

# Introduction to R and R-commander

Curs d'Estadística Bàsica per a la Recerca Biomèdica



UNITAT  
D'ESTADÍSTICA I  
BIOINFORMÀTICA

GRBIO-UEB-VHIR



## Table of contents

- ▮ **Introduction to R (and S)**
- ▮ **Introduction to Rcmdr**
- ▮ **The Rcmdr menus**
- ▮ **Loading data sets**
- ▮ **Working with data sets**
- ▮ **Using R scripts**
- ▮ **Exporting results**
- ▮ **Additional sources**
- ▮ **Practice**

# Introduction to R

- R is a language and environment for **statistical computing** and **graphics**.
- R was implemented in 1993 by **Robert Gentleman** and **Ross Ihaka** as a free alternative to a commercial software with similar capabilities, known as "the S language".
- R is maintained by the *R Development Core Team*.
- It is available from the R-Project website: <http://www.r-project.org/>
- Pushed by a variety of fields –apart of statistics- such as bioinformatics or Ecology R has become a "de facto" standard in many fields for data exploration, manipulation, modelling and analysis.



# Advantages of using R

- It is free
- Multi-platform (Linux, Mac, Windows)
- Powerful in graphics generation
- Powerful statistical tool (top statistical methods)
- Is always growing in users and functionalities
  - Frequent updates
- Flexible, open source programming language
- Useful for repetitive tasks



# Drawbacks of using R

- It is a statistical "language", that is it is based on "commands" and used best in a console.
  - Partially compensated by GUIs such as Rcmdr and Rstudio
- Not so "user friendly" as SPSS or Graphpad
  - Much more powerful than the 2nd
- Supporting documentation is of variable quality
- Frequent updates
- Community-based: Packages may be different depending on who creates them.
  - Partially solved by the tidyverse

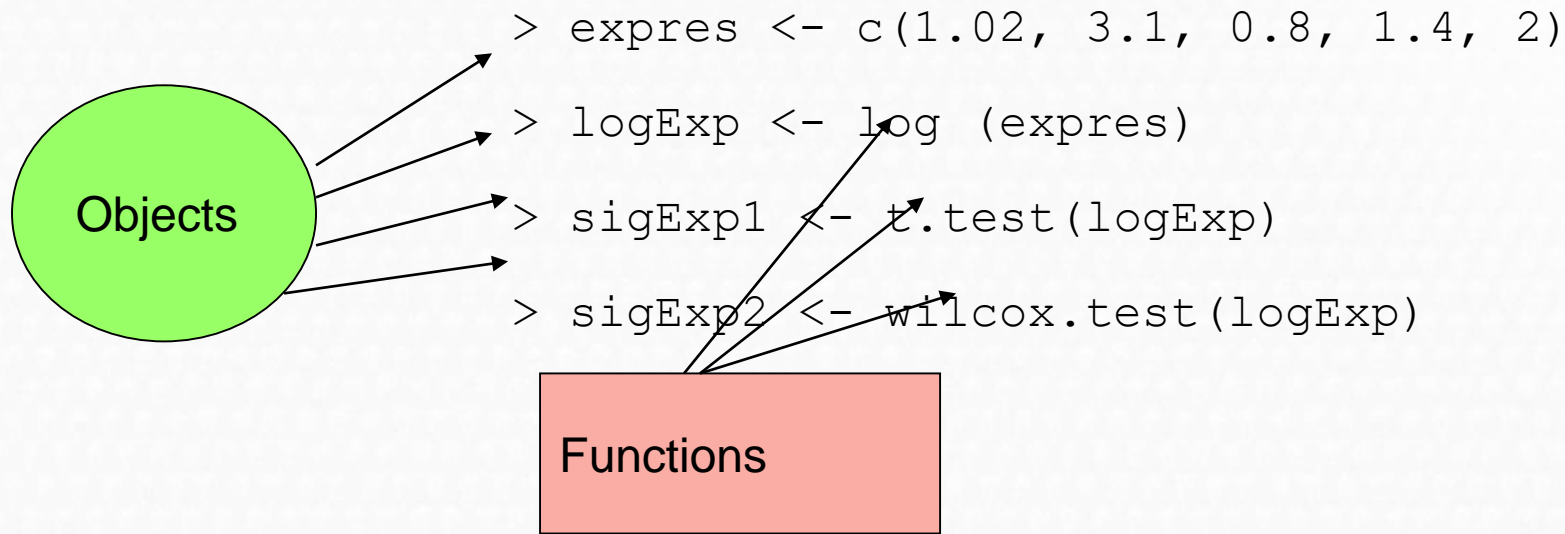
In short, using R consists of:

- Managing the right *objects*
- Using appropriate *functions*.

At the beginning the problema is knowing

- Which type of objects are there
- What can be done with them.

# An example



In short, using R consists of:

- Managing the right *objects*
- Using appropriate *functions*.

At the beginning the problema is knowing

- Which type of objects are there
- What can be done with them.



- R objects can be:
  - Data tables ("datasets"), text, dates,
  - But also more complicated things such as
    - Plots
    - Output of statistical tests
    - Fitted models
- Objects are created
  - Reading data from a file
  - As the result of a computation
  - Assigning them a value

- R functions represent something that can be done with an object :
  - Functions operate with objects
  - Functions can return other objects
- Functions can be
  - Incorporated in R "base"
  - Added to the system using "packages"
  - Created by the user for specific purposes

# Example 2

**# DATA**

```
calcio <-c(11.0, 10.6, 10.5, 10.6, 10.4, 10.2, 9.5,  
           8.2, 7.5, 6.0, 5.0)
```

```
PTH    <- c(0.5, 1.12, 1.23, 1.24, 1.31, 1.33, 2.10,  
           2.15, 2.43, 3.70, 4.27)
```

```
plot(calcio,PTH, main="Hormona Paratiroidea vs [Calcio]")
```

Funciones

```
regres <- lm(PTH ~ calcio) # FIT A MODEL
```

**# RESULTS**

```
summary(regres)
```

```
abline(regres)
```

```
par(mfrow=c(2,2))
```

```
plot(regres)
```

Objetos

# Example 2: results

```
> summary(regres)
```

```
Call:
```

```
lm(formula = PTH ~ calcio)
```

```
Residuals:
```

Min	1Q	Median	3Q	Max
-0.37648	-0.11926	0.08052	0.11177	0.40454

```
Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t )	
(Intercept)	6.88232	0.35283	19.51	1.13e-08	***
calcio	-0.54599	0.03811	-14.33	1.68e-07	***

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.2496 on 9 degrees of freedom
```

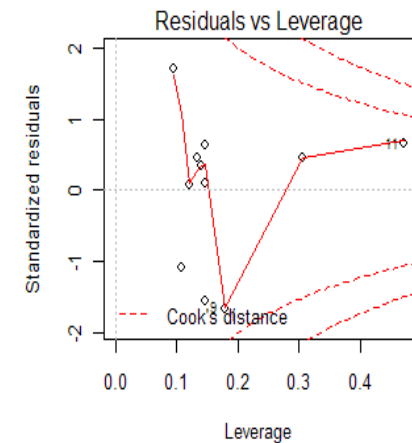
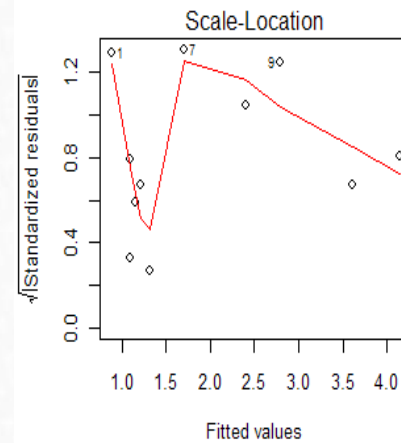
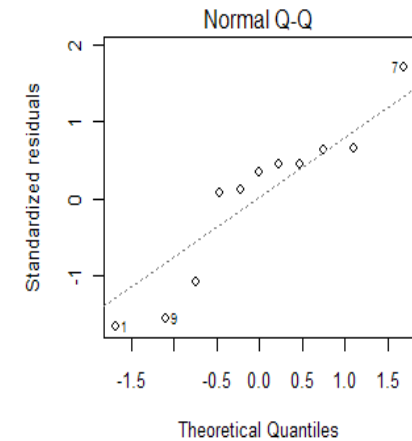
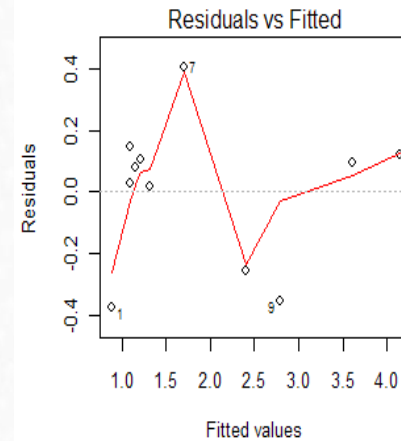
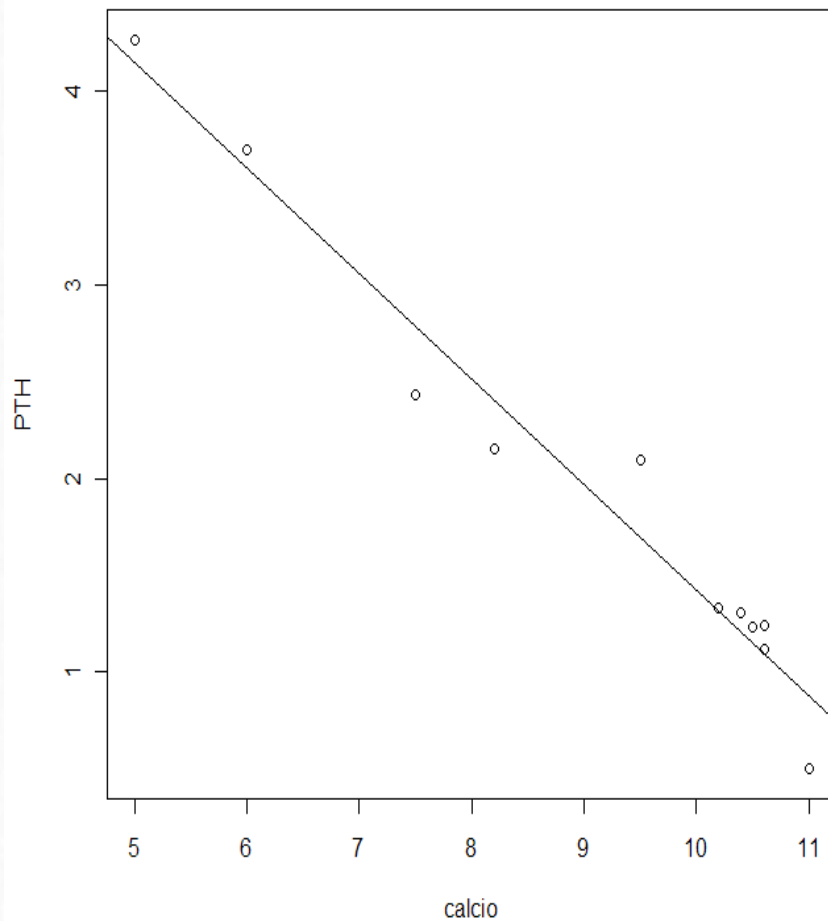
```
Multiple R-squared: 0.958,      Adjusted R-squared: 0.9533
```

```
F-statistic: 205.3 on 1 and 9 DF,  p-value: 1.680e-07
```



# Example 2: plots

Hormona Paratiroidea vs [Calcio]



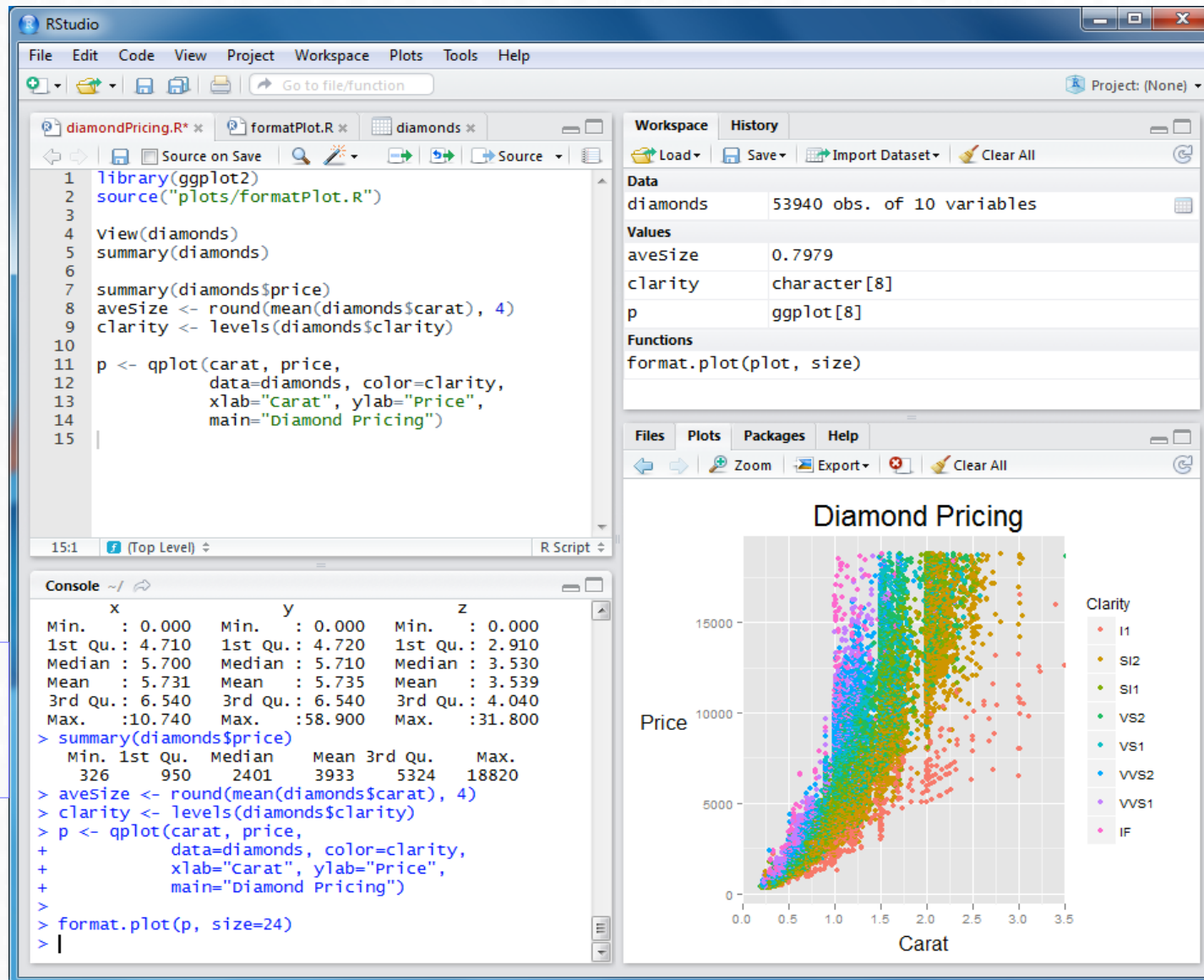
# R GUIs and IDEs

- Using R from the console can have a steep learning curve.
- Simplified with GUIs and IDEs
- Graphical User Interfaces (GUI)
  - Simplify using R in a point&click way
  - Menu-based Statistical analysis: **R-commander**
- Integrated development environments (IDE)
  - Facilitates command-based use of R: **RStudio**

- Free IDE to facilitate Using R from the console.
  - Downloadable from <https://rstudio.com>
- Has become so popular that some people confounds it with R.
- It is only an interface but
  - Very user friendly
  - Specially Good for intermediate-level users

# RStudio

**Source**  
-scripts  
-text edit



**Input data,  
Environment  
& History**

**Console**  
-commands  
-output

**Files, plots,  
packages,  
help**



# R commander

- Free GUI to facilitate doing statistical analyses with R.
- Originally developed for statistics courses where there was no time to learn to use R through the console.
- Has become very popular and has been adopted by many teaching institutions.

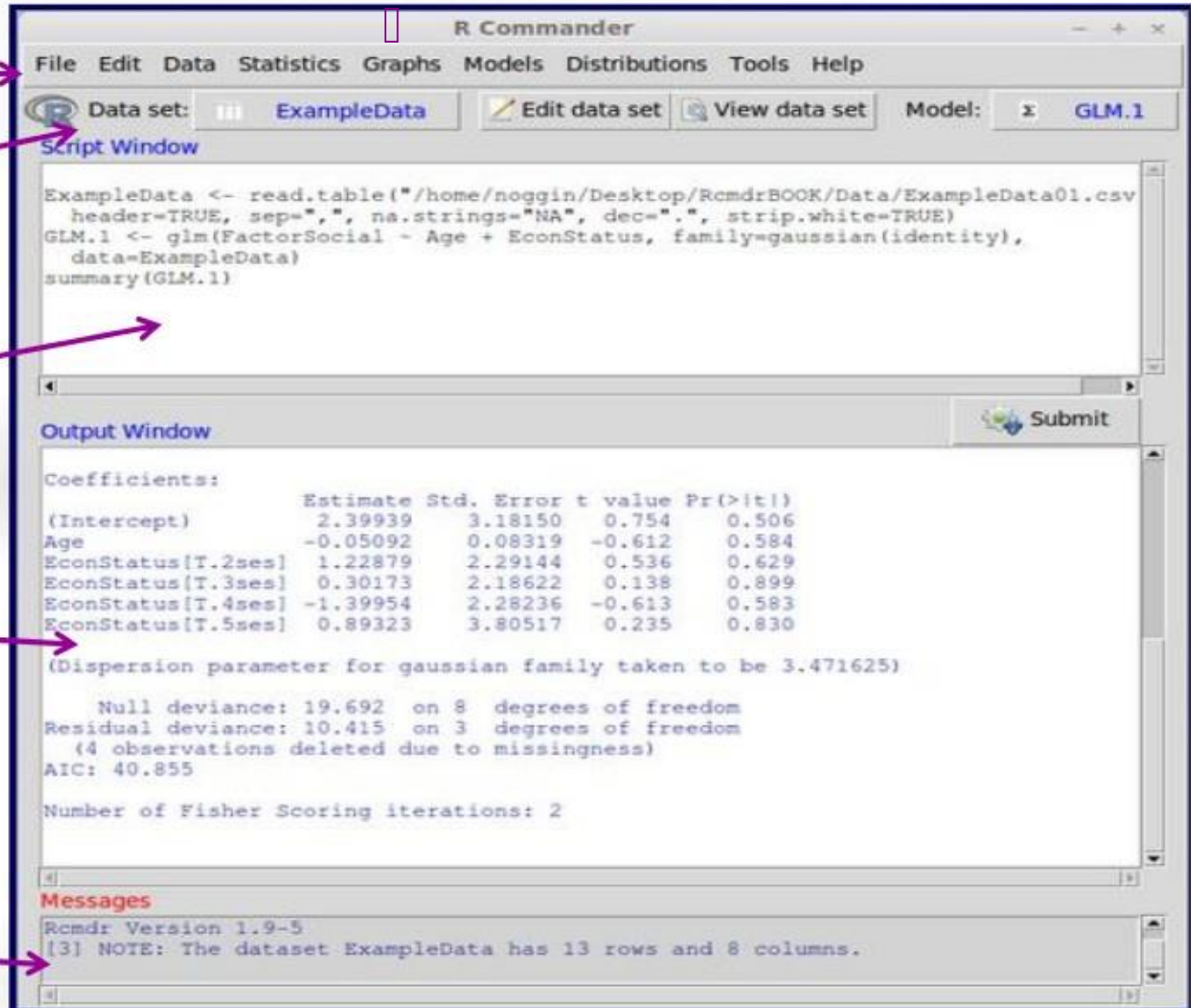
Menu bar

Tool bar

Command window

Text output window

Messages window



The screenshot shows the RCommander window with the following components labeled:

- Menu bar:** File, Edit, Data, Statistics, Graphs, Models, Distributions, Tools, Help.
- Tool bar:** Data set: ExampleData, Edit data set, View data set, Model: GLM.1.
- Script Window:** Contains R code for reading data and fitting a GLM model.
- Output Window:** Displays the results of the model fit, including coefficients, deviance, and AIC.
- Messages window:** Displays messages from RCommander, including the version and dataset information.

```
ExampleData <- read.table("/home/noggin/Desktop/RcmdrBOOK/Data/ExampleData01.csv",
  header=TRUE, sep=";", na.strings="NA", dec=".", strip.white=TRUE)
GLM.1 <- glm(FactorSocial ~ Age + EconStatus, family=gaussian(identity),
  data=ExampleData)
summary(GLM.1)
```

**Output Window**

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	2.39939	3.18150	0.754	0.506
Age	-0.05092	0.08319	-0.612	0.584
EconStatus[T.2ses]	1.22879	2.29144	0.536	0.629
EconStatus[T.3ses]	0.30173	2.18622	0.138	0.899
EconStatus[T.4ses]	-1.39954	2.28236	-0.613	0.583
EconStatus[T.5ses]	0.89323	3.80517	0.235	0.830

(Dispersion parameter for gaussian family taken to be 3.471625)

Null deviance: 19.692 on 8 degrees of freedom  
Residual deviance: 10.415 on 3 degrees of freedom  
(4 observations deleted due to missingness)  
AIC: 40.855

Number of Fisher Scoring iterations: 2

**Messages**

Rcmdr Version 1.9-5  
[3] NOTE: The dataset ExampleData has 13 rows and 8 columns.

## Menu bar



- **File:** contains options to load and save files, define settings and exit.
- **Edit:** options for editing output and log/script window contents.
- **Data:** options to read and modify data.
- **Statistics:** submenu containing options for basic statistical analysis
- **Graphs:** contains options for creating simple statistical graphs
- **Models:** options for obtaining numerical summaries, testing hypotheses and regression models.
- **Distributions:** options to calculate probabilities, obtain quantiles, and get plots of already known statistical distributions.
- **Help:** contains menus with info about how to work with R commander.

Menu bar



- .Create / Load data
- .Editing and inspection of data files.
- .Data transformation / Creation of new variables.
- .Selection of subsets of data or subgroups of variables.
- .Conversion of numerical variables into factors.



Menu bar



- .Summaries
- .Contingency tables
- .Medians
- .Proportions
- .Variants
- .Non-parametric tests
- .Dimensional analysis (A. Multivariate)
- .Model adjustment (Regression)

## Tool bar



- The program always works with a main set of data (active dataset).
- With the active "dataset" we can:
  - edit or visualize it
  - Do analysis
  - Build and use models
- At any moment we can change the active dataset.

## Command window



The screenshot shows the RCommander interface. At the top, there are two tabs: 'R Script' (selected) and 'R Markdown'. Below the tabs is a large text area containing R code. The code reads: `Dataset <- read.table("E:/BioStatFLOSS/dades/osteoporosis.csv", header=TRUE, sep="\t", na.strings="NA", dec="," , strip.white=TRUE)`, `library(abind, pos=14)`, `library(e1071, pos=15)`, and `summary(Dataset)`. A small scroll bar is visible at the bottom left of the text area.

```
Dataset <- read.table("E:/BioStatFLOSS/dades/osteoporosis.csv", header=TRUE,
  sep="\t", na.strings="NA", dec="," , strip.white=TRUE)
library(abind, pos=14)
library(e1071, pos=15)
summary(Dataset)
```

.The menu actions are converted into instructions in the Command window.

## Text output window

Salida Ejecutar

```
> Dataset <- read.table("E:/BioStatFLOSS/dades/osteoporosis.csv", header=TRUE,
+   sep="\t", na.strings="NA", dec=".", strip.white=TRUE)

> library(abind, pos=14)

> library(e1071, pos=15)

> summary(Dataset)
```

registro	area	f_nac	edad	grupedad	peso	talla	imc	bua
Min. : 3.0	Min. :10.00	11509689600: 3	Min. :45.00	45 - 49:378	Min. : 44.00	Min. :138.0	Min. :17.21	Min. : 11.0
1st Qu.: 280.8	1st Qu.:10.00	11718518400: 3	1st Qu.:48.00	50 - 54:233	1st Qu.: 60.50	1st Qu.:153.0	1st Qu.:24.80	1st Qu.: 62.0
Median : 531.5	Median :11.00	11010297600: 2	Median :52.00	55 - 59:176	Median : 68.00	Median :157.0	Median :27.51	Median : 72.0
Mean : 529.9	Mean :11.58	11090822400: 2	Mean :53.42	60 - 64:129	Mean : 69.12	Mean :156.9	Mean :28.11	Mean : 73.3
3rd Qu.: 781.2	3rd Qu.:13.00	11098166400: 2	3rd Qu.:58.00	65 - 69: 84	3rd Qu.: 75.00	3rd Qu.:161.0	3rd Qu.:30.82	3rd Qu.: 84.0
Max. :1033.0	Max. :13.00	11181283200: 2	Max. :69.00		Max. :123.50	Max. :180.0	Max. :48.39	Max. :136.0

(Other) :986

clasific	menarquí	edad_men	menop	tipo_men	nivel_ed
NORMAL :469	Min. : 8.00	Min. :24.00	NO:303	AMBAS : 79	PRIMARIOS :467
OSTEOPENIA :467	1st Qu.:12.00	1st Qu.:46.00	SI:697	HISTERECTOMIA : 63	PRIMARIOS SIN FINALIZAR:212
OSTEOPOROSIS: 64	Median :13.00	Median :51.00		NATURAL :544	SECUNDARIOS :150
	Mean :12.71	Mean :63.04		NO MENOPAUSIA/NO CONSTA:303	SIN ESTUDIOS :122
	3rd Qu.:14.00	3rd Qu.:99.00		OVARIECTOMIA : 11	SUPERIORES : 49
	Max. :17.00	Max. :99.00			

## Message window

Mensajes

```
[1] NOTA: Versión de R Commander 2.3-1: Thu Jan 26 08:42:06 2017
[2] NOTA: El conjunto de datos Dataset tiene 1000 filas y 15 columnas.
```



# How does Rcmdr work

- Similarly to other GUIs:
  - point-and-click
- Without forgetting that it is an interface with R
  - Actions selected in menus
  - Become R commands (in command window)
  - That are automatically executed
- Some "new concepts".
  - Active dataset
  - Active model

# Data input (*the first step!*)

The screenshot illustrates the steps to import an SPSS data set into R Commander. The main window shows the 'Data' menu with 'Import data' selected. An 'Import SPSS Data Set' dialog box is open, with 'Enter name for data set:' set to 'osteo', 'Convert value labels to factor levels' checked, and 'Maximum number of value labels for factor conversion' set to 1. An 'Abrir' (Open) file dialog is also open, showing the file 'osteo.sav' in the 'Real-time PCR Statistics\_archivos' folder. The 'Script Window' at the bottom displays the R code used for the import:

```
osteo <- read.spss("C:/Classes/R/R-commander/osteo.sav",  
  use.value.labels=TRUE, max.value.labels=Inf, to.data.frame=TRUE)  
fix(osteo)
```

The R Commander interface also shows the 'Data set:' field updated to 'osteo' and the 'Model:' field set to '<No active model>'.

- Actions defined through Rcmdr menú system are applied to a privileged dataset: the "active dataset".
- The active dataset can be
  - Edited or visualized
  - Transformed row-wise (cases)
    - Add cases, Subset
  - Transformed column-wise (variables)
    - Add new variables, recode variables
- Any dataset can become "active dataset"

- R commander provides a series of standard statistical analyses that can be applied to the active dataset
  - Selecting options from menus
  - Configuring operations through forms
- If a given analysis cannot be done in "basic" in R commander it may be available in extensions known as "Rcmdr-plugins"
  - Survival analysis
  - Multivariate statistics, ...
  - MORE THAN 30 PLUGINS



- R commander results go either
  - To the output window
  - To the graphics window
- Once the analyses are completed
  - We may need to copy and paste results to a Word document to create a report.
  - We may need to reproduce the analysis step-by-step
    - To check results
    - To extend or change the prior análisis.
- R commander creates an Rmarkdown document
  - That, when run, generates an HTML or Word document
  - Containing the code from the analysis

## R manuals

- \* Intro for beginners [http://cran.r-project.org/doc/contrib/rdebuts\\_es.pdf](http://cran.r-project.org/doc/contrib/rdebuts_es.pdf)
- \* SimpleR <http://cran.r-project.org/doc/contrib/Verzani-SimpleR.pdf>
- \* Quick-R <http://www.statmethods.net/>
- \* Basic statistics with R and R-commander <http://knuth.uca.es/ebrcmdr/>
- \* Statistical methods with R and R-commander  
<http://cran.r-project.org/doc/contrib/Saez-Castillo-RRCmdrv21.pdf>
- \* Try R <http://tryr.codeschool.com/levels/1/challenges/1>

## R books

- \* Introductory Statistics with R
- \* R for SPSS and SAS users