Exercise 3 Report

**Preamble. For this exercise I used Google Colab because I was not able to install correctly Spark on my pc. I tried different times, I followed different tutorials and installed different versions of Python, Java and Spark wasting 3 days and at the end I decided to do it with Colab. Because of this I was not able to train the model with the entire dataset that was too big to handle for the standard plan of Colab and I recognized it with some effort because It gave strange errors whenever I wanted to train the model.**

I started this exercise by looking at the ML notebook from the notes to understand how a dataset can be treated, how a model can work and can be trained. Then I downloaded the flight dataset and I looked into it finding out that there were 32 columns, some of Object type in particular String and the rest Double, so I defined the schema of the dataset and loaded into a Spark Data Frame.

Then I analyzed it understanding the mean of the columns name, like TAXI\_IN that I didn’t know what was, and noticed also that there were values in some column that were derived from others, but at that time I didn’t know if that was useful. Then I calculated some basic statistic values like mean, standard deviations, max and min to have a look on them. I also calculated the correlation between the target columns ARR\_DELAY and others underlining important relationship, for example the correlation with DEP\_DELAY is the biggest one as everyone can expect even without knowing other things. Next, I looked for null values on the column by calculating the percentage of null values in each column. Columns like DELAY\_DUE\_TO\_CARRIER have more than 80% null values so are useless for the model.

Starting from this point I considered which columns to hold and which one to drop. I decided to drop 19 columns that weren’t useful both for null values or for answering to the problem of the delay of 15 minutes and dropped the null values of ARR\_DELAY and DEP\_DELAY that are the most important features for answering the question, also seen by the correlation.

After that, I had a clean dataset ready to be assembled into a vector for the training of the models. First thing I have done is to convert string values in binary values with String Indexer and One Hot Encoder. Then arrived a section in which I used different approaches, the first one was to scale all the variables, binary and double values, and I saw that for the binary ones they were scaled in strange manners, because the aim of the scaler is to standardize all the values into some values that are not far from each other, but for the binary one this didn’t happen, they were not closer to 0 or 1 but much higher. Then I tried scaling only the features with double value and I thought that was the way to go (the code of this section is commented). I went ahead on the next steps but after training the models a bunch of time I tried not to scale any feature and had about the same results.

Then was the time to define the Logistic Regression model and the Random Forest model. I did a research on the models, trying to understand how they work, what are the params and then I started coding them. With them I searched also for the ParamGridBuilder and CrossValidator. Now the hardest part came, because I had a lot of problems with params, number of folds of cross validator and the size of the dataset. Restarting the Colab session, sometimes the models trained and sometimes not. To understand what was going on I spent a lot of time because training the models were very long, on average 15 to 35 minutes for Logistic Regression and 40 to 90 minutes for Random Forest. I could not use a single time the number of folds equals to 5 as the exercise says because it would not be supported by Colab standard plan CPU’s, so I used 2 or 3 for that. Then I faced also the problem of the size of the dataset reducing it from an 80% to a 60% maximum trying different sizes and different training of the models. Every time I went higher with the size of the dataset together with the number of folds or the number of parameters, something went wrong. Even the number of trees and the depth gave me some problems. Anyway, after some effort I did it and trained the models with the best I could do.

Then it was time to test the best model that I got and analyze the results. I didn’t have problems with parameters, but only with the confusion matrix in the case of Random Forest that most of the times gave me only 1 value and not 4. Searching for some solution I found out that was a problem of overfitting so my parameters weren’t good for the training. So, I went back and trained the model a lot with different parameters, always facing the problem with Colab maximum processing capacity. The best parameters for my case are written in the notebook and also the best value for evaluators. This exercise has been very challenging because I had to get into it without lots of notions about machine learning but I learned a lot and managed to do it.

PS. From the Random Forest training to the end, the notebook is not completed because It would have taken too much time and I couldn’t deliver the homework on time, but I have the images from previous sessions, with the models trained with the same parameters, of the ROC, the confusion matrices and all the plots.