M2T2
Lessons Learned Report

Main lessons learned from this experience.

- It was interesting to perform another EDA with a different dataset. Despite the plan of attack is the same, changing the dataset and the main question, you changed everything. It could sound obvious, but it isn't.
- I liked very much using GitHub to share my job with the mentor. I mentioned that in the last lesson learned report but didn’t expect that the next activity will include using it. There are a lot of things I still don’t know how to use and never explored. Hope to learn them soon.
- I enjoyed a lot the pre-processing tasks, particularly the problem of the mixed data types. It took me some long hours of trying different things without being able to solve them. In the beginning, I didn’t size the problem. I thought it was like: 'Ok, easy, just remove the header, that somehow they are interfering with my dataset...” but then I realized there was something else. In the end, with some tips from the mentor, I was able to remove those annoying rows. A very nice task that taught me a lot.
- The task of adjusting types was awesome because led me to read about the impact in the performance that can produce using integers instead of floating-point numbers.
- The method get_dummies was a remarkable finding. After reading about it in the resources, I was doing lots of copies of the categorical variables columns and relabeling them into numbers…. get_dummies solved it in a while. It's a useful tool.
- To perform the visualizations I gave myself a second chance with Matplotlib and another tutorial. In this episode, I found some YouTube tutorials that gently guided me. Now I can understand better the examples provided in the documentation. Also understood how to improve Seaborn graphs. For example, to perform the histograms of the numerical variables, I used a homemade function that allowed me to perform lots of repetitive plots effortlessly.
- The major part of this task was the feature selection. The resources provided to solve this task were scarce, and I was blocked and just didn’t understand what to do neither where to start. The mentor provided a very useful extra resource: https://machinelearningmastery.com/ a website with tons of great information, wrote pleasantly. In a couple of hours, analytically, I solved the problem using the scikit-learn library that measured the correlation between every input variable with the output. Next, I attach to each value the correspondent label and ordered the output in a descendent way. I didn’t use the selectKbest method provided by scikit-learn, I wanted to see the measured correlation of every variable. Finally, I verified the analytic result with the boxplots: now the difference is precise.
- Thanks to the process of selecting the useful features, I learned that having lots of columns with lots of records, doesn't mean that all of them provide meaning to your business question. Having irrelevant features in your data can decrease the accuracy of many models, especially linear algorithms like linear and logistic regression. Some benefits of performing feature selection before modeling your data are: Reduce Overfitting, Improves Accuracy and Reduce Training Time.