Predicting Credit Card Default using XGBoost

Alexander Indrajaya L, Novan Dwi Atmaja, and Siti Fatimah

The following report describes the methodology and exploratory process in the prediction of credit card defaults by using indicators gathered from historical transaction. We also use synthetic minority over sampling (smote) technique to solve the imbalanced data problem. We find that by using smote algorithm the recall value has increased significantly even though this technique caused a slight decrease in the area under curve (auc) and accuracy.

Keywords: credit scoring, xgboost, smote algorithm

Executive Summary

In this report, we will do the credit card default modeling based on data that has been provided by FinHack 2018. We use the XGboost model as the base model. Previously we would do a features selection and solve unbalanced data problems using the smote algorithm. Next, we will measure the performance and stability of the model.

We divide the data to training and testing (80:20). By using the test data, we found that the smote process provided auc and accuracy reduction of 2% - 6% and a recall increase of more than 300%. The most important variable to predict the credit card default is the ratio of the amount paid to the bill in the last month.

The Dataset

The data we use consists of 15,493 data. The data contains related history of credit card usage and the status of whether the credit is good or not. Data details can be seen on the finhack dashboard.

Exploratory Data Analysis

Training data consists of 1,359 (8.77%) bad credit and 14,134 (91.23%) good credit. We find that the data are completed (no missing data).

First, we will measure the predictive power of characteristic using information value (IV). One rule of thumb regarding IV is:

Less than 0.02	: unpredictive
0.02 to 0.1	: weak
0.1 to 0.3	: medium
More than 0.3	: strong

We measure IV using “library (smbinning)”. This algorithm will categorize numeric data into certain bins based on the Conditional Inferences Tree.

```
> iv_table
                                   Char     IV                Process
10                     rasio_pembayaran 0.8233     Numeric binning OK
```

```
8              total_pemakaian_retail 0.8167    Numeric binning OK
18          total_pemakaian_per_limit 0.7214    Numeric binning OK
15                    total_pemakaian 0.6959    Numeric binning OK
6                             tagihan 0.5272    Numeric binning OK
4                         outstanding 0.4698    Numeric binning OK
17             sisa_tagihan_per_limit 0.4575    Numeric binning OK
22                   utilisasi_6bulan 0.3512    Numeric binning OK
19            pemakaian_3bln_per_limit 0.2898    Numeric binning OK
11               persentasi_overlimit 0.2659    Numeric binning OK
21                   utilisasi_3bulan 0.2496    Numeric binning OK
16       sisa_tagihan_per_jumlah_kartu 0.2152    Numeric binning OK
9           sisa_tagihan_tidak_terbayar 0.2117    Numeric binning OK
5                        limit_kredit 0.0103    Numeric binning OK
2                      skor_delikuensi 0.0083     Factor binning OK
3                         jumlah_kartu 0.0062    Numeric binning OK
1                          kode_cabang     NA  Too many categories
7                  total_pemakaian_tunai     NA No significant splits
12               rasio_pembayaran_3bulan     NA No significant splits
13               rasio_pembayaran_6bulan     NA No significant splits
14 jumlah_tahun_sejak_pembukaan_kredit     NA No significant splits
20            pemakaian_6bln_per_limit     NA No significant splits
```

Based on IV, it was found that there were 8 features: rasio_pembayaran, total_pemakaian_per_limit, total_pemakaian_retail, total_pemakaian, outstanding, tagihan, sisa tagihan per_limit, and utilisasi_6bulan was a strong predictor to measure bad credit.

From the table above also obtained that there are 5 features (total_pemakaian_tunai, rasio_pembayaran_3bulan, rasio_pembayaran_6bulan, jumlah_tahun_sejak_pembukaan_kredit, and pemakaian_6bln_per_limit) that significantly do not affect the customer's credit status (the tree is not formed). This feature will be excluded from the model.

Predicting

The available data showed the existence of imbalanced data cases.

```
> table(data_train$flag_kredit_macet)
    0     1
14134  1359
```

However, most of the existing state-of-the-art classification approaches are well developed by assuming the underlying training set is evenly distributed. Thus, they are faced with a severe bias problem when the training set is a highly imbalanced distribution. The resulting decision boundary is severely biased to the minority class, and thus leads to a poor performance according to the receiver operator characteristic (ROC) curve analysis. The synthetic minority oversampling technique (SMOTE) is an important approach by oversampling the positive class or the minority class.

At this stage, we will predict credit default by using XGBoost. We will compare the effect of smote on training data on model performance. The following are the steps we did:

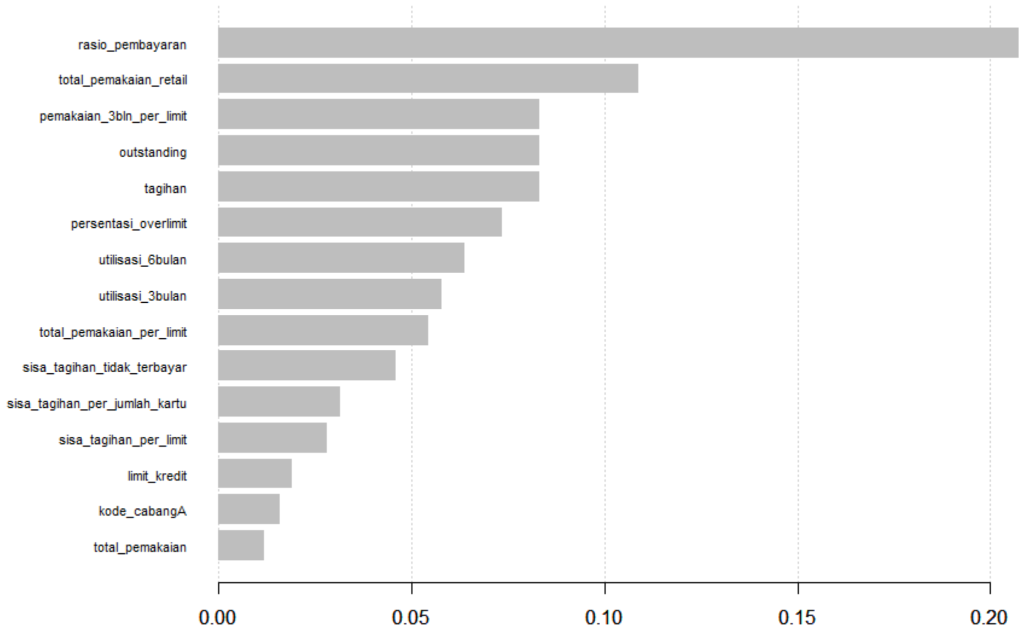
1. Create new data training using SMOTE algorithm
2. convert categorical factor into one-hot encoding
3. construct XGBoost object dengan xgb.DMatrix
4. Construct XGBoost model
5. Predict credit default by using model above
6. Model evaluation

The following is the model comparison:

```
> table_akurasi
          model                 data       auc  accuracy    recall
1       xgboost        train (model) 0.9678323 0.9435028 0.3713504
2       xgboost                 test 0.8444653 0.9181437 0.1102662
3 smote-xgboost smote_train (model) 0.9574799 0.8972019 0.8059611
4 smote-xgboost                 test 0.8241954 0.8643248 0.4524715
```

The smote process slightly reduces the level of accuracy, but can improve recall. By considering that the accuracy is still large, the model with smote will be chosen as the prediction model.

Here the important variables of smote-xgboost model:



The most important variable is still in ratio_pembayaran, it contributes more than 20%.