Get your dataset ready!

Using R and GIS

Rosa Félix Gabriel Valença Rafael Pereira

2024-09-04

Table of contents

1 Introduction

This course aims to provide tools to deal with exploring and treating transportation datasets using R programming, an open-source and widely used tool for data analytics in urban mobility.

Additionally, this course provides guidance towards the use of reproducible methods to deal with large datasets that require manipulation and/or spatial analysis.

The course has a **hands-on** approach, where participants will learn the basics of **coding**, **data manipulation**, and **spatial analysis** for urban mobility and transportation.

1.1 Mobility data

There is an emerging increase in mobility data, through new forms of technology, which result in very large and diverse datasets.

Knowing how to get, treat and analyze complex datasets with the up-to-date technologies is extremely relevant for academia, policy makers and start-ups, since it allows them to:

1. acquire critical view on urban mobility based on data;
2. spatially identify locations in the city that require policy priorities;
3. and improve the efficiency of data analysis processes.

Why R and GIS

Most academic programs focus on teaching modelling and deep analysis of data. However, there is a need to learn how to explore and prepare a dataset for modelling. The use of **programming and GIS** techniques have enormous advantages, including their flexibility; reproducibility; and transparency and understanding the step-by-step process.

The use of GIS techniques in transportation is, traditionally, not considered in transportation learning programs, despite being of enormous relevance when doing accessibility analysis or reeling with georreferenced transportation data, such as bike sharing route trips' datasets, origin-destination flows datasets, home/work locations, GTFS public transit data, and so on. There is a need to learn how to locate these open datasets, how to explore them and

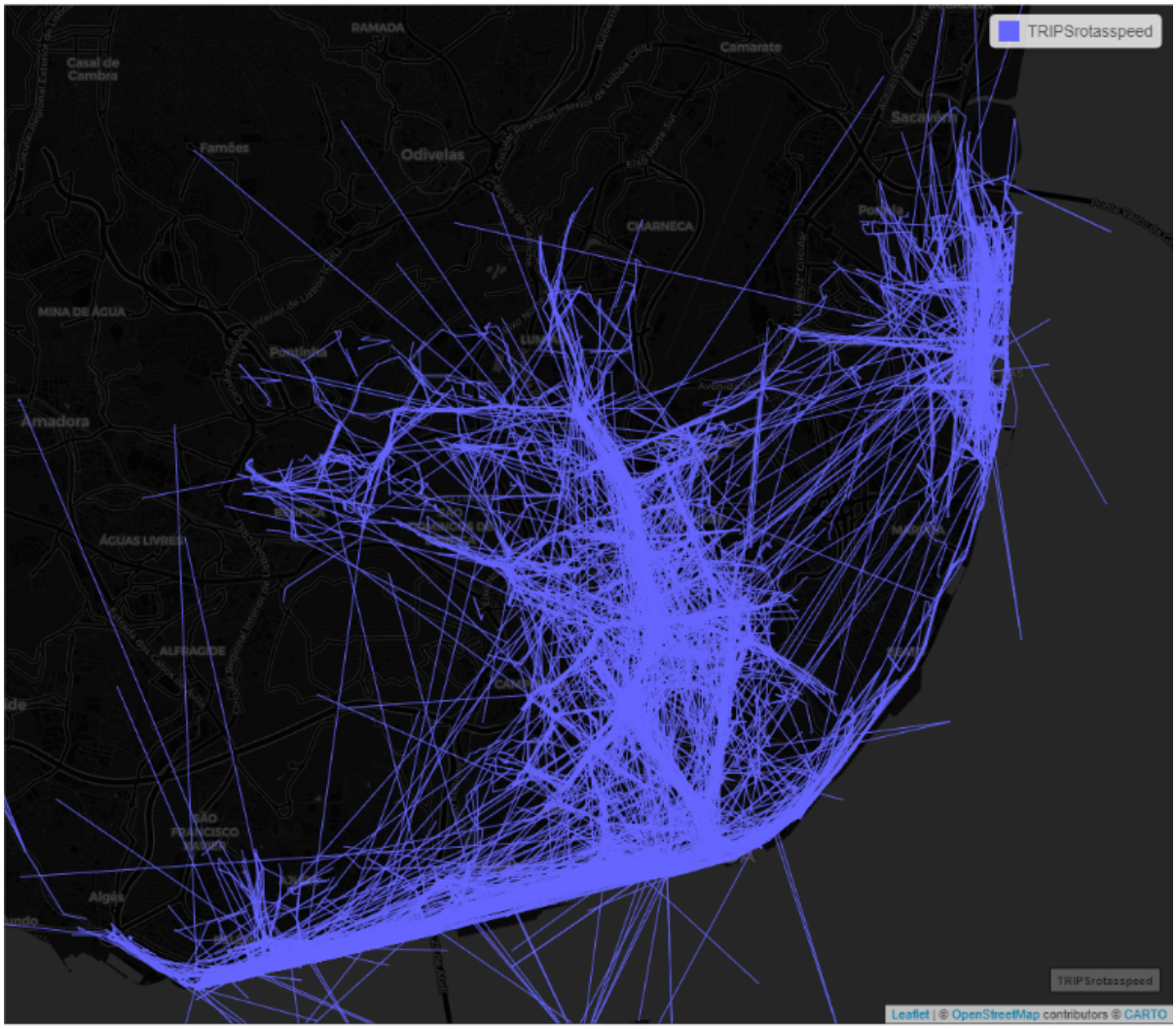


Figure 1.1: E-Scooter trip data in Lisbon. How to deal with it?

how to integrate them into transportation and urban analysis. Additionally, the use of open source software and datasets allows researchers to perform methods that are reproducible and transparent.

TLDR

- Open-source tools widely used in data analytics and spatial analysis
- Flexibility and reproducibility in data manipulation and visualization
- Critical for urban mobility and transportation research, with spatial relevance
- Large transportation datasets are becoming increasingly common

1.2 Course objectives

Introduce R Programming Basics

- Equip participants with foundational skills in R programming
- Emphasize reproducible research practices to ensure transparency and replicability in analyses

Teach Data Manipulation Techniques

- Use key R packages for data cleaning, manipulation, and summarization of datasets
- Enable participants to efficiently handle large and complex transportation datasets

Spatial Data Visualization

- Introduce methods for quick and effective spatial data visualization using R and GIS tools
- Provide hands-on experience with creating interactive maps and visualizations

Perform Basic Spatial Analysis

- Teach participants how to perform spatial analysis of transportation datasets using GIS techniques with R
- Cover practical applications such as georeferencing data, accessibility analysis, and routing ODs
- Utilize real-world transportation data for practical, hands-on learning

1.3 Target audience

- Ph.D. candidates from DTN and other researchers
- Policy makers and practitioners in urban mobility
- Beginners to intermediate R users, no prior experience needed

1.4 Recommended readings

- Engel, Claudia A. (2023) Introduction to R
- Lovelace, Robin, Nowosad, Jakub & Muenchow, Johannes. (2023) Geocomputation with R
- Pereira, Rafael H. M. & Herszenhut, Daniel. (2023) Introduction to urban accessibility: a practical guide with R. Ipea - Institute of Applied Economic Research

2 Course Structure

The course consists of an in-person 2-day course, taking place during the EIT DTN Annual Meeting on the **19th and 20th September 2024**.

The first day will focus on learning the basics of R programming and how to treat and explore datasets. The second day will focus on analyzing spatial datasets, and routing origins to destinations.

2.1 Day 1

Morning

- Introduction to **programming** techniques and **data structures**
- Introduction to R, and RStudio: **software installation** and main packages
- **R base and basics**: examples and exercises

Afternoon

- **Data manipulation**: using the dplyr package to select, filter, left-join, group and summarize
- Introduction to **GIS** and **spatial data**: import and visualize vector data
- R markdown and **interactive maps**

2.2 Day 2

Morning

- **Desire lines** from OD and transport zones
- **Georeference** coordinates: examples from surveys
- **Accessibility analysis**: from buffers to road networks

Afternoon

- **Open Transportation data**: where to find it
- **Routing with R**: multimodal and intermodal (*r5r demo* - Rafael Pereira)
- Group **exercise**

3 Detailed schedule (TBC)

Day 1	
9.30	Introductions and Presentation of the course contents
10.00	Introduction to programming techniques and data structures
10.30	Introduction to R and RStudio: hands-on to install software and main packages
11.00	*Coffee break*
11.15	(cont.)
11.30	R basics: examples and exercises
12.30	*Lunch break*
13.30	Data manipulation: examples and exercises (select, filter, left-join, subset, group and summarize, using dplyr package)
15.00	Introduction to GIS and spatial data: import and visualize vector data
15.30	*Coffee break*
15.45	(cont.)
16.15	View and export interactive maps
17.00	*End of day 1*

Day 2	
9.30	Desire-lines from OD pairs and transport zones: examples and exercises
10.30	Georeferenced coordinates from survey responses: example and exercises
11.00	*Coffee break*
11.15	(cont.)
11.30	Euclidean distance and buffers: example and exercises
12.30	*Lunch break*
13.30	Open Transportation data: where to find it (OSM and GTFS)

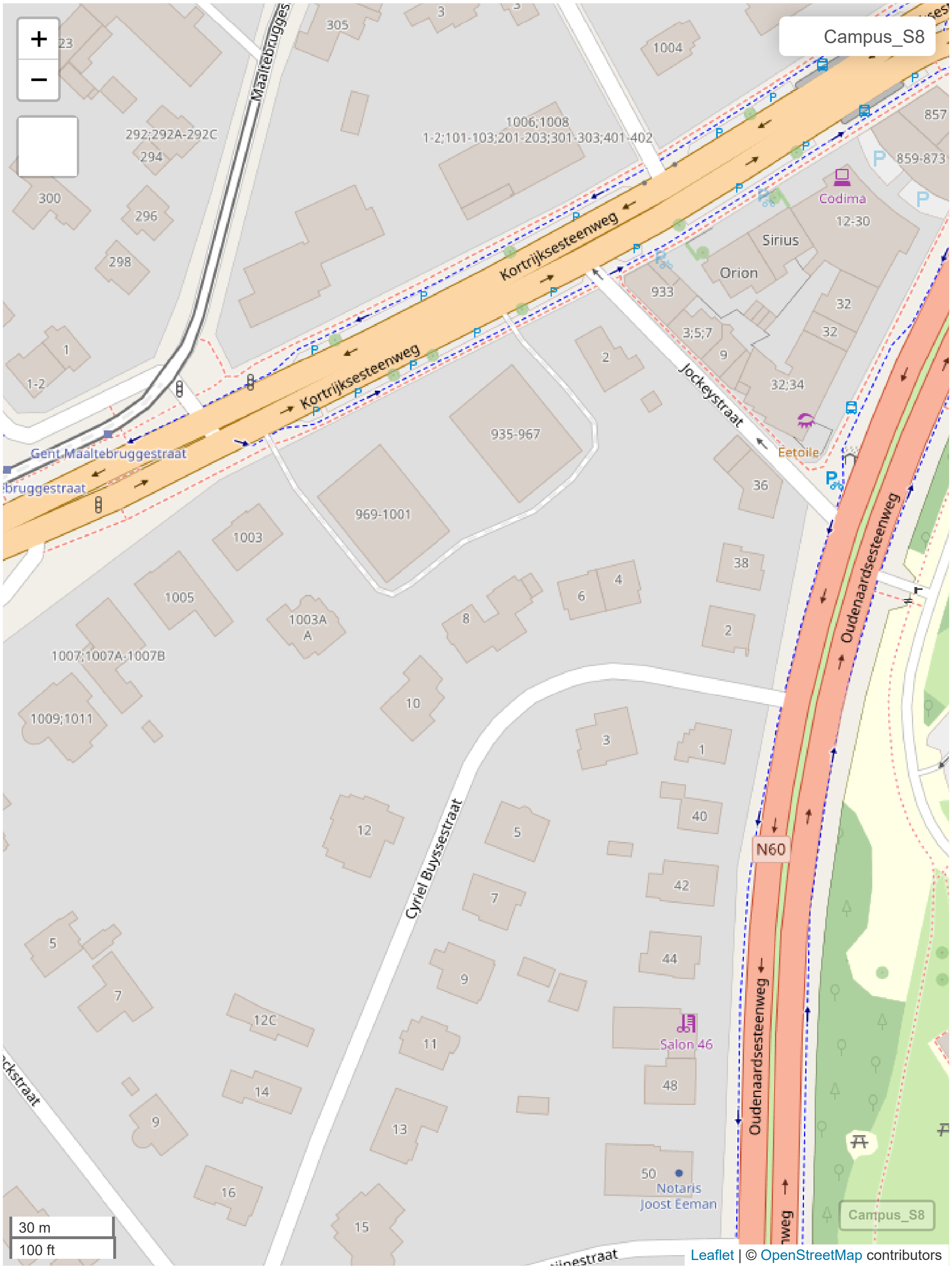
Day 2	
14.30	Uni-modal and Inter-modal Routing with r5r
15.30	Accessibility analysis with r5r
16.00	*Coffee break*
16.15	Using you data: manipulation and spatial analysis methods and further applications
16.45	Survey and feedback from participants
17.00	*End of day 2*

4 Location

The course will take place at Campus Sterre, Building S8, room 2.4.

```
Campus_S8_coord = c(3.7105372, 51.0241258)
Campus_S8 = sf::st_sfc(sf::st_point(Campus_S8_coord)) # create point
Campus_S8 = sf::st_as_sf(Campus_S8, crs = 4326) # assign crs

mapview::mapview(Campus_S8, map.types = "OpenStreetMap") # quick map view
```



Campus_S8
+
−
Kortrijksesteenweg
Kortrijksesteenweg
Maaltebruggestraat
Gent Maaltebruggestraat
Jockeystraat
Oudenaardsesteenweg
Oudenaardsesteenweg
N60
Cyriel Buyssestraat
Codima
Sirius
Orion
Eetoile
Salon 46
Notaris Joost Eeman
1006;1008
1-2;101-103;201-203;301-303;401-402
935-967
969-1001
1003
1003A
1005
1007;1007A-1007B
1009;1011
292;292A-292C
294
296
298
300
305
1004
857
859-873
12-30
933
3;5;7
32;34
12C
30 m
100 ft
Leaflet | © OpenStreetMap contributors
Campus_S8

5 Resources

- You laptop, with any OS
- Github repository with all the materials (data, code and guidelines)
- Survey datasets, school locations and public transport operator datasets

Part I

Day 1

6 Software

In this chapter we will guide you through the installation of R, RStudio and the packages you will need for this course.

R and **RStudio**[1] are separate downloads.

6.1 R

You will need **R** installed on your computer. **R stats** (how it is also known) is a programming language and free software environment for statistical computing and graphics supported by the R Foundation for Statistical Computing.

The download links live at The Comprehensive R Archive Network (aka CRAN). The most recent version is `4.4.1`, but you can use `>= 4.1.x` if you already have it installed.

6.1.1 Windows

Download R-4.4.1 for Windows and run the executable file.

You will also need to install Rtools, which is a collection of tools necessary to build R packages in Windows.

6.1.2 Mac

Download R-4.4.1 for MacOX. You will have to choose between the arm64 or the x86-64 version.

Download the `.pkg` file and install it as usual.

[1] We will use RStudio, although if you already use other studio such as VScode, that's also fine.

6.1.3 Ubuntu

> These are instructions for Ubuntu. If you use other linux distribution, please follow the instructions on The Comprehensive R Archive Network - CRAN.

You can look for R in the Ubuntu **Software Center** or install it via the terminal:

```
# sudo apt update && sudo apt upgrade -y
sudo apt install r-base
```

Or, if you prefer, you can install the latest version of R from CRAN:

```
# update indices
sudo apt update -qq
# install two helper packages we need
sudo apt install --no-install-recommends software-properties-common dirmngr
# add the signing key (by Michael Rutter) for these repos
wget -qO- https://cloud.r-project.org/bin/linux/ubuntu/marutter_pubkey.asc | sudo tee -a /
# add the R 4.0 repo from CRAN -- adjust 'focal' to 'groovy' or 'bionic' as needed
sudo add-apt-repository "deb https://cloud.r-project.org/bin/linux/ubuntu $(lsb_release -c
```

Then run:

```
sudo apt install r-base r-base-core r-recommended r-base-dev
```

[Optional] To keep up-to-date r version and packages, you can follow the instructions at r2u

After this installation, you don't need to open R base. Please proceed to install RStudio.

6.2 RStudio

RStudio Desktop is an integrated development environment (IDE) for R. It includes a console, syntax-highlighting editor that supports direct code execution, as well as tools for plotting, history, debugging and workspace management.

RStudio is available for free download from Posit RStudio.

6.2.1 Windows 10/11

Download RStudio 2024.04 and run the executable file.

6.2.2 MacOS

Download RStudio 2024.04 and install it as usual.

6.2.3 Ubuntu

> These are instructions for Ubuntu **22** / Debian 12. If you use other linux distribution, please follow the instructions on Posit RStudio.

Install it via the terminal:

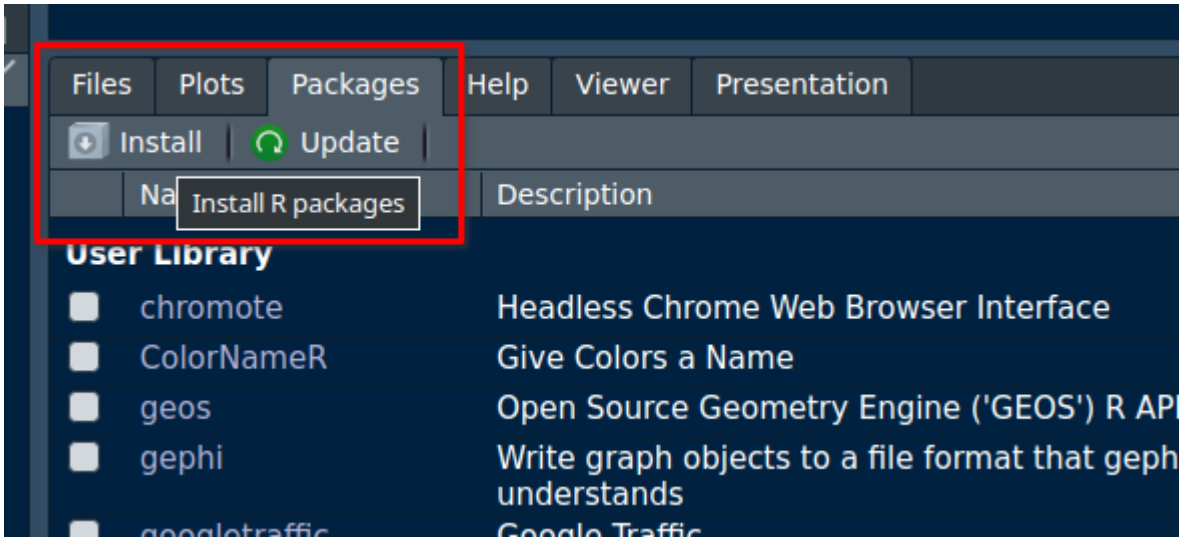
```
sudo apt install libssl-dev libclang-dev
wget https://download1.rstudio.org/electron/jammy/amd64/rstudio-2024.04.2-764-amd64.deb
sudo dpkg -i rstudio*
rm -v rstudio*
```

If you already use Ubuntu **24**, please check and replace the correct url from RStudio Dailies

6.3 R packages

You will need to install some packages to work with the data and scripts in this course.

You can install them in RStudio by searching for them in the **Packages** tab:



or by running the following code in the console:

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

install.packages(c("remotes", "devtools", "usethis"))
install.packages("sf")
install.packages("mapview")
```

6.4 Other software

6.4.1 Java Development Kit 21 and r5r

You will need this to work with the **r5r** package. It is also known as JDK 21.

6.4.2 Windows and MacOS

Go to Java Development Kit 21, download the latest 21 build corresponding to your operating system and run the executable file.

6.4.3 Ubuntu

Install it via the terminal:

```
sudo apt install -y openjdk-21-jdk openjdk-21-jre
java -version
```

Then, in R you will also need rJava package.

```r
install.packages("rJava")
```

Finally, install the **r5r** package:

```r
install.packages("r5r")

# development version:
# devtools::install_github("ipeaGIT/r5r", subdir = "r-package")
```

6.4.4 Open Route Service

Sign up for an account and create a token. Copy your API.

In RStudio console, run:

```
# install.packages("openrouteservice")
openrouteservice::ors_api_key("YOUR-API-KEY")
```

This will store your key on your `.Renviron` file, meaning that every time you open RStudio, you won't need to run this command again.

This is useful also to write your `openrouteservice` scripts without sharing your key with others.

7 R basics

In this chapter we will introduce to the R basics and some exercises to get familiar to how R works.

7.1 Simple operations

7.1.1 Math operations

7.1.1.1 Sum

```
1+1
```

```
[1] 2
```

7.1.1.2 Subtraction

```
5-2
```

```
[1] 3
```

7.1.1.3 Multiplication

```
2*2
```

```
[1] 4
```

7.1.1.4 Division

```
8/2
```

```
[1] 4
```

7.1.1.5 Round the number

```
round(3.14)
```

```
[1] 3
```

```
round(3.14, 1) # The "1" indicates to round it up to 1 decimal digit.
```

```
[1] 3.1
```

```
# You can use help ?round in the console to see the description of the function.
```

7.1.2 Basic shortpaths

7.1.2.1 Perform Combinations

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

```
[1] 1 2 3
```

```
c(1:3) # The ":" indicates a range between the first and second numbers.
```

```
[1] 1 2 3
```

7.1.2.2 Create a comment with ctrl + shift + r

```
# Comments help you organize your code. The software will not run the comment.
```

7.1.2.3 Create a table

A simple table with the number of trips by car, PT, walking, and cycling in a hypothetical street segment at a certain period.

Define the variables

```
modes <- c("car", "PT", "walking", "cycling") # you can use "=" or "<-"
Trips = c(200, 50, 300, 150) # uppercase letters modify
```

Join the variables to create a table

```
table_example = data.frame(modes, Trips)
```

Take a look at the table

Visualize the table by clicking on the "Data" in the "Environment" page or use the following function.

```
View(table_example)
```

7.2 Practical exercise

Dataset: the number of trips between all municipalities in the Metropolitan Area of Lisbon, Portugal (Instituto National de Estatística 2018).

7.2.0.1 Import dataset

```
data = readRDS("data/TRIPSmode_mun.Rds")
```

7.2.0.2 Take a first look at the data

Summary statistics

```
summary(data)
```

```
 Origin_mun        Destination_mun        Total             Walk
Length:315         Length:315         Min.   :     7   Min.   :     0
Class :character   Class :character   1st Qu.:   330   1st Qu.:     0
Mode  :character   Mode  :character   Median :  1090   Median :     0
                                      Mean   : 16825   Mean   :  4033
                                      3rd Qu.:  5374   3rd Qu.:     0
                                      Max.   :875144   Max.   :306289
     Bike              Car             PTransit            Other
Min.   :   0.00   Min.   :     0   Min.   :     0.0   Min.   :    0.0
1st Qu.:   0.00   1st Qu.:   263   1st Qu.:     5.0   1st Qu.:    0.0
Median :   0.00   Median :   913   Median :   134.0   Median :    0.0
Mean   :  80.19   Mean   :  9956   Mean   :  2602.6   Mean   :  152.4
3rd Qu.:   0.00   3rd Qu.:  4408   3rd Qu.:   975.5   3rd Qu.:   62.5
Max.   :5362.00   Max.   :349815   Max.   :202428.0   Max.   :11647.0
```

Check the structure of the data

```
str(data)
```

```
'data.frame':   315 obs. of  8 variables:
 $ Origin_mun     : chr  "Alcochete" "Alcochete" "Alcochete" "Alcochete" ...
 $ Destination_mun: chr  "Alcochete" "Almada" "Amadora" "Barreiro" ...
 $ Total          : num  20478 567 188 867 114 ...
 $ Walk           : num  6833 0 0 0 0 ...
 $ Bike           : num  320 0 0 0 0 0 0 0 91 0 ...
 $ Car            : num  12484 353 107 861 114 ...
 $ PTransit       : num  833 0 81 5 0 ...
 $ Other          : num  7 214 0 0 0 0 0 0 0 0 ...
```

Check the first 10 values of each variable

```
head(data, 10)
```

```
   Origin_mun Destination_mun Total Walk Bike   Car PTransit Other
1   Alcochete       Alcochete 20478 6833  320 12484      833     7
2   Alcochete          Almada   567    0    0   353        0   214
3   Alcochete         Amadora   188    0    0   107       81     0
4   Alcochete        Barreiro   867    0    0   861        5     0
5   Alcochete         Cascais   114    0    0   114        0     0
6   Alcochete          Lisboa  2840   69    0  1994      775     0
7   Alcochete          Loures   634    0    0   634        0     0
```

```
8   Alcochete           Moita   261    0    0   256        5     0
9   Alcochete         Montijo  8714  130   91  7062     1431     0
10  Alcochete        Odivelas   129    0    0   129        0     0
```

Check the number of rows (observations) and columns (variables)

```
nrow(data)
```

```
[1] 315
```

```
ncol(data)
```

```
[1] 8
```

Open the dataset

```
View(data)
```

7.2.0.3 Explore the data

Check the total number of trips

Use '$' to select a variable of the Data

```
sum(data$Total)
```

```
[1] 5299853
```

Percentage of car trips related to the total

```
sum(data$Car)/sum(data$Total) * 100
```

```
[1] 59.17638
```

Percentage of active modes related to the total

```
(sum(data$Walk)+ sum(data$Bike)) / sum(data$Total) * 100
```

```
[1] 24.44883
```

7.2.0.4 Modify original data

Create a column with the sum of the number of trips for active modes

```
data$Active = data$Walk + data$Bike
```

Filter by condition (create new tables)

Filter trips only with origin from Lisbon

```
data_Lisbon = data[data$Origin_mun == "Lisboa",]
```

Filter trips with origin **different** from Lisbon

```
data_out_Lisbon = data[data$Origin_mun != "Lisboa",]
```

Filter trips with origin and destination in Lisbon

```
data_in_Out_Lisbon = data[data$Origin_mun == "Lisboa" & data$Destination_mun == "Lisboa",]
```

7.2.0.5 Modify original data

Create a column

The sum of the number of trips for active modes

```
data$Active = data$Walk + data$Bike
```

Remove a column

Look at the first row

```
data[1,] #rows and columns start from 1
```

```
  Origin_mun Destination_mun Total Walk Bike   Car PTransit Other Active
1  Alcochete       Alcochete 20478 6833  320 12484      833     7   7153
```

Look at first row and column

```
data[1,1]
```

```
[1] "Alcochete"
```

Remove the first column

```
data = data[,-1] #first column
```

Create a table only with origin, destination and walking trips

There are many ways to do the same operation.

```
names(data)
```

```
[1] "Destination_mun" "Total"           "Walk"            "Bike"
[5] "Car"             "PTransit"        "Other"           "Active"
```

```
data_walk2 = data[,c(1,2,4)]
```

```
data_walk3 = data[,-c(3,5:9)]
```

7.2.0.6 Export data

Save data in **.csv** and **.Rds**

```
write.csv(data, 'dataset.csv', row.names = FALSE)
saveRDS(data, 'data/dataset.Rds') #Choose a different file.
```

7.2.0.7 Import data

```
csv_file = read.csv("dataset.csv")
rds_file =readRDS("data/dataset.Rds")
```

Part II

Day 2

8 Introduction

This is a book created from markdown and executable code.

See Knuth (1984) for additional discussion of literate programming.

```
data = readRDS("data/TRIPSmode_mun.Rds")
data_walk = data[,c("Origin_mun", "Destination_mun", "Walk")]
head(data_walk)
```

```
  Origin_mun Destination_mun Walk
1  Alcochete       Alcochete 6833
2  Alcochete          Almada    0
3  Alcochete         Amadora    0
4  Alcochete        Barreiro    0
5  Alcochete         Cascais    0
6  Alcochete          Lisboa   69
```

References

Instituto National de Estatística. 2018. “Mobilidade e Funcionalidade Do Território Nas Áreas Metropolitanas Do Porto e de Lisboa: 2017.” Lisboa. https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_publicacoes&PUBLICACOESpub_boui=349495406&PUBLICACOESmodo=2&xlang=pt.

Knuth, Donald E. 1984. “Literate Programming.” *Comput. J.* 27 (2): 97–111. https://doi.org/10.1093/comjnl/27.2.97.