Data Scientist AltScore - Hackathon

Santiago Pozo Ruiz
santiagopozoruiz@gmail.com

March 30, 2021

Table 1: Number of records in clients database after each filter application

	Brief description	**Number of records**
Raw data	N/A	1545000
Filter 1	Select just clients with contracts since 2015	623242
Filter 2	Remove operations carried out in Italy in 2019	560947
Filter 3	Remove clients with more than 75% of missing information with the database	534073
Filter 4	Remove clients who appear more than once in the database (more than one contract)	518501
Filter 5 - Final clients database	Just clients with more than two years of information within the database are kept	9997

Obtain relevant variables

Table 2: Overview of new relevant variables on four individuals

	credit_score	account_balance	number_of_products	age
1	678	128644.46	1	41
2	647	93960.35	1	45
3	460	102742.91	2	35
4	533	85311.70	1	37

Table 3: Descriptive statistics of new relevant variables

	Account Balance	Credit Score	No of Products	Age
Mean	96453.68	648.61	1.42	41.99
Standard deviation	41793.06	98.77	0.67	10.6
Maximum	238387.56	850.0	4.0	84.0
Minimum	0.0	350.0	1.0	18.0

Building the model

- When all variables are calculated, the dataset still needs to be processed. Categorical variables needed to be enconded; in this case, it was an integer encoding.
- Variables with no predictive power such as *CustomerId* need to be taken out of the dataset.
- A variable which gives us information about the early unsubscription of a clients was needed. In this case, it is the *left_early?*. An overview of the final table is shown below.

Table 4: Overview of the final table for machine learning predictions. Data for three individuals is shown.

	Geography	Gender	HasCrCard	Estimated_Salary	credit_score	account_balance	number_of_products	age	left_early?
1	0	0	0	167673.37	678	128644.46	1	41	0
2	1	1	1	36579.53	647	93960.35	1	45	1
3	2	0	1	189339.6	460	102742.91	2	35	0

Building the model (cont.)

- After characterization, the next step (if needed) is feature selection to improve performance. Correlation plot was generated to check if the variables are correlated and could be removed. The result is that they have not significant correlation.

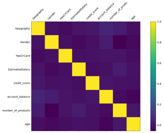
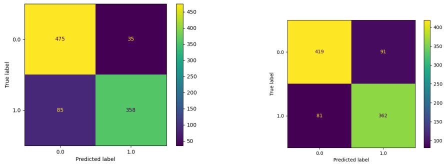


Figure 2: Correlation plot for chosen variables

- It could be noticed that data was unbalanced. A clustering method for under-sampling was used to solve this problem. CCMUT under-sampling
- Then, data was scaled and sub-optimal parameters for the classifiers were chosen. Best results among all classifiers are shown as confusion matrices.



(a) Random Forest

(b) MLP

Figure 3: Confusion matrix for an instance of the classifiers

Building the model (cont.)

- The following classifiers were tested for the given dataset: Ada Boost, Naive Bayes, Quadratic Discriminat Analysis, Support Vector Machine, K-NN, Random Forest and Multilayer Perceptron. So, most of the classifiers available on the *sklearn* library were tested. Deep learning approaches were not taken into account due to the lack of computational resources. It is necessary to mention that CNN and LSTM networks combined could be a good approach, however as deep learning are still considered as black-boxes their usage is not recommended in financial cases like the one presented here.

- At the end, the classifiers which performed better were Random Forest and Multilayer Perceptron, with recall scores of 0.807 (+/- 0.032) and 0.826 (+/- 0.027) respectively. And a general accuracy of 0.872 (+/- 0.013) and Accuracy: 0.821 (+/- 0.015) respectively. The confidence interval is 95%.

- To validate the results and to avoid a single time biased measures, a 10-fold cross-validation process was carried out using the *ShuffleSplit()* routine. Then, a boxplot was generated to better appreciate the performance of each classifier focused on the *recall*, this is important because in this case, recall is the measure that tells us how well the model predicts which clients would leave Kin Safety early.

- As mentioned in the previous slide, Random Forest and Multilayer Perceptron are the classifiers that perform better than the rest.

Building the model (cont.)

- Support Vector Machine and Ada Boost can also be taken into account for further study. However, their recall values are low compared to Random Forest and MLP.
- It is hard to choose between Random Forest and MLP. Random Forest seems to produce higher performance scores, however MLP boxplot seems to be more compact, this means that MLP results could be more trustable. Presented results are not definitive and surely can be improved in further studies.

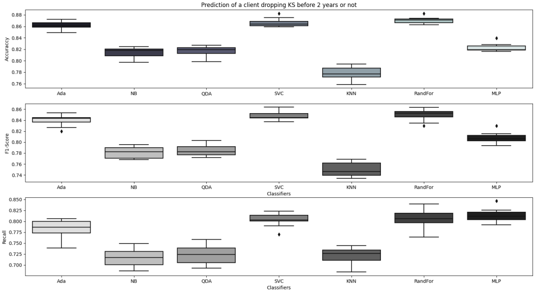


Figure 4: Final table for machine learning predictions