MSBD 5013 Project 1: Home Credit Default Risk

CEN Xinxin, MA Xiaoran and WU Xiang {xcenab, xmabi, xwucb}@connect.ust.hk
Department of Computer Science and Engineering, HKUST

1. Introduction

Providing positive and safe borrowing experience to people who has insufficient or non-existent credit histories is challenging. In our project, we dive into the alternative datasets provided by Home Credit. After feature exploration, data cleansing, model selection, model tuning and performance analysis, we came up with a great solution to this problem. Our best model gained public score=0.77786 and private score=0.77412 on Kaggle.

2. Datasets Aggregation and Data Exploration

We made use of four different datasets in total, since we aimed to depict each applicant with richer information.

- **application_{train|test}.csv:** Static data for all applications. (Main Data)
- **previous_application.csv:** All previous applications of client.
- **credit_card_balance.csv:** Previous monthly balance of credit cards.
- **POS_CASH_balance.csv:** Previous monthly balance of POS and cash loans.

We select these four datasets for certain reasons. We thought previous application history, credit card balance history and POS/Cash loans history are closely related to clients' financial situation, thus would reflect clients' repayment abilities.

Aggregation methods

For the attributes of the three datasets except the main dataset, the most vital attributes are numerical attributes, and each client may correspond to multiple records. Thus, we aggregate these three dataset into the main dataset in the following way: for each client, we found all of his / her record in the dataset, and then try to depict the distribution of each attribute by calculating the maximum, minimum, mean and standard deviation of each numerical attribute, and then merge the resulting table into the main dataset (application.csv) using SK_ID_CURR as the key.

Missing Values & Outliers

There are many missing values in our data, so we tried to delete attributes which have more than 30% missing values, however, the performance dropped. Thus, we think that attributes with many missing values still helps to get better prediction by providing additional valuable information.

The way to deal with missing values depends on the model. For model that accept NAN as input (lightGBM), we left it NAN to avoid confusion to the model. For models such as Logistic Regression, We fill in the missing values by the mean of the attribute.

We also detected several outliers and manually set them to NAN. For example, if DAYS_EMPLOYED=365243, then we it to NAN, since it's unrealistic.

3. Feature Engineering and Selection

Feature Engineering

- ✓ **Numerical Features:** We construct some percentage features. For example, we create AMT_INCOME_TOTAL/CNT_FAM_MEMBERS percentage feature and AMT_INCOME_TOTAL/AMT_CREDIT percentage feature to explore the influence of the average AMT income of each person in a family and the ratio of AMT income to AMT credit on the results.
- ✓ **Categorical Features:** We use different encoding methods for different categorical features. For examples, like CODE_GENDER these kind of features, we use binary encoder with 0/1 to encode them directly. Then, like NAME_EDUCATION_TYPE these hierarchical features, we manually rank the available values of feature and create a mapping dictionary to encode them. Finally, one-hot encoder is used to encode the remaining categorical features.

Feature Selection

- ✓ **Pearson Correlation Analysis**: We expect to use Pearson Correlation Analysis to automatically select features which contribute the most and also reduce the feature dimensions to improve efficiency. So, we select the top 100, 200, 300 and 400 best features to experiment respectively. But the results show that the more features we select, the better results we have, and if we do nothing on the original features, we can get better score than selecting the k best features.
- ✓ **Principal Component Analysis**: Besides, we also use Principal Component Analysis to set components equal to 100, 200, 300 and 400 to experiment respectively. Although it performs better than Pearson Correlation Analysis, the performance of model also dropped compared with without reducing dimensions.

4. Model Construction

We try to use different models to solve this problem.

- **Generalized Linear Model:** Logistic Regression, Support Vector Machine
- **Tree Model:** Random Forest, LightGBM

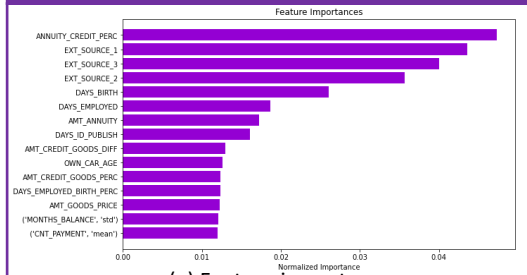
We find Tree Model perform better than Generalized Linear Model, and LightGBM performs best. So we decide to choose LightGBM as our final model. The main idea of LightGBM is to split the feature with largest information gain measured by variance into left and right nodes. The variance gain is defined as:

4. Model Construction (continue)

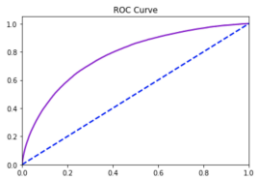
$$\tilde{V}_j(d) = \frac{1}{n}\left(\frac{(\sum_{x_i \in A_l} g_i + \frac{1-a}{b}\sum_{x_i \in B_l} g_i)^2}{n_l^j(d)} + \frac{(\sum_{x_i \in A_r} g_i + \frac{1-a}{b}\sum_{x_i \in B_r} g_i)^2}{n_r^j(d)}\right)$$

Where $\tilde{V}_j(d)$ is variance gain, g_i is the negative gradients of the loss function with respect to the output of the model i in each iteration of gradient boosting.

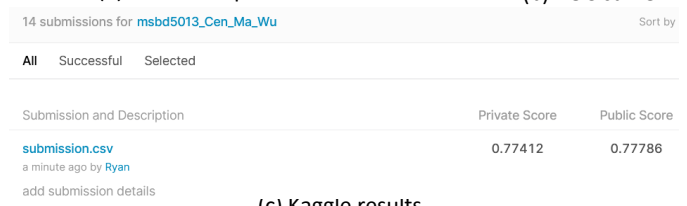
5. Analysis and Conclusion



(a) Feature importance



(b) ROC curve



(c) Kaggle results

- We find that annuity credit and external sources are the most important features for the model. Client's employment days and ages are the second important features.
- Our model's 5-fold overall validation results are 0.777436 and we submit test results to Kaggle. The private and public scores are 0.77412 and 0.77786. They are close to our validation results, which means that our model is robust and has good generalization ability.

6. Contribution

- MA Xiaoran: part 1&2 poster writing and coding
- WU Xiang: part 3 poster writing and coding
- CEN Xinxin: part 4&5 poster writing and coding