RIntro Exam – 1st term 2022/23 – group 4pm

by Maria Kubara

Organisational information: The main purpose of this exam is to show how much you have learned from the RIntro course - or how to code in R. It doesn't really matter whether you use base functions or tidyverse solutions to solve these tasks. Just use the approach that works for you. Think of it as practice before professional work. The main goal here is to do the analysis, in a limited amount of time, not what the codes will look like. However, if you keep your code clean according to the principles you learned in the material on "Clean and Reproducible Code Writing" you can expect to get some bonus points from me to increase your overall score above the 70pct limit. The same applies to the BONUS TASK. It is scored completely outside the 70 point limit. There is no answer key to the bonus task either. If I like your answer (and your thinking), I will give you some bonus points. This assignment is designed to mimic real work situations. Usually you won't get a set of tasks, but rather a puzzle to solve using data and your coding skills. Good luck!

Once you have solved the tasks, please submit your answers via the form below:
https://forms.gle/JPka35AvyBswucJ56

Your tasks:

You have just applied for an internship at ChatXDD - a new, emerging company providing an AI chatbot solution that can chat live with many people around the world. Your chatbot is still in development, but it has already attracted a number of enthusiasts who would like to test it out (especially among students who are trying to use the output of the chatbot as a template for their essays). Due to the high volume of traffic on the site, the ChatXDD chatbot is experiencing numerous errors in its daily operations. As your company is understaffed, it sometimes takes a long time to fix the problems that arise. Your task as an intern is to conduct a detailed analysis of the website crash incidents and understand if there are any patterns behind the time it takes to fix them. Your more experienced colleague has provided a set of initial tasks that can help you prepare such a report. Prepare code in R solving the issues raised in the following tasks.

To access the dataset, please use the following code:

read the csv file chatXDD.csv (be careful of about the separator and decimal mark!)

use filtering on the rows, depending on your student ID number (if you're an Erasmus student, use only the numbers in your ID)

```
data <- read.csv … # READ DATA HERE

# IMPORTANT!! Remember about separator and decimal!!!

id <- 123456 # YOUR ID HERE

set.seed(id)

myData <- as.data.frame(data[sample(1:10000,200,replace=FALSE),])
```

Description of the dataset:

breakdownTime	Day of the website crash (date, when the website error occurred)
employees	Number of employees involved in fixing the incident
siteTraffic	Number of people active on the website at the time of the incident
errorCode	The error code, which includes: the error code number (what the website returned during the incident), and the microservice that was affected by the website failure (microservice 1 - S1, and microservice 2 - S2).
topic	Most popular topic among user queries at the time of the crash (mathematics, economics, medicine)
hoursToImprove	Time (in hours) needed to fix the issue

Tasks:

1. Read the description of the dataset and compare it with the result of the str() function. What types/classes are assigned to the variables (currently) and how they should be represented in R to ensure the most efficient and convenient calculations (eg. factor, numeric, character…)?

	Type (currently)	Target type/class in R
breakdownTime		
employees		
siteTraffic		
errorCode		
topic		
hoursToImprove		

2. Rename variable ‘breakdownTime’ to 'indicentTime’. Correct the error created at the data creation stage. Do not create a new variable, just rename an existing one.
3. Transform the ‘topic’ variable to the appropriate type. Make sure your changes affect the data.frame that you are operating on.
4. Examine the contents of the ‘errorCode’ variable. What type of information does it store? Separate two pieces of information from this variable into the variables ‘errorNumber’ and ‘microservice’.
5. Calculate the maximum number of employees involved in fixing the issues occurred when “economics” was a trending topic.
6. For the issues resolved by at least 16 employees, calculate the maximum time which was needed to restore the normal website activity.
7. Add a new variable to the dataset, which will store the time needed for fixing the issues in minutes. Name it “minutesFixing”.
8. Create a boxplot that shows the distribution of minutes needed for fixing the issues across both microservices. Change title of the x axis to “Microservice type”.
9. Build a linear model “myModel” (function lm()) that explains the time needed for fixing the issues in hours as a function of the available employees, website traffic, trending topic and microservice involved in the crash. Extract the coefficient for the employees number and print it out.
10. Create a group comparison in the form of a summary table (which should look like the one below):

microservice	topic	averageEmployees	maximumFixTimeInMinutes
S1	economics	X	X
S1	mathematics	X	X

S1	medicine	X	X
S2	economics	X	X
S2	mathematics	X	X
S2	medicine	X	X

It should compare the average number of employees involved in fixing the issue and the maximum time in minutes that passed until the website was fully operating, across issues related to both microservices and different topics trending. TIP: ordering of the rows may be different than in the example.

BONUS TASK, BONUS POINTS: Using your expert skills, which do not require AI support, make a recommendation on the number of employees who should be available in-house to fix any errors on the website. Decide whether the number of people waiting should be fixed, or should it depend, for example, on current site traffic or topic trending? Use your R skills and analytical intuition to convince me that your recommendation is valid. Keep your reasoning short – the written part of this assignment (without code) must be no longer than 200 words.