INSTITUTE
OF LAW,
POLITICS AND
DEVELOPMENT

Sant'Anna
School of Advanced Studies – Pisa

Data Analysis for the Social Sciences with R

Introduction

Prof. Kevin Koehler
kevin.koehler@santannapisa.it

The class

Aims:

1. Introduce you to R as a programming language and statistical package (using the RStudio user interface)

The class

Aims:

1. Introduce you to R as a programming language and statistical package (using the RStudio user interface)
2. Implement basic data analysis tasks in R, including data management, visualization, and regression analysis

The class

Aims:

1. Introduce you to R as a programming language and statistical package (using the RStudio user interface)
2. Implement basic data analysis tasks in R, including data management, visualization, and regression analysis
3. Reproduce existing Political Science research

Schedule

1. Getting set up
2. Data management
3. Visualization
4. R programming
5. Exploring data
6. Linear Regression
7. Replication I
8. Logistic regression
9. Replication II
10. Conclusion

Today's class

1. RStudio

Sant'Anna
School of Advanced Studies – Pisa

Today's class

1. RStudio
2. Basic notions in R: Objects, functions, packages

Sant'Anna
School of Advanced Studies – Pisa

Today's class

1. RStudio
2. Basic notions in R: Objects, functions, packages
3. Some statistical notions

Sant'Anna
School of Advanced Studies – Pisa

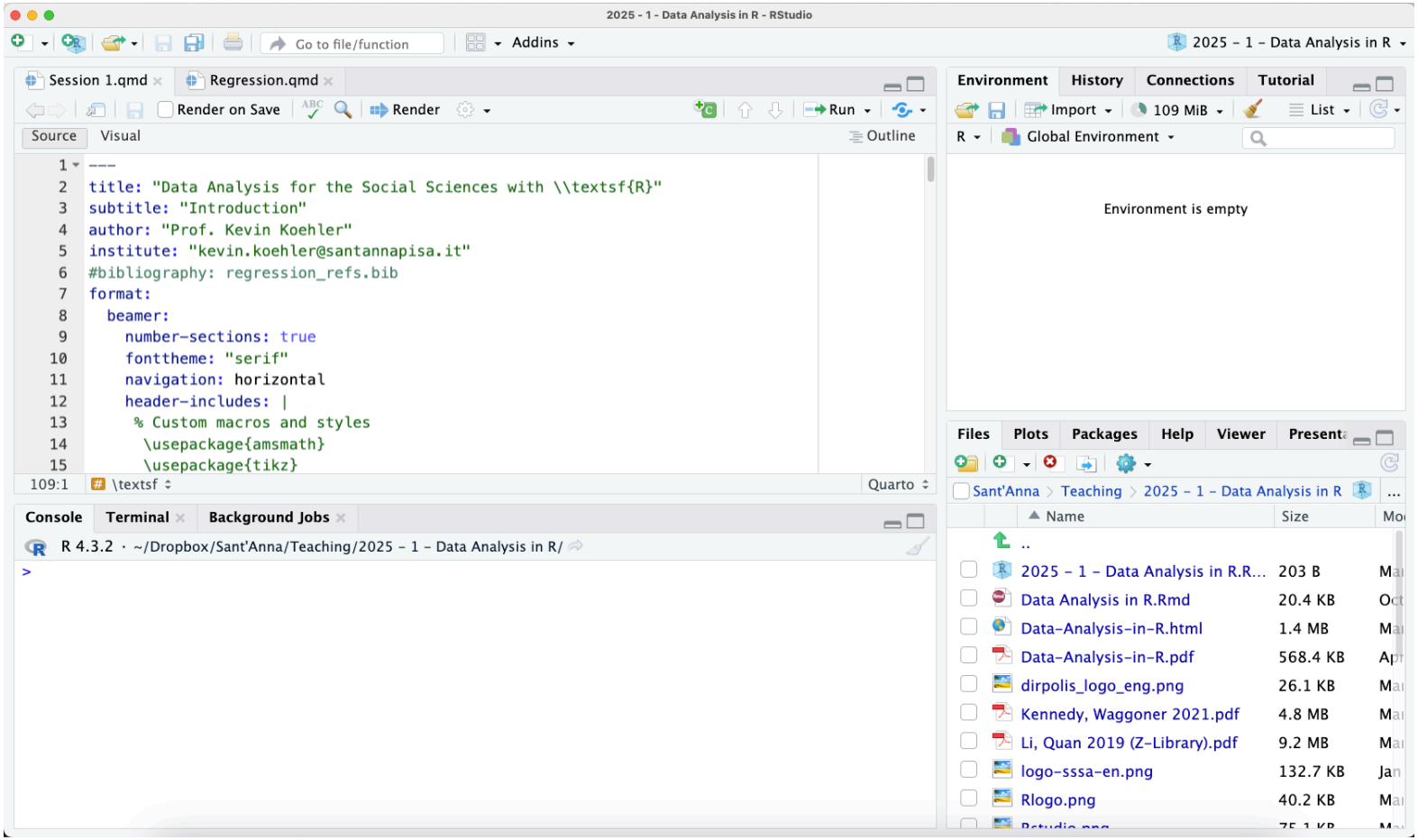
Today's class

1. RStudio
2. Basic notions in R: Objects, functions, packages
3. Some statistical notions
4. Exercises

RStudio

Sant'Anna
School of Advanced Studies – Pisa

RStudio


2025 - 1 - Data Analysis in R - RStudio
Go to file/function
Addins
2025 - 1 - Data Analysis in R
Session 1.qmd
Regression.qmd
Render on Save
Render
Run
Source
Visual
Outline
1 ---
2 title: "Data Analysis for the Social Sciences with \\textsf{R}"
3 subtitle: "Introduction"
4 author: "Prof. Kevin Koehler"
5 institute: "kevin.koehler@santannapisa.it"
6 #bibliography: regression_refs.bib
7 format:
8 beamer:
9 number-sections: true
10 fonttheme: "serif"
11 navigation: horizontal
12 header-includes: |
13 % Custom macros and styles
14 \usepackage{amsmath}
15 \usepackage{tikz}
109:1
\textsf
Quarto
Console
Terminal
Background Jobs
R 4.3.2 · ~/Dropbox/Sant'Anna/Teaching/2025 - 1 - Data Analysis in R/
>
Environment
History
Connections
Tutorial
Import
109 MiB
List
R
Global Environment
Environment is empty
Files
Plots
Packages
Help
Viewer
Sant'Anna
Teaching
2025 - 1 - Data Analysis in R
Name
Size
..
2025 - 1 - Data Analysis in R.R...
203 B
Data Analysis in R.Rmd
20.4 KB
Data-Analysis-in-R.html
1.4 MB
Data-Analysis-in-R.pdf
568.4 KB
dirpolis_logo_eng.png
26.1 KB
Kennedy, Waggoner 2021.pdf
4.8 MB
Li, Quan 2019 (Z-Library).pdf
9.2 MB
logo-sssa-en.png
132.7 KB
Rlogo.png
40.2 KB

RStudio

First, create a **project**. Projects are useful for keeping all files in the same place.

1. Create a directory on your computer where you want to save all files related to this class
2. Go to `> File > New Project` in the R menu and then select "Existing directory"
3. Navigate to the folder and name and create the project

RStudio

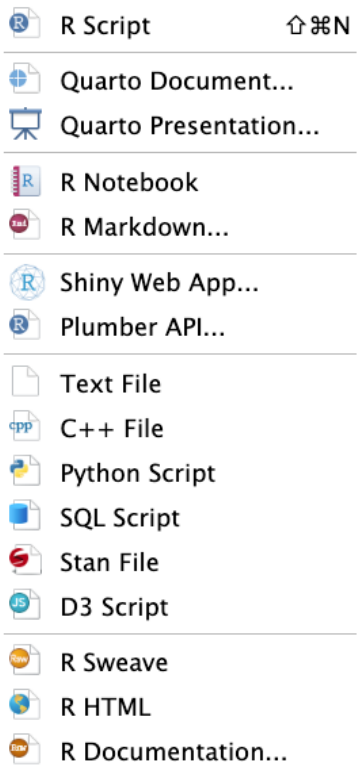
The folder you created is now also your **working directory**. You can see the directory at the top of the console. You can also type:

```
getwd()
```

Basic notions in R

Sant'Anna
School of Advanced Studies – Pisa

Types of R files



R can do many different things, not just statistical analysis.

Consequently, there are many different file types in RStudio:

- R scripts for coding
- Quarto documents and presentations
- R Notebook and R Markdown
- Shiny Apps
- Plumber API
- Files in other languages (C++, Python, SQL...)

R scripts

- Go ahead and open a new R script.
- Type `print("Hello world")`
- With your cursor in the line with the command, press `Ctrl + Enter` (Windows) or `Command + Enter` (Mac)
- The code is executed and the results printed in the Console

objects

Sant'Anna
School of Advanced Studies – Pisa

Vectors

Vectors are the most basic data structure. They hold a series of numeric or character values and are created with the `c()` function (the c in the function stands for concatenate):

```
c(1,2,3,4,5)
```

```
[1] 1 2 3 4 5
```

```
c("a","b","c","d","e")
```

```
[1] "a" "b" "c" "d" "e"
```

Matrices and data frames

```r
matrix(1:6, nrow = 2, ncol = 3)
```

```
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
```

```r
data.frame(name = c("A","B"),
           age = c(24,56))
```

```
  name age
1    A  24
2    B  56
```

The environment

The **Environment** is where R stores objects for the duration of a session. You can assign a vector to the environment by typing:

```
data <- data.frame(name = c("A","B"),
                   age = c(24,56))
```

(you could replace the `<-` with `=`, but I recommend getting used to `<-`)

After running this code, you should have an `object` called "data" in your environment. This object contains two columns (name and age) with two rows each. We call the columns **variables** and the rows **observations**.

You can click on the object to see what it contains.

Working with data frames

```
data
```

```
  name age
1    A  24
2    B  56
```

```
data$name
```

```
[1] "A" "B"
```

```
data$age[1]
```

```
[1] 24
```

```
data$age[data$name=="A"]
```

```
[1] 24
```

Functions

R works with `functions`. A function takes an `object` (or multiple `objects`) as input and does something with it.

Examples include:

- `print("Hello world")` prints "Hello world" to the console
- `c(1,2,3,4)` creates a numerical vector with 1,2,3,4 as elements
- `getwd()` returns the active working directory
- `help(print)` returns the help file for the `print()` function
- `lm(x~y)` performs a linear regression of x on y

Functions in R are a words followed by brackets. You always need to close the brackets, otherwise your code will not run.

User-defined functions

You can write your own functions in R. Here is a function which takes a number as an argument and tells you whether the number is greater than 5:

```
greater5 <- function(x) {
  if (!is.numeric(x)) {
    stop(paste0("Argument must be numeric.\n",
                "You provided an object of class: ",
                class(x)[1],
                ". You moron."))
  } # check if input is numeric, return error if not
  result <- ifelse(x > 5,
                   paste(x, "is greater than 5"),
                   paste(x, "is not greater than 5"))
  return(result)  # Return results
}
```

Packages and CRAN

Functions are part of packages. Your version of R comes with base R, but there are many other packages.

We will use the `tidyverse` family of packages. You can install packages by typing `install.packages("tidyverse")` in the Console. This will download the package and save it on your machine. You need to do this only once.

To use specific packages, you need to load them in the beginning of your R session. It is good practice to include all packages needed to run your code in the beginning of your R script. Packages are loaded typing `library(tidyverse)`.

Some statistical notions

Sant'Anna
School of Advanced Studies – Pisa

Recap on variable types

1. **Nominal Scale**:
   - Categories without a specific order (e.g., gender, color).
2. **Ordinal Scale**:
   - Categories with a defined order but unequal intervals (e.g., rankings, satisfaction ratings).
3. **Interval Scale**:
   - Numeric scales with equal intervals but no true zero (e.g., temperature in Celsius).
4. **Ratio Scale**:
   - Numeric scales with equal intervals and a true zero (e.g., height, weight, age).

How can we describe the typical value for each of these scales?

Measures of Central Tendency

- **Mean**: The average of a data set, calculated by summing all values and dividing by the number of values.

Measures of Central Tendency

- **Mean**: The average of a data set, calculated by summing all values and dividing by the number of values.
  - Mean $= \frac{\sum_{i=1}^{n} x_i}{n}$, where x_i represents each data point, and n is the number of data points.

Measures of Central Tendency

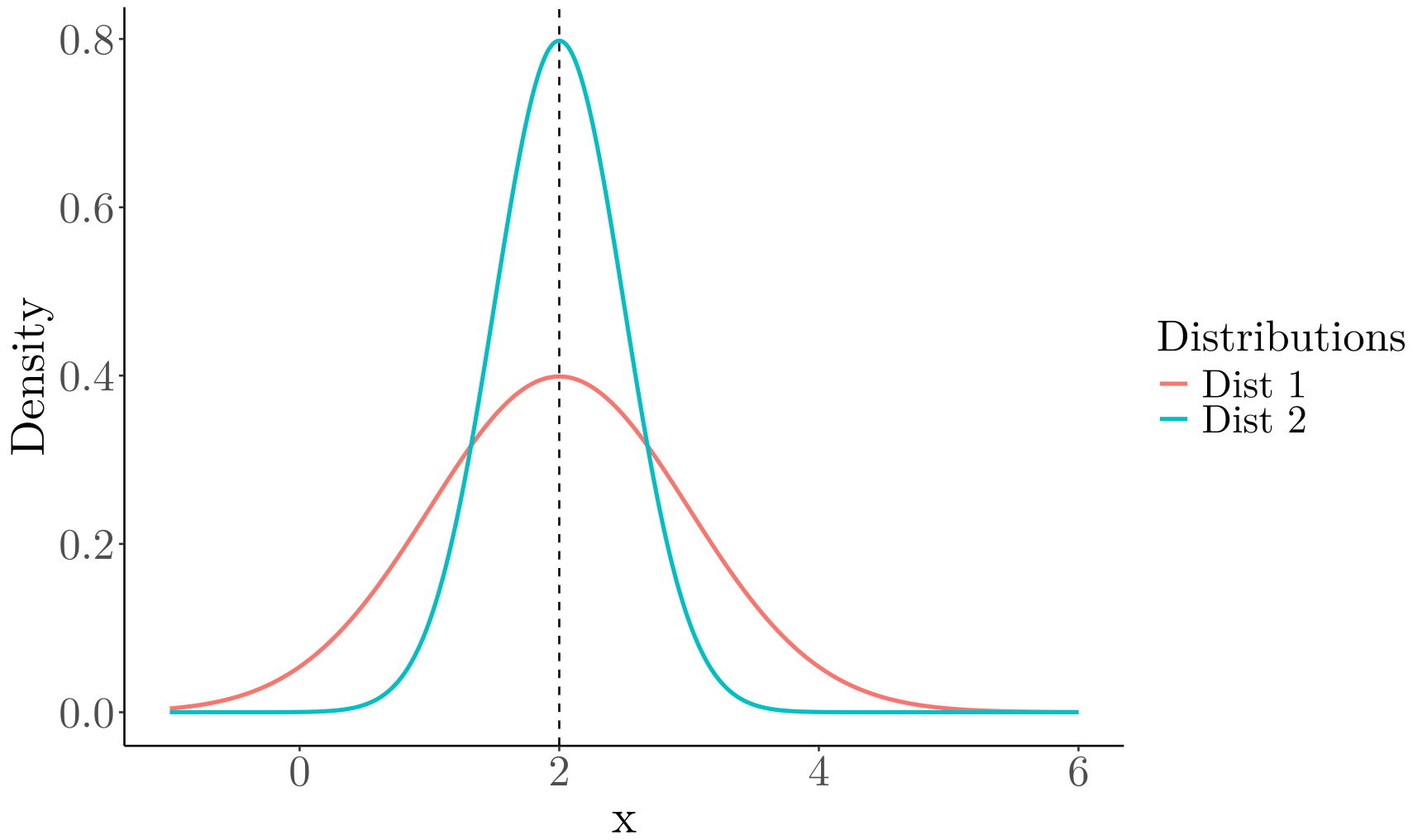
- **Mean**: The average of a data set, calculated by summing all values and dividing by the number of values.
  - Mean $= \frac{\sum_{i=1}^{n} x_i}{n}$, where x_i represents each data point, and n is the number of data points.
- **Median**: The middle value when data is ordered from lowest to highest; 50% of values fall below it.

Measures of Central Tendency

- **Mean**: The average of a data set, calculated by summing all values and dividing by the number of values.
  - Mean $= \frac{\sum_{i=1}^{n} x_i}{n}$, where x_i represents each data point, and n is the number of data points.
- **Median**: The middle value when data is ordered from lowest to highest; 50% of values fall below it.
- **Mode**: The value that occurs most frequently in a data set.

How can we describe variation around a central value?

How can we describe variation around a central value?



Measures of Dispersion

- **Range**: $\text{Range} = \text{Max}(x) - \text{Min}(x)$. The difference between the maximum and minimum values in the data set.

Measures of Dispersion

- **Range**: Range $= \text{Max}(x) - \text{Min}(x)$. The difference between the maximum and minimum values in the data set.
- **Variance**: Variance $= \frac{\sum_{i=1}^{n}(x_i-\mu)^2}{n}$, where x_i represents each data point, μ is the mean, and n is the number of data points.

Measures of Dispersion

- **Range**: Range $= \text{Max}(x) - \text{Min}(x)$. The difference between the maximum and minimum values in the data set.
- **Variance**: Variance $= \frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n}$, where x_i represents each data point, μ is the mean, and n is the number of data points.
- **Standard Deviation**: SD $= \sqrt{\text{Variance}}$. The square root of the variance, which gives the spread of data in the same units as the original data.

Measures of Dispersion

- **Range**: Range $= \text{Max}(x) - \text{Min}(x)$. The difference between the maximum and minimum values in the data set.
- **Variance**: Variance $= \frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n}$, where x_i represents each data point, μ is the mean, and n is the number of data points.
- **Standard Deviation**: SD $= \sqrt{\text{Variance}}$. The square root of the variance, which gives the spread of data in the same units as the original data.
- **Interquartile Range (IQR)**: IQR $= Q3 - Q1$, where $Q3$ is the third quartile and $Q1$ is the first quartile, representing the middle 50% of the data.

Exercises

Exercises

1. Create a data set with crime statistics for all U.S. states in your Environment. Tip: Use the `data()` function to see all data sets included in R.
2. Write code to calculate:
   - 2.1 The typical value of murders per 100,000 inhabitants across all states
   - 2.2 The value such that 50% of states have lower murder rates
3. Describe how much states differ from each other in terms of murder rates. Which measures could you use? Why?
   - 3.1 Write code to calculate these measures
4. Write a function which calculates all of these measures at the same time