Assignment 1

Statistical Computing with R, 2023-24

Introduction

This assignment consists of 2 exercises that have been designed to test the skills you have learned in the first 6 lectures of Statistical Computing with `R`.

When solving the assignment, please bear in mind the following points:

1. you are warmly encouraged to use the packages and functions that you have encountered in the course so far, as well as the ones given as hints in this assignment;
2. the assignment is optional and not graded. You are allowed to discuss your solutions and collaborate with classmates, but please prepare your own solutions, submitting code and answers that you wrote by yourself - this way we can give you feedback about your thought process and code style, which will help you assess your level of preparation so far and will be useful to improve your skills before the exam;
3. try to structure your solutions clearly, and write tidy and commented code.

Preparing your solutions

Please use `R` Markdown to prepare and hand in your solutions:

1. use `R` Markdown to write your solutions to the assignment. Name your `.Rmd` file as follows: `surname_name_SCwR_A1.Rmd`, where `surname` is your surname, and `name` your name;
2. use section and subsection headers to structure the document, so that it is clear which exercise and question you are answering;
3. put your `R` code in code chunks. Keep the argument `echo = TRUE` (= don't hide your code in the compiled pdf!);
4. write your answers inline (= outside code chunks); please don't include your answers as comments inside code chunks!
5. compile your solutions as `.pdf`. If you have issues in compiling your document as pdf, you can reach out to the TAs for help!

Submission

The **deadline** for submissions is **Sunday 15th October 23:59**. To hand in your solutions, submit both your `.Rmd` and `.pdf` files through Brightspace.

Doubts / questions?

If you have any questions about the assignment, feel free to ask the TAs during the coding sessions, the question hour or via email to statcompr[at]gmail.com.

Good luck and have fun!

Exercise 1

In this exercise, we will be looking at a dataset about Coca Cola pricing between 1962 and 2022. This dataset contains 8 variables:

- `Date`: the date at which the observation was collected.
- `Open`: the price of the stock at the time the stock market opened.
- `High`: the highest price the stock reached on that date.
- `Low`: the lowest price the stock reached on that date.
- `Close`: the price of the stock at the time the stock market closed.
- `Volume`: the volume of stocks traded on that date.
- `Dividends`: the profit paid to the shareholders.
- `Stock Splits`: the amounts of splits of the Coca Cola stock.

Data preparation

1. Download the dataset from the Github repository: https://github.com/kalilurrahman/coca-colastockdata, and load the data into `R`.
2. Since this dataset contains a significant number of rows, we will be looking at a subset of this dataset. Make a dataframe that contains only the observations between 01-01-2000 and 31-12-2019. *(Hint: You may find the `as.Date()` function useful.)* Change the column names such that there are no column names that contain the space symbol. For the remainder of this exercise, consider only this subset of the data.

Data manipulation

3. Create a new column taking values `TRUE` / `FALSE` that indicates if the maximum volume sold in one day exceeded $9 \cdot 10^6$. Name this column `HighVolume`.

4. For the days where the volume exceeded $9 \cdot 10^6$, comment on the relationship between the highest price (`High`) and the dividends (`Dividends`), by looking at a scatterplot. What do you see? Is there anything surprising? If so, what could be an explanation?

5. Create a summary dataframe that contains the highest value for each numeric column per year *(Hint: you may find the package `lubridate` useful to answer this question).*

6. Look at the relationship between the highest price (`High`) and the dividends (`Dividends`) of the summary dataframe you created in question 1.5, by looking at a scatterplot. What do you see? What piece of information are you missing in this plot, that was apparent in the plot you made in question 1.4?

7. Create and display in the output a list that for each of the numeric columns contains

   i. the column name;
   ii. the lowest value;
   iii. the earliest date at which this value was measured;
   iv. the difference between the lowest and highest value of that column.

Exercise 2

In programming, the following song is quite famous:

https://www.songfacts.com/lyrics/traditional/99-bottles-of-beer

Write a general function that creates the lyrics of this song along with a header. The function should take as an argument the number of beer bottles as a numeric vector of length one. Make sure to include an `if` statement with a `stop` error message in case the input is not an integer above zero of length one. If you would put 5 as argument in your function, the output should look like this:

> 5 Bottles of Beers Song
>
> 5 bottles of beer on the wall, 5 bottles of beer. Take one down, pass it around, 4 bottles of beer on the wall...
>
> 4 bottles of beer on the wall, 4 bottles of beer. Take one down, pass it around, 3 bottles of beer on the wall...
>
> 3 bottles of beer on the wall, 3 bottles of beer. Take one down, pass it around, 2 bottles of beer on the wall...
>
> 2 bottles of beer on the wall, 2 bottles of beer. Take one down, pass it around, 1 bottle of beer on the wall...
>
> 1 bottle of beer on the wall, 1 bottle of beer. Take one down, pass it around, no more bottles of beer on the wall.

Here are some tips that you could use in your function:

- Check `?is.integer` and you will see in the Note that you cannot use this function directly to test whether the input contains integer numbers. You have to come up with a workaround.
- Use the function `paste()` or `paste0()`to concatenate strings in order to write the text of the header or lyrics.
- If you want some text to start in the next line, use `\n` at the end of the previous string within the quotation marks. You can use `\n\n` for an empty line between the verses.
- Notice that the last two verses for 2 bottles of beer and 1 bottle of beer are slightly different than the rest. The other verses are all the same counting down from the numbers of beer of the input argument. To write these verses, you can use a loop or `if...else` statements.
- You may use `cat()` instead of `print()` to print text to the console, and `\n` to print text on a new line (example: type in the console `cat('Good morning!\nHow are you?')`)

Check whether your function prints the lyrics for 5 beer bottles like the example above. Include your written function and the output for 5 beer bottles, `mybeerbottlefunction(5)`, in your answer.