Practical Statistics in R II

Alboukadel Kassambara

Comparing Groups: Numerical Variables

Summarize, Visualize, Check Assumptions, Run Tests, Interpret, Report

datanovia.com

Edition 1

Practical Statistics in R II - Comparing Groups: Numerical Variables

Alboukadel KASSAMBARA

Published by Datanovia (`https://www.datanovia.com/en`), Alboukadel Kassambara

Contact: Alboukadel Kassambara <alboukadel.kassambara@gmail.com>

For general information contact Alboukadel Kassambara <alboukadel.kassambara@gmail.com>.

Contents

Preface

0.1 What you will learn

This R Statistics book provides a solid step-by-step practical guide to statistical inference for **comparing groups means** using the R software. Additionally, we developed an R package named **rstatix** (`https://rpkgs.datanovia.com/rstatix/`), which provides a simple and intuitive pipe-friendly framework, coherent with the `tidyverse` design philosophy, for computing the most common statistical analyses, including t-test, Wilcoxon test, ANOVA, Kruskal-Wallis and correlation analyses, outliers identification and more.

This book is designed to get you doing the statistical tests in R as quick as possible. The book focuses on implementation and understanding of the methods, without having to struggle through pages of mathematical proofs.

You will be guided through the steps of summarizing and visualizing the data, checking the assumptions and performing statistical tests in R, interpreting and reporting the results.

0.2 Key features of this book

Although there are several good books on statistics and related topics, we felt that many of them are too theoretical. Our goal was to write a practical guide to statistics in R with visualization, interpretation and reporting the results.

The main parts of the book include:

- *statistical tests and assumptions* for the comparison of groups means,
- *comparing two means*,
  - *t-test*,
  - *Wilcoxon test*,
  - *Sign test*,
- *comparing multiple means*,
  - *ANOVA - Analysis of Variance* for independent measures
  - *repeated measures ANOVA*,
  - *mixed ANOVA*,
  - *ANCOVA and MANOVA*,
  - *Kruskal-Wallis test*
  - *Friedman test*

The book presents the basic principles of these tasks and provide many examples in R. This book offers solid guidance in statistics for students and researchers.

Key features:

- Covers the most common statistical tests and implementations
- Key assumptions are presented and checked
- Short, self-contained chapters with practical examples. This means that, you don't need to read the different chapters in sequence.

In each chapter, we present R lab sections in which we systematically work through applications of the various methods discussed in that chapter.

0.3 How this book is organized ?

This book contains 3 parts. After a quick introduction to R (Chapter 1), **Part I** introduces some research questions and the corresponding **statistical tests**, as well as, the **assumptions** of the tests. Many of the statistical methods including t-test and analysis of variance (ANOVA) assume some characteristics about the data, including **normality of the data distributions** and **equality of group variances**. These assumptions should be taken seriously to draw reliable interpretation and conclusions of the research. In Part I, you will learn how to assess normality using the **Shapiro-Wilk test** (Chapter 3) and how to compare variances in R using **Levene's test** and more (Chapter 4).

In **Part II**, we consider how to compare two means using **t-test** (parametric method, Chapter 8) and **wilcoxon test** (non-parametric method, Chapter 9). Main contents, include:

1. **Comparing one-sample mean to a standard known mean**:
   - One-Sample T-test (parametric)
   - Wilcoxon Signed Rank Test (non-parametric)
2. **Comparing the means of two independent groups**:
   - Independent Samples T-test (parametric)
   - Wilcoxon Rank Sum Test (non-parametric)
3. **Comparing the means of paired samples**:
   - Paired Samples T-test (parametric)
   - Wilcoxon Signed Rank Test on Paired Samples (non-parametric)

In this Part, we also described how to check t-test assumptions, as well as, how to compute the t-test effect size (**Cohen's d**). You will also learn how to compute the Wilcoxon effect size. Additionally, we present the **sign test** (Chapter 10), an alternative to the *paired-samples t-test* and the *Wilcoxon signed-rank test*, in the situation where the distribution of differences between paired data values is neither normal (in t-test) nor symmetrical (in Wilcoxon test).

Part III describes how to compare multiple means in R using **ANOVA** (Analysis of Variance) method and variants (Chapters 12 - 18).

Chapter 12 describes how to compute and interpret the different types of ANOVA for comparing independent measures, including:

- **One-way ANOVA**, an extension of the independent samples t-test for comparing the means in a situation where there are more than two groups.
- **two-way ANOVA** for assessing an interaction effect between two independent categorical variables on a continuous outcome variable.
- **three-way ANOVA** for assessing an interaction effect between three independent categorical variables on a continuous outcome variable.

We also provide R code to check ANOVA assumptions and perform Post-Hoc analyses. Additionally, we'll present the **Kruskal-Wallis test** (Chapter 17), which is a non-parametric alternative to the one-way ANOVA test.

Chapter 13 presents **repeated-measures ANOVA**, which is used for analyzing data where same subjects are measured more than once. You will learn different types of repeated measures ANOVA, including:

- **One-way repeated measures ANOVA** for comparing the means of three or more levels of a *within-subjects* variable.
- **two-way repeated measures ANOVA** used to evaluate simultaneously the effect of two within-subject factors on a continuous outcome variable.
- **three-way repeated measures ANOVA** used to evaluate simultaneously the effect of three within-subject factors on a continuous outcome variable.

You will also learn how to compute and interpret the **Friedman test** (Chapter 18), which is a non-parametric alternative to the one-way repeated measures ANOVA test.

Chapter 14 shows how to run **mixed ANOVA**, which is used to compare the means of groups cross-classified by at least two factors, where one factor is a "*within-subjects*" factor (repeated measures) and the other factor is a "*between-subjects*" factor.

Chapters 15 and 16 describe, respectively, some advanced extensions of ANOVA, including:

- **ANCOVA** (analyse of covariance), an extension of the one-way ANOVA that incorporate a covariate variable.
- **MANOVA** (multivariate analysis of variance), an ANOVA with two or more continuous outcome variables.

0.4 Book website

Datanovia: `https://www.datanovia.com/en`

0.5 Executing the R codes from the PDF

For a single line R code, you can just copy the code from the PDF to the R console.

For a multiple-line R codes, an error is generated, sometimes, when you copy and paste directly the R code from the PDF to the R console. If this happens, a solution is to:

- Paste firstly the code in your R code editor or in your text editor
- Copy the code from your text/code editor to the R console

Additionally, if your pdf reader has a select tool that allows you to select text in a rectangle, that works better in some readers.

0.6 Acknowledgment

I sincerely thank all developers for their efforts behind the packages that this book depends on, namely, bookdown and more.

0.7 Colophon

This book was built with R 3.3.2 and the following packages :

```
##        name    version                      source
## 1  bookdown       0.16                        CRAN
## 2     broom      0.5.2                        CRAN
## 3  datarium  0.1.0.999                       local
## 4   emmeans      1.3.3                        CRAN
## 5    ggpubr      0.2.4                        CRAN
## 6   rstatix  0.3.0.999  Github:kassambara/rstatix
## 7 tidyverse 1.2.1.9000 Github:tidyverse/tidyverse
```

About the author

Alboukadel Kassambara is a PhD in Bioinformatics and Cancer Biology. He works since many years on genomic data analysis and visualization (read more: `http://www.alboukadel.com/`).

He has work experiences in statistical and computational methods to identify prognostic and predictive biomarker signatures through integrative analysis of large-scale genomic and clinical data sets.

He is the author of:

1) the bioinformatics tool named **GenomicScape** (www.genomicscape.com), an easy-to-use web tool for gene expression data analysis and visualization.
2) the **Datanovia** (`https://www.datanovia.com/en/`) and **STHDA** (`http://www.sthda.com/english/`) websites, which contains many courses and **tutorials** on data data mining and statistics for decision supports.
3) many popular **R packages** for multivariate data analysis, survival analysis, correlation matrix visualization and basic data visualization (`https://rpkgs.datanovia.com/`).
4) many **books** on data analysis, visualization and machine learning (`https://www.datanovia.com/en/shop/`)

Chapter 1

Introduction to R

R is a free and powerful statistical software for **analyzing** and **visualizing** data. If you want to learn easily the essential of R programming, visit our series of tutorials available on STHDA: `http://www.sthda.com/english/wiki/r-basics-quick-and-easy`.

In this chapter, we provide a very brief introduction to **R**, for installing R/RStudio as well as importing your data into R and installing required libraries.

1.1 Install R and RStudio

1.1.1 Standard installation

R and RStudio can be installed on Windows, MAC OSX and Linux platforms. RStudio is an integrated development environment for R that makes using R easier. It includes a console, code editor and tools for plotting.

1. R can be downloaded and installed from the Comprehensive R Archive Network (CRAN) webpage (`http://cran.r-project.org/`)
2. After installing R software, install also the RStudio software available at: `http://www.rstudio.com/products/RStudio/`.
3. Launch RStudio and start use R inside R studio.

1.1.2 R Online

R can be also accessed online without any installation. You can find an example at `https://rdrr.io/snippets/`. This site include thousands add-on packages.

1.2 Install and load required R packages

An R package is a collection of functionalities that extends the capabilities of base R. For example, to use the R code provided in this book, you should install the following R packages:

- `tidyverse` packages, which are a collection of R packages that share the same programming philosophy. These packages include:

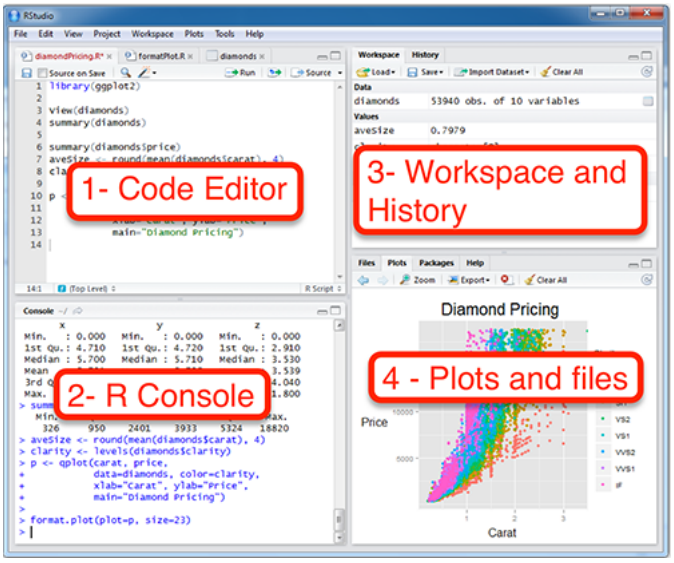

Figure 1.1: Rstudio interface

  - `readr`: for importing data into R
  - `dplyr`: for data manipulation
  - `ggplot2`: for data visualization.
- `ggpubr` package, which makes it easy, for beginner, to create publication ready plots
- `rstatix` provides pipe-friendly R functions for easy statistical analyses
- `datarium`: contains required data sets for this chapter
- `emmeans`: perform post-hoc analyses following ANOVA tests

1. **Install the tidyverse package**. Installing tidyverse will install automatically readr, dplyr, ggplot2 and more. Type the following code in the R console:

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

2. **Install ggpubr, rstatix, datarium and emmeans packages**.

```
install.packages("ggpubr")
install.packages("rstatix")
install.packages("datarium")
install.packages("emmeans")
```

3. **Load required packages**. After installation, you must first load the package for using the functions in the package. The function `library()` is used for this task. An alternative function is `require()`. For example, to load tidyverse and ggpubr packages, type this:

```
library("tidyverse")
library("ggpubr")
```

Now, we can use R functions, such as *ggscatter*() [in the ggpubr package] for creating a scatter plot.

If you want a help about a given function, say ggscatter(), type this in R console: `?ggscatter`.

1.3 Data format

Your data should be in rectangular format, where columns are variables and rows are observations (individuals or samples).

- Column names should be compatible with R naming conventions. Avoid column with blank space and special characters. Good column names: `long_jump` or `long.jump`. Bad column name: `long jump`.

- Avoid beginning column names with a number. Use letter instead. Good column names: `sport_100m` or `x100m`. Bad column name: `100m`.
- Replace missing values by `NA` (for not available)

For example, your data should look like this:

```
  manufacturer model displ year cyl      trans drv
1         audi    a4   1.8 1999   4   auto(l5)   f
2         audi    a4   1.8 1999   4 manual(m5)   f
3         audi    a4   2.0 2008   4 manual(m6)   f
4         audi    a4   2.0 2008   4   auto(av)   f
```

Read more at: Best Practices in Preparing Data Files for Importing into R[1]

1.4 Import your data in R

First, save your data into txt or csv file formats and import it as follow (you will be asked to choose the file):

```
library("readr")

# Reads tab delimited files (.txt tab)
my_data <- read_tsv(file.choose())

# Reads comma (,) delimited files (.csv)
my_data <- read_csv(file.choose())

# Reads semicolon(;) separated files(.csv)
my_data <- read_csv2(file.choose())
```

Read more about how to import data into R at this link: `http://www.sthda.com/english/wiki/importing-data-into-r`

1.5 Demo data sets

R comes with several demo data sets for playing with R functions. The most used R demo data sets include: **USArrests**, **iris** and **mtcars**. To load a demo data set, use the function **data**() as follow. The function `head()` is used to inspect the data.

[1] `http://www.sthda.com/english/wiki/best-practices-in-preparing-data-files-for-importing-into-r`

```
data("iris")   # Loading
head(iris, n = 3)  # Print the first n = 3 rows
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
```

To learn more about iris data sets, type this:

```
?iris
```

After typing the above R code, you will see the description of `iris` data set: this iris data set gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The species are Iris setosa, versicolor, and virginica.

1.6 Data manipulation

After importing your data in R, you can easily manipulate it using the `dplyr` package (Wickham et al., 2019), which can be installed using the R code: `install.packages("dplyr")`.

After loading dplyr, you can use the following R functions:

- `filter()`: Pick rows (observations/samples) based on their values.
- `distinct()`: Remove duplicate rows.
- `arrange()`: Reorder the rows.
- `select()`: Select columns (variables) by their names.
- `rename()`: Rename columns.
- `mutate()`: Add/create new variables.
- `summarise()`: Compute statistical summaries (e.g., computing the mean or the sum)
- `group_by()`: Operate on subsets of the data set.

Note that, dplyr package allows to use the forward-pipe chaining operator (%>%) for combining multiple operations. For example, x %>% f is equivalent to f(x). Using the pipe (%>%), the output of each operation is passed to the next operation. This makes R programming easy.

Read more about Data Manipulation at this link: `https://www.datanovia.com/en/courses/data-manipulation-in-r/`

1.7 Close your R/RStudio session

Each time you close R/RStudio, you will be asked whether you want to save the data from your R session. If you decide to save, the data will be available in future R sessions.

Part I

Statistical Tests and Assumptions

Chapter 2

Introduction

In this chapter, we'll introduce some research questions and the corresponding **statistical tests**, as well as, the **assumptions** of the tests.

2.1 Research questions and statistics

The most popular research questions include:

1. whether **two variables** (n = 2) are **correlated** (i.e., associated)
2. whether **multiple variables** (n > 2) are **correlated**
3. whether **two groups** (n = 2) of samples **differ** from each other
4. whether **multiple groups** (n >= 2) of samples **differ** from each other
5. whether the **variability** of two or more samples differ

Each of these questions can be answered using the following statistical tests:

1. **Correlation test** between two variables
2. **Correlation matrix** between multiple variables
3. **Comparing the means of two groups**:
   - **Student's t-test** (parametric)
   - **Wilcoxon rank test** (non-parametric)
4. **Comparing the means of more than two groups**
   - **ANOVA test** (analysis of variance, parametric): extension of t-test to compare more than two groups.
   - **Kruskal-Wallis rank sum test** (non-parametric): extension of Wilcoxon rank test to compare more than two groups
5. **Comparing the variances**:
   - Comparing the variances of two groups: **F-test** (parametric)
   - Comparison of the variances of more than two groups: **Bartlett's test** (parametric), **Levene's test** (parametric) and **Fligner-Killeen test** (non-parametric)

2.2 Assumptions of statistical tests

Many of the statistical methods including correlation, regression, t-test, and analysis of variance assume some characteristics about the data. Generally they assume that:

- the data are **normally distributed**
- and the **variances** of the groups to be compared are **homogeneous** (equal).

These assumptions should be taken seriously to draw reliable interpretation and conclusions of the research.

These tests - correlation, t-test and ANOVA - are called **parametric tests**, because their validity depends on the distribution of the data.

Before using parametric test, some preliminary tests should be performed to make sure that the test assumptions are met. In the situations where the assumptions are violated, **non-paramatric** tests are recommended.

2.3 Assessing normality

1. With **large enough sample sizes** (n > 30) the violation of the normality assumption should not cause major problems (central limit theorem). This implies that we can ignore the distribution of the data and use parametric tests.
2. However, to be consistent, we can use **Shapiro-Wilk's significance test** comparing the sample distribution to a normal one in order to ascertain whether data show or not a serious deviation from normality (Ghasemi and Zahediasl, 2012).

2.4 Assessing equality of variances

The standard **Student's t-test** (comparing two independent samples) and the ANOVA test (comparing multiple samples) assume also that the samples to be compared have equal variances.

If the samples, being compared, follow normal distribution, then it's possible to use:

- **F-test** to compare the variances of two samples
- **Bartlett's Test** or **Levene's Test** to compare the variances of multiple samples.

2.5 Summary

This chapter introduces the most commonly used statistical tests and their assumptions.

Chapter 3

Assessing Normality

3.1 Introduction

Many of the statistical methods including correlation, regression, t tests, and analysis of variance assume that the data follows a normal distribution or a Gaussian distribution. These tests are called parametric tests, because their validity depends on the distribution of the data.

Normality and the other assumptions made by these tests should be taken seriously to draw reliable interpretation and conclusions of the research.

With large enough sample sizes ($>$ 30 or 40), there's a pretty good chance that the data will be normally distributed; or at least close enough to normal that you can get away with using parametric tests, such as t-test (central limit theorem).

In this chapter, you will learn how to check the **normality of the data in R** by visual inspection (*QQ plots* and **density distributions**) and by significance tests (*Shapiro-Wilk test*).

3.2 Prerequisites

Make sure you have installed the following R packages:

- `tidyverse` for data manipulation and visualization
- `ggpubr` for creating easily publication ready plots
- `rstatix` provides pipe-friendly R functions for easy statistical analyses

Start by loading the packages:

```
library(tidyverse)
library(ggpubr)
library(rstatix)
```

3.3 Demo data

We'll use the `ToothGrowth` dataset. Inspect the data by displaying some random rows by groups: