
```

The tidyverse

- The tidyverse is an opinionated collection of R packages designed for data science.
- All packages share an underlying design philosophy, grammar, and data structures.
- The complete tidyverse collection can be installed with:

```
install.packages("tidyverse")
```

- https://www.tidyverse.org/

Getting data into R

Importing data with Rstudio

Some explanations here

Reading Excel files

Some explanations here

Reading text files

Some explanations here

Interlude: Summarizing data

Some explanations here

Dynamic output with Rmarkdown

Reproducible research with R notebooks

Some explanations here

Dynamic reports with Rmarkdown

Some explanations here