

Tutorial #2: Data visualization and summary statistics

In the previous tutorial, you downloaded datasets from Eurostat, merged them into a single dataset in R, and exported the result as a CSV file.

Objectives of the tutorial

In this tutorial, you will work with the dataset from the second exercise of Tutorial #1 to:

1. explore it through basic summary statistics,
2. create your first data visualizations in R using `ggplot2`.

💡 Clean session

If you need to have a clean session in RStudio, first remove all the objects in memory by evaluating the following instruction:

```
1 rm(list = ls())
```

Then, on the menu “Session”, click on “Restart R”.

Exercise 2.1

Setup

Prepare a project for the second tutorial. Adopt the tree structure presented below. Note that launching your session via `project.Rproj` sets your active working directory to `tutorial-2/`.

```
tutorial-2
├── data
│   ├── raw
│   ├── tmp
│   └── out
│       └── gdp-lifestat.csv
├── scripts
├── figs
├── tables
├── references
├── README.md
└── project.Rproj
```

Q1. Import the dataset you exported at the end of Tutorial #1 (`gdp-lifestat.csv`, assumed to be copied into your data entry structure).

```
1 tb <- read_csv(...)
```

Plot Distributions

Q2. Read the following code, evaluate it, and explain what it does.

¹ dimitri.aiounou[at]univ-amu.fr, ewen.gallic[at]univ-amu.fr, kla.KOUADIO[at]univ-amu.fr

```

1 ggplot(
2   data = tb,
3   mapping = aes(x = gdp)
4 ) +
5 geom_density(fill = "blue", alpha = 0.1) +
6 labs(
7   x = "GDP",
8   y = "Density",
9   title = "Density Plot of GDP"
10 ) +
11 theme_minimal() +
12 theme(
13   panel.grid.major = element_blank(),
14   panel.grid.minor = element_blank()
15 )

```

Q3. Using the `ggsave()` function, export this graph as a PDF file in the `figs` folder of your project environment.

Q4. Plot the distribution of the variable measuring life satisfaction using a histogram:

- Label the x-axis as “Overall life satisfaction”.
- Remove the y-axis title.
- Add the plot title “Histogram of Overall life satisfaction”.

Summary Statistics

Q5. Examine and evaluate the following instructions. Then, explain what they do.

```

1 summary_stats <- tb |>
2   select(gdp, life_satisf) |>
3   summary()
4
5 print(summary_stats)

```

Q6. Create a table of summary statistics for the variables `gdp` and `life_satisf`. The table should include the following statistics: mean, standard deviation, minimum, maximum. Round all values to two decimal places. The resulting table must have two rows (one for `gdp`, one for `life_satisf`). Name this table `summary_stats`.

Q7. Examine and evaluate the following instruction. Then, explain what it does.

```

1 names(summary_stats) <- c("var", "mean", "sd", "min", "max")

```

Q8. Print the content of the object `summary_stats` as a $\text{\LaTeX}$ -formatted table. To do so, use the `{stargazer}` package. Adapt the following code:

```

1 install.packages("stargazer")
2 library(stargazer)
3 ?stargazer
4 summary_stats_latex <- stargazer(...)

```

Save the $\text{\LaTeX}$ table into a file named `summary_stats.tex` inside your output data subfolder.

Rank countries by GDP in 2023

Q9. Create an object called `tb_2023` that contains only the rows of the dataset where the year is 2023.

```
1 tb_2023 <- tb |>
2   ...
```

Q10. Examine and evaluate the following instruction. Then, explain what it does.

```
1 ggplot(
2   data = tb_2023,
3   mapping = aes(x = reorder(country, gdp), y = gdp)
4 ) +
5   geom_bar(stat = "identity") +
6   coord_flip() +
7   labs(
8     title = "GDP by Country in 2023",
9     x = "Country",
10    y = "GDP"
11 ) +
12 theme_minimal()
```

Q11. Create an object called `data_filtered` that contains only the rows of the dataset where the year is either 2014 or 2023.

```
1 data_filtered <- ...
```

Q12. Examine and evaluate the following instruction. Then, explain what it does.

```
1 data_filtered |>
2   arrange(year, desc(gdp))
```

Q13. Create a barplot ranking countries in decreasing order of GDP, with two panels: the left panel for year 2014, the right panel for year 2023. The plot should have:

- (a) the x-axis titled “GDP”,
- (b) no y-axis title,
- (c) the main title “GDP by Country in 2014 and 2023”.

Rank countries by change in GDP between 2014 and 2023

Q14. Create a new variable called `change_gdp` that measures the within-country change in GDP between 2014 and 2023.

```
1 tb <- tb |> ...
```

Q15. Examine and evaluate the following instruction. Then, explain what it does.

```
1 tb_2023 <- tb |> filter(year == 2023)
```

Q16. Create a barplot ranking countries in decreasing order of their change in GDP (between 2014 and 2023). The plot should have:

- the x-axis titled “Change in GDP”,
- no y-axis title,
- the main title “Change in GDP by Country between 2014 and 2023”.

Evolution of GDP in France and Italy

Q18. Create an object called `tb_france_italy` that contains rows of the dataset `tb` where the country is either “France” or “Italy”.

```
1 tb_france_italy <- tb |> ...
```

Q19. Using the dataset `tb_france_italy`, create a line plot showing the year-by-year evolution of GDP for France and Italy (one line per country). Include a legend to indicate which line corresponds to each country. Bonus: use [blue \(#0055A4\)](#) for France and [green \(#009246\)](#) for Italy. Place the legend below the graph. Title the y-axis “GDP”.