Reflective Report

Approaches

In my project, among 7 years of data, I have selected two data which are the first year of enrolments and step activity data. The initial data had more than 100,000 rows so I justified that it is enough to derive the conclusion. However, because the data quality was poor – it had lots of missing values, after data preprocessing, it only remains around 12,000 objectives in enrolments data. Due to the limitations of time, I have decided to explore only the first-year data but I suggest to analyse all the 7 years of data to extract more accurate outcomes.

Tools/Techniques

CRISP-DM

By taking the step of CRISP-DM, I have learnt how to connect the data itself to connect the business problem. I usually know how to generate the plots from data and just treating the data science as a math or coding. However, after learning CRISP-DM, I start to think about what the stakeholders want to find from data and how the data can interpret in relation to real-life problem. It gave me confidence to work in the company dealing with dataset and explaining in the business language.

Nevertheless, I have struggled with the CRISP-DM form because sometimes there are guidance about machine learning/deep learning instead of EDA. For example, model part guided me to compare the model's accuracy and parameters used and conclude which one was the best but there are no guidelines for EDA. I did apply them into EDA problem but it would be better if there was a clear instruction.

ProjectTemplate

ProjectTemplate was very useful to arrange the file and for automation. For instance, I used to feel uncomfortable with R studio since I need to install all the packages to set up the environment everytime I open the R project in the different place such as home or school. However, ProjectTemplate has the function for setting up all the libraries that I wrote in the global.dcf file so that once load.project() is run, every environment is set up. Moreover, not only the environment but also it helps to preprocessing automatically once I wrote the code for data wrangling In the munge directory.

Although ProjectTemplate has these huge benefits, there were couple of things that I felt uncomfortable. When I use it first time, as I am not an expert, it was hard to understand the process of ProjectTemplate. For example, when I have written head() function in the munge directory and ran the load.project(), it gave me infinite runs in the console, which is ended up wasting quite a long time to fix this error. Still, only the first-time encountering the new technology is hard but once I get used to it, it is very convenient tool to use.

dplyr & ggplot

dplyr helps me to wrangle the data easily. Especially, when I wanted to plot only 10 highest and lowest classes in the step activity and enrolments data – such as column of country, employment area and activity time, using mutate() simply extract the data I wanted.

Also, ggplot is helpful to draw a visualisation of data particularly it maximises its advantage when using with ProjectTemplate. For example, as ggplot can save the plots as a name of variable, I can produce a graph in src directory and simply write a graph name in the R markdown report. This might sound simple but because of this, I can attempt to plot different types of graphs and choose the best one to put in a report.

Lastly, both dplyr and ggplot are popular libraries commonly used so it was a great practice to get used to them.

Tasks

I personally think the data was not very handy to work with as there were no explanation but the file name. Also, sometimes it is hard to understand what each column represents. For example, the column name fully_participated_at in enrolment.csv file is quite vague to tell its purpose whether it is fully-participate for registration or entire course. It would be better if there is a brief description about what the data file and column does in the task.