Data manipulation and visualization with R

Welcome to R!

Welcome to R!

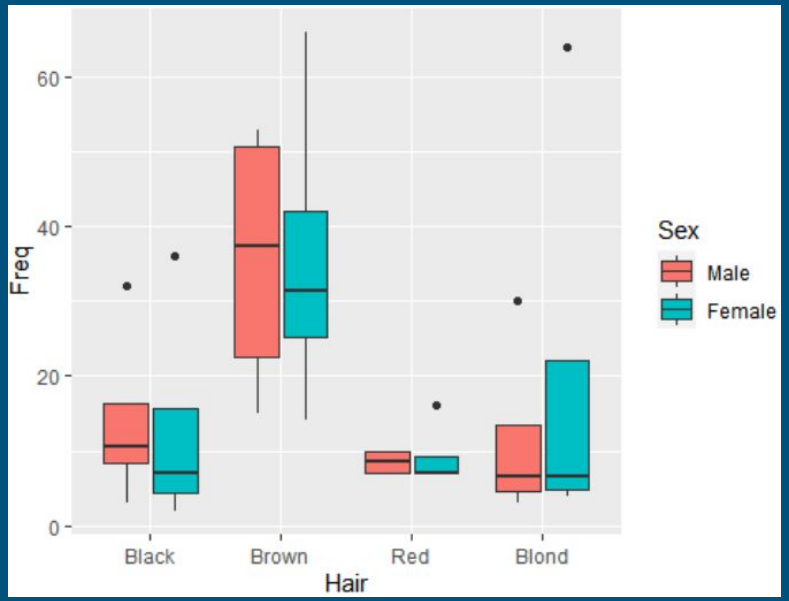
Introduction

Welcome to our course titled “Data manipulation and visualization with R”!

R is a programming language specifically designed for the statistical analysis and visualization of data. In this course we want to teach the principles of R programming and the fundamentals of working with data. The course is made for absolute beginners, who did not work with R (or any other programming language) before and for people who want to peek into the world of data science.

At the end of this course you will be able to read your own data files, manipulate and summarize their data entries and create impressive visualizations.

Both R and the software RStudio are free to download. Some example data files will be provided through this course. You will be able to reproduce everything you see in the learning material provided.



Why R?

Next to Python, R belongs to the most commonly used programming languages in the field of “Data Science”. While R might not be as versatile as Python or C/C++, it doubles down on statistical analyses.

The table shows the TIOBE index of currently popular programming languages, as of July 2020 (https://www.tiobe.com/tiobe-index/). The index takes into account how often the language is sought in search engines and how many skilled engineers use it world-wide. Since 2014, R has had a constant presence in the Top 30 ranking, currently taking 8th place and is continuing to increase in popularity.

Furthermore, R is a rather easy to learn language and can serve as a good introduction to the world of programming.

Jul 2020	Jul 2019	Change	Programming Language	Ratings	Change
1	2	^	C	16.45%	+2.24%
2	1	v	Java	15.10%	+0.04%
3	3		Python	9.09%	-0.17%
4	4		C++	6.21%	-0.49%
5	5		C#	5.25%	+0.88%
6	6		Visual Basic	5.23%	+1.03%
7	7		JavaScript	2.48%	+0.18%
8	20	^^	R	2.41%	+1.57%
9	8	v	PHP	1.90%	-0.27%
10	13	^	Swift	1.43%	+0.31%
11	9	v	SQL	1.40%	-0.58%
12	16	^^	Go	1.21%	+0.19%
13	12	v	Assembly language	0.94%	-0.45%
14	19	^^	Perl	0.87%	-0.04%
15	14	v	MATLAB	0.84%	-0.24%
16	11	vv	Ruby	0.81%	-0.83%
17	30	^^	Scratch	0.72%	+0.35%
18	33	^^	Rust	0.70%	+0.36%
19	23	^^	PL/SQL	0.68%	-0.01%
20	17	v	Classic Visual Basic	0.66%	-0.35%

What will we learn?

Overview

This course is divided into three sections that contain two or three chapters each. We do recommend to take these lessons in their listed order if you have no preceding experience with R programming. All chapters will include code examples that can be copied to, and executed in, the RStudio application installed on your computers.

The first section will look at the very basics of the programming language R and the programming environment RStudio. In the second section we will learn about the aspect of data manipulation. We will show how to import data, how to subset the data and how to do some basic analyses on the data. Eventually, we will learn how to visualize data using either the basic functions provided by the language or advanced techniques using packages, a kind of extension permitting more advanced programming functions.

Let's look at each chapter in detail now!

Section 1: R and RStudio
- 1.1 Introduction to RStudio
- 1.2 Introduction to the R language

Section 2: Data manipulation
- 2.1 Importing data
- 2.2 Selecting data
- 2.3 Data manipulation with dplyr

Section 3: Data visualization
- 3.1 Basic visualization functions
- 3.2 Data visualization with ggplot2

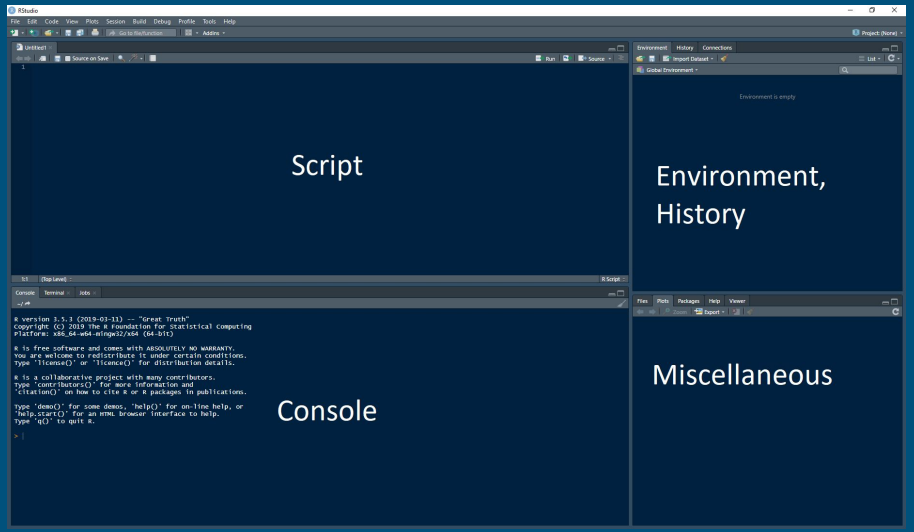
What will we learn?

1.1 Introduction to RStudio

The first chapter will introduce the Integrated Development Environment (IDE) RStudio, which we will use throughout this course to execute scripts written in the R programming language. There are other programs that can execute R code, but we explicitly recommend RStudio for beginners. RStudio is free and can be installed without buying a licence.

We will show all relevant key features of the program and discuss what purpose they serve. The learning material always shows the Windows version of RStudio, however, the software is also available to Mac OS X or Linux. The code examples will not be affected by the operating system.

This chapter also contains links to official installation guides.



What will we learn?

1.2 Introduction to the R language

The second chapter of the first section will introduce the key concepts of the R programming language.

We will learn how the language is structured and how its "grammar" works. A big part of this chapter will be the introduction of data types and structures. As R is mainly used for statistical analysis, a good working knowledge of these is required to proceed with the further chapters. The chapter also contains lessons about iteration loops and conditional programming, two major principles of every programming language.

For each section, exemplary code is provided. All examples are designed so they can be copied into RStudio to be executed. We encourage you to play around with the examples, change a few parameters, and see how the results are affected.

```
numbers <- c(1,3,4,3,4,5,6,1) # creating a vector of numbers

numbers  == 1 # checking each entry if it is a 1 or not
# output:  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE

any(numbers  == 1) # does the vector contain the ANY number 1?
# output:  TRUE (if a clear short statement is wanted)

sum(numbers == 1) # how many 1's does the vector contain?
# output: 2 (as each TRUE is counted as 1 and each FALSE as 0).

mean(numbers == 1) # What is the probability of 1's?
# output: 0.25 (2*TRUE, 6*FALSE;  2 of 8 equals 25%).
```

What will we learn?

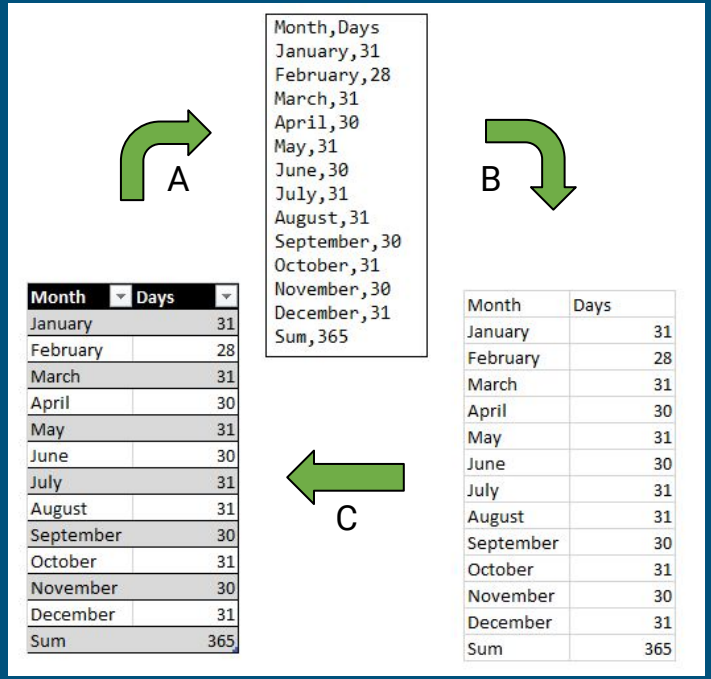
2.1 Importing data

The first chapter of the “Data manipulation” section teaches how to import data from external files.

We will focus on Excel and csv files (comma separated values), as these are the most commonly used data formats. Both formats have advantages and disadvantages which will be discussed in detail.

The chapter also shows how to locate files on your computer and how to set paths to the files that you want to import.

A few data files are provided by this course.



What will we learn?

2.2 Selecting data

The second chapter of the “Data manipulation” section teaches how to select specific data from a data container.

We will go through all data structures that were discussed in chapter “1.2 Introduction to the R language” and show how to select parts of the data. Different selection methods are shown, e.g. selections using data indices or selections using conditional statements.

```
# selecting a single data point
mat[1,2] # output: "d" (row 1, column 2)

# selecting an entire row (by not specifying a column) as vector
mat[1,] # output: "a" "d" (row 1)

# selecting an entire column (by not specifying a row) as vector
mat[,2] # output: "d" "e" "f" (column 2)

# selecting with series
Mat[2:3,1:2] # rows 2 to 3 and columns 1 to 2
# output:
     [,1] [,2]
[1,] "b"  "e"
[2,] "c"  "f"

# using exclusion
Mat[-1,2] # output: "e" "f" (all rows but index 1 and column 2)
```

What will we learn?

2.3 Data manipulation

The third chapter of the “Data manipulation” section teaches how to manipulate data. Under “data manipulation” we understand any action that subsets and filters data, adds more data to existing data structures or summarises data.

In this chapter we will make use of the package “dplyr” which can be downloaded from the internet using RStudio. The package provides advanced functions that make the process of data manipulation much easier.

With the skills taught there, you should be ready to do simple summaries and analyses of data.

```
# summary of cars
mtcars %>%
   summarise(hpMean = mean(hp),
             hpMedian = median(hp),
             hpSD = sd(hp))
    hpMean    hpMedian    hpSD
1 146.6875    123         68.56287

# summary of cars, grouped by their number of cylinders
mtcars %>%
   group_by(cyl) %>%
   summarise(count = n(),
             hpMean = mean(hp),
             hpMedian = median(hp),
             hpSD = sd(hp))

   cyl  count  hpMean hpMedian  hpSD
1   4    11      82.6     91      20.9
2   6     7     122.     110      24.3
3   8    14     209.     192.     51.0
```

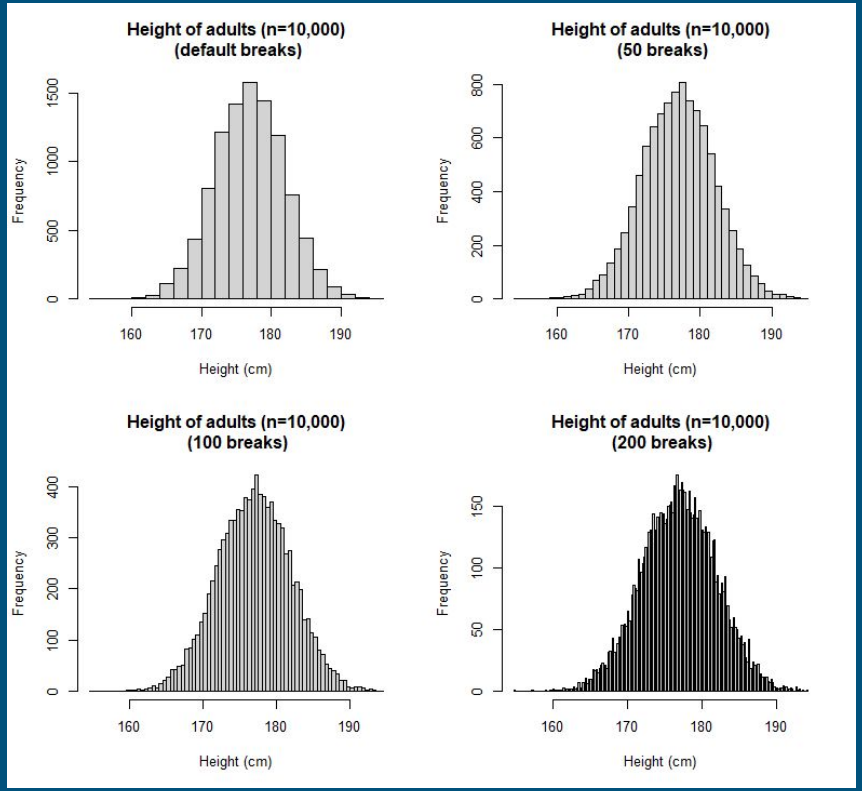
What will we learn?

3.1 Basic visualization functions

The first chapter of the “Data visualization” section introduces the basic functions used to visualize data.

We will learn how to create simple graphs from data, e.g. x/y plots, histograms, bar plots or pie charts. The graphs can also be used to visualize mathematical functions.

Additionally, we will show how to combine multiple graphs to a single visualization. The chapter also shows ways to export the graphs, created in RStudio, as files to your computer.



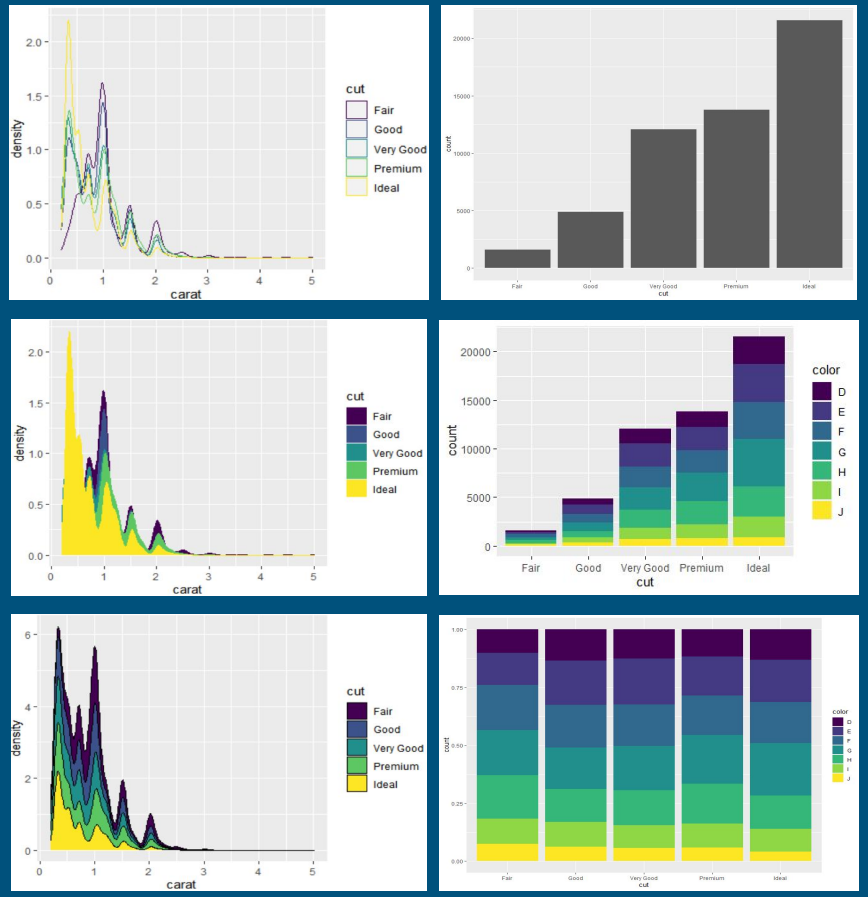
What will we learn?

3.2 Basic visualization functions

The second chapter of the “Data visualization” section shows more advanced techniques using the “ggplot2” package.

The package “ggplot2” is designed to give graphs a more consistent and aesthetically pleasing look. We will show many different visualization types (e.g. scatterplots, histograms, stacked bar plots, box plots, …) and under which circumstances they should be used.

The chapter also teaches the use of legends, the adjustment of coordinate systems, the application of design themes, and more concepts to increase the readability of a graph.



What will we learn?

Let’s begin!

Now it is time to get started with the lesson!

At the end of the course, you might not be an expert in R programming. This requires many years of practice and dedication. Our goal, however, is to provide a solid understanding of the programming language R and the program RStudio. The course is also a short introduction to data science.

You will be able to read your own data files, manipulate and summarize their data entries and create impressive visualizations.

We hope you will enjoy our course and obtain new skills that might help you advance in your careers!