Credit One

Memo

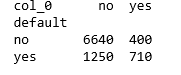
|  |  |
| --- | --- |
| To: | Guido Rossum |
| From: | Yiching Huang |
| Date: | 2018/06/02 |
| Re: | Build and Evaluate Models |

As you requested, this memo presents the result of models that I built and evaluated in order to solve the problem that Credit One faces – an increasing trend in customer default rate. I will detail the key takeaways that I gained from the process and potential business value from this analysis. I will separate the following discussion into different sections based on results.

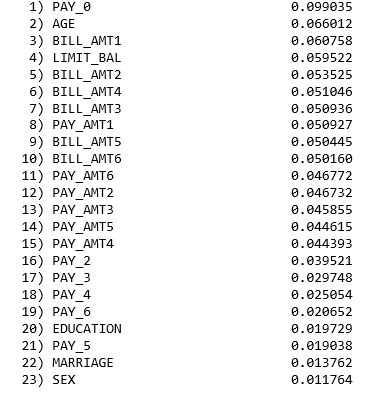
**Build and Evaluate models**

## As always with the pipline process, after we executed cleaning, preparing and EDA process in our last task, we built a model and evaluated the performance to see if the training model can help us to reduce the customer default rate since it is important for Credit One to not approve the loan for customers that have a high chance to default the payment. We tried different models in Python – random forest, support vector machine for classification and KNN. We also used PCA to create another data set to compare with the original one in order to increase the performance metrics.

* We first trained the three models with the original data. We did not include the ID column as an independent variable, and we also recoded the education 0,4,5,6 all into 0 since they all represent others, and we changed the dependent variable from integer (0,1) into object (Yes, No) to better represent customer default or not default. We still split the data set as 70% for training data and 30% for testing data. We also created the data set with the PCA feature engineering method to reduce variables dimensions and see if it can help us find new attributes that better describe our data.
* After trying three models with different tuned parameters with two different data sets, we can see the random forest with the original data set and SVC with the PCA dataset both achieve 81% accuracy rate. We then used these two models to predict the testing data to see which models will have better results. We noticed the random forest model has a slightly higher accuracy than SVC model and helped us get 81.9% accuracy rate. This indicates that the model can predict which customers will default and which will not 81.9% of the time.
* We then looked into the problem of if we can approve customers with high certainty. In order to answer this question, we need to look at the confusion metrics that are generated from our prediction. We can see the model successfully predicts 6,640 customers who will not default on the loan and 710 customers who will default on the loan. It seems that the model helps us get the right prediction most of the time. However, the question we need to focus on is the type error, which are those that will default the loan but our model predicts the customers as no default. This is more serious than our model mistakenly predicts customers who will not default the loan to default since our goal here is to reduce the wrong customer default rate. As we can see in the confusion matrix, there are 1,250 customers whose the loan application will be authorized by Credit One when applying this model. Thus, even our model achieves 81% accuracy, we still need to improve to reduce the type one error.



* Next, we want to investigate how we can ensure that customers will pay their loans. There are no models that are 100% sure for future predictions. All we can do is to raise the model performance and see which attributes take an important role for deciding customer default rate. As we can see in the table below, the top three important variables are PAY\_0, AGE and BILL\_AMT1 and the least important variable is SEX. Thus, we can understand which are the driving factors for our dependent variables allowing us to better observe customers in the future.
* Last, we recommend that even if we train and evaluate the model to help us to achieve high accuracy, we still need to try different models with different parameter combinations to see if there is one that can help us to reduce the Type 1 error for us to retain our partners and the business.



## .