**Credit Card Approval**

Changuk Lee

**Introduction**

The United States is a credit-based economy. For instance, the landlords and insurers look at the credit score based on the credit card. The rent fee and insurance fee can go down when one’s credit score is high. In addition, the bank can also benefit from high credit score clients. One of the advantages can be collecting interest fees from credit card owners. In contrast, when there are increasing numbers of bad clients, there will be economic chaos in the financial chaos. The bad credit card owners have fewer options for rent and insurance due to low credit scores. From the bank’s perspective, the bank is not able to collect money that the bad client borrowed. The main purpose of the credit card approval project is to determine whether the credit card owner is a good client or a bad client based on the client’s information submitted by the credit cardholders.

**Data Wrangling and Cleaning**

Data Source: <https://www.kaggle.com/rikdifos/credit-card-approval-prediction>

There were 0, X, C, 1, 5 values of the status. Since 1 and 5 mean that client’s payment was overdue, 0, X, and C were considered as 0 which is a good client. The 1 and 5 were classified as a bad client. After process of data wrangling and cleaning, project\_data.pickle file was created by merging two datasets called application\_record and credit\_record using client’s ID. After printing out the information of the dataset, no missing values were found. There were duplicated IDs for some clients. Therefore, the last values of the client’s ID information were kept in the dataset.

**EDA**

**Figure 1**

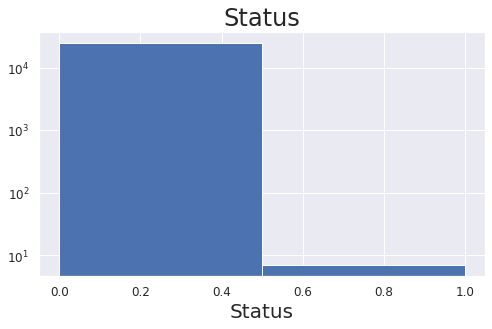


Figure 1 represents the labels of status. There are two statuses which are 0 and 1. As it was mentioned in the data wrangling section, 0 means good clients while 1 is the representation of bad clients. To be more specific, the first bar represents a good client while the second bar represents a bad client. As it was shown in the figure 1, there were over 10000 good clients but less than 10 bad clients. One can assume that the dataset is a highly unbalanced dataset. Therefore, both accuracy and sensitivity should be used to get the correct interpretation.

**Figure 2**

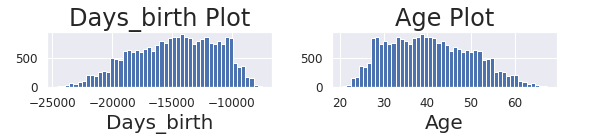
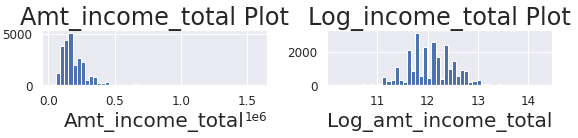


Figure 3

In figure 2, the original days\_birth column is negative because 0 represents today. For instance, if the client’s day\_birth is -365, the client is 1 year old. To find out the client’s age, the days\_birth was divided by -365. The age plot showed a better picture of the client’s ages. In figure 3, the amt\_income\_total was largely right skewed. The log transformation was used to solve the right-skewed problem. Taking the log of amt\_income\_total made amt\_income\_total more normally distributed graph.

**Inference**

Significance: ∗p < 0.05, ∗∗p < 0.01, ∗∗∗p < 0.001

|  |  |
| --- | --- |
|  | Status |
| Log Name\_family\_status\_Separated  Log Name\_housing\_type\_Municipal apartment  Log Cnt\_children | -2.764 β1\* +  (249.85)  -0.084 β2\*\* +  (28.92)  -0.018 β3\*\* +  (13.69) |
| Features  Obs  R2 | 51  438,558  0.99706 |

There were 28 variables such as name\_income\_type\_Pensioner, occupation\_type\_Drivers, and name\_education\_type\_Lower secondary removed because every 28 variables had coefficients of 0 which means that variables are not affecting the model. The 0.01 alpha was obtained as the best alpha for the inference model using the grid search method.

Interpretation on three significant variables

* One standard deviation increase in the name\_family\_status\_Separated is associated with an decrease probability of having status by -2.764.
* One standard deviation increase in the name\_housing\_type\_Municipal apartment

is associated with an decrease probability of having status by -0.084.

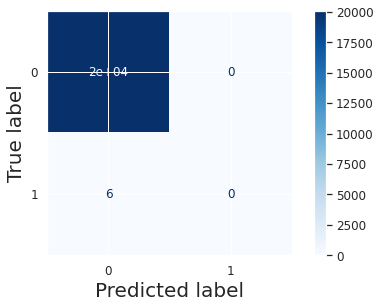
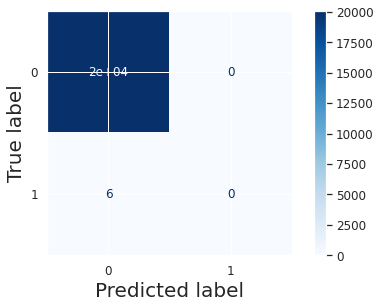
* One standard deviation increase in the cnt\_children is associated with an decrease probability of having status by -0.018.

**Prediction**

|  |  |
| --- | --- |
| XGBoost | N\_estimators = 30  Max\_depth = 2  Learning\_rate = 0.5  Random\_state = 490  Early\_stopping\_rounds = 5 |
| AdaBoost | Base\_estimator = DecisionTreeClassifier(max\_depth = 1)  N\_estimators = 25  Learning\_rate = 0.1  Random\_state = 490 |

To find the best hyperparameter, both the XGBoost model and AdaBoost model used the same random\_state of 490. XGBoost model was first ran using n\_estimators of 250 with early\_stopping\_rounds of 5 but the best\_iteration came out as 30. Therefore, n\_estimators was changed to 30 in the final model with no early\_stopping\_rounds. Using the grid\_search, AdaBoost model was run through n\_estimators of 25, 50, 75, 100, 125 and learning\_rate of 0.1, 0.5, 1, 1.25, 1.5. The best hyperparameter was using n\_estimators of 25 and learning\_rate of 0.1. In addition, the base\_estimator for AdaBoost model was DecisionTreeClassifier using max\_depth of 1.

**Comparison**

XGBoost Model AdaBoost Model

The accuracy for the XGBoost model and AdaBoost model came out to be 0.9997 so both confusion matrixes look the same. In addition, both models had the same sensitivity which is 1. The model did well on predicting the true positive of the confusion matrix because both models predicted every client as 0 which is a good client. However, the models are inflexible and hard to interpret because the dataset is unbalanced. For instance, there was a bias on predicting statuses of clients because there were 25,127 good clients but only 7 bad clients. Even though both the XGBoost model and AdaBoost model had exceedingly high accuracy and sensitivity, there should be more bad client data to have better interpretation.

**Conclusion**

The purpose of the credit card approval is to evaluate the client’s status using the client’s information that was submitted by each credit card owner. The dataset was collected through the Kaggle website (Data Source: <https://www.kaggle.com/rikdifos/credit-card-approval-prediction>). There were three models used in the credit card approval prediction which were the inference model, XGBoost model, and AdaBoost model. The inference model had the lowest accuracy of 0.9973. On the other hand, the XGBoost model and AdaBoost model had the tie highest accuracy of 0.9997. Even though the XGBoost model and AdaBoost model did well in predicting good clients, it is hard to say that both models are good because there were nearly 25,000 more good clients compared to bad clients. To have a better interpretation on determining if the client is a good client or a bad client, the bank should collect more data on the bad clients who are overdue on the credit card payment.