Exercise Sheet #3

Fortgeschrittene Statistische Software für NF - SS 2022/23

Jacob Beck & Jan Simson

2023-05-26

General Remarks

- You can submit your solutions in teams of up to 3 students.
- Include all your team-member's names and student numbers (Martrikelnummern) in the `authors` field.
- Please use the exercise template document to work on and submit your results.
- Use a level 2 heading for each new exercise and answer each subtask next to it's bullet point or use a new level 3 heading if you want.
- Always render the R code for your solutions and make sure to include the resulting data in your rendered document.
  - Make sure to not print more than 10 rows of data (unless specifically instructed to).
- Always submit both the rendered document(s) as well as your source Rmarkdown document. Submit the files separately on moodle, **not** as a zip archive.

Exercise 1: Initializing git (4 Points)

For this whole exercise sheet we will be tracking all our changes to it in git.

a) Start by initializing a new R project with git support, called `exeRcise-sheet-3`. If you forgot how to do this, you can follow this guide.
b) Commit the files generated by Rstudio.
c) For all of the following tasks in this exercise sheet we ask you to always commit your changes after finishing each subtask e.g. create a commit after task *1d*, *1e* etc.

   Note: This applies only to answers that have text or code as their answer. If you complete tasks in a different order or forget to commit one, this is no problem. If you change your answers you can just create multiple commits to track the changes.

d) Name 2 strengths and 2 weaknesses of git. (Don't forget to create a commit after this answer, see *1c*)
e) Knit this exercise sheet. Some new files will automatically be generated when knitting the sheet e.g. the HTML page. Ignore these files, as we only want to track the source files themselves.

Exercise 2: Putting your Repository on GitHub (3.5 Points)

For this task you will upload your solution to GitHub.

a) Create a new repository on GitHub in your account named `exeRcise-sheet-3`. Make sure you create a **public repository** so we are able to see it for grading. Add the link to the repository below:

b) Push your code to this new repository by copying and executing the snippet on github listed under `...or push an existing repository from the command line`.
c) Regularly push your latest changes to GitHub again and especially do so when you are finished with this sheet.

Exercise 3: Baby-Names in Munich (4.5 Points)

Download the latest open datasets on given names ("Vornamen") from the open data repository of the city of Munich for the years `2022` and `2021`.

Link: https://opendata.muenchen.de/dataset/vornamen-von-neugeborenen

a) Download the data for both years and track it in git. For small datasets like these adding them to git is not a problem.

b) Load the data for both years into R. Check the type of the count variable ("Anzahl") and look into the data to determine why it is not numeric? Fix the problem in an appropriate manner, it is OK if some of the counts are inaccurate because of this. Explain your solution and the repercussions.

c) Calculate the total number of babies born in Munich in 2022 and 2021. Which year had the bigger baby-boom?

d) Add a new column `year` to both datasets which holds the correct year for each.

e) Combine both datasets into one using `bind_rows()`.

f) Combine the counts for same names to determine the most popular names across both years. Print out the top 10 names in a nicely formatted table for both years. Include a table caption.

Exercise 4: Chat GPT + apply (3 points)

For this task: Specifically use ChatGPT to solve the task and submit your prompts in addition to the solution

a) The code below does not work because the wrong apply function has been used. Find out which apply function would be correct and why it did not work. Correct the code. Also calculate the rowwise means.

###Create a sample data frame

tax_data <- data.frame(Name = c("Munich GmbH", "ABC Inc.", "Backpacks 1980", "Bavarian Circus"), Tax_2019 = c(5000, 4000, 6000, 3500), Tax_2020 = c(4800, 4200, 5800, 3700), Tax_2021 = c(5200, 3800, 5900, 3400))

###Calculate column-wise means

column_means <- lapply(tax_data[, -1], 2, mean)

column_means

b) Using ChatGPT try to understand what the rapply() function does. Create an easy example with mock data where the function is used and explain it in your words.

Final Note

Make sure to push all your commits and changes to GitHub before submittining the exercise sheet.