MSBD 5013 Project 1: Home Credit Default Risk Prediction

Zhaoyang Deng* and Congjian Chen* {zdengao, cchenci}@connect.ust.hk
*MSc Candidate in Big Data Technology, Deptartment of Computer Science and Engineering, HKUST

1. Introduction and Task Description

Home Credit, a bank that aims to provide safe loans to people, is making use of a variety of data to predict their clients' repayment abilities. They are convinced that the data from not only the current application but also previous ones in both Home Credit and other financial institutions will be helpful.

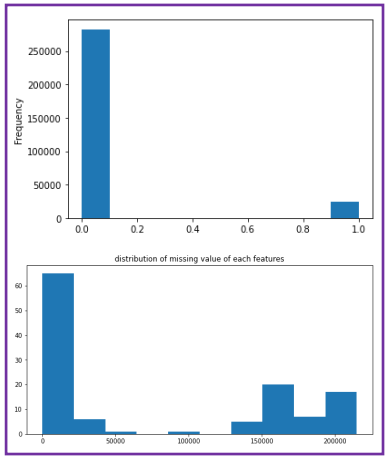
Thus, in this Kaggle competition, we will predict the default risk by the probability for each loan record in the Home Credit Sample. Essentially, this is a regression problem with output between 0 and 1.

2. Dataset

We roughly divide 7 csv files into 4 groups as follows (a detailed relationship graph of tables is available here):

- Main (*application_{train, test}*)
- Bureau (*bureau, bureau_balance*)
- Previous (*previous_application*)
- Personal records (*POS_CASH_balance, installments_payments, credit_card_balance*)

The training and test set contain around 308k and 49k unique samples, respectively.



In the *application_train* csv, the ground truth indicated by the TARGET attribute of 0 or 1 is given for each sample (0 for default and 1 for non-default). The left upper graph shows the distribution of TARGET.

The left lower graph shows the distribution of missing values in different columns. We delete the columns that have missing values with a percentage over 90%, then fill the remaining missing values with 0.

3. Feature Engineering

For the Main group data, since SK_ID_CURR is unique for each row and the meaning of each variable is clear, we simply impletement the one-hot embedding for categorical attributes.

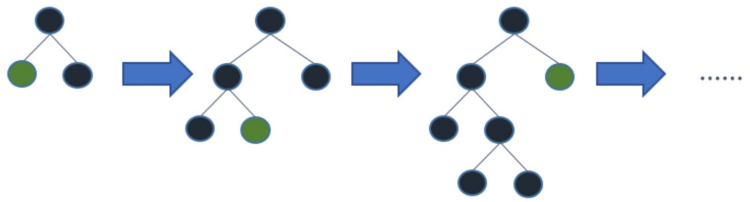
For the other groups of data, features are created on either loan level or client level (achieved by *groupby* function). For the numerical attributes, we exact 5 features: count, mean, max, min and sum. For categorical attributes, after one-hot encoding, we exact 3 features: count, mean and sum.

After that, all features are joined on SK_ID_CURR to create the input table.

4. Models

For this probability-oriented problem, logistic regression is a great choice to create a baseline. Then we focus on tree-based models/gradient boosting frameworks, which are widely considered dominating the Kaggle competitions.

Note that our dataset after feature engineering takes up a huge memory, LightGBM, a fast, distributed, high-performance gradient boosting framework based on decision tree algorithm, would be useful.



In LightGBM, the leaf-wise tree growth pattern (as above) leads to increase in complexity and may lead to overfitting and it can be overcome by specifying another parameter max-depth which specifies the depth to which splitting will occur. Therefore, comparing with other gradient boosting frameworks, it ensures the faster training speed and lower memory usage.

5. Experiment Results

Logistic regression on Main is applied for baseline. LightGBM on different features are applied for improvements 5-fold cross validation and grid search tuning is applied for model selection.

The model is trained on Colab. Although we have reduced the memory usage by converting type of variables and deleting raw data, the RAM restriction (free 12 GB on Colab) still prevents us from training features from all csv files together. Thus, we try several combinations of features.

The AUC in different models are as follows:

Model	Feature	Train	Validation	**Test (Kaggle Score)**
Logistic Regression	Main	0.75346	0.69864	0.68453
LightGBM	Main + Bureau	0.82120	0.76626	0.75661
LightGBM	Main + Previous	0.83245	0.77550	0.76281
LightGBM	Main + Previous + Cash	0.83530	0.77229	**0.76614**

6. Conclusion

The results indicate that Main+Privous+Cash feature combination achieves a best score of 0.76614. Also, it can be inferred that Previous dataset provides better features than the Bureau group, probably resulting from the fact that it contains more financially significant attributes (e.g., annuity data).

7. References

https://www.analyticsvidhya.com/blog/2017/06/which-algorithm-takes-the-crown-light-gbm-vs-xgboost/
https://www.kaggle.com/code/willkoehrsen/start-here-a-gentle-introduction/notebook

8. Contribution

Zhaoyang Deng: Data preprocessing and feature engineering for Main and Bureau group, Modelling
Congjian Chen: Data preprocess and feature engineering for Previous and Personal records group, Poster making