JORGE NAVARRO LÓPEZ • ALBERTO TURÓN LANUZA •
JUAN AGUARÓN JOVEN

STATISTICS AND DATA SCIENCE WITH R

DIAZ DE SANTOS

JORGE NAVARRO LÓPEZ
ALBERTO TURÓN LANUZA
JUAN AGUARÓN JOVEN

STATISTICS AND DATA SCIENCE WITH R

Madrid • Buenos Aires • México • Bogotá

Ediciones Díaz de Santos
Internet: http//www.editdiazdesantos.com
E-mail: ediciones@editdiazdesantos.com

ISBN: 978-84-9052-578-4 (print edition)
e-ISBN: 978-84-9052-579-1 (digital edition)
Legal Deposit: M-2182-2026

Photocomposition and cover design: P55 Servicios Culturales

Contents

Authors' Notes **vii**

Introduction **ix**

I Working with R and R Commander **1**

1 Introduction to R **3**
1.1 The R Language 3
1.2 Installing R 4
1.2.1 Windows Users 5
1.2.2 macOS Users 8
1.3 Basic Use of `R` 10
1.4 Installing `R Commander` 10
1.4.1 Installing the `TeachStat` Library 12
1.5 Working with `R Commander` 12

2 Working with Data **15**
2.1 Importing External Data 15
2.1.1 Importing Data from a URL 16
2.2 Saving Data 17
2.3 Data Structures 19
2.3.1 Vectors 19
2.3.2 Matrices 20
2.3.3 Arrays 22
2.3.4 Factors 22
2.3.5 Lists 23
2.3.6 Data Frames 25
2.4 Data Transformation 27

2.4.1 Filtering Rows . . . 27
2.4.2 Selecting Columns . . . 28
2.4.3 Creating New Columns . . . 28
2.4.4 Sorting by Rows . . . 28
2.4.5 Recoding Variables . . . 29
2.5 Worked Examples . . . 29

II Descriptive Statistics 33

3 Univariate Descriptive Statistics 35
3.1 Basic Concepts . . . 35
3.1.1 Tabulation . . . 35
3.1.1.1 Frequencies . . . 36
3.1.1.2 Frequency Distribution . . . 37
3.1.2 Graphical Representation . . . 39
3.1.3 Numerical Description . . . 39
3.2 Univariate Statistics with `R` . . . 43
3.2.1 Tabulation and Graphical Representation with `R` . . . 44
3.2.2 Numerical Description with `R` . . . 52
3.3 Univariate Statistics with `R Commander` . . . 56
3.4 Univariate Statistics with `TeachStat` . . . 62

4 Bivariate Descriptive Statistics 71
4.1 Basic Concepts . . . 71
4.1.1 Tabulation of Bivariate Data . . . 72
4.1.1.1 Joint Frequencies . . . 72
4.1.1.2 Marginal Frequencies . . . 73
4.1.1.3 Conditional Frequencies . . . 75
4.1.1.4 Independence . . . 76
4.1.2 Correlation Theory . . . 77
4.1.3 Regression Theory . . . 80
4.1.3.1 Goodness of Fit . . . 81
4.1.3.2 Prediction and Reliability . . . 83
4.2 Bivariate Statistics with `R` . . . 83
4.2.1 Tabulation and Graphical Representation . . . 83
4.2.2 Correlation . . . 87
4.2.3 Linear Regression . . . 88
4.3 Bivariate Statistics with `R Commander` . . . 92
4.3.1 Tabulation and Graphical Representation . . . 92

4.3.2 Correlation 96
4.3.3 Linear Regression 98

III Probability Theory **101**

5 Probability and Random Variables **103**
5.1 Basic Concepts 103
5.1.1 Notable Characteristics 104
5.1.1.1 Mathematical Expectation 104
5.1.1.2 Variance of a Random Variable 105
5.1.1.3 Standard Deviation of a Random Variable . 105
5.1.2 Notable Discrete Distributions 106
5.1.2.1 Discrete Uniform Distribution, $U_D(1, N)$ . . 106
5.1.2.2 Bernoulli Distribution, $Be(p)$ 106
5.1.2.3 Binomial Distribution, $Bi(n, p)$ 107
5.1.2.4 Hypergeometric Distribution, $H(N, n, D)$ . . 108
5.1.2.5 Geometric Distribution, $G(p)$ 108
5.1.2.6 Poisson Distribution, $\wp(\lambda)$ 108
5.1.3 Notable Continuous Distributions 109
5.1.3.1 Uniform Distribution, $U(a, b)$ 109
5.1.3.2 Exponential Distribution, $\text{Exp}(\lambda)$ 109
5.1.3.3 Gamma Distribution, $\text{Gamma}(a, \lambda)$ 110
5.1.3.4 Normal Distribution, $N(\mu, \sigma)$ 110
5.2 Probability Calculations with `R` and `R Commander` 112
5.3 Worked Examples 118
5.4 Proposed Exercises 132
5.5 Theoretical-Practical Questions 137
5.6 Solutions 142
5.6.1 Solutions to the Proposed Exercises 142
5.6.2 Solutions to the Theoretical-Practical Questions . . . 147

IV Inferential Statistics **149**

6 Sampling **151**
6.1 Basic Concepts 151
6.1.1 Simple Random Sampling 151
6.1.1.1 Simple Random Sampling with Replacement 152

6.1.1.2 Simple Random Sampling without Replacement . . . 152
6.1.2 Sample Statistics . . . 153
6.1.2.1 Sample Mean . . . 153
6.1.2.2 Sample Proportion . . . 154
6.1.2.3 Sample Variance and Corrected Sample Variance . . . 154
6.1.2.4 Sample Total . . . 155
6.1.3 Distribution of Sample Statistics . . . 155
6.1.3.1 Exact Distributions . . . 155
6.1.3.2 Approximate Distributions . . . 156
6.2 Worked Examples . . . 158
6.3 Proposed Exercises . . . 166
6.4 Theoretical-Practical Questions . . . 169
6.5 Solutions . . . 174
6.5.1 Solutions to the Proposed Exercises . . . 174
6.5.2 Solutions to the Theoretical-Practical Questions . . . 178

7 Point Estimation and Confidence Intervals 181
7.1 Basic Concepts . . . 181
7.1.1 Point Estimation . . . 181
7.1.1.1 Methods for Constructing Estimators . . . 181
7.1.1.2 Properties of Estimators . . . 182
7.1.2 Interval Estimation . . . 183
7.1.3 Distributions Related to Confidence Intervals . . . 183
7.1.3.1 Pearson's Chi-Squared Distribution . . . 183
7.1.3.2 Student's t Distribution . . . 184
7.1.4 Notable Confidence Intervals for a Single Sample . . . 184
7.1.4.1 Confidence Interval for the Mean of a Normal with Known Variance . . . 185
7.1.4.2 Confidence Interval for the Mean of a Normal with Unknown Variance . . . 185
7.1.4.3 Confidence Interval for the Proportion p of a Bernoulli Distribution with Large Sample Size 185
7.1.4.4 Confidence Interval for the Variance of a Normal Distribution . . . 186
7.1.5 Determining the Sample Size . . . 186
7.1.5.1 Estimating the Mean of a Normal Distribution 186

7.1.5.2 Estimating the Parameter p of a Bernoulli Distribution with a Large Sample 187
7.1.5.3 Finite Population Correction 187
7.2 Worked Examples 188
7.3 Proposed Exercises 210
7.4 Theoretical-Practical Questions 214
7.5 Solutions 222
7.5.1 Solutions to the Proposed Exercises 222
7.5.2 Solutions to the Theoretical-Practical Questions . . . 225

8 Parametric Hypothesis Tests 227
8.1 Basic Concepts 227
8.1.1 Hypothesis Tests 227
8.1.2 Types of Errors 229
8.1.3 Methodology of Hypothesis Testing 230
8.1.4 Parametric Hypothesis Tests 230
8.1.4.1 Examples of Parametric Tests 230
8.1.5 Applications 231
8.1.6 Most Common Tests for a Parameter of a Variable . . 232
8.1.6.1 Test for the Mean of a Population with Unknown Variance 232
8.1.6.2 Test for the Proportion of a Bernoulli Variable 233
8.1.6.3 Test for the Variance of a Normal Variable . 233
8.2 Worked Examples 235
8.3 Proposed Exercises 257
8.4 Theoretical-Practical Questions 263
8.5 Solutions 270
8.5.1 Solutions to the Proposed Exercises 270
8.5.2 Solutions to the Theoretical-Practical Questions . . . 276

9 Parametric Analysis of Two Populations 279
9.1 Basic Concepts 279
9.1.1 Types of Tests for Two Populations 279
9.1.2 Applications 280
9.1.3 Snedecor's F Probability Distribution 281
9.2 Parametric Analysis of Two Independent Samples 281
9.2.1 Difference Between the Means of Two Normal Populations with Unknown but Equal Variances 282
9.2.2 Difference Between the Means of Two Normal Populations with Unknown and Unequal Variances 282

9.2.3 Difference Between the Proportions of Two Populations with Large Samples . . . 283
9.2.4 Ratio of the Variances of Two Normal Populations . . 284
9.3 Parametric Analysis of Two Paired Samples . . . 284
9.3.1 Difference Between the Means of Two Populations . . 284
9.3.2 Difference Between the Proportions of Two Populations 285
9.4 Worked Examples . . . 285
9.4.1 Independent Samples . . . 285
9.4.2 Paired Samples . . . 306
9.5 Proposed Exercises . . . 312
9.6 Theoretical-Practical Questions . . . 315
9.7 Solutions . . . 328
9.7.1 Solutions to the Proposed Exercises . . . 328
9.7.2 Solutions to the Theoretical-Practical Questions . . . 332

10 Non-Parametric Hypothesis Tests 335
10.1 Basic Concepts . . . 335
10.1.1 Chi-Square Tests . . . 336
10.1.2 Goodness-of-Fit Tests . . . 337
10.1.2.1 Chi-Square Goodness-of-Fit Test . . . 337
10.1.2.2 Goodness-of-Fit Tests for a Normal Distribution . . . 338
10.1.3 Chi-Square Test of Independence . . . 338
10.1.4 Applications . . . 339
10.2 Worked Examples . . . 340
10.2.1 Goodness-of-Fit Test for a Normal Distribution . . . 340
10.2.2 Chi-Square Test of Independence . . . 344
10.3 Proposed Exercises . . . 353
10.4 Theoretical-Practical Questions . . . 354
10.5 Solutions . . . 357
10.5.1 Solutions to the Proposed Exercises . . . 357
10.5.2 Solutions to the Theoretical-Practical Questions . . . 358

Authors' Notes

Structure and Organization of the Content
This book primarily focuses on probability calculations and statistical inference. Consequently, the chapters devoted to these topics follow a coherent and instructional structure, which includes a theoretical exposition, solved examples, proposed exercises, combined theoretical-practical exercises, and, finally, the corresponding solutions. However, to provide an accessible introduction and broaden the scope of the text, two initial chapters have been added, focused on univariate and bivariate descriptive statistics using `R`. In these chapters, a different, more direct and practical approach is adopted, presenting theoretical-practical concepts accompanied by illustrative examples, to facilitate an exploratory approach to the data from the very beginning.

Numerical Notation and Consistency Criteria
Throughout the book, the use of the comma as a thousands separator has been avoided, following the recommendations of the International System of Units (SI). Instead, a space is used to group digits in thousands, which improves readability and avoids possible ambiguities with the decimal separator.

Supplementary Materials and Acknowledgments
The Excel files containing the datasets used in the book are available for consultation and download at the following link:

`https://www.editdiazdesantos.com/libros/9788490525784`

These resources allow readers to reproduce the analyses presented and to experiment independently with the data, applying the statistical techniques and tools described in the text.

Finally, the authors wish to express their sincere gratitude to Professors Tomás R. Cotos Yañez, Manuel A. Mosquera Rodríguez, Ana Pérez González, and Benigno Reguengo Lareo, from the Department of Statistics and Operations Research at the University of Vigo, for the development of the `RcmdrPlugin.TeachStat` package. This tool, the result of their academic and research work, represents a valuable contribution to the teaching of statistics, as it facilitates the practical application of theoretical concepts within an accessible environment integrated into `R Commander`.

Introduction

What are the odds that one of my customers will switch to another company? Should I sell my shares? How much money could I sell my house for? Is this email *spam*? How much should I charge this client for their insurance policy? Which products should I place together on my supermarket shelves? Should I grant a loan to this client? Questions like these are asked all over the world thousands of times every day. They all have one thing in common: there's no manual with the answer. Usually, these questions precede a decision that, at best, we won't know if it was right or wrong until it's too late, and we may never truly know whether the decision we made was good or bad.

To make the best possible decision, we strive to collect and analyze all available information rigorously. Statistics is the science that helps us process this information and use it in the search for solutions. Although none of this is new, the reality of the 21st century is that students of the *Big Data* era have at their fingertips statistical and computational methods that would have been unimaginable just a few decades ago. *Data Science*, as we like to call it today, combines classical Statistics with innovative techniques developed in the field of Computer Science, offering our graduates a hopeful future in a world where Google interprets user queries through various interfaces (browser, *Hey Google...*), credit institutions analyze the potential risk of their clients based on detailed profiles elaborated from highly diverse information including even their activity on social networks, entertainment companies use streaming-based predictive techniques to recommend movies and series to their subscribers, governments use biometric models to identify terrorism suspects, fact-checking agencies analyze message transmission processes in networks to detect *fake news*, outlier detection techniques are used to identify suspicious credit card activity, computer system intrusions, anomalies in sensor data, and more. These are just a few examples in which statistical methods, combined with the almost infinite capacity to process large volumes of data at high speed, provide solutions to problems we would

not even have considered until recently.

The explosion of data generated in recent decades—driven by the internet, mobile devices, sensors, social networks, among other factors—has fostered the development of artificial intelligence systems that today affect practically all areas of our lives. This transformation has created the need to develop new statistical techniques and computational tools, specific to Data Science, capable of storing, processing, and analyzing large volumes of information. All of this has accelerated the learning of artificial intelligences and facilitated their application in fields such as medicine, marketing, or finance, among others.

Among double degree students, about 40% currently choose programs combining Economics and Data Science. Data Scientists apply economic theory to real-world economic situations and exploit large databases to guide economic decisions in finance, government, leisure, or industry. A quick glance at job postings is enough to see that many companies are replacing traditional CEOs with CDOs: *Chief Digital Officers*. The dizzying evolution that Statistics is experiencing, and which shows no signs of slowing down in the coming decade, offers students of these specialties an extensive field in which to develop successful professional careers.

Here are some facts that support these statements[1]:

- For several years, the demand for analysts and data scientists has consistently outpaced supply. The salaries of a recent graduate in Data Science are well above average.
- Companies of all sectors and sizes have accumulated enormous amounts of data since the implementation, twenty or thirty years ago, of database management systems. Now they are looking for professionals with these profiles to interpret their data and extract useful information from them.
- Employment of mathematicians and statisticians is projected to grow 11 percent between 2023 and 2033, much faster than the average for all occupations. For this decade, the creation of about 3 900 jobs per year for mathematicians and statisticians is expected.

Reaching this promised land requires students to have a solid mathematical background and good knowledge of statistical methods, but also some

[1]Source: U.S. Department of Labor, Bureau of Labor Statistics, Occupational Outlook Handbook, Mathematicians and Statisticians (visited March 19, 2025). [https://www.bls.gov/ooh/math/mathematicians-and-statisticians.htm]

skill in using computational techniques that allow them to manage large amounts of data to which these methods can be applied. That is why our goal with this collection of problems is to teach statistical methods while introducing students to the use of R, a programming language oriented toward Statistics.

R is an open-source programming environment that includes virtually all statistical tools developed to date, while at the same time offering great flexibility to create new tools or adapt existing ones to a specific problem. At the same time, R is designed for data handling, providing means both for acquiring data from external sources (relational databases, *open data* sources, web application APIs, *web scraping*...) and locally stored files in a wide range of formats. Finally, R implements numerous graphical tools that allow powerful graphical data analysis.

Introducing Statistics students to the use of the R language is a necessary step to complement their statistical training in preparation for the kinds of careers described above. That is why the challenge we propose in this book is not only that they learn Statistics: it is that they learn it through a computational tool that allows them to open doors to promising opportunities in the field of Data Science. Upon graduation, the student should be prepared to pursue one of the increasingly numerous master's degrees and specialization courses in Data Science that will open the doors to a brilliant professional future.

Part I

Working with R and R Commander

Chapter 1

Introduction to R

1.1 The R Language

`R` is a programming language oriented towards statistical data analysis. Nowadays, along with `Python`, it is the most widely used language by the scientific community. However, unlike Python, `R` is especially oriented towards the field of statistics and financial mathematics.

As an open-source language, `R` allows its functionalities to be extended through packages or libraries created by third parties. Thus, `R` offers numerous graphical visualization libraries that facilitate the analysis of data not only numerically but also visually.

Its ease and versatility for accessing external data sources allow `R` to read and manage large volumes of data. In addition, it is an interactive language, making it relatively easy to learn.

In this first section, we will see how to install `R` on our computer. We will also see how to install `R Commander`, a graphical interface that will allow us to apply the statistical techniques covered in this book without needing to know the `R` language in detail. The examples in this book are solved using both methods: with `R` commands and via `R Commander`. The latter greatly simplifies the work for those without programming knowledge and is sufficient for everything we need to do. However, we recommend students to learn the fundamentals of the `R` language, as it can be very useful for their future studies or professional career.

1.2 Installing R

The `R` software can be downloaded free of charge from the official R project website, `http://www.r-project.org` (Figure 1.1). On this site, you can find versions for Linux, macOS, and Windows platforms, both as self-executable binaries and as the original source code, in addition to abundant information and manuals. At the time of writing, the stable version of `R` is R 4.4.3, *Trophy Case*, published on February 28, 2025.

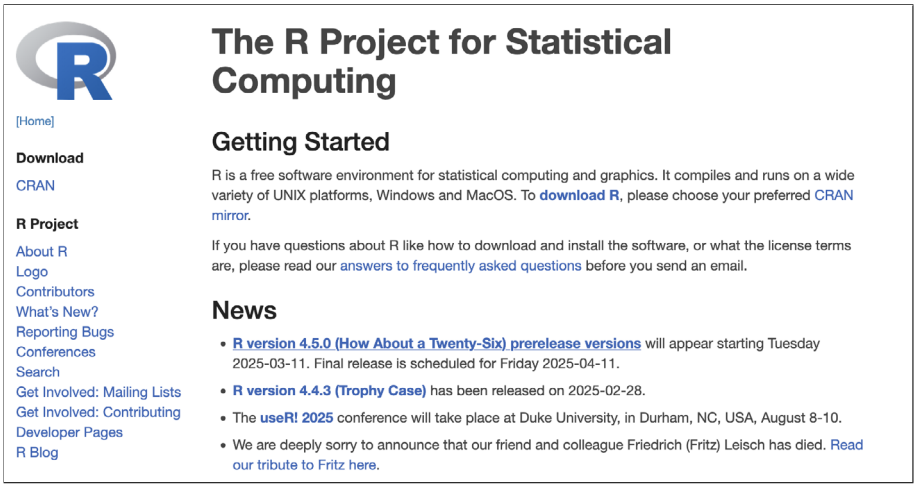


Figure 1.1: Main page of the R project.

By selecting the **download R** link, we access a page where we must choose the mirror (repository) from which to download the software (Figure 1.2). The list is quite extensive, with servers located in different geographic areas. We can select, for example, `0-Cloud`, which automatically redirects us to one of the available repositories.

Regardless of the chosen repository, we then access a page that allows us to choose between downloading binaries (installers) for different operating systems, or the source code (Figure 1.3). We must select the link corresponding to the operating system of the computer where the software will be installed. Below, we will describe the installation process for Windows and macOS.

Usually, four or five new versions are released each year; each brings small improvements over the previous one, but it is not necessary to install them all. We recommend installing the most recent version once a year.

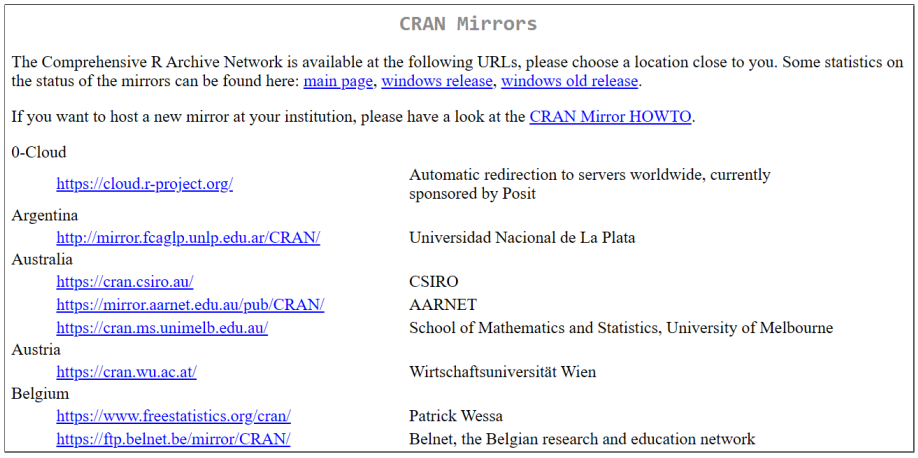


Figure 1.2: `R` repositories.

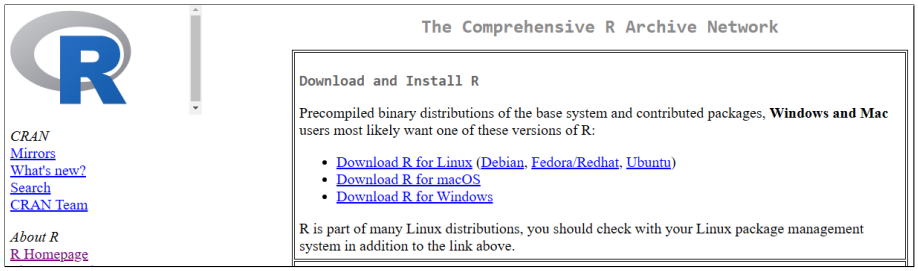


Figure 1.3: Main downloads page.

1.2.1 Windows Users

Selecting the option **Download R for Windows** leads to another page where we must choose (Figure 1.4) the link **install R for the first time**. This link takes us to a new page (Figure 1.5), where we find the link **Download R-4.4.3 for Windows**, which downloads the installation program we must execute.

The installation wizard window appears, asking us to select the language. Next, it shows us the license type, and we press **Next**. It then indicates the folder where the files will be installed, providing one by default. The recommendation is not to modify it and press **Next**.

Next, we must select the components we want to install. It is recommended to keep the default selection (all elements) and press **Next**. This

leads to a dialog that allows us to indicate whether we want to use configuration options (Figure 1.6).

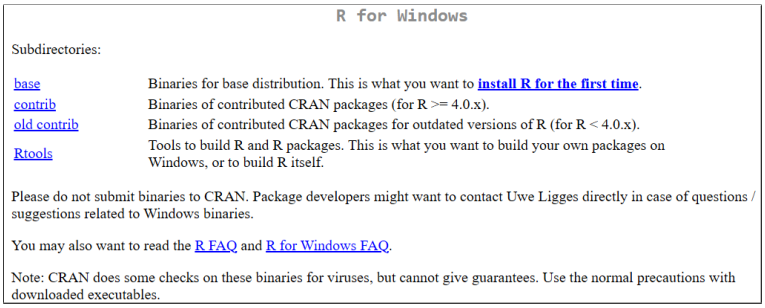


Figure 1.4: Windows downloads page.

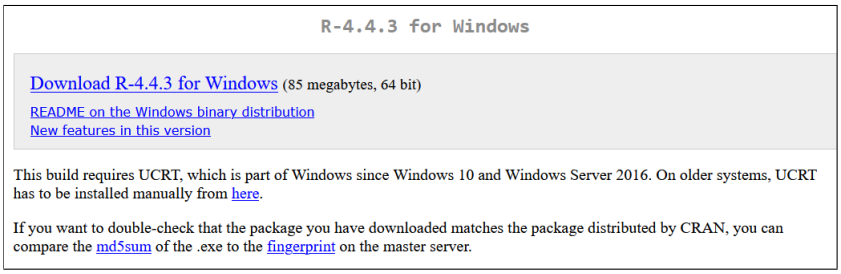


Figure 1.5: Download page for version 4.4.3.

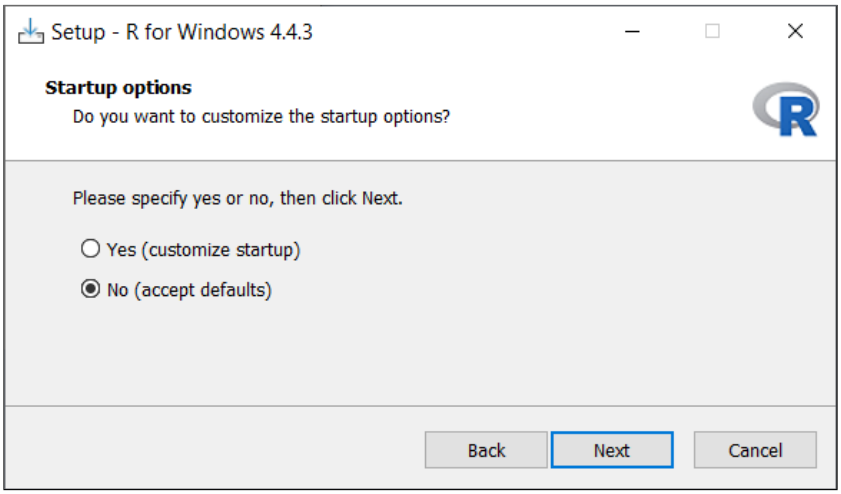


Figure 1.6: Using configuration options.

We can skip these options and continue with the default installation, in which case only a couple of dialogs remain. The first asks us for the name of the Start Menu folder (default: `R`), and the second allows us to indicate whether we want to create a desktop shortcut and associate `.RData` files with `R`. It is recommended to keep these options. After that, the installation process will be complete.

If we decide to use the configuration options shown in Figure 1.6, we will need to go through two additional dialogs. In the first, we must select the display mode (Figure 1.7), choosing between MDI or SDI. When working directly with `R`, it is usually more convenient to choose MDI. On the other hand, it is better to select SDI if we are going to work frequently with `R Commander`[1].

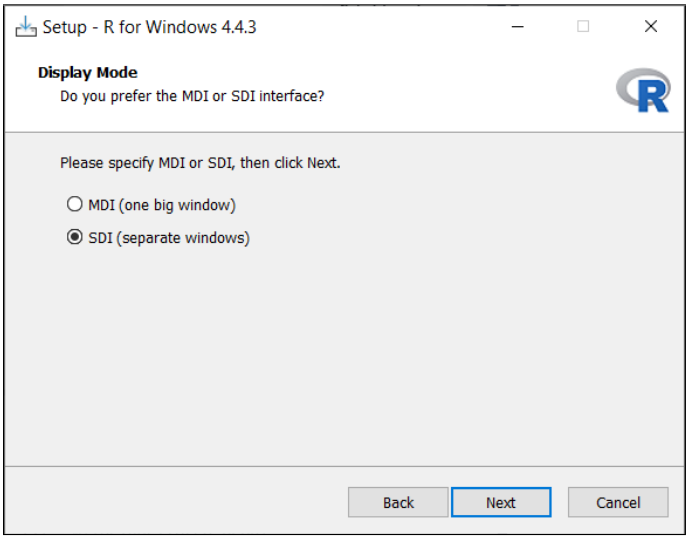


Figure 1.7: Display mode options.

The next dialog shows the second configuration option, in which we can select the help style, choosing between HTML format or plain text (Figure 1.8). The recommendation is to select HTML format. With this, the configuration options (if chosen) are complete, and we proceed to the final dialogs already described (Start Menu folder name, desktop icons, etc.).

[1]If MDI is selected, graphs will be displayed within the main `R` window. This is convenient when working with `R`, but if working with `R Commander`, it forces us to constantly switch between the `R Commander` and `R` windows. By selecting SDI, graphs are displayed in a window separate from the `R` interface.

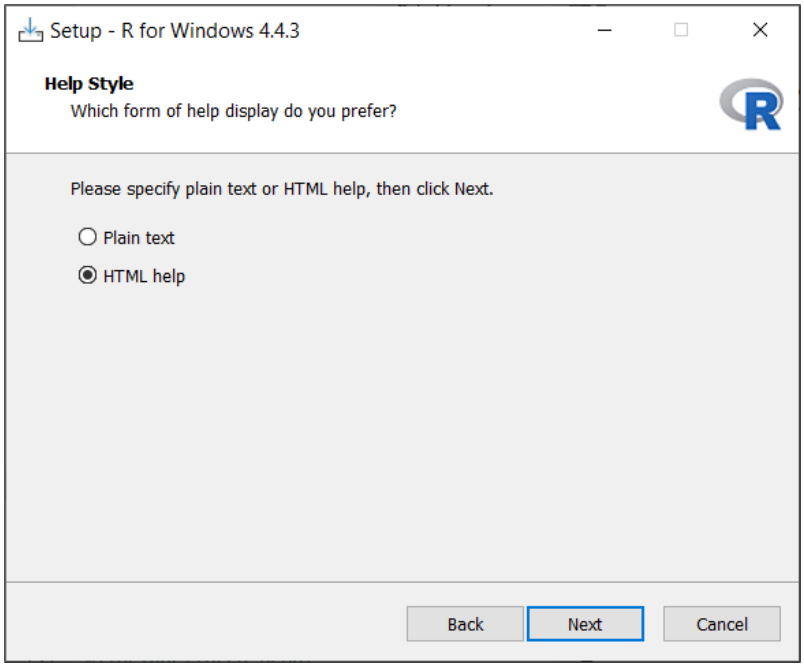


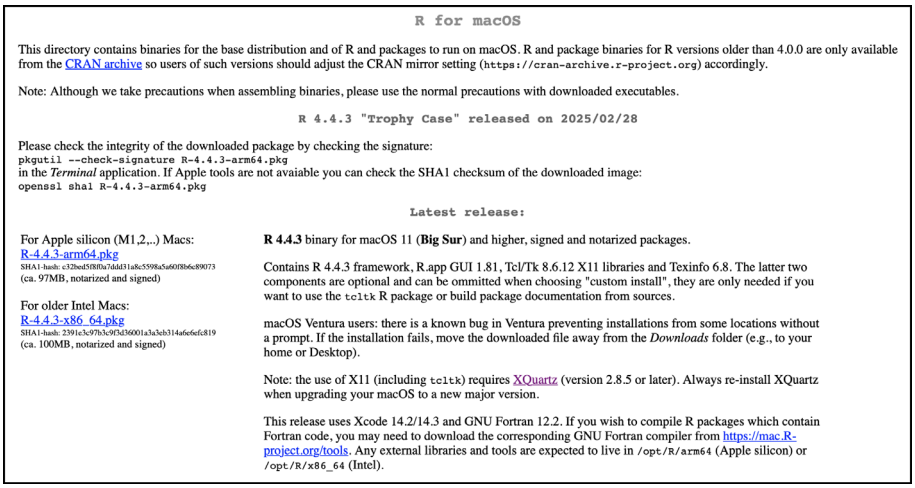
Figure 1.8: `R` help options.

Once the installation process is complete, the application icon will appear on the desktop.

1.2.2 macOS Users

Version 4.4.3 works with any version of OS X starting from version 11, known as *Big Sur*. On the main downloads page (Figure 1.3), we click on the link **Download R for macOS**, and on the page that appears we will see two links corresponding to two different installation versions (Figure 1.9): the first, *R-4.4.3-arm64.pkg*, corresponds to computers equipped with M1, M2, or M3 processors; the second, *R-4.4.3-x86_64.pkg*, corresponds to older computers equipped with Intel processors. We select the one corresponding to our computer, and the download will begin.

Once the installation package is downloaded, we must locate it where it has been saved. We execute it, and the installation process will begin. In this case, the process is very simple, as it is enough to accept all the default options. At the end of the process, the program icon will appear in the Applications folder.



R for macOS

This directory contains binaries for the base distribution and of R and packages to run on macOS. R and package binaries for R versions older than 4.0.0 are only available from the CRAN archive so users of such versions should adjust the CRAN mirror setting (`https://cran-archive.r-project.org`) accordingly.

Note: Although we take precautions when assembling binaries, please use the normal precautions with downloaded executables.

R 4.4.3 "Trophy Case" released on 2025/02/28

Please check the integrity of the downloaded package by checking the signature:
`pkgutil --check-signature R-4.4.3-arm64.pkg`
in the *Terminal* application. If Apple tools are not avaiable you can check the SHA1 checksum of the downloaded image:
`openssl sha1 R-4.4.3-arm64.pkg`

Latest release:

For Apple silicon (M1,2,..) Macs:
R-4.4.3-arm64.pkg
SHA1-hash: e32bed50f0fa7ddd31a8c5598a5a60f8b6e89073
(ca. 97MB, notarized and signed)

For older Intel Macs:
R-4.4.3-x86_64.pkg
SHA1-hash: 2391e3c97b3c90f3d36001a3a3eb314a6e6efc819
(ca. 100MB, notarized and signed)

R 4.4.3 binary for macOS 11 (**Big Sur**) and higher, signed and notarized packages.

Contains R 4.4.3 framework, R.app GUI 1.81, Tcl/Tk 8.6.12 X11 libraries and Texinfo 6.8. The latter two components are optional and can be ommitted when choosing "custom install", they are only needed if you want to use the `tcltk` R package or build package documentation from sources.

macOS Ventura users: there is a known bug in Ventura preventing installations from some locations without a prompt. If the installation fails, move the downloaded file away from the *Downloads* folder (e.g., to your home or Desktop).

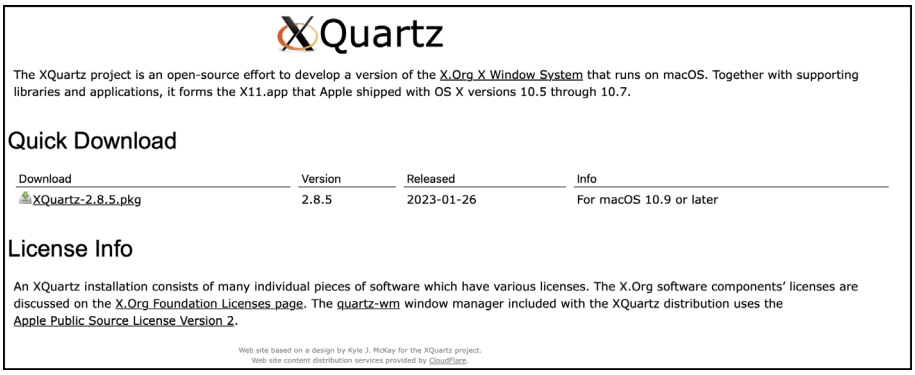
Note: the use of X11 (including `tcltk`) requires XQuartz (version 2.8.5 or later). Always re-install XQuartz when upgrading your macOS to a new major version.

This release uses Xcode 14.2/14.3 and GNU Fortran 12.2. If you wish to compile R packages which contain Fortran code, you may need to download the corresponding GNU Fortran compiler from https://mac.R-project.org/tools. Any external libraries and tools are expected to live in `/opt/R/arm64` (Apple silicon) or `/opt/R/x86_64` (Intel).

Figure 1.9: Download links for R for macOS.

However, the `R Commander` interface requires the use of `X11` graphical libraries, which are not natively installed on Mac. Therefore, we must install the `XQuartz` package, whose latest version can be downloaded from `https://www.xquartz.org` (Figure 1.10).

By clicking on the download link, we obtain the installer for the latest version of `XQuartz`. The downloaded package is compressed; when opening it, we will see the installer, a file called *XQuartz.pkg*. Next, it is enough to run this installer and accept all the proposed options. At the end of the process, the libraries are installed and ready to be used by `R Commander`.



XQuartz

The XQuartz project is an open-source effort to develop a version of the X.Org X Window System that runs on macOS. Together with supporting libraries and applications, it forms the X11.app that Apple shipped with OS X versions 10.5 through 10.7.

Quick Download

Download	Version	Released	Info
XQuartz-2.8.5.pkg	2.8.5	2023-01-26	For macOS 10.9 or later

License Info

An XQuartz installation consists of many individual pieces of software which have various licenses. The X.Org software components' licenses are discussed on the X.Org Foundation Licenses page. The quartz-wm window manager included with the XQuartz distribution uses the Apple Public Source License Version 2.

Web site based on a design by Kyle J. McKay for the XQuartz project.
Web site content distribution services provided by CloudFlare.

Figure 1.10: Download page for the `XQuartz` graphical library.

1.3 Basic Use of R

R is an interactive programming language, which means that commands can be written directly in the console, and when pressing the Enter key, they are executed, showing the result (if any). Commands are written on the last line of the R window, right after the > symbol (called the *prompt*).

For example, if we write the instruction `print("Hello, world!")` and press Enter, the result will be:

```
> print("Hello, world!")
[1] "Hello, world!"
```

The `print()` command is a **function**; it tells R to display on the following line(s) whatever is inside the parentheses, which in this case is a text string.

Arithmetic operations can be performed directly in the console:

```
> 3 + 5    # Addition
[1] 8

> 10 / 2   # Division
[1] 5

> sqrt(16) # Square root
[1] 4

> 2^3      # Power
[1] 8
```

or values can be assigned to variables using either the <- symbols or the = symbol, to be used later:

```
> a <- 10           # Or also a = 10
> b <- 20           # Or also b = 20
> suma <- a + b     # Or also suma = a + b
> suma
[1] 30
```

Note that the character # is used in R to insert comments.

1.4 Installing R Commander

R allows the installation of third-party scripts designed to perform various tasks (financial calculations, mathematical computations, data visualization,

access to external application APIs, and a wide range of other functionalities) in the form of packages, which once integrated into the application are called *libraries*.

To make statistical data analysis easier for people who are not experts in the `R` programming language, the `R Commander` library was created. It is a graphical interface in which the main statistical functions can be selected via menu options, instead of having to remember `R` commands or know their syntax in detail.

Before using `R Commander` for the first time, it is necessary to install the package, which is called `Rcmdr`. To install a package in `R`, go to the **Packages** option and select **Install packages**. The first time we install a package, we will be asked to choose the mirror from which to perform the installation. Just like with the download of the `R` installer, we can select different locations, but usually selecting the first one (`0-Cloud`) is sufficient.

Next, a new window opens with the available packages on this server; we must select the `Rcmdr` package, and it will automatically be installed on `R`.

Once installed, the `Rcmdr` package will be available for use whenever we need it. To open `R Commander`, there is the menu option **Load package**, where we can select `Rcmdr`. The first time it is loaded, it may request the installation of some auxiliary library, which we authorize by pressing the **Accept** button. Finally, the graphical window of `R Commander` will open (Figure 1.11).

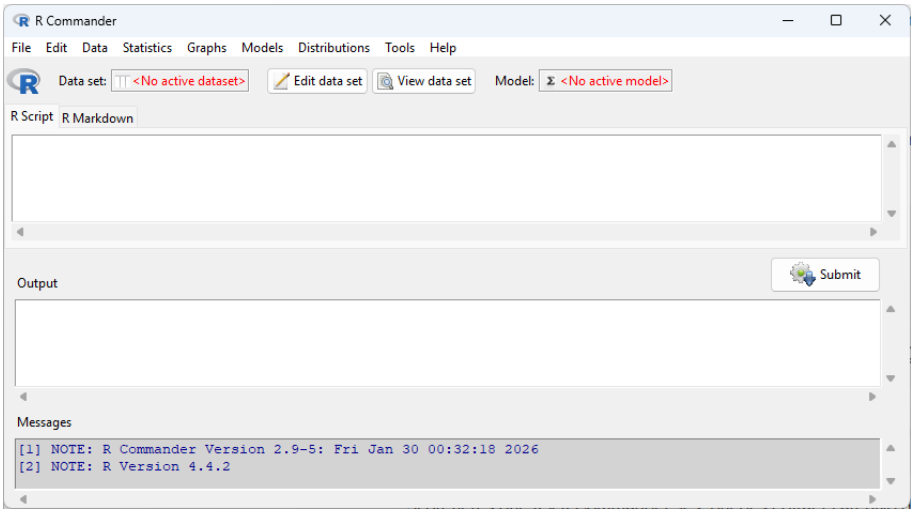


Figure 1.11: Graphical interface of `R Commander`.

We also have the option to open `R Commander` by running the command:

```
> library(Rcmdr)
```

1.4.1 Installing the `TeachStat` Library

Some of the techniques we will use in this book are implemented in an auxiliary package for `R Commander` called `TeachStat`. In order to use it with the examples in the book, we must install it beforehand, once we have installed `R Commander`. To do this, we can proceed in the same way as for installing `R Commander`: from `R`, go to the **Packages** menu, choose the option **Install packages**, select a mirror, and finally choose the package `RcmdrPlugin.TeachStat` from the list of all available packages.

Alternatively, we can do this from the `R` console by running the following command:

```
> install.packages("RcmdrPlugin.TeachStat")
```

If the installation is successful, the plugin will be ready for use. We then have several options:

- From `R` we can go to the menu **Load package** and select the package `RcmdrPlugin.TeachStat`.
- From the `R` console, we can type:

  ```
  > library(RcmdrPlugin.TeachStat)
  ```

These two options automatically load both `R Commander` and the `TeachStat` plugin.

- If we have already loaded `R Commander`, the package is accessible from the menu **Tools** → **Load Rcmdr plugin(s)...** (see Figure 1.12). If we select it from the list of plugins that appears, we can work with it during the entire `R Commander` session. Before doing so, however, we must restart `R Commander`.

1.5 Working with `R Commander`

When opening `R Commander`, we find a main window (Figure 1.11) divided into several sections:

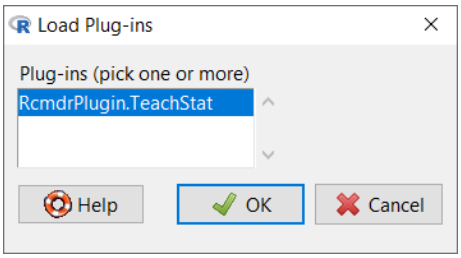


Figure 1.12: Activation of the `TeachStat` plugin.

1. **Menu Bar**. At the top of the window are the drop-down menus that allow us to access the different functionalities of `R Commander`.

2. **R Script Tab**. In this area, we can see the `R` commands that are automatically executed when we use the menus and dialog boxes. We can also write `R` commands directly here and run them using the `Run` button. We can edit scripts, which are sequences of commands that can be saved and executed later.

3. **R Markdown Tab**. This is used to generate dynamic reports, combining R code, results (tables, graphs), and explanatory comments in a single document (HTML, PDF, Word, ...).

4. **Output Window**. Below the console and the script window, there is an output window where the results of the commands we execute are displayed.

5. **Messages Window**. Displays informational notes, warnings, and errors related to the executed commands.

The main options accessible from the `R Commander` menu are:

1. **File:** create a new `R` script, open an existing script, or save the current script or results.

2. **Edit:** open a data editor to modify the active dataset or clear the `R` console.

3. **Data:** manage datasets, create, modify or delete variables in the dataset, transform datasets, or import data from Excel or various formats.

4. **Statistics:** provide basic statistical summaries (mean, median, standard deviation, etc.), apply inferential parametric and nonparametric statistical tests (t-test, chi-square test, ANOVA...), dimensional analysis, or fit various statistical models (linear, generalized linear models, etc.).

5. **Graphs:** create bar charts, scatter plots, box plots, Q-Q plots, etc.

6. **Tools:** configure `R` options such as the working directory or script editor, or load additional `R` packages.

7. **Help:** access online documentation, tutorials, and help on `R` and `R Commander`.

In addition, if we install the `TeachStat` plugin, we will see a new menu option called **Basic Statistics** from which we can run most of the commands we will use in this book.

It should be noted that `R Commander` makes the use of `R` easier, since we do not need to write code to perform basic analyses, and it is also interactive, so we can see the `R` commands generated when using the menus and thus learn `R` through them. However, `R Commander` does not include all of `R`'s functionalities, especially the more advanced ones. For complex analyses, we may need to write code directly in `R`.

If necessary, we can turn to various sources of help. For example, the official `R` documentation includes several manuals that cover everything from the basics to advanced topics. These can be accessed from the `R` console using the command:

```
> help.start()
```

We can also find information on the CRAN project website: `https://cran.r-project.org/manuals.html`.

For information on specific commands, we can access the documentation of any `R` function by using the command `?` followed by the function name. For example, the command:

```
> ?mean
```

will give us information about the `mean` function, which calculates the mean of a series of values.

From the **Help** option in `R Commander`, we can access various websites with information about `R` and `R Commander`.