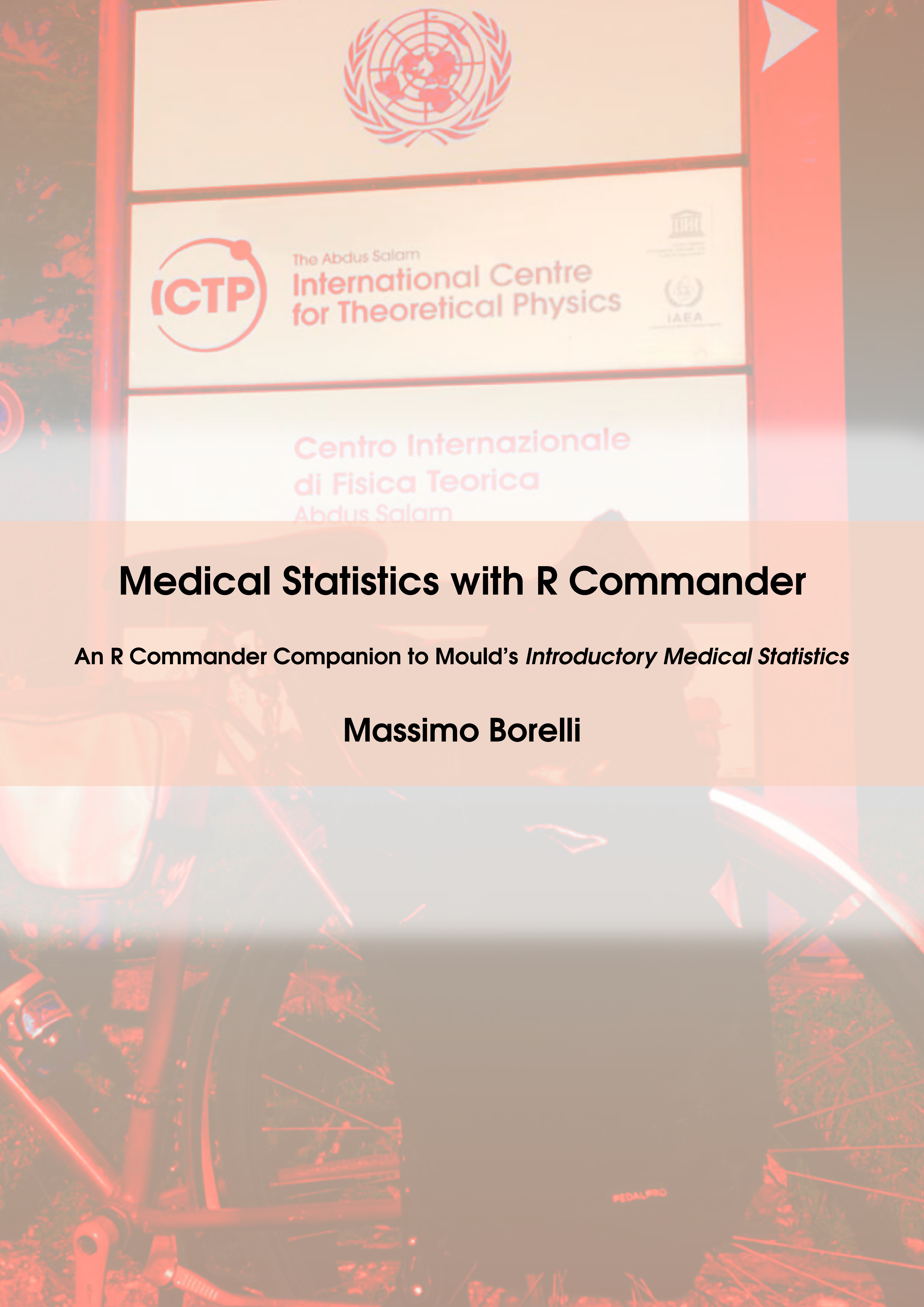


Medical Statistics with R Commander

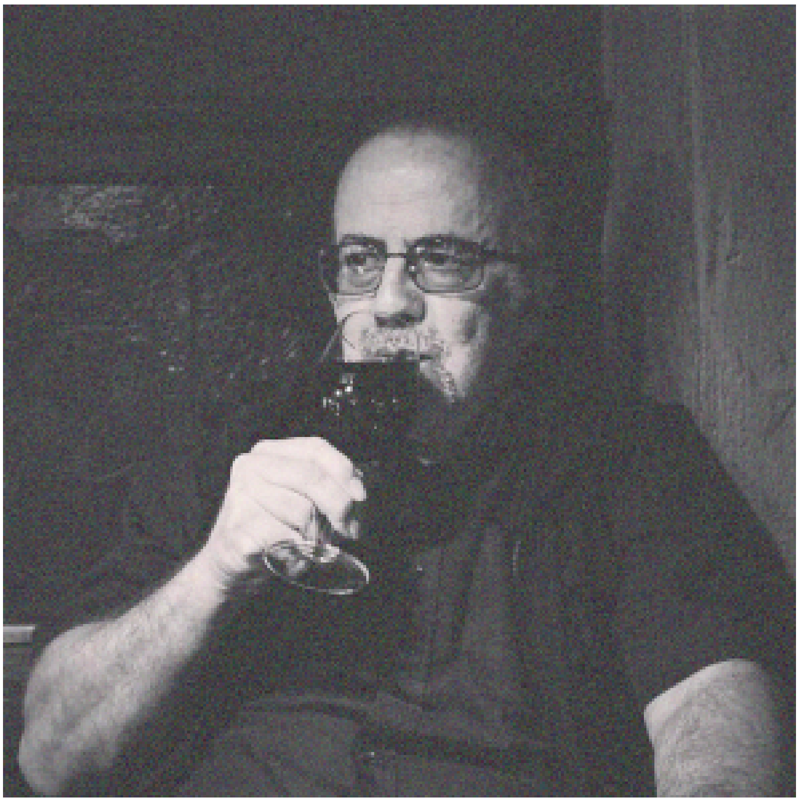
An R Commander Companion to Mould's *Introductory Medical Statistics*

Massimo Borelli

LECTURE NOTES FOR THE ICTP MASTER IN MEDICAL PHYSICS, TRIESTE, ITALY

`http://ictpmmp.weebly.com`

First printing, January 2020



in memoriam
professor Gianni Morrone, MD.

Contents

II Classical Inference

III Statistical Models

To describe data

1. Data Presentation

1.1 Introduction

This book collects the lectures of a short course in Medical Statistics held during the 2019 and 2020 editions of the Master in Medical Physics by the *Abdus Salam* International Center of Theoretical Physics in Miramare, Trieste, Italy. Our lecture notes are a ’computer-based companion’ to the course textbook [34] by Richard Francis Mould, *Introductory Medical Statistics*, available at the *Marie Curie* ICTP library with the 51-76 MOU identification code. To type this book we exploited, with thankfulness, the Mathias Legrand template available at `https://www.latextemplates.com/cat/books`.

1.1.1 The R Language

R is an open source software environment for statistical computing and graphics, which can be freely downloaded from a variety of CRAN (the Comprehensive R Archive Network) world-wide mirrors: `https://cran.r-project.org/mirrors.html`. R runs on UNIX/Linux, Windows and MacOS platforms. You can also exploit the cloud computing facilities, and compile online your script into `https://rdrr.io/snippets/`.

Nick Thieme has recently published an article[44] which recalls the astonishing success history of R, born twenty years ago in Auckland University by the ideas of two statistics professors: Ross Ihaka and Robert Gentleman. Other details are provided by Carlos Alberto Gómez Grajales in his *Created by statisticians for statisticians: How R took the world of statistics by storm* appeared on `http://www.statisticsviews.com/view/index.html`.

R is very well documented on the web; for instance, you can find free on line introductory books, as the Hadley Wickham and Garrett Grolemund textbook [52] *R for data science*, available at `https://r4ds.had.co.nz/`, or as the Kim Seefeld and Ernst Linder textbook *Statistics Using R with Biological Examples*, available at `https://cran.r-project.org/doc/contrib/Seefeld_StatsRBio.pdf`. There are also lots of webpages, blogs and Moocs concerning R; for instance:

- `http://ncss-tech.github.io/stats_for_soil_survey/chapters/`
- `http://www.sthda.com/english/wiki/r-software`

- Quick-R, `https://www.statmethods.net/`

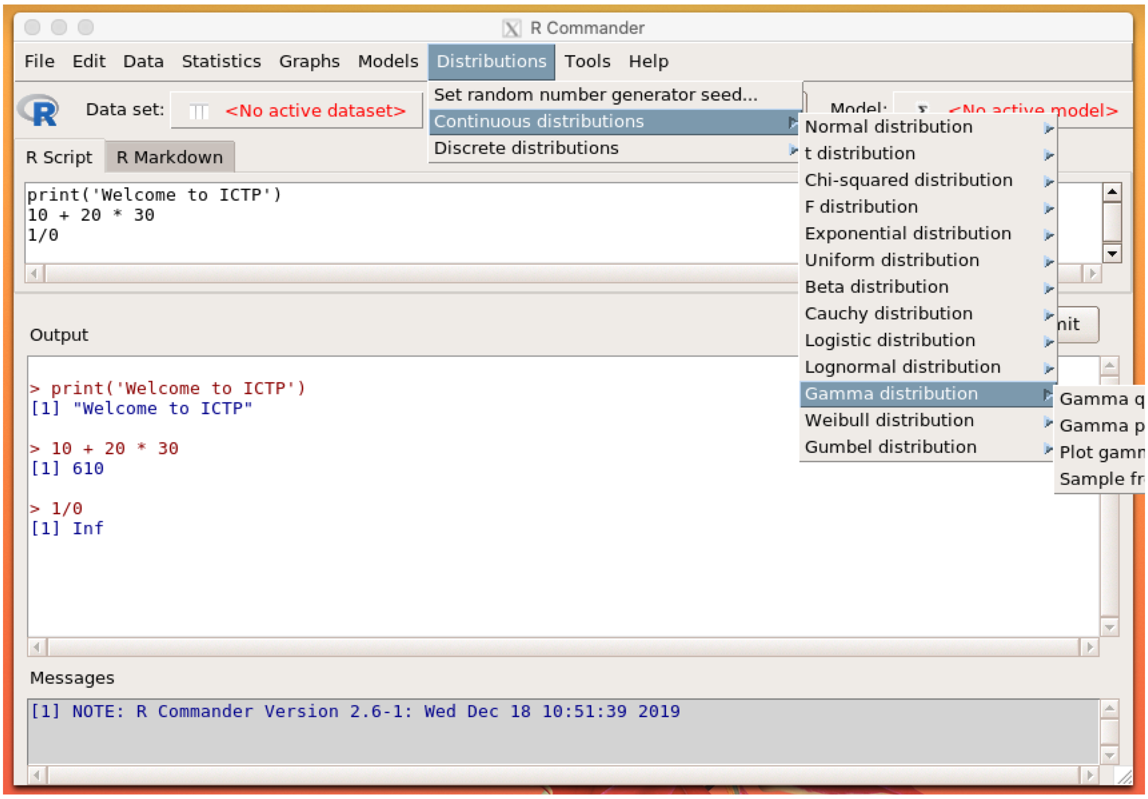
Many video tutorials are also available on YouTube, following the query `https://www.youtube.com/results?search_query=R+tutorial`.

1.1.2 The R graphical interfaces

Instead of working directly on the R Console, many scientists prefer to use R Studio `https://www.rstudio.com/` Integrated Development Environment (IDE). Beginners often find sufficient to access to a selection of commonly-used R commands using ’familiar’ graphical user interfaces, as R Commander, `https://www.rcommander.com/`, or as RKWard, `https://rkward.kde.org/`. In the present book we explain how to work with R Commander; for simplicity we suppose that both R and R Commander are already installed in your computer (otherwise, go back to previous section and search within YouTube tutorials how to do it: there are some differences between Windows or Linux procedures, which are simpler, and Mac, which needs a further little effort). Simply activate the R Commander within the R Console typing:

```
> library(Rcmdr)
```

Here an example R Commander appearance:



```
print('Welcome to ICTP')
10 + 20 * 30
1/0
```

```
> print('Welcome to ICTP')
[1] "Welcome to ICTP"

> 10 + 20 * 30
[1] 610

> 1/0
[1] Inf
```

```
[1] NOTE: R Commander Version 2.6-1: Wed Dec 18 10:51:39 2019
```

Figure 1.1: The R Commander environment, with its menus, the input section, the output section and the alert messages section.

Typically, you see a menu environment with an input section named R Script which has been created by the File New Script procedure, an Output section which lists the input commands and produces the outputs, and a Messages section which eventually alerts you for mismatches.

1.1.3 Possible alternatives

Of course, different laboratories and different scientists choose different softwares to analyse their data. For instance, commercial packages like SAS, SPSS, or S+. Or other programming languages, like Python. My favours also goes to the Slovenian python-based visual interface free software Orange, `https://orange.biolab.si/`.

Data Mining
Fruitful and Fun

Open source machine learning and data visualization for novice and expert. Interactive data analysis workflows with a large toolbox.

Figure 1.2: Orange is an open-source data visualization, machine learning and data mining toolkit. It features a visual programming front-end for explorative data analysis and interactive data visualization, and can also be used as a Python library.

1.1.4 Let us start making some practice with R

Let us start exploring R capabilities by loading and manipulating the famous Ronald Fisher [18] / Edgar Anderson [4] `iris` didactical dataset.

We start a new script, simply entering the command:

```
> iris
```

Scrolling up and down the output, we recognise a dataset of 150 rows and 5 columns, named respectively `Sepal.Length`, `Sepal.Width`, `Petal.Length`, `Petal.Width` and `Species`. The first four columns provide numerical data, while the last column provide qualitative information about the (three) different species of flowers considered.

Exercise 1.1 Try to understand what happens when inserting the following commands (similarly to what professor Michael Crawley does in his reknowned reference book [12], *TheRBook*) :

```
> iris[1,]
> iris[1:6,]
> head(iris)
> iris[145:150,]
> tail(iris)
> iris[,1]
> iris[,c(3,4,5)]
> iris[,3:5]
> names(iris)
```

■

Observe that:

```
> length(iris)
```

returns the number of the columns in the dataset, while if one tries, for instance:

```
> length(Species)
```

obtains a not-found error.

Exercise 1.2 Try what follows:

```
> length(iris$Species)
> with(iris, length(Species))
> attach(iris)
> length(Species)
```

Basic iris information can also be immediately retrieved by exploring the dataset structure:

```
> str(iris)
```

As we see, a typical bio/medical dataset collects both numeric (i.e. **quantitative**) and factor (i.e. **qualitative**, **nominal**), and the different 'groups' inside the factor are called levels.

Exercise 1.3 Try what follows, and discover at the end the R boolean constants:

```
> levels(Species)
> levels(Species)[2]
> is.numeric(Species)
> is.factor(Species)
> is.factor(Petal.Length)
```

Datasets can be easily sorted and filtered in R.

Exercise 1.4 Try what follows and discuss the output:

```
> iris[order(Sepal.Length), ]
> iris[order(Sepal.Width), ]
> iris[order(Sepal.Length, Sepal.Width), ]
> iris[rev(order(Sepal.Length)), ]
> iris[Species == "virginica",]
> iris[(Species == "virginica") & (Sepal.Length == 6.3),]
> iris[(Species == "virginica") & (Sepal.Length != 6.3),]
```

The iris dataset is an example of what is called a *complete dataset*, as all columns ('fields', 'variables', 'instances') and all rows ('records', 'observations') has a known value. On the contrary, let us consider the New York spring-summer 1973 airquality dataset.

```
> ? airquality
> attach(airquality)
> head(airquality)
```

Vocabulary 1.1 — incomplete dataset. A dataset is said to be **incomplete** when we observe some **missing data**, or **missing values**, that R represents with the symbol `NA`. In the example above, we note that during May 5th neither the `Ozone` nor the `Solar.Radiation` have been collected.

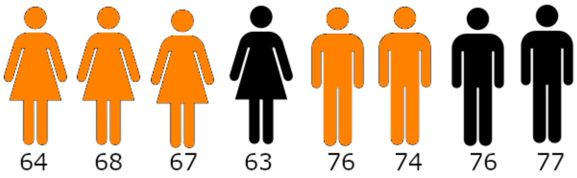
In order to manage the missing values, we could decide to drop them away (this is a typical need when working with regression trees or when training machine learning algorithms), or to identify records in which missing values are present. Here we go:

Exercise 1.5 Try what follows and discuss the output:

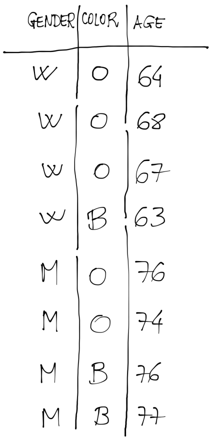
```
> na.omit(airquality)
> complete.cases(airquality)
> which(complete.cases(airquality) == TRUE)
```

1.1.5 Let us start making some practice with R Commander

In this paragraph we want to learn ho to create a simple dataset and how to import it into R Commander. Consider a didactical situation of eight patients, of different gender (women and men), with a different health situation (orange and black) and of different age.

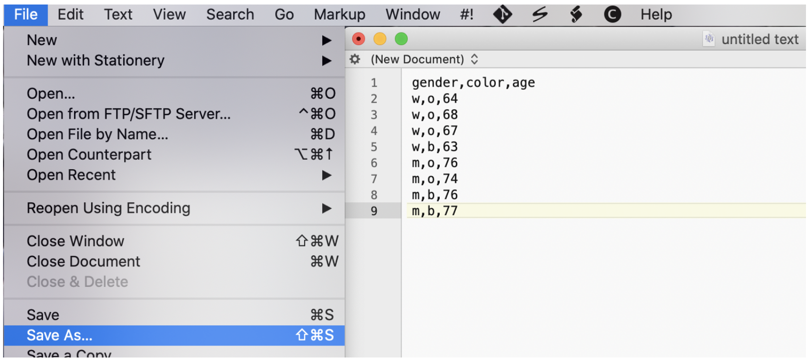


Just to start, we create 'by hand' a dataset to describe the above depicted situation. To create it, we need to arrange the data into a rectangular table, similar to the following:



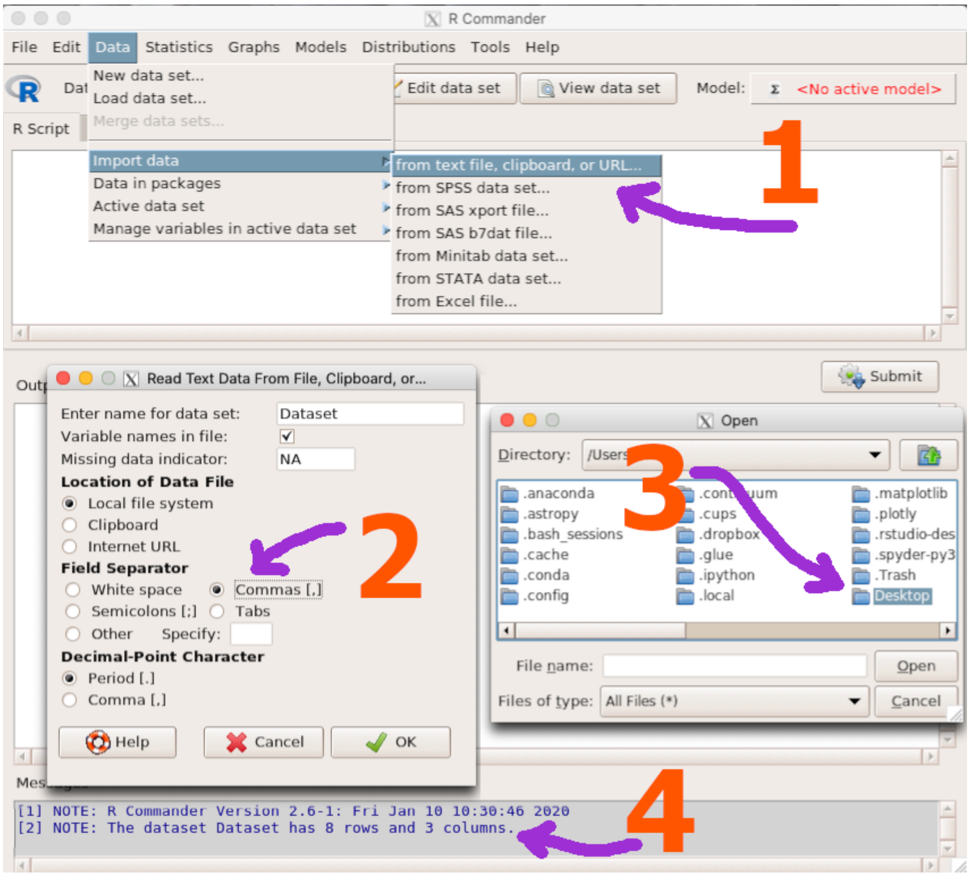
GENDER	COLOR	AGE
W	O	64
W	O	68
W	O	67
W	B	63
M	O	76
M	O	74
M	B	76
M	B	77

Usually this task is performed using a spreadsheet (like MS Excel, Open Office - Libre Office Calc, Google Sheet and so on) or by means of a 'table' exported from some database. For simplicity, we type the information into a text editor (for instance Windows Notepad, Bare Bones BBEdit, SciTE / Geany, ..), using the comma symbol to separate the data:



```
gender,color,age
w,o,64
w,o,68
w,o,67
w,b,63
m,o,76
m,o,74
m,b,76
m,b,77
```

After saving this simple text with the file name `example.csv` (it means that data are separated by the comma: csv = comma separated values), we can import the dataset `example.csv` into R Commander following these steps. We (1) *import data from text file* from the *Data* menu; (2) we specify that our field separator is the comma; (3) we scroll the dialogue window to search where `example.csv` is located - in our case, the Desktop; and after opening it we obtain a blue message confirming that an 8 rows and 3 columns dataset has been imported into R Commander.

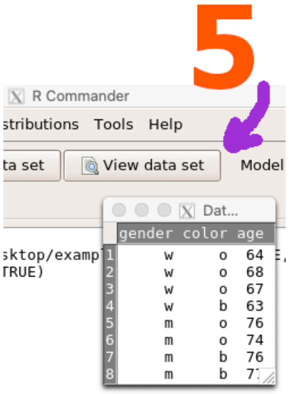


Enter name for data set: Dataset
Variable names in file: ☑
Missing data indicator: NA
Location of Data File
● Local file system
○ Clipboard
○ Internet URL
Field Separator
○ White space ● Commas [,]
○ Semicolons [;] ○ Tabs
○ Other Specify:
Decimal-Point Character
● Period [.]
○ Comma [,]

Directory: /Users
File name:
Files of type: All Files (*)

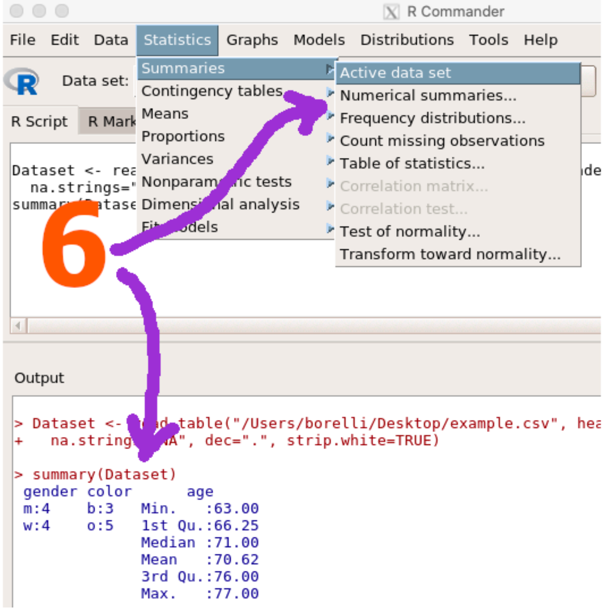
```
[1] NOTE: R Commander Version 2.6-1: Fri Jan 10 10:30:46 2020
[2] NOTE: The dataset Dataset has 8 rows and 3 columns.
```

To scroll the dataset (5) one can click the *View data set* button; and to quickly obtain some descriptive statistics, as described in the next Chapter, one can follow the path (6) *Statistics / Summaries / Active data set*, as depicted in the next page figure.



```
sktop/examp   E,
TRUE)
```

	gender	color	age
1	w	o	64
2	w	o	68
3	w	o	67
4	w	b	63
5	m	o	76
6	m	o	74
7	m	b	76
8	m	b	77



```
Dataset <- rea                                      ide
  na.strings="
summary(Datase
```

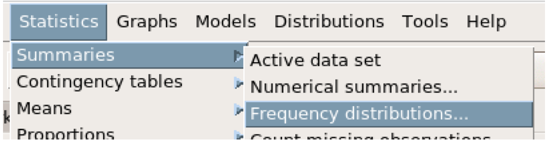
```
> Dataset <- read.table("/Users/borelli/Desktop/example.csv", hea
+   na.strings="NA", dec=".", strip.white=TRUE)

> summary(Dataset)
 gender color      age
 m:4    b:3    Min.   :63.00
 w:4    o:5    1st Qu.:66.25
               Median :71.00
               Mean   :70.62
               3rd Qu.:76.00
               Max.   :77.00
```

1.2 Charts and Tables

After having read the Chapter 1 of Richard Mould textbook [34], we can discover the powerful graphical capabilities of R; for instance, have a look to `https://www.r-graph-gallery.com/`. To start, when having qualitative/nominal data, i.e. a `factor`, it is very simple to obtain a table reporting the **absolute frequencies** encountered. Here we have an example:

```
> table(Species)
```

In R Commander one can follow the *Statistics / Summaries / Frequency distributions* menu:

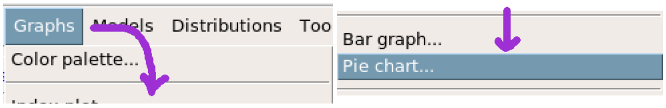

Statistics Graphs Models Distributions Tools Help
Summaries
Contingency tables
Means
Proportions
Active data set
Numerical summaries...
Frequency distributions...
Count missing observations

Vocabulary 1.2 — balanced dataset. The `iris` dataset is said to be **balanced** as we observe data with the same absolute frequencies in each group considered. In our example, fifty flowers belonging to each of the (`levels` of the) `Species` `setosa`, `versicolor` and `virginica` have been measured.

After creating a table, it is straightforward to draw a **pie chart** or a **barplot** - **bar graph** (and have a look to professor Tian Zheng 'Colors in R' `http://www.stat.columbia.edu/~tzheng/files/Rcolor.pdf` to choose the perfect chromatic outfit of your wonderful picture):

```
> pie(table(Species))
> barplot(table(Species))
> barplot(table(Species), col = c("orange3", "orange2", "orange1"))
```

Wtih R Commander one simply scroll the *Graphs* menu:



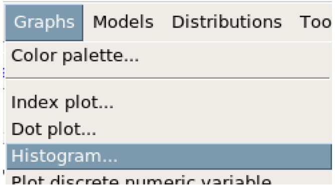
Exercise 1.6 What about a table in which not absolute but **relative frequencies** appears? (*Hint: try to exploit the* `length` *and the* `round` *commands.*) ■

R **Further readings.** To study in deep the statistics of categorical data we suggest to start with the classical Alan Agresti textbook [1], *An introduction to categorical data analysis*, or the recent M. Friendly and D. Meyer *Discrete data analysis with R: visualization and modeling techniques for categorical and count data* [20]. If you need to do HTML or LaTeXtables, read `https://www.r-bloggers.com/getting-tables-from-r-output/`.

1.3 Histograms

Let us move from `factor` to `numeric` data, and let us carefully read what Richard Mould [34] writes in his 1.4 paragraph:

> In a histogram, the height of each vertical block does not always represent the value of the variable of interest (unless the width of the block is unity), as is the case of a bar in a bar chart. Also, in a histogram, the horizontal scale is continuous and not, like the bar charts, discrete. Also, unlike a bar chart width, a histogram block width *does have a meaning*.



Let us explain in a precise way the following idea of relative frequency histogram, which is a central concept linked to the Mould's 2.6 Probability Density Function paragraph. We now follow the Sergio Invernizzi (italian language written) textbook [25]. Let $x = (x_1, x_2, \ldots, x_n)$ be the n `numeric` data considered and let $c_1 < c_2 < c_3 < \ldots < c_r$, $2 \leq r < n$, a class partition with **cut-off** c_j's, such that $c_1 = min(x)$ and $c_r = max(x)$. We obtain $r-1$ limited disjoint **classes** (or **bins**):

$$C_1 = [c_1, c_2]\ ,\ C_2 = (c_2, c_3]\ ,\ C_3 = (c_3, c_4]\ , \ldots,\ C_{r-1} = (c_{r-1}, c_r]$$

Denote with n_j the absolute frequencies of the x data falling into each class C_j, and let $f_j = n_j/n$ the relative frequencies $(1 \leq j \leq r-1)$. With these choices, the **relative frequency histogram** is made by $r-1$ rectangles of bases C_j and heights:

$$h_j = \frac{n_j/n}{c_{j+1} - c_j}$$

Let's make a step-by-step check, assuming to be x the `Petal.Length` in `iris` and wishing to draw a simple $r = 2$ column histogram, with $c_2 = 5.0$ as cut-off.

To start, we search for the minimum and maximum value of `Petal.Length` (note that if we insert R sintax enclosed in round brackets we immediately obtain the screen printed output):

```
> (c1 = min(Petal.Length))
> (c3 = max(Petal.Length))
> hist(Petal.Length, breaks = c(1, 5, 6.9))
```

Now we compute the bases and the heights of the two rectangular blocks:

```
> (n1 = sum(Petal.Length <= 5))
> (n2 = sum(Petal.Length > 5))
> (C1 = (5 - c1))
> (h1 = (n1/150) / C1 )
> (C2 = (c3 - 5))
> (h2 = (n2/150) / C2)
```

Exercise 1.7 Please, visually check the values obtained by algebraic calculations and confront them with the histogram. Then, quickly answer to the following question:

$$4 \cdot 0.18 + 1.9 \cdot 0.1473684 = ?$$

Now go back to Mould's quotation and reflect on those words. ■

1.4 Scatter Diagrams

It is simple to draw a cartesian x-y **scatter plot**:

```
> plot(Petal.Length, Petal.Width)
```

and to add colors information with the `col` option:

```
> plot(Petal.Length, Petal.Width, col = Species)
```

On the cartesian plane you can also sketch mathematical functions: suppose you are interested in drawing the $y = 2.37 \cdot e^{-0.68 \cdot x}$ function on the interval $x \in [-4, 4]$. One possibility is to define an R **function**; here it is the code:

```
> expo = function(s){ 2.37 * exp(-0.68 * s) }
> x = seq(from = -4, to = 4, by = 0.01)
> y = expo(x)
> plot(x,y)
```

Exercise 1.8 Explore the help menu to discover a number of graphical possibilities:

```
> ? lines
> ? points
> ? par
```

■

R **Further readings.** To obtain very elegant and readble scatter plots, even handling big data, try to use the ggplot2 library, which belongs to the tidyverse 'ecosystem' `https://ggplot2.tidyverse.org/`. In detail, refer to the online free book `https://r4ds.had.co.nz/` *R for Data Science*, by Garrett Grolemund and Hadley Wickham [52]. Moreover, Selva Prabhakaran in his *r-statistics.co* blog offers a free tutorial, `http://r-statistics.co/Complete-Ggplot2-Tutorial-Part1-With-R-Code.html`.

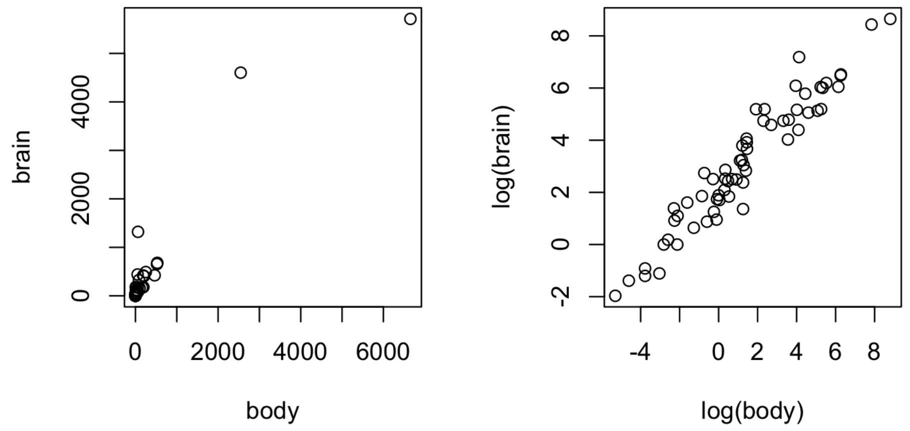
1.5 Linear and Logarithimic Axes

One of the 'historical' R reference books is the *Modern Applied Statistics with S* [46] of Bill Venables & Brian Ripley. They have also released an add-on R package collecting various dataset discussed into their textbook, and we need to load the mammals one:

```
> library(MASS)
> attach(mammals)
```

Scrolling the dataset, you will find the huge 6 tons african elephant and the minuscule 5 grams canadian shrew, and clearly it is difficult to depict the situation using `plot(body, brain)`, as you see on the left panel below. Therefore, changing the coordinate axes by the log function is a natural idea:

```
plot(log(body), log(brain))
```



1.6 More plots

While pie charts and histograms can be used to depict a situation (respectively described by `factor` and `numeric` variables) in an **univariate** statistical analysis, the scatter plot is a typical graph used in the **bivariate** statistical analysis within two `numeric` variables. In the next Chapters, we are going to introduce more useful graphs, as boxplots (section 2.4), mosaic plots (section 3.1) and ROC curves (section 3.1.2).

2. Describing Distributions

2.1 Introduction

Besides simple graphical information, we shall need to 'summarise', to 'abridge' a dataset by means of numerical information. To practice, we try to analyze a dataset concerning the incidence of leukemia, lymphoma and multiple myeloma among atomic bomb survivors (1950-2001), as reported in the 22.4 paragraph of Mould's textbook. We therefore retrieve (after having registered our names and affiliations) the 5.1 Mb `lsshempy` dataset, published in `https://www.rerf.or.jp/en/library/data-en/`. After having downloaded and unzipped the folder, you can import the dataset into R with the `read.csv` command:

```
> lsshempy = read.csv(file.choose(), header = TRUE)
> attach(lsshempy)
```

2.2 Mean, Mode and Median: the measure of central tendency

In his 2.2 paragraph, Richard Mould [34] recalls three **measures of central tendency** (or as he and Bernard Rosner [38] prefer, three **measures of location**).

To find the **mode** of a `factor` variable, we simply use the `table` instruction:

```
> table(sex)
> max(table(sex))
> which( max(table(sex)) == table(sex))
```

and we see that the 2's (Female) appears 19827 times, against the 18751 1's (Male); therefore, Female is the modal character of the distribution.

Exercise 2.1 Discover what happens in R if you invoke the `mode` function:

```
> ? mode
```

Note that the mode is the typical central tendency measure of `factor` data. Nevertheless, it is also correct to deal with the mode in a `numeric` variable; as an example, in the 3.2.1 section we will discuss of the famous bimodal female vs. male height distribution.

Now, explore `agxcat` (Age at exposure categories) and `mar_ag` (Person year weighted mean weighted adjusted truncated DS02 Bone Marrow Gamma) variables: they are both `numeric` variables, but they belong, in a sense, to different mathematical sets - the latter belongs to the continuous line $\mathbb{R}$, the former to the ordered ring $\mathbb{Z}$ (some authors, like Martin Bland[7], use the term **discrete** versus **continuous** data).

Therefore, to 'summarise' the continuous `mar_ag` variable the arithmetic `mean` (approximatively 544.7) appears to be suitable:

```
> mean(mar_ag)
```

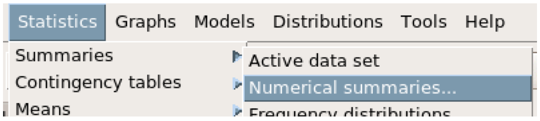
If you need to quickly calculate the mean within groups (for instance, the means of the bone marrow gamma rays person-year in `city` 1 (Hiroshima) and 2 (Nagasaki)), one can exploit the `tapply` function:

```
> tapply(mar_ag, city, mean)
```

On the contrary, to 'summarise' the discrete `agxcat` variable the arithmetic mean is not a clever idea; it is wise to exploit the `median`, which is 6:

```
> median(agxcat)
```

2.3 Standard Deviation, Variance and other measure of dispersion



In his 2.4 paragraph, Richard Mould [34] use the term **measures of shapes** (many other authors prefer to say **measures of dispersions**) to introduce the standard deviation and the variance of a `numeric` variable, which can be easily calculated with:

```
> sd(mar_ag)
> var(mar_ag)
```

Exercise 2.2 Guess what happens when in R you type:

```
> sd(mar_ag)^2 == var(mar_ag)
> sd(mar_ag) == sqrt(var(mar_ag))
```

R **Remark.** Observe that R possess only the `sd` function to compute the standard deviation, while the spreadsheets (MS Excel, O.O. Calc, Google Sheets, ..) have – at least! – a couple of functions, usually named STDEV.S and STDEV.P. Doing statistics, we are almost always involved with data coming from a **sample**, i.e. a subset of the whole **population**. And, having $x = (x_1, x_2, \ldots, x_n)$ a **sample size** n, to compute the standard deviation we need to calculate in advance the sample mean $M = \sum x_j / n$; therefore, having estimated M, it is necessary and sufficient to know only $(x_1, x_2, \ldots, x_{n-1})$ to algebrically deduce x_n – exploiting the M information. This is the reason why Statisticians refers to Physics in saying that there are only $n-1$ **degrees of freedom** in estimating the sample standard deviation: and this is the reason why in the formula of the 2.4 Mould's[34] paragraph that 'strange' $n-1$ denominator appears. While, working with a whole population an n denominator is required (but in biostatistics this seldom occurs).

When we deal with ordered (discrete) data, as in `agxcat` variable, there are many possibilities to choose. Suppose [25] , without loss of generality, that the sample $x = (x_1, x_2, \ldots, x_n)$ is already ordered, $x_1 \leq x_2 \leq \ldots \leq x_n$. We introduce here the **quantiles**.

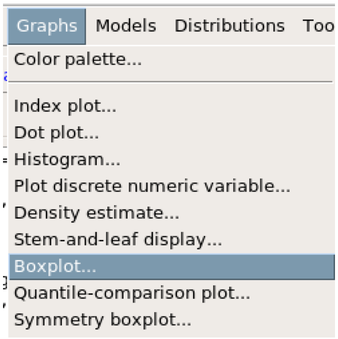
Vocabulary 2.1 — Quantiles. Let us denote with L the median of x: L divide the sample x into two subsets, the first half and the second half. If we compute the medians of those two halves we obtain respectively the **first quartile** $Q1$ and the **third quartile** $Q3$ (being the median L the second quartile, $\min(x)$ the zeroth quartile and $\max(x)$ the fourth quartile). If we split x in ten sections instead of two, one can define the first, second, ... **deciles**. And again, splitting x in one hundred sections, we compute the **percentiles**. Quartiles, deciles and percentiles are examples of **quantiles**.

Exercise 2.3 Type in R what follows and discuss the results:

```
> median(agxcat)
> quantile(agxcat, 0.50)
> summary(agxcat)
> quantile(agxcat, 0.25)
> quantile(agxcat, 1)
```

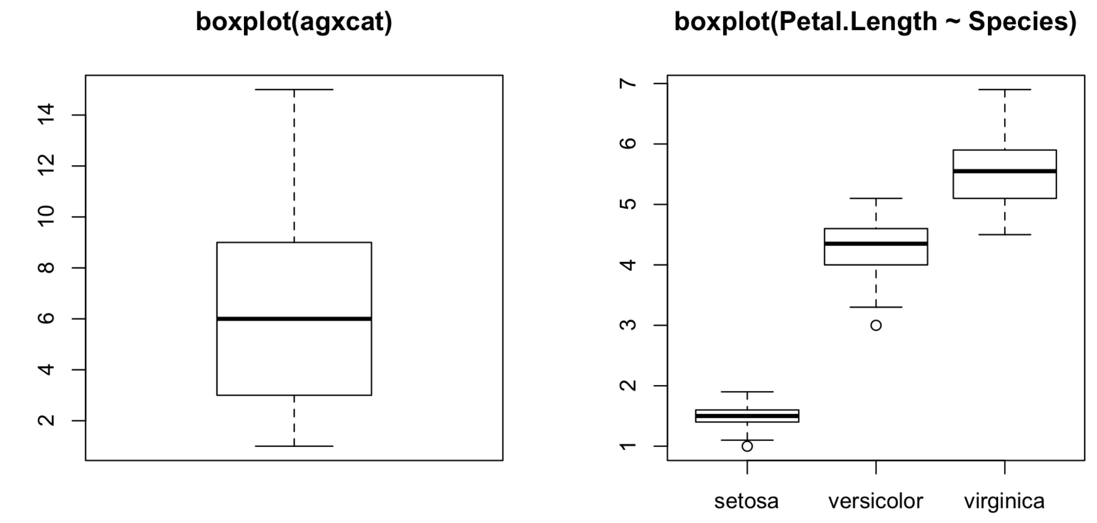
■

2.4 Box and Whiskers Plot



The legendary chemist, mathematician and statistician John W. Tukey, `https://en.wikipedia.org/wiki/John_Tukey`, introduced this type of data visualization:
The left panel has been obtained with the command:

```
> boxplot(agxcat)
```



it depicts the elements provided by the `summary`, i.e. `min(agxcat)`, `quantile(agxcat, 0.25)`, `quantile(agxcat, 0.50)` in **bold**, `quantile(agxcat, 0.75)` and `max(agxcat)`. The spacings between the different parts of the box (which, of course, encompasses the 50 per cent of the data) indicate the 'degree' of dispersion (spread) and the skewness – as discussed in the next 2.5 paragraph – in the data. The wiskers describe the **tails** of the distribution.
The right panel refers to `iris` dataset and it has been drawn with the command (note the use of the tilde character, ∼, which for example can be recalled by the keyboard numeric pad with the combination ALT+126):

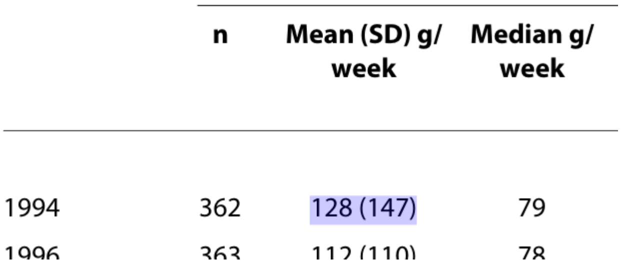
```
> boxplot(Petal.Length ~ Species)
```

As you see, in `setosa` and in `versicolor` boxplots some isolated points appear. They are the so-called **outliers**, as defined by Tukey himself: consider the **interquartile range** , $IRQ = Q3 - Q1$, 'amplify' it by a 50% , $1.5 \cdot IRQ$, and search if there are points $x_j \in x$ such that $x_j < Q1 - 1.5 \cdot IRQ$ or $x_j > Q3 + 1.5 \cdot IRQ$. It can be shown (e.g. [24, page 29]) that outliers are not so rare in experimental measures: asymptotically, 0.7% of data.

The boxplot is an univariate graph. But there exist – although not so frequently used (unfortunately, I say) – the Rousseeuw & Ruts & Tukey bivariate **bagplot**, `https://en.wikipedia.org/wiki/Bagplot`, available in the 'Another PLot PACKage' `aplpack` [39].

2.5 Skewness

Once upon a time, the **skewness** (`https://en.wikipedia.org/wiki/Skewness`) measure of asymmetry and the **kurtosis** (`https://en.wikipedia.org/wiki/Kurtosis`) measure of 'fat tails' were commonly calculated and used in literature to describe data distribution. Nowadays these concepts seems to be buried in dust: ever R do not possess 'standard' function to compute them, but you need – for instance – to load the `library(fBasics)` and to call the `basicStats` function.
Nevertheless, skewness plays an important role – and a boxplot reveals it immediately – when our mind try to perceive the data distribution only knowing some descriptive statistics, as it happens when reading literature. Let us make an example to be more clear.



	n	Mean (SD) g/ week	Median g/ week
1994	362	128 (147)	79
1996	363	112 (110)	78

Consider the study of professor Kersti Pärna ahd her colleagues regarding the alcohol consumption in Estonia and Finland, `https://doi.org/10.1186/1471-2458-10-261`. Have a look to their Table 5:
Unaware readers might not perceive – as very often in literature it is (mis)used the symbol 'Mean ± SD' – the skewness of the data, and (joking) they could mantein that, as $128 - 147 = -19$, in Estonia there exist some drinkers whose body do not consume, but 'produce', alcohol. This is the reason why, when data are skewed, I prefer to avoid to describe them by mean(sd), preferring to use the Tukey five numbers `summary`.
Skewness, and kurtosis, are in effect 'the responsibles' for the fact that the famous Čebišev inequality (`http://mathworld.wolfram.com/ChebyshevInequality.html`) is so 'poor': it would be possible to create artificial data x, all of them very far from the mean, such that $P(|x - M| \geq S) \leq 1$.

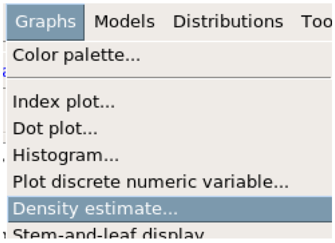
2.6 Coefficient of Variation

> Coefficients of variation are particularly useful when observations with different dimensions are being compared, such as UK sterling and US Dollars. A dimensionless measure of dispersion is then very convenient. (R. Mould, 2.5 [34])

Remembering how the R `function` was introduced in section 1.4, we propose you to solve this exercise:

Exercise 2.4 Write an R `function` called `cv` that computes the coefficient of variation of a numeric vector (*Hint: exploit the inner functions* `sd` *and* `mean`). Check that `agxcat` and `mar_ag` have a coefficient of variation of, respectively, 59.3% and 143%. ■

2.7 Probability Density Function



In the next chapter 3.1 we will recall some basic aspects of probability. But, in his 2.6 paragraph[34], R. Mould recalls the meaning of a density function in view to discuss further the concept of **continuous random variable**, from the analytical point of view (i.e. on vector space $\mathbb{R}$). Let us now recall that in Medical Statistics very often one deals also with **finite random variables**. Consider for instance the Table 4.3 of the *Fundamentals of Biostatistics* of Bernard Rosner [38, page 84], concerning the number of episodes of otitis media in the first two years of life. The finite random variable associated is represented by a 'two rows matrix':

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 \\ 0.129 & 0.264 & 0.271 & 0.185 & 0.095 & 0.039 & 0.017 \end{pmatrix}$$

The first row describe all the possible **events**, while the second row precise their single success probability; and the function which associates the event to its probability is called **probability mass function**, or *discrete density function*. We will discuss better those aspects later in paragraph 3.2 , talking of binomial and of Poisson random variables.

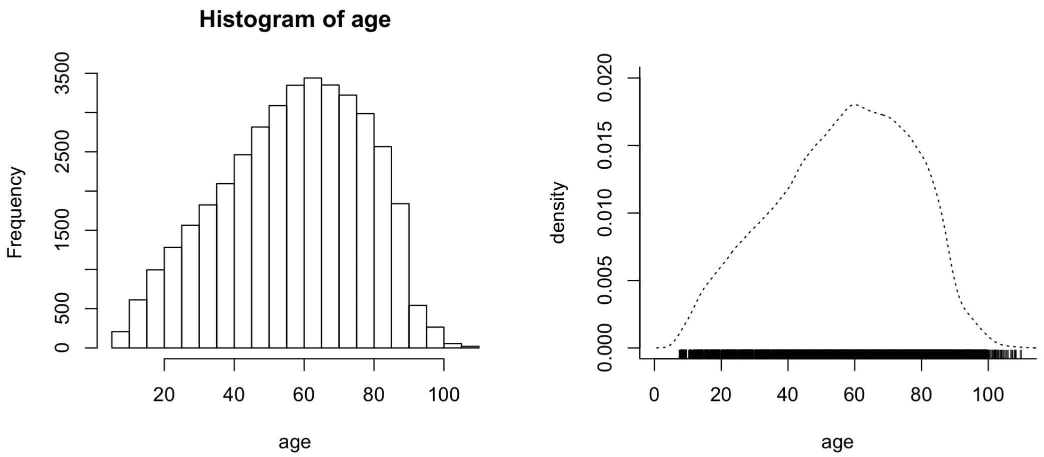


Figure 2.1: Estimating a continuous `density` function

Turning back to continuous density functions, let us recall what Venables and Ripley explain very well in their Figure 5.8 [46, pages 127-128]:

> The histogram with `probability = T` is of course an estimator of the (*continuous*) density function. The histogram depends on the starting point of the grid of bins. The effect can be surprisingly large.

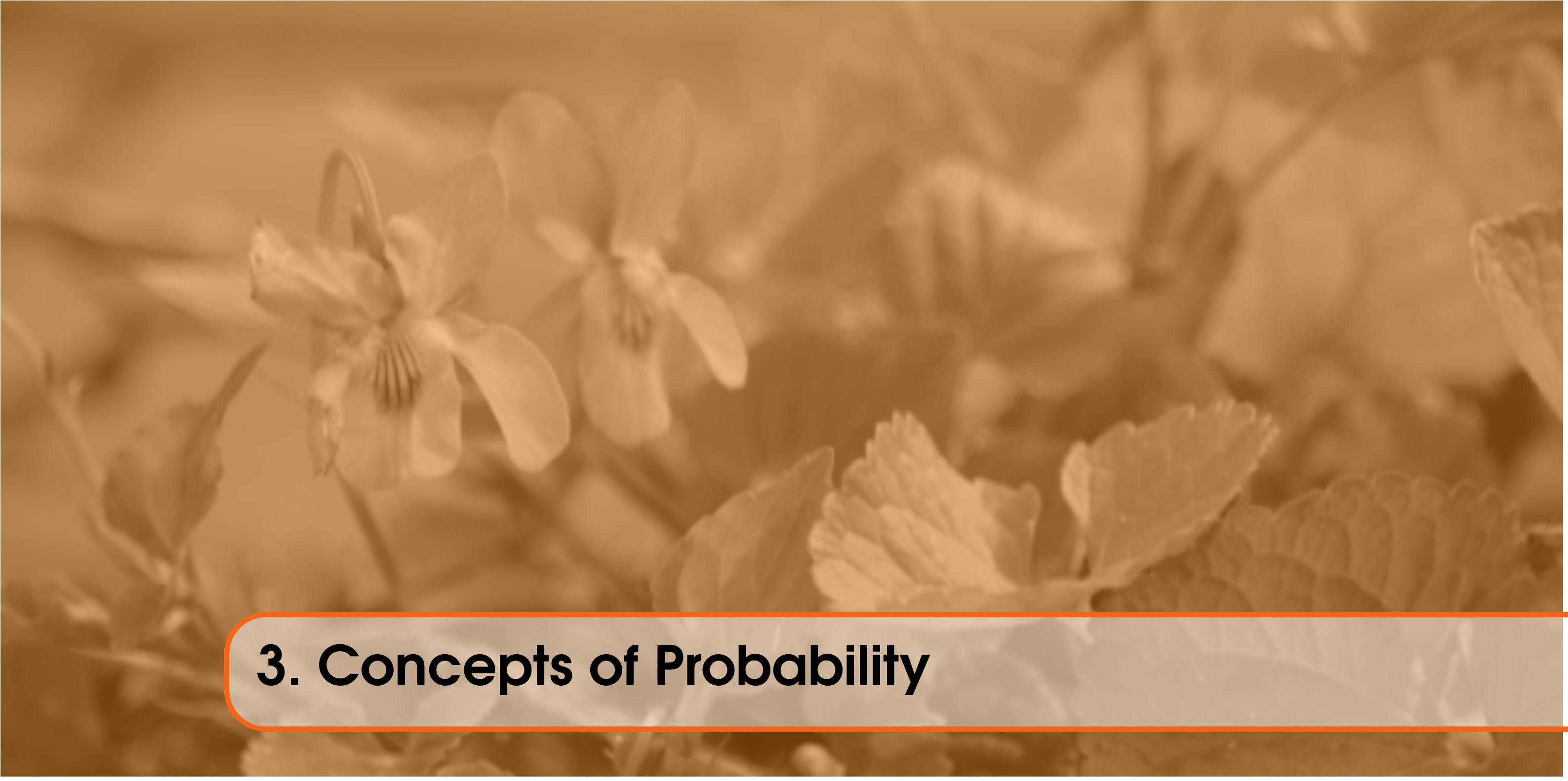
R language possesses the `density` function which (depending on the user's choice of a bandwidth and of a kernel) fits a numerically estimated density function, as in Figure 2.1.

Exercise 2.5 Referring to the `lsshempy` person-year weighted mean attained `age`, insert in R the following code, and discuss step-by-step its syntax:

```
> par(mfrow = c(1,2))
> hist(age)
> plot(x = c(0, 110), y = c(0, .02), type = "n", bty = "l", xlab = "age",
ylab = "density")
> lines(density(age), lty = 3)
> rug(age)
```

II

Classical Inference



3. Concepts of Probability

3.1 Overview of Probability

Usually, a graduate Physicist knows a lot of probability (on average, more than a graduate Mathematician, at least here in Italy) and therefore

Here we are simply going to recall some practical aspects of applied probability; we are moving ourselves in the framework of a **cross-section** experimental design, as explained in the 23.5 paragraph of Mould's textbook [34].
Ten years ago, Richard Moore et al. published [33] the 'R.O.M.A., Risk of Ovarian Malignancy Algorithm', a method to estimate benign vs. malignant probability in an ovarian mass. Shadi Najaf, a gynæcologist now at the Kantonsspital Baden, Zürich (Swiss), explored the possibility to enhance that algorithm. Her `roma` dataset can be loaded and explored in R directly from the web with these instructions:

```
> www = "http://www.biostatisticaumg.it/dataset/roma.csv"
> roma = read.csv(www, header = TRUE)
> attach(roma)
> head(roma)
> tail(roma)
```

As you see with the `tail` command, Shadi Najaf observed 210 patients with an ovarian mass, and she researched whether the `Histology` may be associated, in a statistical sense, to `AgePatients`, to their `Menopause` status, and to four candidate biomarkers (logaritmic transformed): `logHE4`, `logCA125`, `logCA19.9` and `logCEA`. Let us start exploring `Menopause` and `Histology` with a **contingency table**:

```
> table(Histology, Menopause)
```

	ante	**post**	sum
benign	106	65	171
malignant	12	27	39
sum	118	92	210

Table 3.1: Menopausal status as a possible predictor of malignancy in ovarian cancer in the Shadi Najaf research.

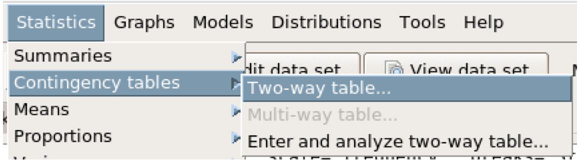


Figure 3.1: How to create a contingency table with R Commander

Exercise 3.1 Explore the output of the following instructions:

```
> thm = table(Histology, Menopause)
> thm
> thm[2,]
> thm[,2]
> thm[1:4]
> thm[4]
> thm[2,2]
> sum(thm[1:4])
> sum(thm)
> plot(thm, col = c("orange", "violet"))
```

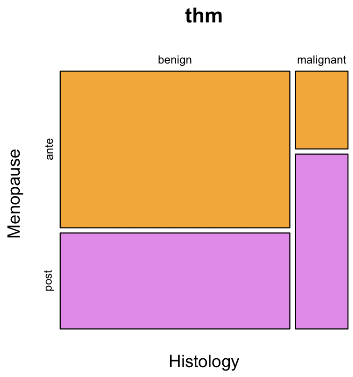


Figure 3.2: A mosaic plot.

Figure 3.2 is called a **mosaic plot**, and the area of the four rectangles is proportional to the counts in the cells of (the transpose of) Table 3.1; orange coloured rectangles depict the `ante` `Menopause` women. In Table 3.1 we see that 39 women over 210 has been diagnosed with a malignant ovarian tumor; so one could estimate the **relative frequency**, i.e. an estimate of the disease **(frequentist) probability** to be around the 19 percent (of course not within the whole healthy population, but within women with certain precise symptoms known to the Gynæcologists):

$$P(malignant) = \frac{39}{210} = 0.186...$$

Vocabulary 3.1 — Prevalence. In a cross-section design, the **prevalence** of the disease into a selected subpopulation described by some precise **inclusion criteria** is represented by its (frequentist marginal) probability.

Exercise 3.2 Try what follows:

```
> sum(thm[2,])/sum(thm)
> 39/210
```

■

Such marginal probability does not distinguish whether women are in their ante-menopausal or post-menopausal status. So we look to the inner columns of the table, i.e. we estimate the **conditional probability**:

$$Pr(malignant|ante) = \frac{12}{118} = 0.102...$$
$$Pr(malignant|post) = \frac{27}{92} = 0.293...$$

Those numbers appears to be different (and in the mosaic plot 3.2 the 'horizontal aisle' is not straight): a post-menopausal woman appears to have a triple risk than an ante-menopausal woman. Therefore, we can argue that `Menopause` and `Histology` are not **independent events**, but they are (in a statistical sense to be better precised later, in section 5.3.4) **associate events**.
By the way, we recall here two commonly used **association measure**; the first is the **odds ratio**:

$$O.R. = \frac{106 \cdot 27}{65 \cdot 12} = 3.67$$

and when O.R. is 'far away from' 1 (i.e. close to 0 or to $+\infty$), then rows – and columns – are 'far away' from proportionality, and therefore one event (e.g. menopausal status ante / post) provide 'a certain quantity of information' to the other event (e.g. to be `ante` / `post` inform us on `benign` / `malignant` response). Another common association measure is the **relative risk** (i.e. the ratio of the conditional probabilities):

$$RR = \frac{\frac{27}{92}}{\frac{12}{118}} = \frac{27}{92} \cdot \frac{118}{12} = 2.89$$

Exercise 3.3 The odds ratio can be found by the following command, later explained:

```
> fisher.test(thm)
```

■

Exercise 3.4 Relative risk and odds ratio are dependent. Explain:

```
> OR = (27*106)/(12*65)
> RR = (27*118)/(92*12)
> RR
> OR + (1-OR)*(27/92)
> OR*(65/92) + (27/92)
```

In a contingency table, marginal probabilities and conditional probabilities are ruled by the famous **Bayes theorem**:

$$P(malignant|ante) = \frac{P(ante|malignant)}{P(ante)} \cdot P(malignant)$$

The proof is straightforward. Let us simply check the relation in our example:

$$\frac{12}{118} = \frac{(12/39)}{(118/210)} \cdot \frac{39}{210}$$

$$\frac{12}{118} = \frac{12}{39} \cdot \frac{210}{118} \cdot \frac{39}{210}$$

$$\frac{12}{118} = \frac{12}{118}$$

3.1.1 Sensitivity, specificity and predictive values

Let us read the Mould's paragraph 19.1 definitions [34] in view of our Table 3.1.

> **Sensitivity** is the probability of a positive test in people with the disease. (...) **Specificity** is the probability of a negative test in people without the disease.

In our Table 3.1, sensitivity and specificity are the conditional probabilities $P(post|malignant)$ and $P(ante|benign)$, $Sens = 27/39 = 69\%$, while $Spec = 106/171 = 62\%$. Sensitivity and specificity are characteristics of a test and are not affected by the prevalence of the disease [6]. But those quantities are not suitable in assessing the 'quality', the 'usefulness' of a clinical test (i.e to answer to the question '*is it relevant to know about the menopausal status in order to foresee malignancy*?'). Therefore one considers [34]:

> **Positive predictive value** (PPV) is the probability of the person having the disease when the test is positive. (...) **Negative predictive value** (NPV) is the probability of the person not having the disease when the test is negative.

In our Table 3.1, PPV $= P(malignant|post) = 27/92 = 29\%$ and NPV $= P(benign|ante) = 106/118 = 90\%$. Unfortunately, although the PPV and NPV give a direct assessment of the usefulness of the test, they are affected by the prevalence of the disease [6]. This is the reason why often researchers move to the **likelihood ratioes** [6].

R To discuss better the relation between PPV, NPV and other concepts as likelihood ratios, pre-test probability, post-test odds, Youden's index see Viv Bewick, Liz Cheek and Jonathan Ball *Statistics review 13: Receiver operating characteristic curves*[6] paper, published on line in the Medical Statistics series of paper, `https://www.biomedcentral.com/collections/CC-Medical`

3.1.2 the ROC curve

All previous quantities (sens, spec, PPV, NPV, ..) can be easily calculated inRby several packages: for instance ROCR[41] and pROC[36]. Such packages are suitable to draw a particular graph, called the **receiver operating characteristic**, i.e the **ROC curve**. Its history is recalled in:
https://en.wikipedia.org/wiki/Receiver_operating_characteristic
It is known[40] that HE4 can be considered an ovaric tumor biomarker. Suppose we are interested in finding a proper logHE4 cut-off value, within the roma dataset, in order to 'maximize' the sensitivity and the specificity with respect to the Histology outcome. Let us make three naive attempts, choosing as cut-off respectively 3, 4 and 5:

```
> cutoff3 = logHE4 > 3
> table(Histology, cutoff3)
> cutoff4 = logHE4 > 4
> table(Histology, cutoff4)
> cutoff5 = logHE4 > 5
> table(Histology, cutoff5)
```

	≤ 3	> 3	sum		≤ 4	> 4	sum		≤ 5	> 5	sum
benign	1	170	171		128	43	171		170	1	171
malignant	0	39	39		7	32	39		23	16	39
sum	1	209	210	sum	135	75	210	sum	193	17	210

Note that on the left table, the 3 cut-off is too low: the sensitivity is perfect, $39/39 = 100\%$, but the specificity is nearly null, $1/171 < 0.1\%$. On the contrary, the 5 cut-off is too high: now the sensitivity is about like tossing a coin, $16/39 = 41\%$, while the specificity is nearly optimal in terms of a population screening test, $170/171 > 99\%$. So we could ideally search for an optimal cut-off just 'pinching' 3 and 5 somewhere toward 4. Using for instance library(pROC) everything is simple:

```
> library(pROC)
> roccurve = roc(Histology ~ logHE4)
> roccurve
> coords(roccurve, "best")
```

We re-discover that there are 171 controls versus 39 cases, and that fixing a cut-off in 4.1 we obtain the 'best' trade-off in sensitivity, 82.1%, and in specificity, 82.5%. Figure 3.3 depicts the situation; the black/orange points is the nearest point, in term of Euclidean metric (i.e. Pythagorean theorem), to the top-left corner – which represents the theorical optimal point with no false positive and no false negative (i.e. sensitivity = specificity = 100%). The output provides also the area under the curve (i.e. the L^1 norm of the graph).

```
> plot(roccurve)
> points(0.8205, 0.8246, lwd = 2, pch = 21, col = "black", bg = "orange")
```

3.2 Commonly used random variables

In paragraph 2.7 the concept of finite random variable has been introduced. Let us show how to simulate with R the otitis variable; here a **sampling with replacement** experiment of 1000 random extraction, and the relative barplot are coded. We shall use sample:

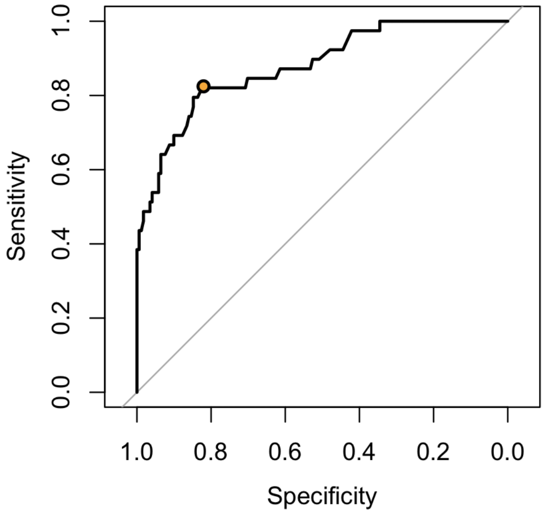


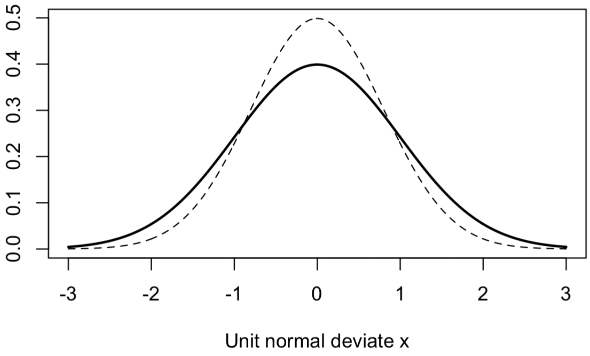
Figure 3.3: ROC curve analysis

```
> otitis = 0:6
> simulation = sample(otitis, size = 1000, replace = TRUE,
              prob = c(0.129, 0.264, 0.271, 0.185, 0.095, 0.039, 0.017))
> simulation
> barplot(table(simulation))
```

Exercise 3.5 Try to modify the above code in order to simulate 100 tossing of a fair coin. ■

Richard Mould's chapters 3, 6 and 7 [34] are devoted to discuss some typical random variables commonly used in biostatistics. Let us now learn the meaning of the d, q, p and r prefix symbols combined to some random variables defined in R, starting from the most important case.

3.2.1 The Normal Distribution



To start, let us start drawing the 3.3 Mould's paragraph figure, as above. Here you find the code:

```
> x = seq(from = -3, to = 3, by = 0.01)
> ysolid = dnorm(x, mean = 0, sd = 1)
> ydash = dnorm(x, mean = 0, sd = 0.8)
> plot(x, ydash, lty = 2, "l", xlab = "Unit normal deviate x", ylab = "")
> lines(x, ysolid, lwd = 2)
```

Here the key instruction is `dnorm`, (i.e. the density of the normal random variable, as tabulated in his Table 3.1(b)), while to compute the probabilities of his Table 3.1(a) – Area P beneath the standard normal curve, between the limits $X = \zeta$ and $X = +\infty$ – we can use the instruction `pnorm` (p stands for probability). For instance, to check the words written by R. Mould:

> Thus for $\zeta = 2.54$, the required area is P = 0.0055426 (R. Mould, [34] Table 3.1(a)).

we use:

```
> 1 - pnorm(2.54)
```

Lastly, to compute exactly the **quantiles**, i.e. the multiples of standard deviations (i.e. Number of SDs, also called the **deviates**) in his Figure 3.3, we exploit the `qnorm` command:

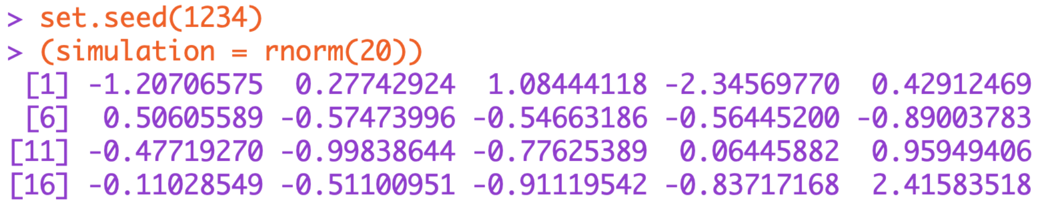
```
> qnorm(0.025)
> qnorm(0.05)
```

Very often we have to decide whether some data are or not distributed according a normal distribution; R. Mould [34] claims:

> In the first paragraph of this section it was stated that a more rigorous test than visual assessment of a normal probability graph plot should be used.

In the next chapters we will discuss about **testing for normality** using the test of Samuel Shapiro and Martin Wilk (see section 5.3.2). Meanwhile, we introduce a very useful graph called the **quantile - quantile plot** (i.e. the **Q-Q plot**). For this, we generate twenty (pseudo)random numbers normally distributed with the `rnorm` command (the `set.seed` is here for reproducibility):

```
> set.seed(1234)
> simulation = rnorm(20)
```



```
> set.seed(1234)
> (simulation = rnorm(20))
 [1] -1.20706575  0.27742924  1.08444118 -2.34569770  0.42912469
 [6]  0.50605589 -0.57473996 -0.54663186 -0.56445200 -0.89003783
[11] -0.47719270 -0.99838644 -0.77625389  0.06445882  0.95949406
[16] -0.11028549 -0.51100951 -0.91119542 -0.83717168  2.41583518
```

When data are normally distributed, they (approximately) tends to lay on the 'diagonal' of the Q-Q plot (i.e. the violet dashed line intersecting the first and third quartile of the orange triangle shaped sample.)

```
> qqnorm(simulation, col = "orange", pch = 17)
> qqline(simulation, col = "violet", lty = 2)
```

To read in deep the details, see for instance `https://en.wikipedia.org/wiki/Q%E2%80%93Q_plot`. The Q-Q plot will be very useful in assessing the 'quality' of the linear models in the forthcoming Subsection 6.2.3.

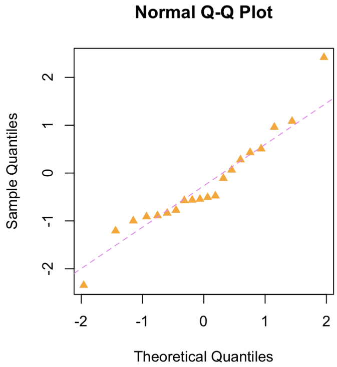
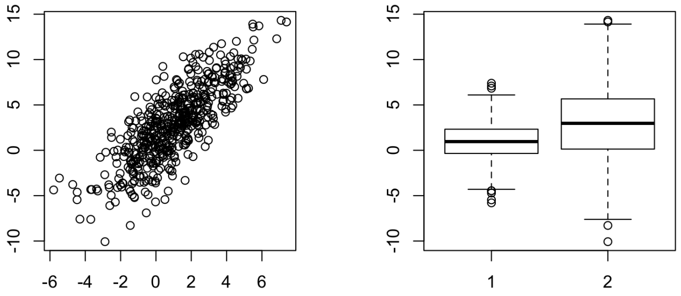


Figure 3.4: The quantile - quantile normal plot

R The rnorm command generates one dimensional normal data. Often it is required to deal with **bivariate normally distributed** random points: in that case the notion of correlation (between the marginal one-dimensional distributions) is required. Here the code to generate 500 random bivariate points, respectively of mean 1 and 3, and standard deviation 2 and 4, on the x and y axes, with correlation of 75%:

```
> mx = 1; sx = 2
> my = 3; sy = 4
> n = 500; r = 0.75
> n1 = rnorm(n); n2 = rnorm(n)
> n3 = r * n1 + sqrt(1-r^2) * n2
> xx = mx + sx * n1; yy = my + sy * n3
> par(mfrow = c(1,2))
> plot(xx, yy); boxplot(xx, yy)
```



The sum of normal variables is, or is not, normal?

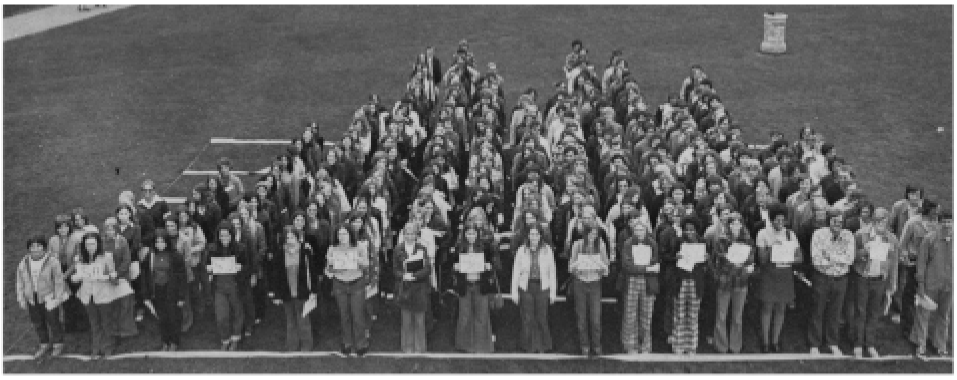
Do two dromedaries make a camel? It's a funny question, but there is in literature a bit of mess about the 'sum' of two normal variables. Let us read the authoritative Bernard Rosner [38, page 135]

> .. linear combination of normal random variables are often of specific concern. It can be shown that any linear combination of normal random variables is itself normally distributed.

And now, let us move to Martin Bland [7, page 111]:

> ... If we add two variables from Normal distributions together, even with different means and variances, the sum follows a Normal distribution.

The two statements are misleading; it seems that there is a confusion between things happening $\mathbb{R} \times \mathbb{R} = \mathbb{R}^2$ or in $\mathbb{R}$. As a famous counterexample, we recall the *Living histograms* of Brian Joiner [25, 27], in which the tallers (mostly, boys) stay on the right of the photo of the next page, while the smallers (mostly, girls) are on the left: the distribution suggests an immediate bimodality, and therefore normality is clearly excluded (i.e. two dromedaries do not make a camel). We will discuss such important case in 6.2.3 subsection.



In particular, in a 1947 number of *Nature*, S. Vaswani [45] provide a counterexample, recalled and enlarged by C. Kowalski in his 1973 *Non-Normal Bivariate Distributions with Normal Marginals* [29]. And in 1982, E. Melnick and A. Tenenbein, with their *Misspecifications of the Normal Distribution* [31], provide a clear response:

> Question 3: are linear combinations of normally distributed random variables always normal? The answer to this question is no and it can be illustrated by using the example in Question 2 ... linear combinations of normal random variables need not themselves be normal. The correct statement is that any linear combination of random variables from a multivariate normal distribution is normally distributed.

3.2.2 The Lognormal Distribution

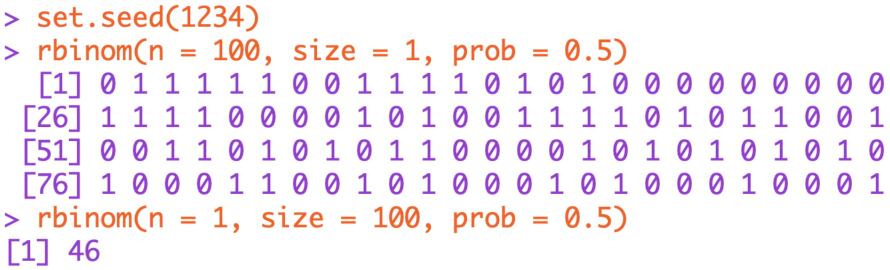
In his paragraph 3.6, R. Mould [34] (and in particular in Figure 3.7) introduces the lognormal distribution, which in R is managed by the commands `dlnorm`, `plnorm`, `qlnorm`, and `rlnorm`. Typical log-normal distributed data are patients' body mass indexes [19] (and this represents a typical pitfall in recent literature). We suggest to read the nice paper by the swiss scientists Limpert, Stahel and Abbt, *Log-normal Distributions across the Sciences: Keys and Clues*, [30].

Exercise 3.6 Simulate 10000 throws of three dice (hint: use `sample`), and draws two barplot: one relative to the sum of their faces, one to the product of their faces. What do you see? ■

3.2.3 The Binomial Distribution

Let us re-examine the exercise in paragraph 3.2 of tossing 100 times a fair coin: instead of using `sample`, we can exploit one of the four commands `dbinom`, `pbinom`, `qbinom`, and `rbinom`. But have a look to the syntax:

```
> set.seed(1234)
> rbinom(n = 100, size = 1, prob = 0.5)
> rbinom(n = 1, size = 100, prob = 0.5)
```



```
> set.seed(1234)
> rbinom(n = 100, size = 1, prob = 0.5)
  [1] 0 1 1 1 1 1 0 0 1 1 1 1 0 1 0 1 0 0 0 0 0 0 0 0 0
 [26] 1 1 1 1 0 0 0 0 1 0 1 0 0 1 1 1 1 0 1 0 1 1 0 0 1
 [51] 0 0 1 1 0 1 0 1 0 1 1 0 0 0 0 1 0 1 0 1 0 1 0 1 0
 [76] 1 0 0 0 1 1 0 0 1 0 1 0 0 0 1 0 1 0 0 0 1 0 0 0 1
> rbinom(n = 1, size = 100, prob = 0.5)
[1] 46
```

Exercise 3.7 In Mould's 6.3 paragraph we read: *A binomial situation of historical importance is the work of Sir Edward Jenner on smallpox vaccination (an enquiry into the causes and effects of the variolae vaccinae, 1798). A sample of 23 people was infected with cowpox (n = 23). The probability of contracting smallpox when inoculated with the virus was some 90% (p = 0.9), but none of the previously vaccinated 23 people did in fact contract smallpox (r = 0). The binomial probability of such an event occurring is exceedingly small, and the observations are therefore definitely not random.* Compute with R such 'exceedingly small' probability. ■

3.2.4 The Poisson Distribution

Born as a distribution ot the number of occurences of a rare event, i.e. with 'small' probability p in n independent trials and closely connected to the binomial distribution [37], the Poisson distribution is nowadays applied not only to rare events but to generic ’count’ problems. Let us move, for instance, to the Figure 7.4 of Daniel Zips, *Tumour growth and response to radiation*, collected in [28]. Let us read his words about the local tumour control:

> If not a single tumour but a group of tumours (or patients) is considered, the local tumour control probability (TCP) as a function of radiation dose can be described statistically by a Poisson distribution of the number of surviving clonogenic tumour cells (...). As an illustration, one might imagine that a given radiation dose causes a certain amount of ’lethal hits’ randomly distributed within the cell population. Some cells will receive one ’lethal hit’ and will subsequently die. Other cells will receive two or more ’lethal hits’ and will also die. However, some cells will not be hit, will therefore survive and subsequently cause a local failure. According to Poisson statistics, a radiation dose sufficient to inflict on average one ’lethal hit’ to each clonogenic cell in a tumour (number of ’lethal hits’ per cell, *m*, = 1) will result in 37 per cent surviving clonogenic cells.

1				1	2
2	3	1	2	1	
	1		2		1
1	1	2		4	1
1		1		3	1
	2	1	1		

In that example, the Poisson distribution has the intensity parameter $\lambda = 0.37$ (or using Mould’s notation of paragraph 7.1, the mean number *m* of events). Let us try to simulate the situation with the syntax: `rpois(36, lambda = 0.37)`

Exercise 3.8 Use the R command `matrix` to transform the 36 length vector in a square table. ■

3.2.5 The Uniform Distribution

To conclude this random variables review, remember that with `runif` we can generate numbers uniformly distributed. For instance, instead of using `sample` or `rbinom` to toss a fair coin, we can try: `trunc(2 * runif(100))` .

4. Introduction to Sampling

4.1 Sampling Distribution of the Sample Mean

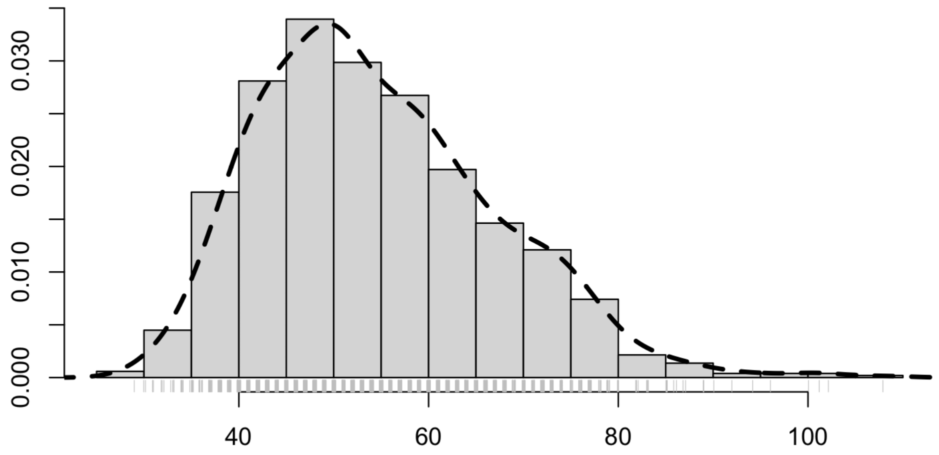
Let us carefully read R. Mould's words written in his 4.1 paragraph:

> In statistical parlance the term population refers to the group of objects, events, results of procedures or observations (rather than the geographical connotation of population relating only to persons in a country or state etc) which is so large a group that usually it cannot be given exact numerical values for statistics such as the population mean μ or the population standard deviation σ. These statistics therefore can only be estimated.
>
> To obtain for example, an estimate of the population mean μ of a certain characteristic x of the population, *sampling* must first take place because all the values of x for the entire population cannot be measured. Only a small part of the population can be surveyed and that part is called a *sample*.
>
> There are various methods of sampling, including *random sampling*, which for clinical trials is discussed in a later chapter as simple randomisation, stratified randomisation and balanced randomisation. (...)
>
> From a knowledge of this sample the sample mean x_m can be found and a *statistical inference* (i.e. drawing a conclusion about a population from a sample) can be drawn about 'how good' is this value x_m as an estimate of the true population mean μ. The phrase 'how good' can be stated in terms of *confidence limits*.
>
> The standard deviation s_m of the sample mean x_m tells you about the spread of the measured sample values $x_1, x_2, \ldots, x_i, \ldots$. (...) If the *sampling experiment* to measure x_m is then repeated N times, with the sample size n always remaining the same, a total of N values of x_m will be obtained. If these are then averaged, then M , which is the *mean of means* or *grand mean* is obtained.
>
> The standard deviation of the mean of means M is given a special name: standard error of the mean, where $SE = Sample\ Standard\ Deviation\ /\sqrt{n}$

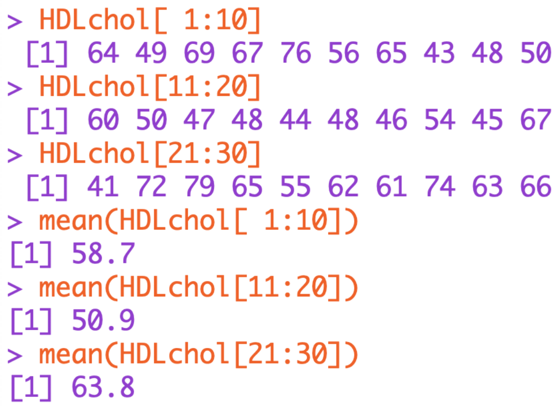
We want to be sure to understand all these details, which are related to some ’epic fail’ made for about six centuries by the London Royal Mint, or to the kidney cancer rate and rural american lifestyle, or to the USD 1.7 billion spending by Bill and Melinda Gates Foundation in support to small schools, as Richard Wainer tells in his *The most dangerous equation* [48].

Let us import into R data concerning 1025 Triestiners healthy blood donors, in particular relative to their HDL `cholesterol` .

```
> www = "http://www.biostatisticaumg.it/dataset/cholesterol.csv"
> cholesterol = read.csv(www, header = TRUE)
> attach(cholesterol)
> tail(cholesterol)
> hist(HDLchol, freq = FALSE, col = "lightgrey")
> rug(jitter(HDLchol), col = "grey")
> lines(density(HDLchol), lwd = 3, lty = 2)
```



We are interested in estimating the unknown HDL cholesterol mean μ of the whole Triestine healthy population. Let us ’extract’ the first sample of ten donors (i.e. $n = 10$), and then the second sample of the next ten donors, and then the third, and again and again; and then we compute the means:



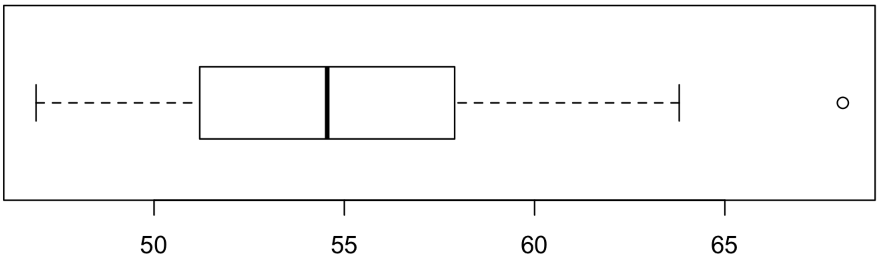
```
> HDLchol[ 1:10]
 [1] 64 49 69 67 76 56 65 43 48 50
> HDLchol[11:20]
 [1] 60 50 47 48 44 48 46 54 45 67
> HDLchol[21:30]
 [1] 41 72 79 65 55 62 61 74 63 66
> mean(HDLchol[ 1:10])
[1] 58.7
> mean(HDLchol[11:20])
[1] 50.9
> mean(HDLchol[21:30])
[1] 63.8
```

Actually, we are required to repeat $N = 102$ times the *sampling experiment* to measure the $N = 102$ sample means, with the sample size $n = 10$ always remaining the same. Let us use a `for` cycle, storing in the vector `storesamplemean` the $N = 102$ sample means:

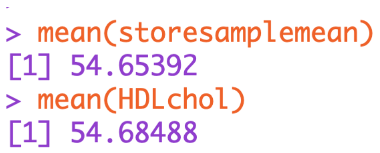
```
> storesamplemean = numeric(102)
> for(i in 1:102)
> {
>  inf = (10*(i-1)+1)
>  sup = (10*(i-1)+10)
>  storesamplemean[i] = mean(HDLchol[inf:sup])
> }
```

Now, what about the distribution of `storesamplemean` ?

```
boxplot(storesamplemean, horizontal = TRUE)
```

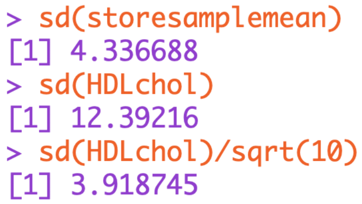


And now, what about the mean of `storesamplemean`, with respect to the mean of `HDLchol`?



```
> mean(storesamplemean)
[1] 54.65392
> mean(HDLchol)
[1] 54.68488
```

Well, this could be quite a surprise, but it is the Jakob Bernoulli Weak Law of Large Numbers Theorem. And, lastly: what about the standard deviation of `storesamplemean`, with respect to the standard deviation of `HDLchol`? Here it arises the definition of **standard error of the mean**, $s/\sqrt{n}$:

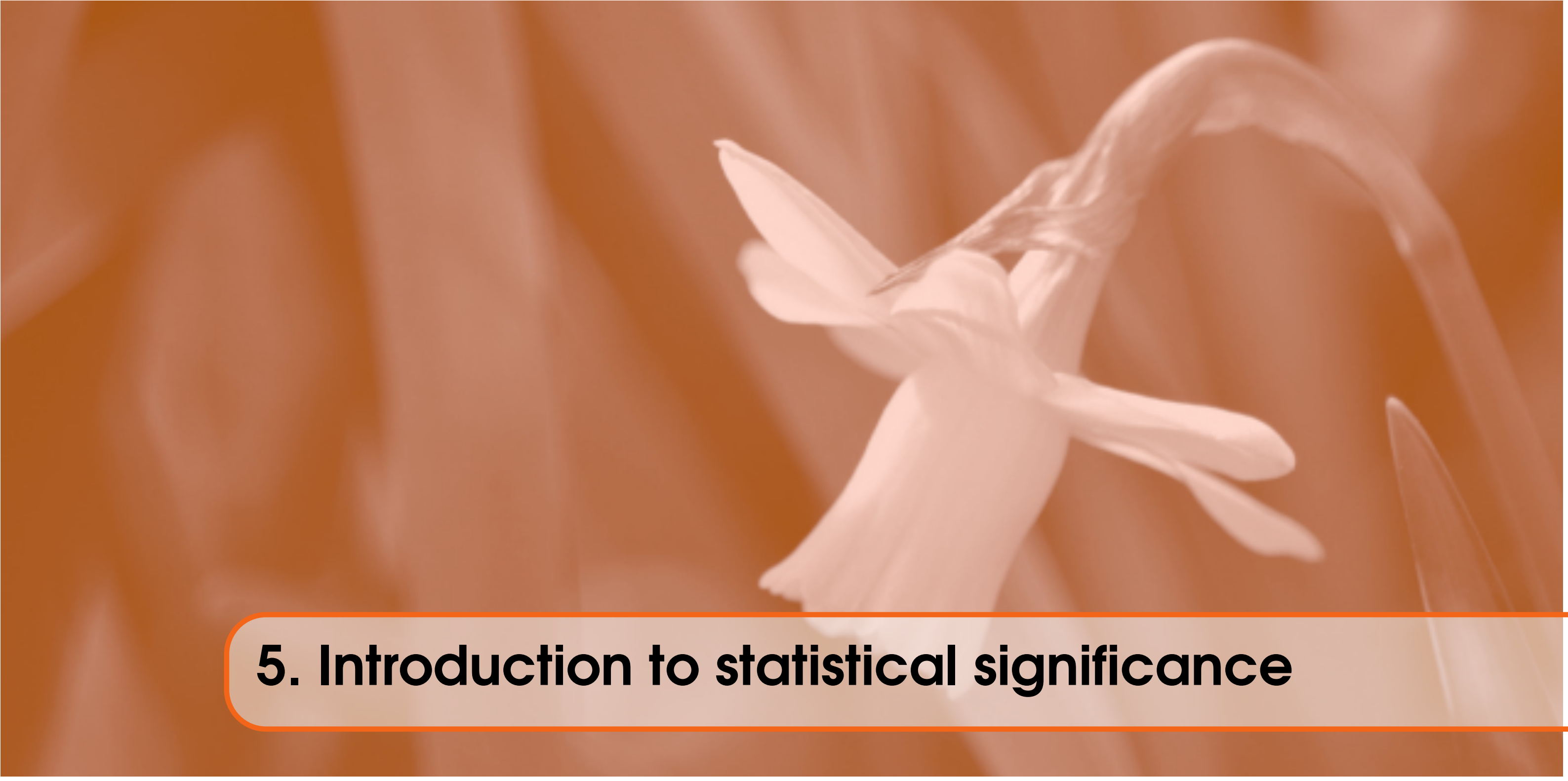


```
> sd(storesamplemean)
[1] 4.336688
> sd(HDLchol)
[1] 12.39216
> sd(HDLchol)/sqrt(10)
[1] 3.918745
```

This is astonishing, and it is the Jarn Lindenberg and Paul Lévy Central Limit Theorem, which also justifies the nice symmetry of the above boxplot:

Theorem 4.1.1 — Lindenberg-Lévy Central Limit Theorem. Suppose $(X_i)_{i\in\mathbb{N}}$ is a sequence of independent and identically distributed random variables with $E[X_i] = \mu$ and $Var[X_i] = \sigma^2 < +\infty$. Then as n approaches infinity, the random variables $\sqrt{n}(S_n - \mu)$ converge in distribution to a **normal** $N(0, \sigma^2)$.

In biomedical literature often there is unclearness, or even misuse, regarding the two concepts of dispersion, measured by the standard deviation s of the sample, and the 'pratical' meaning of the standard error $s/\sqrt{n}$, which is a **measure of reliability** [7, 12].



5. Introduction to statistical significance

Let us 'melt' the Chapters 8 and 11 of R.Mould's textbook [34], recalling and interpreting step by step the R code to analyse the historical example of the mathematician and chemist William Gosset a.k.a. 'Student'. The history is recalled for instance in: `https://www.encyclopediaofmath.org/index.php/Gosset,_William_Sealy`.

Not Kiln-Dried	Kiln-Dried	Difference
1903	2009	+106
1935	1915	-20
1910	2011	+101
2496	2463	-33
2108	2180	+72
1961	1925	-36
2060	2122	+62
1444	1482	+38
1612	1542	-70
1316	1443	+127
1511	1535	+24

Table 5.1: The original data of Student published in Biometrika [43, page 24].

Let us describe the William Gosset/Student question, as reported in his fundamental paper [43]; he had in particular to decide whether an *ante-litteram* 'agricultural biotechnology' treatment is useful, or not, in increasing the production of Dublin's Guinness beer, i.e. to dry seeds into a special oven before seeding. Here Gosset's words:

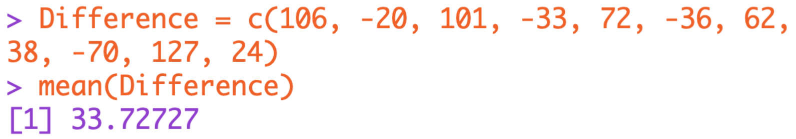
> To test whether it is advantage to kiln-dry barley seed before sowing, seven varieties of barley were sown (both kiln-dried and not kiln-dried) in 1899 and four in 1900; the results are given in the table (5.1), expressed in Lbs. head corn per acre.

To summarize, Gosset noted that pnorm provides too optimistic probability estimates when examining data coming from 'small-sized' samples – like these 11 records –, as it often occurs in biology labs or in medical pilot studies. Let us discuss two possible (and in this case equivalent) methodological choice to analyse these data, considering only the difference in yields, or considering the couples of not-kiln and kiln data.

5.1 One-sample t test

Let us consider the average $x_m = 33.7$ of the Difference between treated (kiln-dried) and not treated (not kiln-dried) seeds.

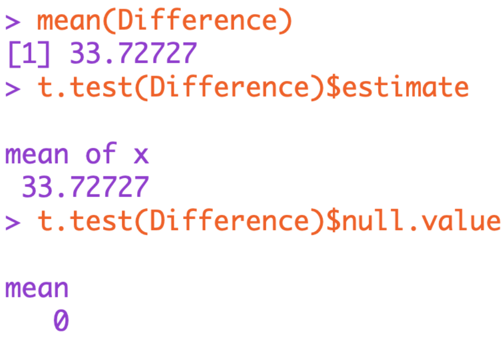
```
> Difference = c(106, -20, 101, -33, 72, -36, 62, 38, -70, 127, 24)
> mean(Difference)
```



```
> Difference = c(106, -20, 101, -33, 72, -36, 62,
38, -70, 127, 24)
> mean(Difference)
[1] 33.72727
```

One wants to compare such **experimental result** $x_m = 33.7$ with the theoretical hypothesis that to treat or not to treat provide the same effect: this is the so-called **null hypothesis**, i.e. $\mu = 0$. One therefore is interested in evaluating the 'distance' of these two quantities, $x_m - \mu$, from a statistical point of view; that is, to decide if $|x_m - \mu|$ could be considered a null distance, or not.

```
> mean(Difference)
> t.test(Difference)$estimate
> t.test(Difference)$null.value
```



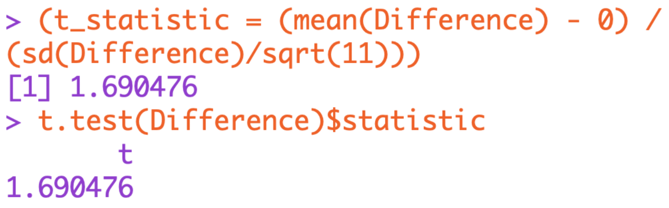
```
> mean(Difference)
[1] 33.72727
> t.test(Difference)$estimate

mean of x
 33.72727
> t.test(Difference)$null.value

mean
   0
```

The idea is to exploit the concept of **signal to noise ratio**, as correctly discussed by Stephen Ziliak and Deirdre McCloskey in their *The Cult of Statistical Significance* magistral paper [54]:

$$t = \frac{x_m - \mu}{s/\sqrt{n}}$$

The quantity t is usually called **test statistic**, and this is a sort of pun, and source of confusion, in various language of the World: while in English and in Spanish the words 'Statistics' and 'Estadistica' means the science, and 'the test statistic' and 'el estadistico de test' means the t – and the word 'statistic' is a sinonymous of 'summary' –, in French and in Italian 'Statistique' and 'Statistica' do not differ from 'la statistique test' and 'la statistica test'.

```
> (t_statistic = (mean(Difference) - 0) /(sd(Difference)/sqrt(11)))
> t.test(Difference)$statistic
```



```
> (t_statistic = (mean(Difference) - 0) /
(sd(Difference)/sqrt(11)))
[1] 1.690476
> t.test(Difference)$statistic
       t
1.690476
```

Now, how to interpret such ’noise to signal ratio’ of 1.69? Is it ’close’ or ’far away’ from zero? Resorting to the normal distribution probability, pnorm, one could compute the area of the two violet (see Figure 5.1) tails encompassing t’s far away – in absolute value – from 1.69..., which is about 9.1%, according to (recall paragraph 3.2.1):

```
> 2 * (1 -  pnorm(1.690476))
```

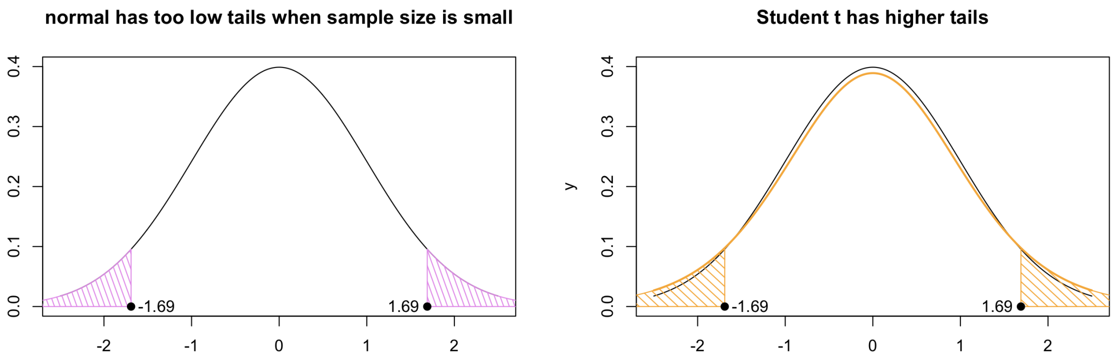
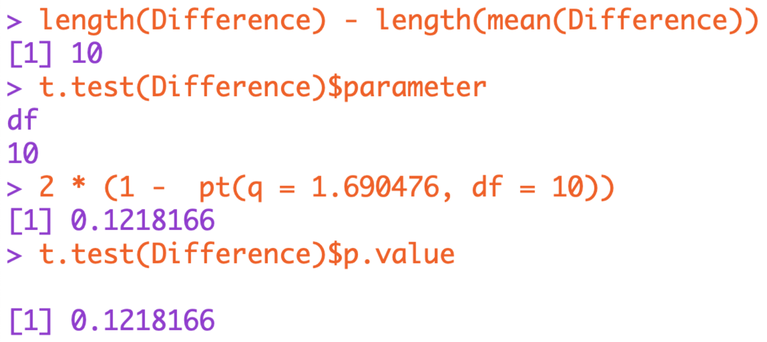


Figure 5.1: On the left panel, the probability to observe by chance a mean value greater or equal to $x_m = 33.7$, computed according the normal distribution, is painted in violet and it is approximately equal to 9%. On the right panel, the probability painted in orange increases to about 12%, according to the Student t distribution with 10 degrees of freedom: ’tails are heavier when sample size is small’ .

In his paper [43], William Gosset - Student provides the integral relations to correctly evaluate such probability; those integrals depends of course on how many free parameters, i.e. degrees of freedom, possess data: in this example they are 10. So, the probability increases to about 12.2%.



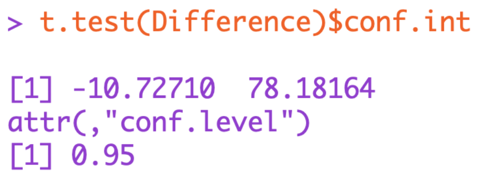
```
> length(Difference) - length(mean(Difference))
[1] 10
> t.test(Difference)$parameter
df
10
> 2 * (1 -  pt(q = 1.690476, df = 10))
[1] 0.1218166
> t.test(Difference)$p.value

[1] 0.1218166
```

So, we conclude that if the null hypothesis is true (i.e. there is no effect in drying or not the seeds) then there is a probability of 12.2% that the observed effect $x_m = 33.7$ is simply due to chance. This probability is known as the **p-value** of the test, and quite commonly scientits adopt the 'fideistic' – and misinterpreted – rule originally stated by Ronald Fisher:

- p-value $< \alpha \equiv 0.05$? Reject null hypotheses, there is an effect
- p-value $> \alpha \equiv 0.05$? Accept null hypotheses, no effect at all

An equivalent method to decide, introduced by the polish mathematician Jeržy Neyman, is to look to the 95% **confidence interval** and to verify if $\mu = 0$ belongs, or not, to the interval:

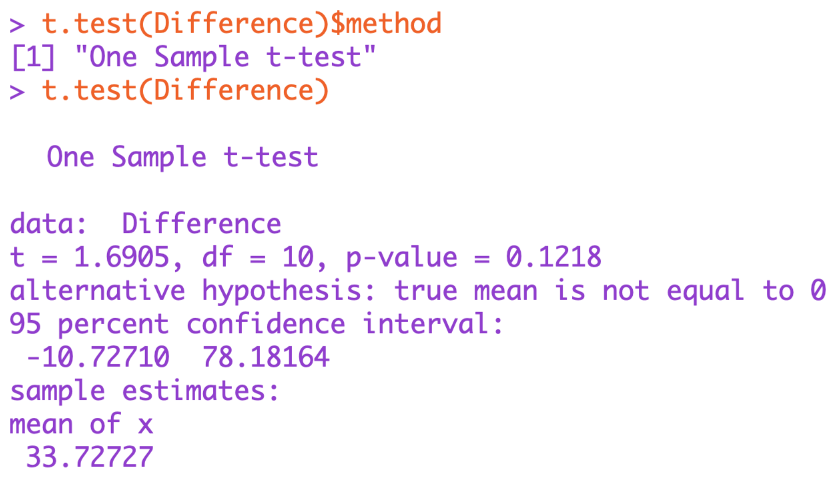
```
> t.test(Difference)$conf.int
```



```
> t.test(Difference)$conf.int

[1] -10.72710  78.18164
attr(,"conf.level")
[1] 0.95
```

As $\mu = 0 \in [-10.7, 78.2]$, then we accept the null hypotheses and we are lead to decide that the drying treatment is unuseful. So, now we are able to understand the whole output of the **One Sample t-test**:

```
> t.test(Difference)$method
> t.test(Difference)
```



```
> t.test(Difference)$method
[1] "One Sample t-test"
> t.test(Difference)

	One Sample t-test

data:  Difference
t = 1.6905, df = 10, p-value = 0.1218
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -10.72710  78.18164
sample estimates:
mean of x 
 33.72727 
```

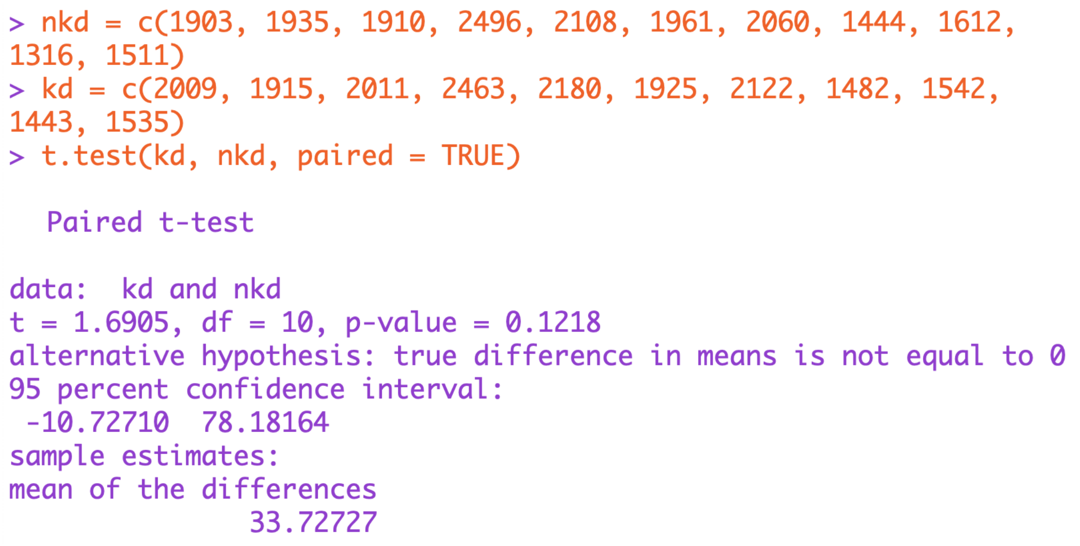
Around the question of 'statistical significance' $p < 0.05$ there is a wide and profound discussion. We will discuss again the topic in section , but to have an initial idea we suggest reading:

- the Douglas Curran-Everett and Dale J. Benos *Guidelines for reporting statistics in journals published by the American Physiological Society* [13]
- the Stephen Ziliak and Deirdre McCloskey *The cult of statistical significance* [54]
- the *Why Most Published Research Findings Are False* of John Ioannidis [26], and all the subsequent debate which led to `https://metrics.stanford.edu/`
- *The ASA's statement on p-values* [49]

5.1.1 Two-sample paired t test

Here we discuss the other proper methodology to decide about kill-drying. Let us refer again to Table 5.1, considering the 11 couples of **paired data** Not-Kiln Dried and Kiln-Dried seeds. Check that the following **Two Sample paired t-test** is in agreement with what we have just seen in the One Sample t-test:

```
> nkd = c(1903, 1935, 1910, 2496, 2108, 1961, 2060, 1444, 1612, 1316, 1511)
> kd = c(2009, 1915, 2011, 2463, 2180, 1925, 2122, 1482, 1542, 1443, 1535)
> t.test(kd, nkd, paired = TRUE)
```



```
> nkd = c(1903, 1935, 1910, 2496, 2108, 1961, 2060, 1444, 1612,
1316, 1511)
> kd = c(2009, 1915, 2011, 2463, 2180, 1925, 2122, 1482, 1542,
1443, 1535)
> t.test(kd, nkd, paired = TRUE)

	Paired t-test

data:  kd and nkd
t = 1.6905, df = 10, p-value = 0.1218
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -10.72710  78.18164
sample estimates:
mean of the differences
               33.72727
```

A paired t-test is a simple but typical statistical procedure exploited in the **longitudinal** experimental design (see section 7.4), where (a couple of) repeated measures are collected on the same subject.

5.2 Power of a test

Let us quote Mould's paragraph 8.4 [34] words:

> There are two types of error which can be made in arriving at a decision about the null hypothesis, H_0. A type-I error is to *reject* H_0 *when in fact it is true* and a type-II error is to *accept* H_0 *when in fact it is false*. By convention the probability of a type-I error is usually denoted by α and the probability of a type-II error by β. (...) The probability $1-\beta$ is defined as the *power* of the test of the hypothesis H_0 against an alternative hypothesis.

By analogy, a judge starts from the hypothesis $H_0 =$ 'this defendant is innocent'; the type-I error is to *reject* innocence *when in fact it is true* and to imprison an innocent. And a type-II error is to *accept* innocence *when in fact it is false*, i.e. to release a culprit.

```
> power.t.test(n = 11,  delta = (mean(Difference) - 0),
             sd = sd(Difference), sig.level = 0.1218,
             power = NULL, type = "one.sample")
```

```
> power.t.test(n = 11,  delta = (mean(Difference) -
0), sd = sd(Difference), sig.level = 0.1218, power =
NULL, type = "one.sample")

     One-sample t test power calculation

              n = 11
          delta = 33.72727
             sd = 66.17113
      sig.level = 0.1218
          power = 0.516215
```

R holds the possibility to estimate $1-\beta$ with the above function `power.t.test`, which depends on four information linked together: the numerator of the t statistic, $x_m - \mu$, and the denominator which requires estimation of σ and knowledge on sample size n. So, in the particular example of William Gosset / Student's barley seeds, the situation is summarized in this table (confront Mould's Table 8.2):

	Experimental Result	
Actual Truth	To dry 'works'	To dry 'not works'
To dry 'works'	correct, $1-\beta = 0.52$	wrong, $\beta = 0.48$
To dry 'not works'	wrong, $\alpha = 0.12$	correct, $1-\alpha = 0.88$

The power calculation here shown has only a didactical interest, but is is uneuseful – see John Hoenig and Dennis Heisey, *The Abuse of Power: The Pervasive Fallacy of Power Calculations for Data Analysis* [22].

To estimate power when designing a clinical trial is crucial. For an introductory review on this topic, see Elise Whitley and Jonathan Ball, *Statistics review 4: Sample size calculations*, [51]:

`https://ccforum.biomedcentral.com/articles/10.1186/cc1521`

5.3 Two sample tests

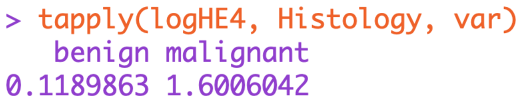
We provide here a brief survey of some classical tests concerning two indipendent samples, following the Michael Crawley comprehensive *The* R *Book* [12, pages 289-298]. We are interested in:

- comparing two variances (Fisher / Snedecor F test, `var.test`)
- comparing two (unpaired) sample means with normal errors (Student's t test, `t.test`)
- comparing two means with non-normal errors (Wilcoxon's rank test, `wilcox.test`)
- testing for independence of two variables in a contingency table (chi-squared, `chisq.test`, or Fisher's exact test, `fisher.test`).

5.3.1 Testing two variances

Let us revise the ovarian cancer `roma` Shadi Najaf dataset, and observe that the variance of the biomarker `logHE4` differs a lot between the `benign` (0.12) and `malignant` (1.60) groups.

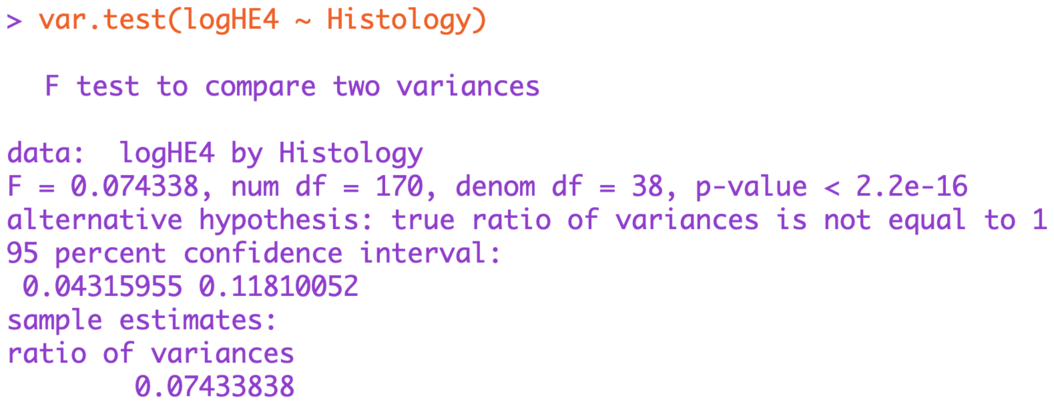
```
> www = "http://www.biostatisticaumg.it/dataset/roma.csv"
> roma = read.csv(www, header = TRUE)
> attach(roma)
> tapply(logHE4, Histology, var)
```



```
> tapply(logHE4, Histology, var)
   benign malignant
0.1189863 1.6006042
```

Therefore we proceed assessing whether those data comes from two distinct populations, differring in dispersion; this is the goal of the `var.test` Fisher and Snedecor F test ratio of variances:

```
> var.test(logHE4 ~ Histology)
```

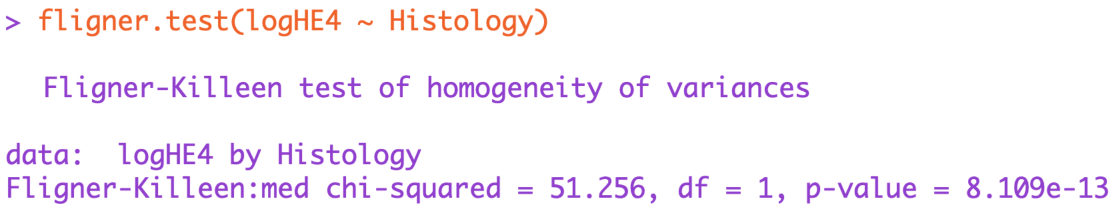


```
> var.test(logHE4 ~ Histology)

   F test to compare two variances

data:  logHE4 by Histology
F = 0.074338, num df = 170, denom df = 38, p-value < 2.2e-16
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.04315955 0.11810052
sample estimates:
ratio of variances
        0.07433838
```

The output shows that the ratio of the two variances is 0.074 (i.e. the $F = 0.074$ statistic), really far away from 1 (the p-value is practically null, and the 95% confidence interval do not cover 1). Such computations come from two distinct sample of dimension 171 and 39 (in fact, after estimating two variances they remain df = 170 and df = 38 degree of freedom). We conclude therefore that the two samples are different, in a statistical sense: data are **heteroskedastic**. The game is over, as we remember Michael Crawley words [12, page 290]:

> Because the variances are significantly different, it would be wrong to compare the two sample means using Student's t-test.

Remembering also that `var.test` 'is highly sensitive to outliers' [12, page 291], one can consider the non-parametric equivalent **Fligner - Killeen test**, `fligner.test(logHE4` $\sim$ `Histology)`, most robust against departures from normality [11]:



```
> fligner.test(logHE4 ~ Histology)

   Fligner-Killeen test of homogeneity of variances

data:  logHE4 by Histology
Fligner-Killeen:med chi-squared = 51.256, df = 1, p-value = 8.109e-13
```

5.3.2 Testing two (unpaired) sample means with normal errors

Suppose that we are interested in exploring the role of biomarker `logCA19.9` with respect to `Menopause`. We start with a little exploratory analysis, observing the `boxplot(logCA19.9` $\sim$ `Menopause)`. The boxes are symmetric, the whiskers are 'short', not showing evident departure from normality, and **homoskedasticity** is confirmed by `var.test(logCA19.9 ~ Menopause)` – or better by `fligner.test(logCA19.9 ~ Menopause)`, having considered in the boxplots the presence of a number of outliers, both in `benign` and `malignant` cases. By the way, we recall that one can also rely on a formal test to decide if data are normally distributed, i.e. the Samuel Shapiro and Martin Wilk's test, `shapiro.test` [12].

So, if we were required to decide whether `logCA19.9` levels varies in `Menopause`, we would provide a negative answer, according to **Two Sample t-test**: the two means 2.42 and 2.67 are so close, and their difference (-0.24, not reported) is the center of the 95% confidence interval $[-0.61, 0.12]$ which covers 0.

```
> t.test(logCA19.9 ~ Menopause)

	Welch Two Sample t-test

data:  logCA19.9 by Menopause
t = -1.3337, df = 177.44, p-value = 0.184
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.6051626  0.1170528
sample estimates:
mean in group ante mean in group post 
          2.426271           2.670326 
```

Note that the numbers of degrees of freedom is not anymore an integer (df = 177.44), and this is related to the fact the two groups have different dimensions and we are resorting the proper **Welch - Satterthwaite relation**:

`https://en.wikipedia.org/wiki/Welch%E2%80%93Satterthwaite_equation`

5.3.3 Testing two (unpaired) sample means with non-normal errors

Richard Mould [34] recalls in his Table 11.1 that in order to properly apply the t-test, several hypotheses have to be fulfilled:

1. The observations must be independent in order to avoid bias
2. The observations must be drawn from normal populations
3. These normal populations must have the same variance (or in special circumstances, a known ratio of variances)
4. The variables involved must have been measured in an interval scale, so that it is possible to use arithmetical operations (e.g. add, divide, obtain means) on the values of the variables

Despite the fact that in 1969 Bradley Efron [15] has proved that some mild 'orthant symmetry condition' instead of normality and homoskedasticity can be sufficient, suppose to be interested to confirm the biomarker `logHE4` ability in predicting `Histology` outcome. The `boxplot(logHE4 ~ Histology)` exhibit a very skewed distribution, with a long whisker, and we are surely doubtful about normality.

In this case it is proper to resort to the non-parametric **Wilcoxon test**, which consider data ordered along their rank [11]. No doubt, here: a so small p-value confirms our expectation.

```
> wilcox.test(logHE4 ~ Histology)
> wilcox.test(logHE4 ~ Histology)

	Wilcoxon rank sum test with continuity correction

data:  logHE4 by Histology
W = 808, p-value = 1.616e-13
alternative hypothesis: true location shift is not equal to 0
```

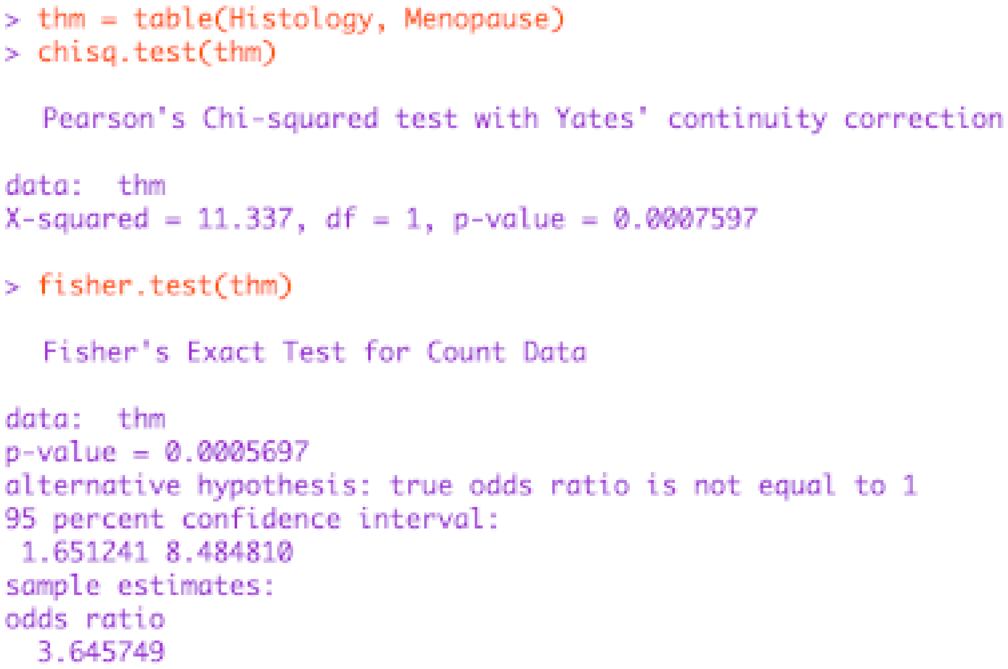
5.3.4 Testing for independence in a 2 by 2 contingency table

In section 3.1, we were discussing about menopausal status as a possible predictor of malignancy in ovarian cancer, and we printed the Table 3.1. Now we try to decide if `Histology` and `Menopause` are **independent event**, or **associated event**, intepreting in a statistical – and not purely mathematical – sense the 'crude' definition:

$$P(malignant) \equiv P(malignant|post\ menopausal)$$

The **Pearson's chi-squared test**, `chisq.test`, is a classical tool, but currently many authors recommend to exploit the **Fisher's exact test**, `fisher.test`. In any case, again no doubt here: `Histology` and `Menopause` are associated events.

```
> thm = table(Histology, Menopause)
> chisq.test(thm)
> fisher.test(thm)
```



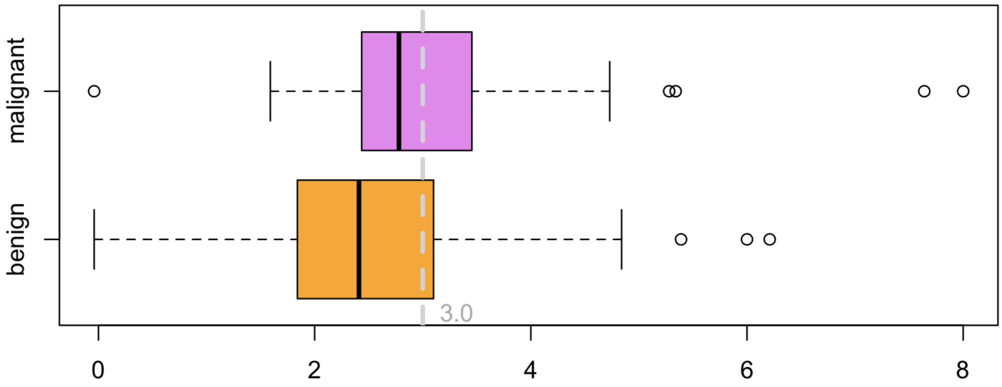
```
> thm = table(Histology, Menopause)
> chisq.test(thm)

	Pearson's Chi-squared test with Yates' continuity correction

data:  thm
X-squared = 11.337, df = 1, p-value = 0.0007597

> fisher.test(thm)

	Fisher's Exact Test for Count Data

data:  thm
p-value = 0.0005697
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 1.651241 8.484810
sample estimates:
odds ratio
  3.645749
```

5.4 Significance: statistical or clinical?

We try to clarify the point with an example. Suppose that we want to assess the role of the carbohydrate antigen 19-9, `logCA19.9`, as a predictor of the ovarian cancer. There is no doubt about its *statistical significance*, the t-test exhibit a smashing p-value = 0.004:

```
> t.test(logCA19.9 ~ Histology)

	Welch Two Sample t-test

data:  logCA19.9 by Histology
t = -3.0102, df = 49.114, p-value = 0.004114
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -1.3062020 -0.2604107
sample estimates:
   mean in group benign mean in group malignant
               2.387719                3.171026
```

Nevertheless, a simple boxplot enlighten the fact that although CA19-9 may be ’significant’ it is not ’useful’, i.e. *clinically significant* in detecting ovarian pathology. Suppose for instance that a woman with symptoms has `logCA19.9` = 3.0. Of course, such a value is closer to the malignant group mean 3.2 than to the benign group mean 2.4, but basing on the 3.0 information to guess histology is nothing more than looking into a crystal ball:



Let us in conclusion read what Richard Mould claims in his 8.3.2 paragraph [34]:

> One of the problems encountered by those involved with statistics is how, and with what accuracy, inferences can be drawn about the nature of a population when the only evidence which exists is that from samples of the population. In order to solve this problem an understanding of *statistical significance* is essential and it should be immediately recognised that this is not necessarily the same as *clinical significance* when the statistics refer to medicine. (...) It is an absolute priority for those using tests for statistical significance that they understand the conditions which must apply for a particular test to be valid and that they have a clear understanding of the hypotheses which are being tested.

III

Statistical Models

6. The linear model

6.1 Overview

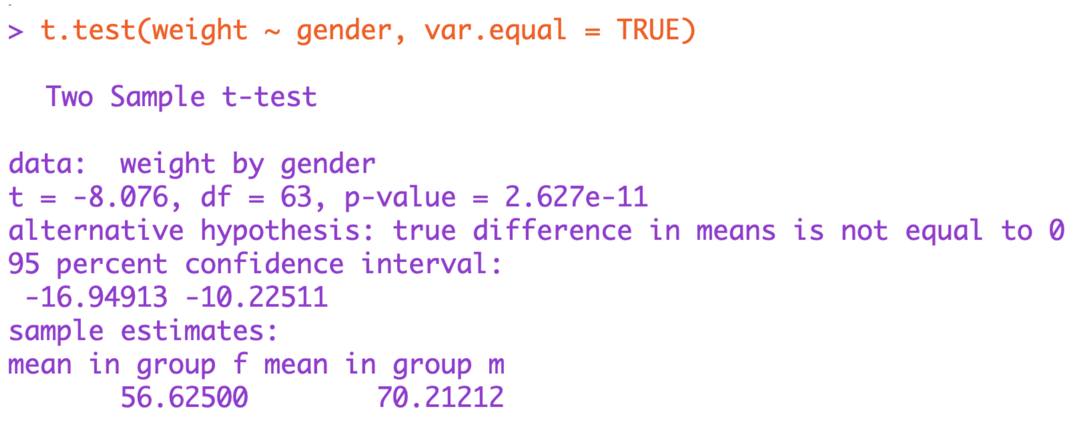
In this Chapter we introduce a modern and powerful statistical tool widely used in the cross-sectional studies: the **linear model**. Typically, in the statistical textbooks, this argument is introduced talking about the sir Francis Galton **regression** 'towards mediocrity' **line** 'in hereditary stature' [21], and at a first sight the two arguments perfectly overlap. We are going here to show that the linear model encompasses a variety of important and classical statistical tools, as the **Anova**, or the **Ancova**; or even the **t-test**.

Consider in fact the `fresher` cross-section dataset, relative to a cohort of medicine and surgery first year Trieste university students: the command `str(fresher)` discloses that they are 65, and we collected their `gender` (a factor variable with `f` and `m` levels), their `height`, `weight` and `shoesize` (numeric variables), along with their `smoke` habits (a factor with levels `no` and `yes`), and their `gym` physical activity (classified as a three level alphabetically ordered factor `not` < `occasional` < `sporty`).
Suppose we are interested in assessing differences in `weight` with respect to the `gender` of our students. Assuming by hypothesis that we can exploit the gaussian normal framework, both `var.test` and `fligner.test` confirm an homoskedastic situation, so we resort to the proper Welch/Student t-test:

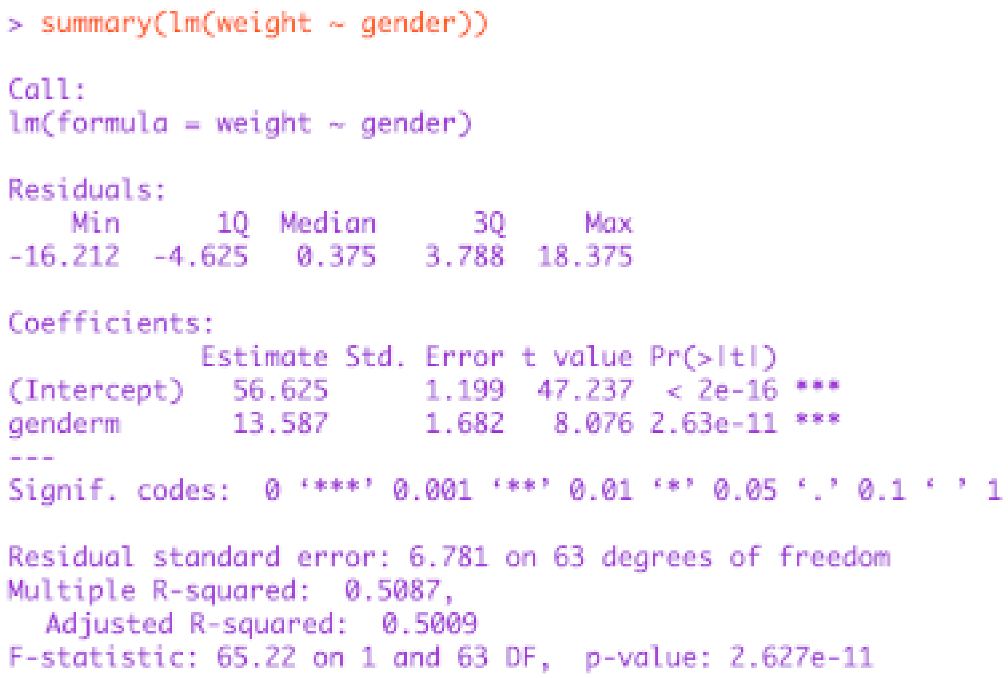
```
t.test(weight ~ gender, var.equal = TRUE)
```

The figure of the next page provides the familiar, usual, output: females have a 56.6 Kg mean weight, while male on average are heavier, 70.2 Kg; such a difference is statistically significant, with a p-value `2.627e-11` very close to zero, as computed on the $t = 8.076$ statistic on 63 degree of freedom. But, look what happens if instead of typing `t.test` we exploit the `lm` linear model R command, and we ask for its `summary`:

```
summary(lm(weight ~ gender))
```



```
> t.test(weight ~ gender, var.equal = TRUE)

	Two Sample t-test

data:  weight by gender
t = -8.076, df = 63, p-value = 2.627e-11
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -16.94913 -10.22511
sample estimates:
mean in group f mean in group m 
       56.62500        70.21212 
```



```
> summary(lm(weight ~ gender))

Call:
lm(formula = weight ~ gender)

Residuals:
    Min      1Q  Median      3Q     Max 
-16.212  -4.625   0.375   3.788  18.375 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   56.625      1.199  47.237  < 2e-16 ***
genderm       13.587      1.682   8.076 2.63e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.781 on 63 degrees of freedom
Multiple R-squared:  0.5087,
  Adjusted R-squared:  0.5009 
F-statistic: 65.22 on 1 and 63 DF,  p-value: 2.627e-11
```

Comparing the two images above, we note that the `summary` last lines offer the same p-value of the t-test, and the 63 degrees of freedom as well. Also the t statistic 8.076 is reported (with a change of sign in the latter) in a column denoted `t value`, but there are lots of further information in the second output, and the next sections are devoted to clarify most of them.

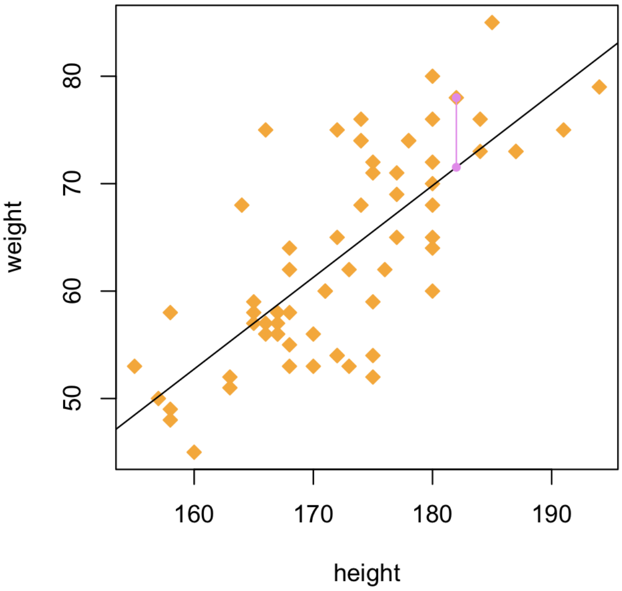
6.2 The regression line

Suppose we are interested in assessing the possible relation that interlaces `fresher`'s `weights` with their `heights`. It is a relation between two numeric variables, and we stress the role that `height` assumes as a possible **predictor** of (i.e. a dataset covariate significantly associated to) the `weight`. In this sense, we pose the following:

```
> relation = weight ~ height
```

This position implies that `height` represents the input, the independent variable located on the abscissa x, while `weight` is thought to be the output, the dependent variable located on the ordinate y. Always remember that a statistical relation **is not** a cause-effect relation at all. Just for fun, look to the `http://www.tylervigen.com/spurious-correlations` in which for instance the divorce rate in Maine is put in relation with consumption of margarine. More seriously, remember that:

> The objective (..) is to show that a relationship exists between these two variables, so that having demonstrated the existence of this relationship, it can be used within some theoretical framework. Blind use of regression formulae, just because they exist, can be very misleading. If Y = a cause and X = an effect, one must be careful not to draw too many conclusions if there may be several other possible causes. Cause and effect in medicine are seldom so simple as to be explained by a single straight line. (R. Mould [34], section 16.1)



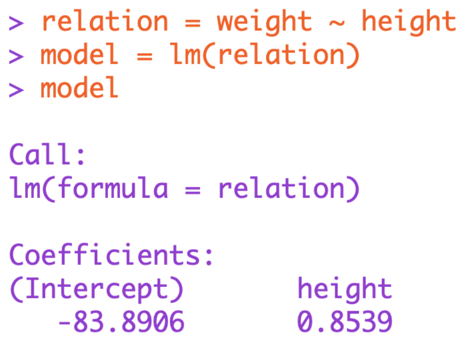
When looking for a regression line $y = a + b \cdot x$ we need to precise how to choose the intercept a and the slope b, in a way that the line crosses the point cloud in the 'best possible way'. This can always be achieved as demonstrated in the **Gauss - Markov theorem** (e.g. [17, page 18]): the regression line is the Best Linear Unbiased Estimate ('BLUE') according to the Ordinary Least Square (OLS) estimation, a method explored since 1755 by the dalmatian Ruggero Boscovich / Ruđer Bošković [42]. Simply, one consider all the **residuals** (one of them is the violet segment in the above figure) and, likewise in the Pythagorean theorem, one consider the sum of the squared residuals (i.e the sum of the squared 'vertical' distances from each cloud point (x_i, y_i) and its vertical projection of the line, i.e. the point $(x_i, a + b \cdot x_i)$). In other words, the residuals are defined as $\varepsilon_i = y_i - (a + b \cdot x_i)$ and defining the vector $\varepsilon = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_i, \ldots, \varepsilon_n)$ one computes the scalar product $\varepsilon^T \cdot \varepsilon \equiv < \varepsilon | \varepsilon >$ and search the parameters a and b which minimize such scalar product.

R On the web there are very nice pages showing this:

- `https://setosa.io/`
- `https://seeing-theory.brown.edu`

It is easy to calculate such $a = -83.89$ and $b = 0.85$ with R:

```
> model = lm(relation)
> model
```



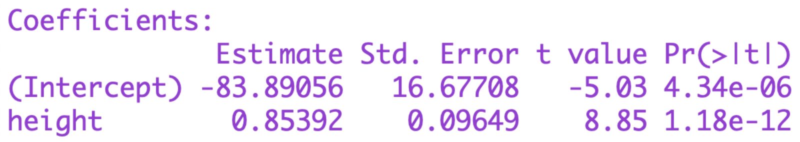
```
> relation = weight ~ height
> model = lm(relation)
> model

Call:
lm(formula = relation)

Coefficients:
(Intercept)       height
   -83.8906       0.8539
```

After having discovered a and b, which represent the **fixed effects** of the linear model, we have to describe also the **stochastic component** of the linear model, or **random effect**. The theory requires in fact that residuals ε_i have to be independent and normally distributed, with zero mean, and with a constant standard deviation σ (remember section 5.3.3). To estimate that $\sigma = 6.46$, we can type:

```
> summary(model)$sigma
```

Now, a question: how to be 'statistically sure' that the `weight` increases with `height`? In other words, how to be 'statistically sure' that the $b = 0.85$ differs from zero and the regression line is not nearly horizontal? This question is similar to that exposed in the 'signal to noise ratio' of Section 5.1; we look to the coefficients in the `summary`:

```
> summary(model)$coefficient
```



```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -83.89056   16.67708   -5.03 4.34e-06
height        0.85392    0.09649    8.85 1.18e-12
```

If we use the $t = \frac{x_m - \mu}{s/\sqrt{n}}$ relation, we obtain exactly the statistic $t = \frac{0.85392-0}{0.09649} = 8.849\ldots$. Being more than 8 deviates far away from 0, we are sure (i.e. p value `1.18e-12` < 0.001) that the line has a not null slope, i.e. that `weight` is predicted by `height`.

Exercise 6.1 Check the meaning of a horizontal regression line. Generate 100 normally distributed random points `x`, and generate other 100 normally distributed random points `y` with a certain off-set, say 13:

```
> x = rnorm(100)
> y = rnorm(100) + 13
```

Of course, we are not able to predict `y` knowing `x`: they are random. Check that computing the `summary` of the `lm` linear model applied to the `formula = y ~ x` the slope of the regression line is not significant and close to zero; and in the `plot` the regression line is horizontal, which is the peculiarity of **uncorrelated data**.

```
> formula = y ~ x
> model = lm(formula)
> summary(model)
> plot(x,y)
> abline(model)
```

Exercise 6.2 Think a simple way to check that the regression line $y = a + bx$ passes through the **mass center** of the point cloud, i.e. the point (`mean(height)`, `mean(weight)`).

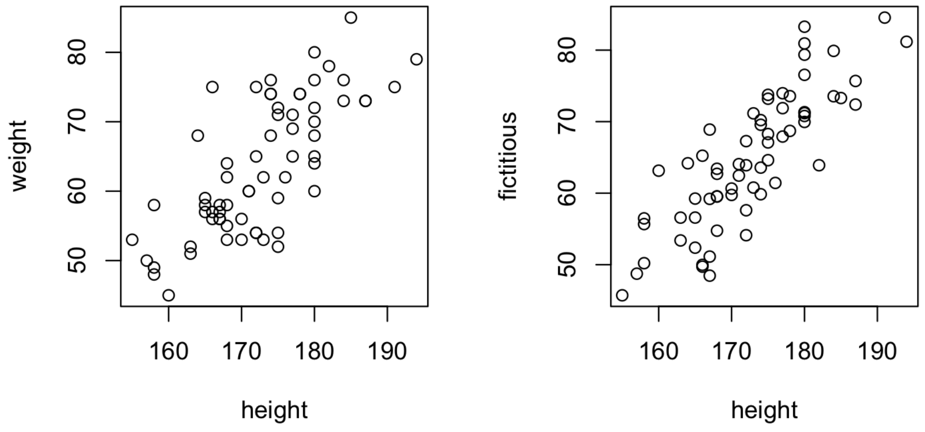
6.2.1 Understanding random effect

Let us explain better the concept of the stochastic component of the linear model, as measured in the `summary(model)` by the `residual standard error`. The regression line is a statistical model which conveys three parameters – and a fourth parameter taken for granted: two of them are the fixed effects a and b, the intercept and the slope of the line. The 'taken for granted' parameter is the null mean of the residuals, that is the normal distribution $N(\mu, \sigma)$ describing the residuals is always of the form $N(0, \sigma)$. Lastly, the third parameter is the standard deviation σ quantifying the dispersion of the sampled residuals along the null mean (remember the standard error of the mean in Section 4.1).

In our example, the `summary(model)$sigma` provides the $\sigma = 6.46$ estimate. Now look to the below picture: in the left panel, the true `weight` vs. `height`; the right panel instead shows a plot of 65 `fictitious` weights random generated, according to the estimated regression line perturbed by a `randerror` normally distributed with null mean and 6.46 standard deviation.

```
> set.seed(4321) # for reproducibility
> randerror = rnorm(65, 0, 6.46)
> fictitious = -83.891 + 0.854 * height + randerror
> par(mfrow = c(1,2))
> plot(height, weight)
> plot(height, fictitious)
```

The fact that two panels resemble each other suggests that our linear model is well posed. We will discuss better this issue in the following chapters.



6.2.2 Measuring point cloud disorder

The (positive) slope $b = 0.85$ is also a 'proxy' measure of the **(positive) correlation** which intervenes between `height` and `weight`. But $b = 0.85$ do not indicate whether the points in the cloud are more or less 'adherent', 'snug-fitting' to the line: we need a way to measure the 'disorder' around the regression line caused by that unruly cloud. Being b a slope, i.e. a quantity of the type $\Delta y/\Delta x$, it is natural to multiply it by a quantity $\Delta x/\Delta y$ to obtain a pure number; in fact:

$$\rho = b \cdot \frac{\sigma_x}{\sigma_y}$$

is defined as the (**linear**) Auguste Bravais - Karl Pearson **correlation coefficient** (also called the product - moment correlation coefficient). In our example, $\rho = 0.74$:

```
> cor.test(height, weight)
> b = summary(model)$coefficient[2]
> b * sd(height) / sd(weight)
```

The squared value of the correlation coefficient, ρ^2, is called **coefficient of determination**, usually noted as R^2

```
> cor(height, weight)^2     ## 0.554184
> summary(model)$r.squared  ## 0.554184
```

The notion of coefficient of determination is linked to that of **Kullback - Leibler information measure**, which happens to be the negative of **Boltzmann's entropy** [9]:

```
> install.packages("rsq")
> library(rsq)
> rsq.kl(model)           ## 0.554184
```

All these ideas are related to the statistical concept of **model deviance**, which in the linear model is nothing but the pythagorean sum of the squared residuals:

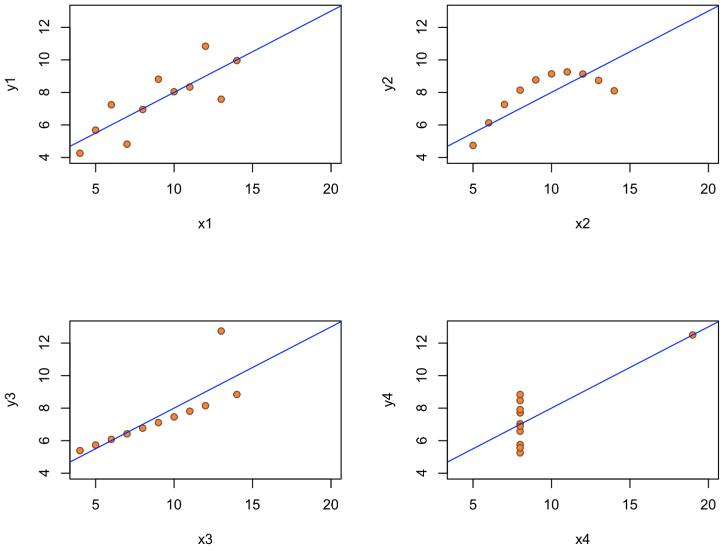
```
> sum(summary(model)$residuals^2)
> deviance(model)
```

In our example, the model deviance is 2628.63. One can check that assuming that there is not any predictor to `weight`, i.e. considering the **null model**, `lm(weight ∼ 1)` – which is indeed the mean and the standard deviation of `weight` – and computing its deviance, 5896.22, one can again obtain the R^2 determination coefficient, as one minus the deviances' ratio:

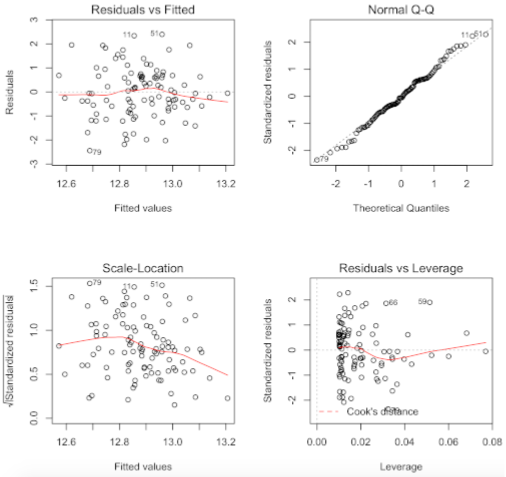
```
> nullmodel = lm(weight ~ 1)
> sum(summary(nullmodel)$residuals^2)
> deviance(nullmodel)
> 1 - deviance(model)/deviance(nullmodel)
> summary(model)$r.squared
```

6.2.3 The diagnostic plots

In a remarkable 1973 paper [5], Francis Anscombe exhibited four artificial datasets, indeed very different one another, but characterized to have the same regression lines and the same R^2 determination coefficients. You can import them in R with the syntax `attach(anscombe)` and check what plotted on Wikipedia: `https://en.wikipedia.org/wiki/Anscombe%27s_quartet`



We therefore need some tools to judge the 'quality' of our model, to decide if our linear model fits well data, or not. This can be done by means of the so-called **diagnostic plots**, a group of four panels enlightening the linear model mathematical hypotheses fulfillment. Michaael Crawley [12, page 357] explains how to understand these plots very well the details; we quote professor Crawley's words, adapting them to our case. The first two graphs are the most important. First, you get a plot of the residuals against the fitted values (above left plot) which do not shows a very pronounced curvature. Remember, this plot should look like the sky at night, with no pattern of any sort. Second, you get a QQ plot as introduced in Subsection 3.2.1, which do not indicate pronounced non-normality in the residuals (the line is straight, not banana-shaped or humped).



The third graph is like a positive-valued version of the first graph; it would good for detecting non-constancy of variance (heteroscedasticity), if showing up as a triangular scatter (like a wedge of cheese). The fourth graph shows a pronounced pattern in the standardized residuals as a function of the leverage. The graph also shows **Cook's distance**, which combine leverage and residuals in a single measure, highlighting the identity of particularly influential data points, which – if present – appear as isolated numbered points outside some red hyperbolas.

R To discuss further aspect of diagnostic plots have a look to the .pdf *Data Quality Assessment Statistical Methods for Practitioners* provided by EPA (on `https://nepis.epa.gov`, or simply on Google)

Exercise 6.3 Check that in the `anscombe` dataset, the third example exhibit a relevant isolated point. Check in particular that in the diagnostic plots, most of the residuals are negative with a constant drift, and the positive residuals are concentrated on the left. This suggests systematic inadequacy in the structure of the model. Moreover, the fourth graph shows a pronounced pattern in the standardized residuals as a function of the leverage, enlightening the role of the 3rd record.

```
> attach(anscombe)
> m3 = lm(y3 ~ x3)
> par(mfrow = c(2,2))
> plot(m3)
```

■

Exercise 6.4 Check that in the `airquality` dataset, it is not sufficient to model `Ozone`'s level in function only of `Temperature`. Check in particular that you get a plot of the residuals against the fitted values (above left plot) which shows very pronounced curvature; most of the positive residuals are on the left and on the right, while the negatives one are concentrated in the central part of the plot. This suggests systematic inadequacy in the structure of the model, which can be improved considering maybe a log-transform of `Ozone`, or a quadratic term for `Temperature`.

```
> attach(airquality)
> relationwrong = Ozone ~ Temp
> wrongmodel = lm(relationwrong)
> par(mfrow = c(2,2))
> plot(wrongmodel)
```

■

Exercise 6.5 In Section 6.3 we approached the t-test as a particular case of linear model. Check that in the `fresher` dataset, the t-test was 'proper' to assess differences in `weight` vs. `gender`: residuals appears to be normally distributed in the QQ plot.

```
> relation2 = weight ~ gender
> discovermodel = lm(relation2)
> par(mfrow = c(2,2))
> plot(discovermodel)
```

■

Let us summarise the question raised in Section 3.2.1 where we introduced the misleading sentences concerning the sum of normal variables. Check that in the `fresher` dataset, the t-test can be considered 'proper' to assess differences in `weight` vs. `gender`: residuals appears to be quite normally distributed in the Q-Q plot. But 'the sum' of two normals is not 'normal', as it appears in the bimodal histogram of the weight:

```
> hist(weight)
```

6.3 The Ancova

In the previous sections and 6.2 we have observed that in the fresher dataset the weight can be predicted, in a statistical sense, by gender and by height. The question now is: how can we 'melt' together those two predictors performing a **multivariate** (or, better to say, **multivariable**) regression? The classical approach to this question is known with the term **Ancova**, which stands for 'analysis of covariance'.

Start exploring what happens with the command split(weight, gender). As you see, the 65 weights are splitted into two groups, the first belonging to the girls, the other to the boys. So, one viable idea could be to split also the heights according to gender, and to study the two separate regression lines, one for the girls, one for the boys. Here you have one possible R code:

```
> weightf = split(weight, gender)$f
> weightm = split(weight, gender)$m
> heightf = split(height, gender)$f
> heightm = split(height, gender)$m
> plot(height, weight, col = "white")
> points(heightf, weightf, col = "orange", pch = 19)
> abline(lm(weightf~ heightf), col = "orange")
> points(heightm, weightm, col = "violet",  bg = "violet", pch = 22)
> abline(lm(weightm ~ heightm), col = "violet", lty = 2, lwd = 2)
```

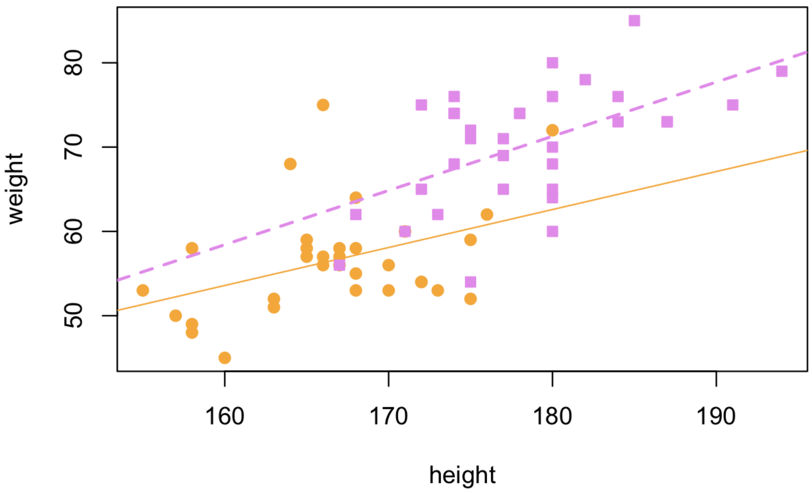
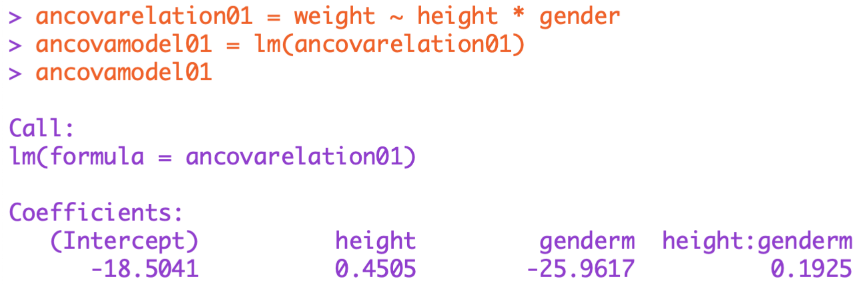


Figure 6.1: An example of Ancova: females and males are respectively painted in orange, $y = -18.50 + 0.45 \cdot x$, and violet, $y = -44.47 + 0.64 \cdot x$. The natural question arising is: do the orange regression line differ, in statistical sense, from the violet line by slope, by intercept, or both?

We have a very compact and elegant way to analyse such a situation in terms of linear models. The only point to focus is the 'strange' **Wilkinson and Rogers notation** adopted to describe the statistical relations; when we want to investigate on weight versus height relation, imagining that gender may influence (or better, interact with) that relation, we adopt the cross operator, *, and we talk of **Ancova with interaction** analysis:

```
> ancovarelation01 = weight ~ height * gender
> ancovamodel01 = lm(ancovarelation01)
> ancovamodel01
```



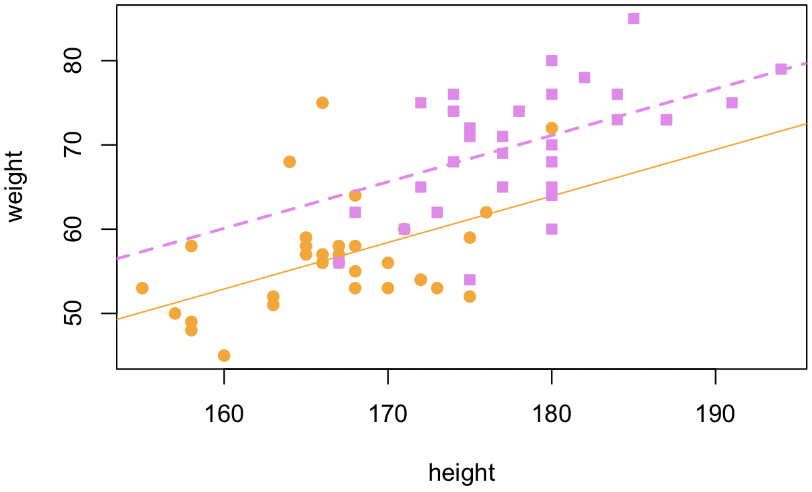
```
> ancovarelation01 = weight ~ height * gender
> ancovamodel01 = lm(ancovarelation01)
> ancovamodel01

Call:
lm(formula = ancovarelation01)

Coefficients:
   (Intercept)          height         genderm  height:genderm
      -18.5041          0.4505        -25.9617          0.1925
```

Now we can interpret that output, recognising the equations of the orange and violet regression lines in Figure 6.1: the females (according to the R alphabetical order) have Intercept -18.50 and slope 0.45, i.e. $y = -18.50 + 0.45 \cdot x$. The last two terms (i.e. genderm and – according to the 'strange' Wilkinson and Rogers notation – height:genderm) modify the coefficients of the girls regression line, in an additive way; therefore, for males, Intercept is -18.50 - 25.96 and slope is 0.45 + 0.19 , i.e. $y = -44.47 + 0.64 \cdot x$.

But there is another possibility to consider, which is known as the **Ancova without interaction**, which can be called using the plus operator, +:

```
> ancovarelation02 = weight ~ height + gender
> ancovamodel02 = lm(ancovarelation02)
> ancovamodel02
```

The output not reported shows only three parameters, and not four as in the previous model: the females have Intercept -35.37 and slope 0.55, i.e. $y = -35.37 + 0.55 \cdot x$, while the last term genderm is an offset for the males Intercept, i.e. -35.37 + 7.20, $y = -28.18 + 0.55 \cdot x$. In this model, gender has an effect in moving up or down the regression line, but not in changing its slope: that's the reason why we see two parallel lines.



Of course, we urge to define a criterion in order to decide whether ancovarelation01 or ancovarelation02 is the 'best' statistical relation to explain that point cloud. One possible choice is to investigate on the deviances of the two linear models, as explained in previous section 6.2.2. In such a case, R has the (badly named, if I might say) **analysis of deviance** `anova()` function, which performs an F test similar to `var.test`. Anyway, many authors [3, 16] manteins that there are convincing reasons to avoid such an approach: the best thing to do is to adopt the Akaike Information Criterion, as we are going to see.

6.3.1 The Akaike Information Criterion

In Section 6.2.2 we tried to measure the disorder in a point cloud, exploiting the concept of deviance, which in turn is related to that of residuals. And we remember that those residuals have to be independent and normally distributed, with null mean and some constant standard deviation. There, we also evaluated that standard deviation $\sigma = 6.64$ with these commands:

```
> relation = weight ~ height
> model = lm(relation)
> sigmamodel = summary(model)$sigma
> sigmamodel
```

But actually, the proper standard deviation to consider in the present case is the maximum likelihood estimate [32], which takes in account the fact that the 65 original `heights` of the students had already provided 2 information (i.e. the intercept and the slope of the regression line) and then we have only 63 free information to exploit. The result is $\sigma_{ML} = 6.36$:

```
> sigmaMLmodel = sigmamodel * sqrt((65-2)/65)
> sigmaMLmodel
```

Now we use the notion of independence: can we evaluate the probability, or better the **likelihood**, to observe by chance exactly these residuals? The independence guarantees that we can multiply the single probability, concentrated on each residual thought to be a random event. But multiplying 65 probabilities yield a result very close to zero; this is the reason why usually one passes to the logarithms, in order to transform products of probabilities into their sums:

```
> sum(log(dnorm(resid(model), mean = 0, sd = sigmaMLmodel)))
```

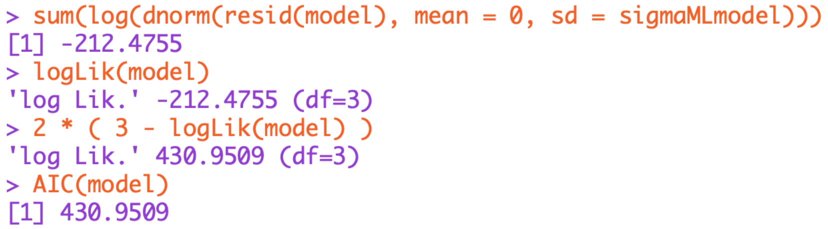
The quantity here obtained, -212.48, is defined to be the **log-likelihood** of the model, and it can be calculated by the command:

```
> logLik(model)
```

And now, the great intuition of professor Hirotsugu Akaike[2]: to penalize the log-likelihood of the model with the ’cost’ of the total parameter used (in our example, 2 fixed effects + 1 random effect = 3), in agreement to the Kullback - Leibler theoretical framework [10, pages 28-30]. Here we have the Akaike information criterion:

```
> 2 * ( 3 - logLik(model) )
> AIC(model)
```

Now we have a reliable tool to select the proper model in the cross-section studies datasets. Let us see how this stuff works.



```
> sum(log(dnorm(resid(model), mean = 0, sd = sigmaMLmodel)))
[1] -212.4755
> logLik(model)
'log Lik.' -212.4755 (df=3)
> 2 * ( 3 - logLik(model) )
'log Lik.' 430.9509 (df=3)
> AIC(model)
[1] 430.9509
```

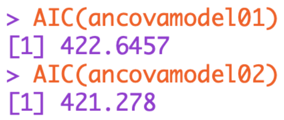
6.3.2 The Model Selection

We follow professor Michael Crawley's crystal clear words [12, page 353]:

> The more parameters that there are in the model, the better the fit. You could obtain a perfect fit if you had a separate parameter for every data point, but this model would have absolutely no explanatory power. There is always going to be a trade-off between the goodness of fit and the number of parameters required by parsimony. AIC is useful because it explicitly penalizes any superfluous parameters in the model, by adding $2(p+1)$ to the deviance.
> When comparing two models, the smaller the AIC, the better the fit.

Let us return to the unsolved question of section 6.3: we have to decide if it is proper to adopt the `ancovamodel01` which 'costs' four fixed effects (two slopes and two intercepts), or the 'less expensive' `ancovamodel02` with a common slope is sufficient. We compute their Akaike Information Criteria:

```
AIC(ancovamodel01)
AIC(ancovamodel02)
```



```
> AIC(ancovamodel01)
[1] 422.6457
> AIC(ancovamodel02)
[1] 421.278
```

We are done. We will completely appreciate the efficacy of this method in the next two topics, devoted to the Anova and to the binomial regression.

6.4 The Anova

Recall that t-test is able to detect differences in means between two groups, as its test statistic is defined as:

$$t = \frac{m_1 - m_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$$

Having to treat for instance three groups, it would be easy to modify the denominator, $\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2} + \frac{s_3^2}{n_3}}$. But the numerator would be undefined: $m_1 - m_2 - m_3$? $m_1 + m_2 - m_3$? $m_1 - m_2 + m_3$? And so on. It is possible to overcome this difficulty observing that when differences in mean are present, also the data dispersions, i.e. the variances, decrease. We know that `gender` is a predictor of `weight`, and we see (Figure 6.2 above left panel) that the `weight` variances of girls and boys are, respectively, 41.1 and 50.7: a great reduction with respect to the 92.1 variance of the whole `weight` data. On the contrary (Figure 6.2 above right panel), there is not any significant difference in mean of `weight` according to `smoke` (p-value = 0.52); and splitting the `weight` into the two groups of smokers ($\sigma^2 = 106.4$) and not smokers ($\sigma^2 = 89.4$) do not sensibly reduce the 92.2 variance. And this is the reason why Anova (= *An.o.va.*, *An*alysis *o*f *Va*riance) is an indirect way to discover differences between means.

```
> t.test(weight ~ gender)$p.value
> t.test(weight ~ smoke)$p.value
> tapply(weight, gender, var)
> tapply(weight, smoke, var)
> var(weight)
```

Now, if you like the horror movies, go on reading the next subsection 6.4.1. But if you love the k.i.s.s. (i.e. keep it simple, stupid!), proceed without doubts to the subsection 6.4.2

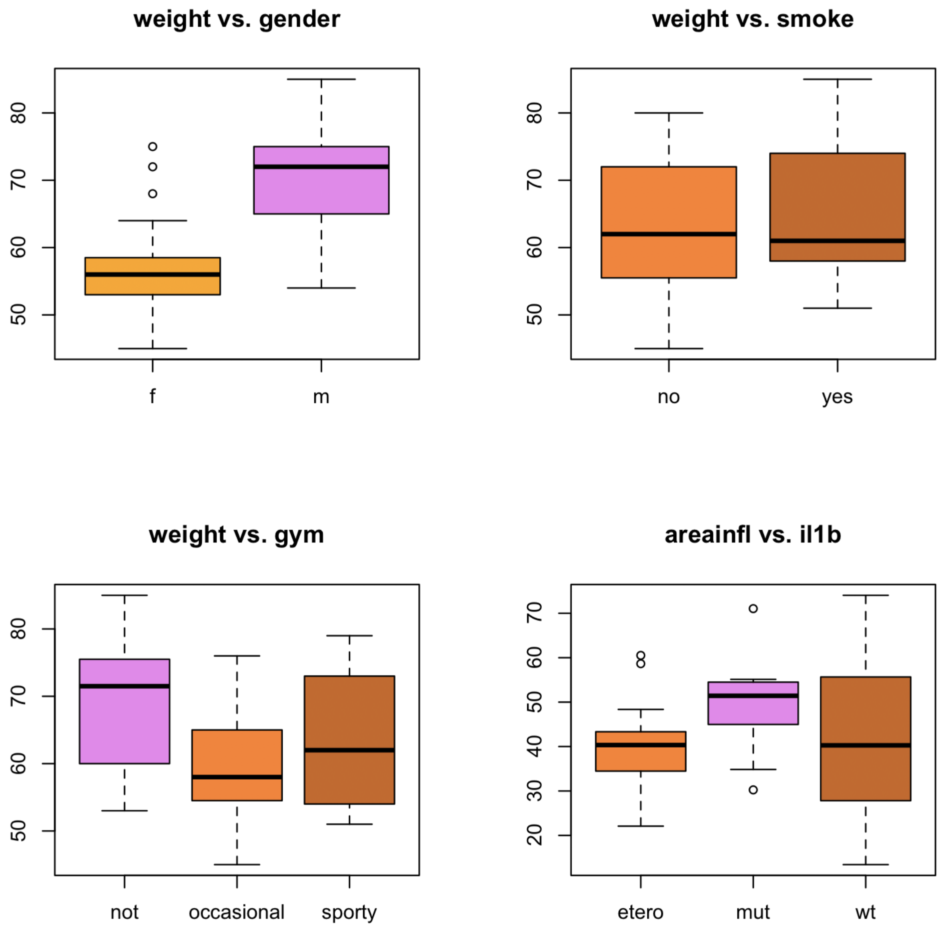
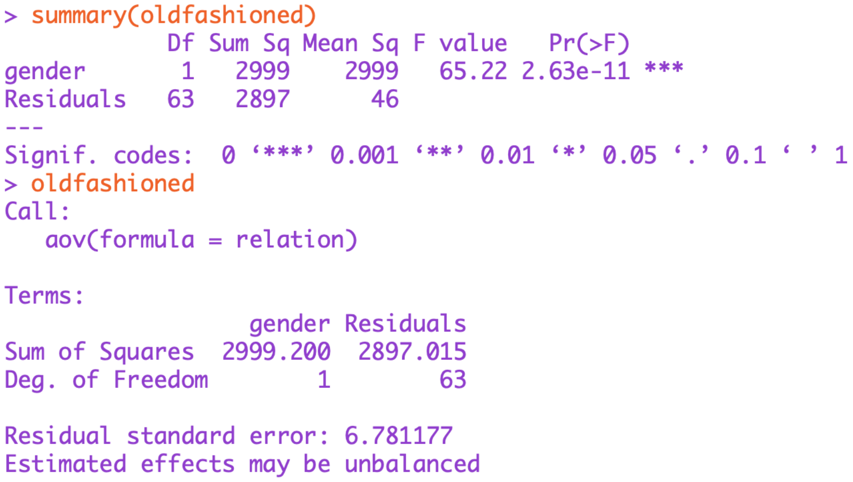


Figure 6.2: In the above left panel, `gender` is a predictor of `weight`. On the right, `smoke` is not a predictor of `weight`: the boxes are wider than those in the left panel - the variance has not reduced, and therefore there is no difference in mean. In the below left panel, `fresher`'s `weight` of `not` gymming differs from the others two levels, which are equivalent in statistical sense. In the below right panel, despite a strong heteroskedastic situation, `mutated` patients have a different gengival `areainflamation` that `heterozygotes` or `wild-type` patients, the latter two levels not differing each other.

6.4.1 The old-fashioned approach

The classical approach to the **one-way Anova** analysis in R – i.e. one `numeric` response and one `factor` covariate – exploits the `aov` command; let us check its 'poor' output in a case that we know very well since the 6.3 section: the t-test.

```
> relation = weight ~ gender
> oldfashioned = aov(relation)
> summary(oldfashioned)
> oldfashioned
```



```
> summary(oldfashioned)
            Df Sum Sq Mean Sq F value   Pr(>F)    
gender       1   2999    2999   65.22 2.63e-11 ***
Residuals   63   2897      46                     
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> oldfashioned
Call:
   aov(formula = relation)

Terms:
                  gender Residuals
Sum of Squares  2999.200  2897.015
Deg. of Freedom        1        63

Residual standard error: 6.781177
Estimated effects may be unbalanced
```

The calculation performed are similar when F-testing two variances, as in subsection 5.3.1, being the variance splitted to compute the deviance as shown in subsection 6.2.2:

```
> var(weight)*(length(weight)-1)-deviance(anovamodel)
> deviance(anovamodel)
```

But the summary reports only that there is a significant difference (p-value `1.18e-12`), that the normally distributed residuals have dispersion $\sigma = 6.46$ and that 'Estimated effects may be unbalanced' in the sense that the dimensions of the two group do not coincide. The summary of the `lm` command was much more complete.
This lack of information in particular penalizes the analysis when we try to predict `weight` in function of `gym`, a three level (Figure 6.2 below left panel) alphabetically ordered factor: `not`, `occasional`, `sporty`.

```
> relation3 = weight ~ gym
> anovamodel = aov(relation3)
> summary(anovamodel)
```

We can discover only that p-value = 0.003, so that `gym` has a statistically significant effect on `weight`, and stop! But at least two open questions remain, discussed in the next two paragraph.

1. The mathematical hypotheses of the Anova

Vijay Rohatgi [37] correctly states that:

> Let $X_{11}, X_{12}, ..., X_{1n_1}, X_{21}, X_{22}, ..., X_{2n_2}$ and $X_{31}, X_{32}, ..., X_{3n_3}$ be independent random samples from three normal populations with respective parameters μ_1 and σ_1^2, μ_2 and σ_2^2 and μ_3 and σ_3^2. Suppose $\sigma_1 = \sigma_2 = \sigma_3$. ...

Therefore if one wants to perform an Anova according the traditional way, it is required to check whether in `weight`:

1. all three groups `not`, `occasional`, `sporty` are normally distributed (by means of `qqnorm` or `shapiro.test`)
2. their dispersions are homoskedastic, i.e. in statistical sense $\sigma_1 = \sigma_2 = \sigma_3$.

To check for differences in dispersion when groups are mode than two one can use the Maurice Bartlett's test, `bartlett.test(weight ~ gym)`, but this test is very sensitive to outliers or to normality violations, and therefore many scientists prefer to use the Howard Levene's test, `leveneTest(areainfl, il1b)` (but you are required to download and install in advance the package `car`).

1.1 How to mend not-normality

In presence of not normality, [46], [12], often it is exploited the George Box and sir David Cox transformation:

```
> library(MASS)
> boxcox(anovamodel)
```

The parabola depicted is the log-likelihood [16, 46] of the model, varying over the transformation of `weight` powered to an exponent λ. The graph suggest in this case to adopt (approximatively) an exponent 1/4 (i.e. to perform an Anova on `weightboxcox = sqrt(sqrt(weight)))`

1.2 How to mend heteroskedasticity

If we want to test differences in means exploiting reduction on variances, but the variances are very different (Figure 6.2 below right panel), we are in deep trouble. Consider in fact the Sara De Iudicibus [14] `tooth` dataset, in which the three group patients (`heterozygous`, `mutated`, `wild-type`) according the Interleukin-1 beta (IL-1β) – a cytokine mediating inflammatory response – are related to the gingival inflammation area (`areainfl`) measured on a digital image. While the variance of `areainfl` is $\sigma = 205.3$, splitting it according to `il1b` yields for `etero`, `mut` and `wt` respectively 88.5, 127.2 and 294.1. This trouble heavily affects the following question of multiple comparisons.

```
> www = "http://www.biostatisticaumg.it/dataset/tooth.csv"
> tooth = read.csv( www, header = TRUE )
> var(tooth$areainfl)
> tapply(tooth$areainfl, tooth$il1b, var)
```

2. The multiple comparison issue

We see that in `relation3`, i.e. `weight` versus `gym`, the Anova p-value is significant. But such a p-value do not disclose which group is different from the other (spoiler alert: in Figure 6.2 the solution is revealed by the violet colour). Richard Mould's words in his chapter 17.1 [34] are clear:

> With more than two means it is of course technically possible to make multiple t-tests on all possible pairs of means, but *making multiple tests increases the probability of making a type I error*.

In fact, suppose to choose an α level of 5%; then, the probability to commit an error of the fisrt type is about the 14% (independent events, product of probabilities):

$$1-(1-\frac{5}{100})\cdot(1-\frac{5}{100})\cdot(1-\frac{5}{100}) = 1-(1-\frac{5}{100})^3 = 0.143$$

One 'radical' solution is to exploit the Bernoulli inequality $1+nh < (1+h)^n$, i.e. if we have $n = 3$ groups and therefore $n\cdot(n-1)/2 = 3$ comparisons, then one fix $h = \alpha/3$, i.e. $\alpha = 0.05/3 = 0.017$.

This is the famous **Carlo Bonferroni correction**[35] . Here, not gym versus sporty has a p-value = 0.08.

```
> weightboxcox = sqrt(sqrt(weight))
> pairwise.t.test(weightboxcox, gym, p.adj="bonferroni")
```

One milder and elegant approach is to trust in John Tukey and adopt his Honest Significant Differences multiple comparison test[12]:

```
> TukeyHSD(anovamodel)
```

Here, not gym versus sporty has a p-value = 0.060. As you see, everything appears to be shaky and slippery. And when the situation is heteroskedastic, the things are even worse.

2.1 How to mend heteroskedasticity - conclusion

Let us go back to the tooth question. We need to install two packages, sandwich [53] and multcomp [23], as magistrally explained in the *Multiple comparisons using R* by Bretz, Hothorn and Westfall [8]. We shall use the glht general linear hypotheses test. In this way, one discover that hetero versus mut has p-value = 0.024.

```
> library(multcomp)
> library(sandwich)
> posthoc = glht(anovamodel, linfct = mcp(gym = "Tukey"), vcov = sandwich)
> summary(posthoc)
```

Again, everything do not appear simple at a first glance. We hope to have convinced readers to skip all this stuff, and to proceed to the following simpler approach.

6.4.2 The Anova with AIC Model Selection

Let us recap: we try to predict weight in function of gym, a three level (Figure 6.2 below left panel) alphabetically ordered factor: not, occasional, sporty. We set the linear model and evaluate its Akaike Information Criterion, which is equal to 473.1

```
> relation3 = weight ~ gym
> linearmodel = lm(relation3)
> AIC(linearmodel)  ## 473.1
```

Now (the 'multiple comparisons issue') we have to decide if linearmodel is the minimal adequate model and all the three different levels provide different information; or some levels can be joined together. Let us make some attempts, 'melting' together the gym factor levels in all the possible two by two manners, as shown in Table 6.1.

```
> gymNO = gym
> levels(gymNO)[1] = "notoccasional"
> levels(gymNO)[2] = "notoccasional"
> gymNS = gym
> levels(gymNS)[1] = "notsporty"
> levels(gymNS)[3] = "notsporty"
> gymOS = gym
> levels(gymOS)[2] = "occasionalsporty"
> levels(gymOS)[3] = "occasionalsporty"
```

weight	gym	gymNO	gymNS	gymOS
53	not	notoccasional	notsporty	not
58	not	notoccasional	notsporty	not
50	occasional	notoccasional	occasional	occasionalsporty
49	occasional	notoccasional	occasional	occasionalsporty
73	sporty	sporty	notsporty	occasionalsporty
79	sporty	sporty	notsporty	occasionalsporty

Table 6.1: Example of six `freshers` whose `gym` activity has been re-grouped in a two by two manner.

To be clear, this can be thought as a procedure which 'glues' to the dataset three new columns, `gymNO`, `gymNS` and `gymOS`; and all the new columns instead of having three levels has only two of them – see Table 6.1.

Now, it is sufficient to compute the Akaike Information Criterion for all these new linear models, and to choose the smaller. And Bob's your uncle!

```
> AIC(lm(weight ~ gymNO))
> AIC(lm(weight ~ gymNS))
> AIC(lm(weight ~ gymOS))
```

We observe that the latter linear model, `weight` versus `gymOS`, has the lowest AIC = 472.7. Therefore it can be choosen as the minimal adequate model, and we interpret this saying that, in the `fresher` dataset, those who `not` practice `gym` has a `weight` significantly different from those who practice it in an `occasional` manner, or those who are `sporty`; and the latter two conditions do not differ between them (now, check again the orange/brown boxplots in the below left panel of Figure 6.2).

Exercise 6.6 Do you remember `iris` dataset? Decide whether `Sepal.Width` differs between `Species` by means. ■

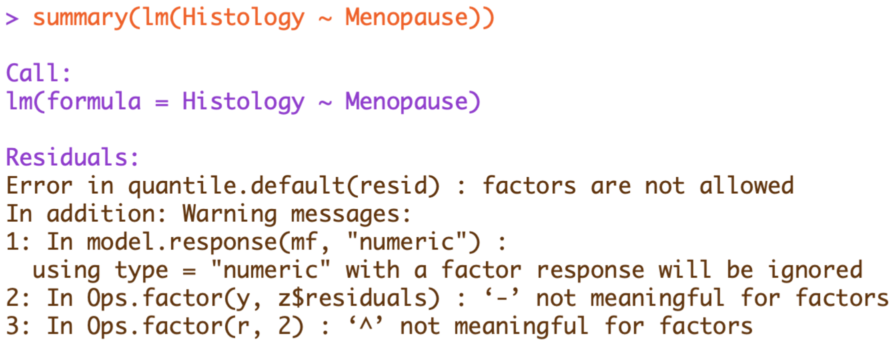
7. The generalized linear models

7.1 Overview

Biostatisticians often are consulted by Biologists or Physicians when seeking for reliable oncological biomarkers. In such a case, the typical response comes from a retrospective cross-section dataset whose response is of binomial type (benign/malignant, positive/negative, alive/dead, ...). You remember that in section 3.1 we introduced the Shadi Najaf `roma` dataset, in which 210 patients with a known `Histology` response (`benign` or `malignant`) were studied in association to four candidate biomarkers (logaritmic transformed) – `logHE4`, `logCA125`, `logCA19.9` and `logCEA` –, along with their `AgePatients` and their `Menopause` status:

```
> www = "http://www.biostatisticaumg.it/dataset/roma.csv"
> roma = read.csv(www, header = TRUE)
> attach(roma)
> head(roma)
> tail(roma)
```

As explained in subsection 3.2.3, the `Histology` response is not a `numeric` variable, but a `factor` response, mathematically modelled by a binomial random variable as discussed in subsection 3.2.3. Therefore, the `lm` machinery can not work at all:



```
> summary(lm(Histology ~ Menopause))

Call:
lm(formula = Histology ~ Menopause)

Residuals:
Error in quantile.default(resid) : factors are not allowed
In addition: Warning messages:
1: In model.response(mf, "numeric") :
  using type = "numeric" with a factor response will be ignored
2: In Ops.factor(y, z$residuals) : ‘-’ not meaningful for factors
3: In Ops.factor(r, 2) : ‘^’ not meaningful for factors
```

The same problems occur for instance when we have a count response, as described in the Poisson random variable 3.2.4 subsection. In these situations we exploit the **generalized linear models** theory.

Vocabulary 7.1 — Generalized linear model. A generalized linear model is a set of three statistical tools composed by:

1. a **relation**, named the **linear predictor**, between the dataset response and one or more dataset covariates (as seen in previous chapter 6).
2. a (family of) **random variable** able to model the response (or, to say better, to model the residuals)
3. a **link function** which transforms ('injects') the expected value of the random variable modelling the response into the mean of the linear predictor.

In the `roma` dataset a possible relation to investigate is suggested by Moore et al. [33]:

```
moorerelation = Histology ~ Menopause + logHE4 + logCA125
```

In the following section we will discuss step by step all the details of the powerful and comprehensive R `glm` function.

7.2 From the maximal model to the minimal adequate model

To start, it is clear the in the sequel we will concern with the `family = binomial` of random variables, having to model the `Histology`. So, we have set the first of the three ingredients of our seeked generalized linear model. A priori, we do not know which are the `Histology` predictors, i.e. the statistically significant covariates within the `roma` dataset. We start to explore the **maximal additive model**, in which all the covariates are present:

```
> maximalrelation = Histology ~ logHE4 + logCA125 + logCA19.9 + logCEA
                      + AgePatient + Menopause
```

Having set the `maximalrelation` we compute the maximal additive model with the `glm` command, specifying the `family = binomial` of `Histology` response:

```
> maximalmodel = glm(maximalrelation, family = binomial)
> summary(maximalmodel)
```

The `summary` is, as always, very complete; but we immediately recognize that there are some covariates 'full of stars', and other not. Our next goal is to throw away uneuseful variables; we exploit the `step` function, which analyse the AIC criteria when inside a model a single covariate is dropped away, halting when a minimum AIC occurs (the convexity of AIC guarantees the success):

```
> step(maximalmodel)
```

After a while, a very long output will be provided. The last lines provide the clue – the `moorerelation` seems to be the right one:

```
Call:  glm(formula = Histology ~ logHE4 + logCA125 + Menopause, family = binomial)

Coefficients:
  (Intercept)        logHE4       logCA125  Menopausepost
     -14.3770        2.3382         0.6845         0.9378

Degrees of Freedom: 209 Total (i.e. Null);  206 Residual
Null Deviance:      201.6
Residual Deviance: 107.3    AIC: 115.3
```

Now we explain how to use those `Coefficients`, taking as an example the 210-th dataset patient:

```
> roma[210,]
    logHE4 logCA125 logCA19.9 logCEA AgePatient Menopause Histology
210   3.96     4.03      1.67   0.71         63      post malignant
```

Substituting the patient information into the model we evaluate the relation, obtaining what sometimes it is called a **predictive index**, *PI*:

$$P.I. = -14.3770 + 2.3382 \cdot 3.96 + 0.6845 \cdot 4.03 + 0.9378 \approx -1.42$$

Now it is the turn of the third ingredient of the generalized linear models, the **link function**, which transform the $P.I. = -1.42$ into an evaluation of probability. The standard choice is the famous **logit** transformation, a bijective map from $p \in [0,1]$ into $y \in \mathbb{R}$:

$$y(p) = logit(p) \equiv \log(\frac{p}{1-p})$$

Thinking $y(p)$ to be the $P.I.$ result, it is sufficient to compute the inverse function of the *logit* to obtain the probability p. This is the famous sigmoidal function from $P.I. \in \mathbb{R}$ into $p(P.I.) \in [0,1]$ known as the **logistic** function (and this is the reason why often the binomial distributed generalized linear model is called **logistic regression**):

$$p = \frac{e^{P.I.}}{1+e^{P.I.}}$$

In the present example, the 210-th patient had an estimated probability of being `malignant` $p = \frac{e^{P.I.}}{1+e^{P.I.}} = \frac{e^{-1.42}}{1+e^{-1.42}} = 0.19$ (remember, R adopts the alphabetical order, therefore in the binomial (correctly, Bernoulli) random variable `Histology` `benign` is 0 and `malignant` is 1).

Exercise 7.1 Try to write two functions in R, say `probante` and `probpost`, to evaluate the probability of malignancy knowing as input the `logHE4` and `logCA125` values, respectively for a woman in ante or post menopausal status. ■

7.3 Model checking

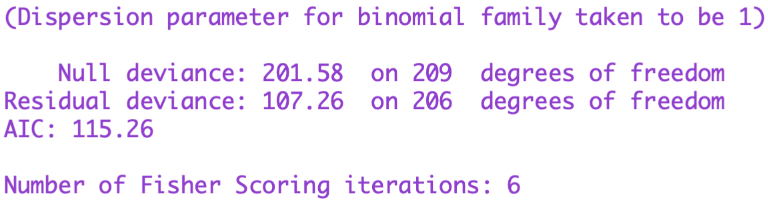
The first thing to check in the `moorerelation`, which is purely additive, is that there is not **interaction** between the predictors. We make three attempt, and we confront their AIC to the `mooremodel` AIC:

```
> attempt1 = Histology ~ Menopause * logHE4 + logCA125
> AIC(glm(attempt1, family = binomial))
> attempt2 = Histology ~ Menopause * logCA125 + logHE4
> AIC(glm(attempt2, family = binomial))
> attempt3 = Histology ~ Menopause + logHE4 * logCA125
> AIC(glm(attempt3, family = binomial))
> mooremodel = glm(moorerelation, family = binomial)
> AIC(mooremodel)
```

These three models respectively have AIC = 115.9, 116.7 and 116.4, while `mooremodel` maintains the lowest AIC = 115.3. Therefore we decide not to consider any interaction within predictors. Another thing to check is the possible presence of **curvature** in the predictors. This can be done inserting in the linear predictor a squared term, according the following syntax:

```
> attempt4 = Histology ~ Menopause + logHE4 + logCA125 + I(logCA125^2)
> summary(glm(attempt4, family = binomial))
> AIC(glm(attempt4, family = binomial)) # 116.9
> attempt5 = Histology ~ Menopause + logHE4 + I(logHE4^2) + logCA125
> summary(glm(attempt5, family = binomial))
> AIC(glm(attempt5, family = binomial)) # 117.1
```

Let us give now a closer look to the last lines of `summary(mooremodel)`:



```
(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 201.58  on 209  degrees of freedom
Residual deviance: 107.26  on 206  degrees of freedom
AIC: 115.26

Number of Fisher Scoring iterations: 6
```

To compute the deviance into a `glm` is not an algebraic straightforward task, but an iterative method is adopted: to maximize the likelihood on the model according to the Newton-Raphson's derivative method [16]. This is what is called the **Fisher Scoring** procedure.

Another possible issue in estimating `glm`'s is the phenomenon of **overdispersion**; remember that the normal distribution `dnorm` depends on two 'free' parameters, the mean μ and the standard deviation σ. On the contrary, in `dbinom` (mean = $n \cdot p \equiv$ variance$/(1-p)$) and `dpois` (mean = $\lambda \equiv$ variance) variance and mean are algebraically related in a fixed manner: as a consequence the residual deviance has an implicit relation with the dimension n of the dataset, and therefore with the degrees of freedom of the model. In our example, the 107.26 residual deviance is **less** than the 206 degrees of freedom: good news, no overdispersion. On the contrary, when in the model `summary` you detect overdispersion, i.e. residual deviance $>$ degrees of freedom, you can act in the `glm` call invoking `family = quasibinomial`. The result will be a computation of a **dispersion parameter** different from 1.

Also in the `glm` it is possible to make some dignostic, by means of the commands `residuals` and `influence`; we recommend reading the milestone-book of Julian Faraway [16].

7.4 Conclusions

Time has gone and our course has ended. But many other important topics are covered into Richard Mould's textbook [34]. For instance, his Chapter 14 is devoted to **survival analysis** and to Kaplan-Meier estimators. If you prefer, you can have a look to those general introductory surveys:

- `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3071962/`
- `https://ccforum.biomedcentral.com/articles/10.1186/cc2955`

There are also many tutorials on the web devoted to survival analysis performed with R. For instance:

- `https://www.datacamp.com/community/tutorials/survival-analysis-R`
- `https://www.emilyzabor.com/tutorials/survival_analysis_in_r_tutorial.html`
- `https://www.r-bloggers.com/steps-to-perform-survival-analysis-in-r/`

One important (and difficult, I say) argument not covered by Mould's text concerns **longitudinal experimental design**, in which for instance we collect **repeated measures** on the same patient. We introduce the difficulty by a simple didactical example. Alice and Ellen are twin, and they have a silly question: *have Alice and Ellen the same weight?* They decide to measure themselves each day at the same time with the same dress with the same weight scale, and the first day the situation is: Alice, 73.60; Ellen, 73.80. So, *have Alice and Ellen the same weight?* Well, no from a pure mathematical point of view. But, repeating for five days the experiment, the situation:

```
> alice = c(73.6, 73.4, 74.1, 73.5, 73.2)
> ellen = c(73.8, 73.5, 74.6, 73.8, 73.6)
> t.test(alice, ellen, var.equal = TRUE)
```

provide the 73.56 mean weight of Alice and 73.86 for Ellen, but such 0.30 Kg is not a significant difference (t = -1.2227, df = 8, p-value = 0.2562). But, repeating for three weeks the measures, as reported in the table below, the 73.66 Kg mean weight of Alice and 73.94 for Ellen provide a 0.28 Kg significant difference (t = -2.4594, df = 40, p-value = 0.01834).

```
> alice = c(73.6, 73.4, 74.1, 73.5, 73.2, 74.0, 73.6, 73.3, 74.2, 73.6,
73.4, 74.1, 73.6, 73.4, 74.1, 73.5, 73.2, 74.0, 73.6, 73.3, 74.2)
> ellen = c(73.8, 73.5, 74.6, 73.8, 73.6, 74.4, 73.8, 73.5, 74.3, 73.9,
73.6, 74.6, 73.8, 73.6, 74.4, 73.7, 73.5, 74.4, 73.9, 73.6, 74.5)
> t.test(alice, ellen, var.equal = TRUE)
```

Let us recap this strange situation: on the first day the difference was of 0.20 Kg, and we decided this was a difference. After five days we decided that the 0.30 Kg was not a difference; after 21 days 0.28 Kg is a difference. All very strange! The explanation is that Alice and Ellen's weights represent a time series, as it happened in the `airquality` dataset; but in the latter, the measurments appeared to be uncorrelated, while the Alice weight time series is obviously compounded by **correlated data** [47] (and the same obviously occurs for Ellen): it is natural to expect that tomorrow's Alice weight will resemble the current value. The proper tools to manage these kind of data are the **linear mixed effects models** [16, 50], in which the pseudoreplication is managed adding a further random effect.

Bibliography

Articles

[2] Hirotugu Akaike. “A new look at the statistical model identification”. In: *Automatic Control, IEEE Transactions on* 19.6 (1974), pages 716–723 (cited on page 67).

[3] DR Anderson and K Burnham. “Model selection and multi-model inference”. In: *Second. NY: Springer-Verlag* (2004) (cited on page 67).

[4] Edgar Anderson. “The species problem in Iris”. In: *Annals of the Missouri Botanical Garden* 23.3 (1936), pages 457–509 (cited on page 13).

[5] Francis J Anscombe. “Graphs in statistical analysis”. In: *The American Statistician* 27.1 (1973), pages 17–21 (cited on page 63).

[6] Viv Bewick, Liz Cheek, and Jonathan Ball. “Statistics review 13: receiver operating characteristic curves”. In: *Critical care* 8.6 (2004), page 508 (cited on page 32).

[14] Sara De Iudicibus et al. “Effect of periodontal therapy on the course of cyclosporin-induced gingival overgrowth: role of ABCB1 and PAI-1 gene polymorphisms.” In: *Quintessence International* 44.3 (2013) (cited on page 71).

[15] Bradley Efron. “Student’s t-test under symmetry conditions”. In: *Journal of the American Statistical Association* 64.328 (1969), pages 1278–1302 (cited on page 52).

[18] Ronald A Fisher. “The use of multiple measurements in taxonomic problems”. In: *Annals of eugenics* 7.2 (1936), pages 179–188 (cited on page 13).

[19] Gregg C Fonarow et al. “An obesity paradox in acute heart failure: analysis of body mass index and inhospital mortality for 108 927 patients in the Acute Decompensated Heart Failure National Registry”. In: *American heart journal* 153.1 (2007), pages 74–81 (cited on page 38).

[21] Francis Galton. “Regression towards mediocrity in hereditary stature.” In: *The Journal of the Anthropological Institute of Great Britain and Ireland* 15 (1886), pages 246–263 (cited on page 57).

[22] John M Hoenig and Dennis M Heisey. “The abuse of power: the pervasive fallacy of power calculations for data analysis”. In: *The American Statistician* 55.1 (2001), pages 19–24 (cited on page 50).

[23] Torsten Hothorn, Frank Bretz, and Peter Westfall. “Simultaneous Inference in General Parametric Models”. In: *Biometrical Journal* 50.3 (2008), pages 346–363 (cited on page 72).

[26] John PA Ioannidis. “Why most published research findings are false”. In: *PLoS medicine* 2.8 (2005), e124 (cited on page 48).

[27] Brian L Joiner. “Living histograms”. In: *International Statistical Review/Revue Internationale de Statistique* (1975), pages 339–340 (cited on page 37).

[29] Charles J Kowalski. “Non-normal bivariate distributions with normal marginals”. In: *The American Statistician* 27.3 (1973), pages 103–106 (cited on page 38).

[30] Eckhard Limpert, Werner A. Stahel, and Markus Abbt. “Log-normal Distributions across the Sciences: Keys and Clues”. In: *BioScience* 51.5 (2001), pages 341–352 (cited on page 38).

[31] Edward L Melnick and Aaron Tenenbein. “Misspecifications of the normal distribution”. In: *The American Statistician* 36.4 (1982), pages 372–373 (cited on page 38).

[33] Richard G Moore et al. “Comparison of a novel multiple marker assay vs the Risk of Malignancy Index for the prediction of epithelial ovarian cancer in patients with a pelvic mass”. In: *American journal of obstetrics and gynecology* 203.3 (2010), 228–e1 (cited on pages 29, 76).

[36] Xavier Robin et al. “pROC: an open-source package for R and S+ to analyze and compare ROC curves”. In: *BMC Bioinformatics* 12 (2011), page 77 (cited on page 33).

[39] Peter J Rousseeuw, Ida Ruts, and John W Tukey. “The bagplot: a bivariate boxplot”. In: *The American Statistician* 53.4 (1999), pages 382–387 (cited on page 24).

[40] Michèl Schummer et al. “Comparative hybridization of an array of 21 500 ovarian cDNAs for the discovery of genes overexpressed in ovarian carcinomas”. In: *Gene* 238.2 (1999), pages 375–385 (cited on page 33).

[41] T. Sing et al. “ROCR: visualizing classifier performance in R”. In: *Bioinformatics* 21.20 (2005), page 7881. URL: `http://rocr.bioinf.mpi-sb.mpg.de` (cited on page 33).

[43] Student. “The probable error of a mean”. In: *Biometrika* (1908), pages 1–25 (cited on pages 45, 47).

[44] Nick Thieme. “R generation”. In: *Significance* 15.4 (2018), pages 14–19 (cited on page 11).

[45] Sundri P Vaswani. “A pitfall in correlation theory”. In: *Nature* 160 (1947), pages 405–406 (cited on page 38).

[48] Howard Wainer. “The most dangerous equation”. In: *American Scientist* 95.3 (2007), page 249 (cited on page 42).

[49] Ronald L Wasserstein, Nicole A Lazar, et al. “The ASA’s statement on p-values: context, process, and purpose”. In: *The American Statistician* 70.2 (2016), pages 129–133 (cited on page 48).

[51] Elise Whitley and Jonathan Ball. “Statistics review 4: sample size calculations”. In: *Critical care* 6.4 (2002), page 335 (cited on page 50).

[53] Achim Zeileis. “Econometric computing with HC and HAC covariance matrix estimators”. In: (2004) (cited on page 72).

[54] Stephen T Ziliak and Deirdre N McCloskey. “The cult of statistical significance”. In: *Ann Arbor: University of Michigan Press* 27 (2008) (cited on pages 46, 48).

Books

[1] Alan Agresti. *An introduction to categorical data analysis*. Wiley, 2018 (cited on page 18).

[7] Martin Bland. *An introduction to medical statistics.* Ed. 3. Oxford University Press, 2000 (cited on pages 22, 36, 43).

[8] Frank Bretz, Torsten Hothorn, and Peter Westfall. *Multiple comparisons using R*. CRC Press, 2010 (cited on page 72).

[9] Kenneth P Burnham and David R Anderson. *Model selection and multimodel inference: a practical information-theoretic approach*. Springer Science & Business Media, 2003 (cited on page 62).

[10] Gerda Claeskens and Nils Lid Hjort. *Model selection and model averaging*. Cambridge University Press, 2008 (cited on page 67).

[11] William Jay Conover. *Practical nonparametric statistics*. Wiley New York, 1980 (cited on pages 51, 52).

[12] Michael J Crawley. *The R book*. John Wiley & Sons, 2012 (cited on pages 13, 43, 50–52, 63, 68, 71, 72).

[16] Julian J Faraway. *Extending the linear model with R: generalized linear, mixed effects and nonparametric regression models*. Chapman and Hall/CRC, 2016 (cited on pages 67, 71, 78, 79).

[17] Julian J Faraway. *Linear models with R*. Chapman and Hall/CRC, 2016 (cited on page 59).

[20] Michael Friendly and David Meyer. *Discrete data analysis with R: visualization and modeling techniques for categorical and count data*. Chapman and Hall/CRC, 2015 (cited on page 18).

[24] Sergio Invernizzi. *Matematica nelle Scienze Naturali*. Trieste: Edizioni Goliardiche, 1996. ISBN: 8886573170 (cited on page 24).

[25] Sergio Invernizzi, Maurizio Rinaldi, and Federico Comoglio. *Moduli di matematica e statistica – Con l'uso di R*. Zanichelli, 2018 (cited on pages 18, 23, 37).

[28] Michael C Joiner and Albert Van der Kogel. *Basic clinical radiobiology*. Volume 1. CRC press, 2016 (cited on page 39).

[32] Russell B Millar. *Maximum likelihood estimation and inference: with examples in R, SAS and ADMB*. Volume 111. John Wiley & Sons, 2011 (cited on page 67).

[34] Richard F Mould. *Introductory medical statistics*. CRC Press, 1998 (cited on pages 11, 17, 18, 21–23, 25, 29, 32, 34, 35, 38, 45, 49, 52, 54, 59, 71, 78).

[35] Dalgaard Peter. *Introductory statistics with R*. Springer Verlag New York Inc, 2002 (cited on page 72).

[37] Vijay K Rohatgi. *Statistical inference*. Jonh Wiley & Sons, 1984 (cited on pages 38, 70).

[38] Bernard A Rosner. *Fundamentals of biostatics*. Duxbury Press, 1995 (cited on pages 21, 25, 36).

[42] Stephen M Stigler. *The history of statistics: The measurement of uncertainty before 1900*. Harvard University Press, 1986 (cited on page 59).

[46] William N. Venables and Brian D. Ripley. *Modern Applied Statistics with S*. Fourth. ISBN 0-387-95457-0. New York: Springer, 2002. URL: `http://www.stats.ox.ac.uk/pub/MASS4` (cited on pages 20, 26, 71).

[47] Geert Verbeke and Geert Molenberghs. *Linear mixed models for longitudinal data*. Springer Science & Business Media, 2000 (cited on page 79).

[50] Brady T West, Kathleen B Welch, and Andrzej T Galecki. *Linear mixed models: a practical guide using statistical software*. CRC Press, 2014 (cited on page 79).

[52] Hadley Wickham and Garrett Grolemund. *R for data science: import, tidy, transform, visualize, and model data*. O'Reilly Media, Inc., 2016 (cited on pages 11, 19).

Index