Introduction to Econometrics
Session 4 – Data Wrangling: `tidyverse` and `data.table`

September 2025

The goal of this exercise session is to solve the exercises and the problem twice: once using the `tidyverse`, and once using the `data.table` package.

1 Exercise 1

Create a table with the integers from 1 to 10. Select only the even numbers.

2 Exercise 2

Create a table with the integers from 1 to 5. Add a column containing their squares.

3 Exercise 3

Create a table with a column `group = c("A","A","B","B","B")` and a column `value = c(1,2,3,4,5)`. Compute the mean of `value` by group.

4 Exercise 4

Create a table with `id = c(1,1,2,2)`, `fuel = c("SP95","Diesel","SP95","Diesel")`, `price = c(1.5,1.7,1.6,1.8)`. Reshape the table so that each fuel has its own column.

5 Exercise 5

Start from a table with `id = c(1,2)` and two price columns: `SP95 = c(1.5,1.6)`, `Diesel = c(1.7,1.8)`. Reshape the table to obtain three columns: `id`, `fuel`, `price`.

6 Exercise 6

Create a first table `stations` with `id = c(1,2)`, `city = c("Paris","Lyon")`. Create a second table `prices` with `id = c(1,2)`, `Diesel = c(1.7,1.8)`. Merge the two tables by `id`.

7 Exercise 7

Create a table with `id = c(1,2,3)` and three numeric columns: `a = c(1,2,3)`, `b = c(4,5,6)`, `c = c(7,8,9)`. For each row, compute the mean of all numeric columns except the identifier.

8 Exercise 8

Create a table with `store = c("A","A","B","B")`, `product = c("apple","orange","apple","orange")`, `sales = c(10,5,8,7)`, `price = c(1.2,1.5,1.1,1.4)`. For each store, compute the mean and the standard deviation of all numeric variables.

9 Problem

1. Download the file `prix-carburants-quotidien.csv` from the page https://www.data.gouv.fr/datasets/prix-des-carburants-en-france-flux-quotidien-1/ and load the data into R.

2. Standardize all column names by converting them to lowercase, removing accents and punctuation, and replacing spaces with `"_"`.

3. Create a table `stations`, which for each station identified by its `id`, provides information on its postal code (`Code postal` in the original table), address (`adresse`), municipality (`com_arm_code` and `Commune / Arrondissement Municipal`), department (`Numéro Département` and `Département`), region (`Code Officiel Région` and `Région`), and finally its coordinates (`geom`).

4. In the table `stations`, create two variables `latitude` and `longitude` from the variable `geom`. These correspond to the numerical values separated by `","` in the variable `geom`.

5. From the original table, create a new table `prix` that lists, for each station identified by `id`, the prices of the different fuel types in separate columns.

6. Merge the two tables `stations` and `prix` using the identifier `id`.

7. From this merged table, create a new table `prix_moyens` that provides, for each department (identified by its code and label), the average latitude

and longitude of the stations, as well as the mean price and the quartiles of the price for each type of fuel.

8. Using the `ggplot2` package and the previous table, represent the geographic variations of the average diesel price.

9. In the merged table of stations and prices, identify for each department whether there are stations where the price of SP98 deviates by more than 10% from the average price in that department.