Home Credit Default Risk - P15

Sravan Kumar Matta
Department of Computer Science, NCSU
skmatta@ncsu.edu

Shivaprakash Balasubramanian
Deaprtment of Computer Science, NCSU
sbalas22@ncsu.edu

1 Background and Introduction

A tedious process every loan provider face is to assess the creditworthiness of the applicant. Is the applicant worth the requested amount, will the candidate be able to repay the loan amount and installments in time. Credit score of the applicant might give an overview of the attitude of applicant towards borrowed credit but not a complete picture and the candidate's ability to pay bigger amounts. The problem becomes even worse if the applicant has little or no credit history at all, it becomes extremely difficult to evaluate the candidate and vice versa, many worthy candidates struggle a lot to secure loans because of less or no credit history and are often exploited by untrustworthy lenders.

In [2] the author introduces an effective prediction model for predicting the credible customers who have applied for bank loan. To gain insight into the credibility of the user, a decision tree based model is used. This prototype model can be used to sanction the loan request of the customers or not. The model proposed in [3] has been built using data from banking sector to predict the status of loans. This model uses three classification algorithms namely j48, bayesNet and naive Bayes. The model is implemented and verified using Weka. The best algorithm j48 was selected based on accuracy. An improved Risk prediction clustering Algorithm that is Multi-dimensional is implemented in [4] to determine bad loan applicants. In this work, the Primary and Secondary Levels of Risk assessments are used and to avoid redundancy, Association Rule is integrated. The proposed method predicts with better accuracy and consumes less time than previous methods. The work in [5] proposes two credit scoring models using data mining techniques to support loan decisions for the Jordanian commercial banks. Considering the rate of accuracy, the results indicate that the logistic regression model performed better than the radial basis function model. The work in [6] builds several non-parametric credit scoring models. These are based on the multi layer perceptron approach. The work benchmarks their performance against other models which applies the traditional linear discriminant analysis, logistic regression and quadratic discriminant analysis techniques. The results show that the neural network model outperforms the other three techniques. This survey comparison was completely based on [8].

The main goal of this project is to develop a loan approval model to predict the creditworthiness of the applicants who have very less, or non-existent credit histories based on the available demographic data and other factors. This helps the financial institutions save a lot of time in assessing the creditworthiness of the applicants, avoid losses and incur huge profits and broaden their target audience by addressing the unbanked population, people with less credit history.

The data we used for developing this model is provided by Home Credit. Home Credit is a financial institution which provides loans of various types like personal loans, home appliances, automobiles, mobile phones, laptops etc.

Problem Description and Data Set

Project Link: Home Credit Risk

2 Method

For the applicant data we have to predict if they will be able to repay the loan or not. The is basically a binary classification problem.

The first step is data exploration and we went through all the csv files to figure out how much information we have about the applicant and did a rough analysis of what the useful features will be for this classification. This data set consists of 307511 rows and 122 columns, with a target value of 0 or 1 which basically identifies if the loan will be payable or not. One of the first things we observed was that the data was sparse in some columns and contained a lot of missing values.

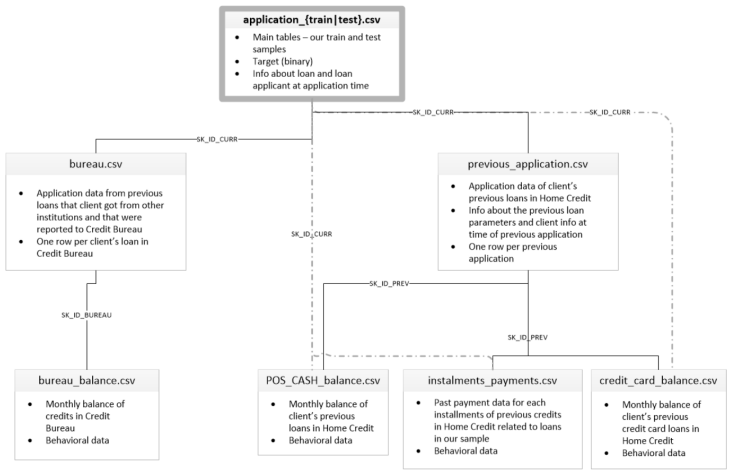


Figure 1: Overview of Data

2.1 Data Exploration

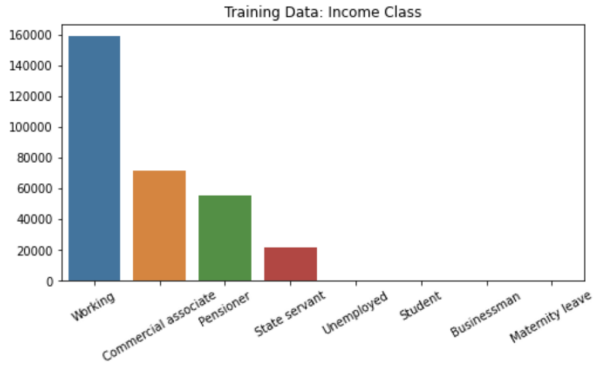
1. Application: Static data for all applications. One row represents one loan in our data sample.Each row is a loan data which gives information about the applicant and also if the loan will be repaid or not.This is the main table, broken into two files for Train (with TARGET) and Test (without TARGET).
2. Bureau: In this data file, we have information about the client's previous credit history from all financial institutions they are associated with.
3. Bureau Balance: It consists of monthly data about the previous credits in the bureau.This table has one row for each month of history of every previous credit reported to Credit Bureau.
4. Previous Application: The data of previous applications for loans at Home Credit of clients who have loans in the application data. Each current loan in the application data can have multiple previous loans.
5. POS Cash Balance: Monthly data about Cash loans with home credit. This table has one row for each month of history of every previous credit in Home Credit (consumer credit and cash loans) related to loans in our sample.
6. Credit Card Balance: The monthly data about previous credit cards clients have had with Home Credit. This table has one row for each month of history of every previous credit in Home Credit (consumer credit and cash loans) related to loans in our sample.
7. Installments Payment: The data of payment history for previous loans at Home Credit. There is a) one row for every payment that was made plus b) one row each for missed payment. One row is equivalent to one payment of one installment OR one installment corresponding to one payment of one previous Home Credit credit related to loans in our sample.

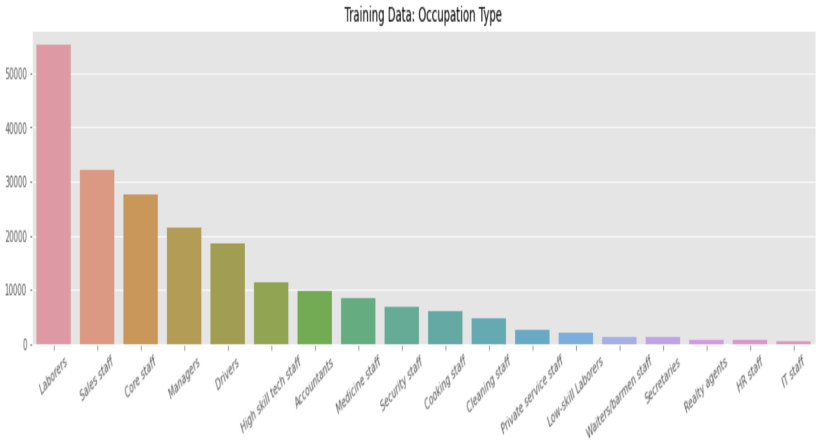
2.2 Data Analysis

After doing a primary data analysis and prepossessing, we imputed the missing values with median values and standardized the data before feeding it to the model. we checked how the data is distributed to choose a corresponding model. The result indicated that the data is imbalanced with a lot of data pointing to loans being repaid. This has to be taken care of since skewness in data can lead to bad interpretations and results. We used log transformation to make the data normally distributed. Some other observations we made were that people who are taking credit for large amounts are very likely to repay the loan.

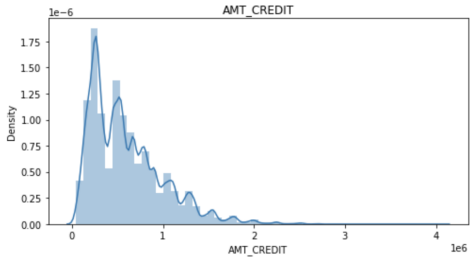
Since we decided to use a dense neural network as one of the models, it was necessary to balance out the skewness. To get more samples of the 1 target class, we used SMOTE to over sample and balance the skewness.

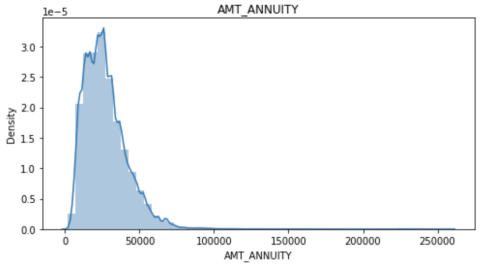
We believe income source is a key contributor to figuring out if the loan is repaid or not. When we plotted the chart to see the same, we figured that students and businessmen are more likely to repay the loan compared to other groups.





We observe the distributions of credit and annuity, which are primary factors in deciding the validity of the loan credibility score. An annuity is a contract between you and an insurance company in which you make a lump-sum payment or series of payments and, in return, receive regular disbursements, beginning either immediately or at some point in the future. Since these are continuous variables, we analyse the distribution and observe both of them are left skewed. This gives us insight that people who are taking credit for large amount are very likely to repay the loan. Since the distribution is skewed, we can log transformation to make it normal distributed to obtain better results while training.





2.3 Feature Engineering

SK-ID-CURR is the foreign key that holds all the tables in the given data. The raw application data contains 122 columns without considering the foreign tables and not all the features directly attribute to the goal of the project. Hence, we have analyzed the data and modified it to generate the best possible results. The raw columns by themselves do not add much importance and we have engineered some new metrics to help improve the model.

1. credit income percent
2. annuity income percent
3. credit term
4. days employed percent
5. number of previous loans from each applicant
6. different types of loans of each applicant
7. debt over credit ratio
8. overdue over debt ratio
9. raw number of previous applications
10. length of credit
11. total amount in debt
12. total credit amount
13. average term of previous application
14. average installment payment
15. average number of installments left
16. number of active credit accounts

2.4 Model Selection

Since we are using 2 models, we decided to use one standard machine learning model and a neural network and compare the performances of the same.

2.4.1 LightGBM

Light GBM is a fast, distributed, high-performance gradient boosting framework based on the decision tree algorithm, used for ranking, classification and many other machine learning tasks. Since it is based on decision tree algorithms, it splits the tree leaf wise with the best fit whereas other boosting algorithms split the tree depth wise or level wise rather than leaf-wise. So when growing on the same leaf in Light GBM, the leaf-wise algorithm can reduce more loss than the level-wise algorithm.

LightGBM is a gradient boosting framework that uses tree based learning algorithms. It is designed to be distributed and efficient with the following advantages:

1. Faster training speed and higher efficiency
2. Lower memory usage
3. Better accuracy
4. Support of parallel and GPU learning
5. Capable of handling large-scale data

2.4.2 Neural Network

The second model we use is a simple neural network with standardization, categorical encoding and building the tensors that will feed the neural network. The function and its derivative both are monotonic. Softmax is a very interesting activation function because it not only maps our output to a [0,1] range but also maps each output in such a way that the total sum is 1. The output of Softmax is therefore a probability distribution.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 31)                992
_________________________________________________________________
dense_1 (Dense)              (None, 100)               3200
_________________________________________________________________
dense_2 (Dense)              (None, 64)                6464
_________________________________________________________________
dense_3 (Dense)              (None, 2)                 130
=================================================================
Total params: 10,786
Trainable params: 10,786
Non-trainable params: 0
_________________________________________________________________
```

We built a fully connected network and fine tuned the hyper-parameters, the number of layers and the optimizer for the gradient by trying out on different values using grid search. Grid search is a method used to find the optimal parameters. It extensively searches through a manually specified hyper-parameter values by considering all possible combinations, evaluates the model for each combination and select the best set of hyper-parameter values based on the performance metric used. We got the best outputs when we used a neural net with 4 dense layers all of which used relu activation functions except for the final layer which used softmax.

As mentioned before, the data set contains an imbalanced distribution of data and we used SMOTE to over fit the 0 target class to ensure better classification results.

3 Experimental Setup

Both the models are being developed in python. We have used pandas to read the given data as data frames for easy and efficient processing of the data, matplotlib and seaborn libraries for the visualizations and presenting the insights from data in a more concise and understandable way. Numpy library for multidimensional mathematical calculations and pickle library to save the trained model for future predictions. We have used scikit-learn library for building the lightGBM model and tensorflow keras libraries for building the neural network.

3.1 Evaluating Models

Since it's a binary classification problem we can calculate the confusion matrix of both the model and compare different metrics like accuracy, precision, recall, f1- Score. The problem is to figure out bad loans and the primary goal should be to reduce the false positives because these are the loans if not flagged might turn bad. As mentioned before, the data is imbalanced by a huge margin with 91.9 percent of applicants belonging to class 0 (Loan repaid) and the rest belonging to class 1 (Loan not repaid) Accuracy won't be a great measure to the performance of both the models. To better compare the models, we will use False Positives as our primary driver as they have a higher cost to the banks in the given scenario. We also also use log loss and ROC curve metrics to handle the imbalance in data.

3.2 Hypothesis

We have completed the exploratory data analysis, data cleaning and data wrangling, relevant data has been aggregated from all the data files for each applicant to prepare the data to feed to the model. The primary goal of the model is to predict if the loan is repayable by the candidate or not. Apart from that, we also tried to find the impact of the features engineered against the target variable and validated the following hypothesis.

1. The data is highly imbalanced with 91.9 percent belonging to class Loan repaid (class value = 0) and the rest 8.07 percent belonging to class Loan not repaid (class value = 1)
2. People with high income values (Approximately greater than 1 million) are likely to repay the loan.
3. People who have academic degree are more likely to pay loans.
4. People who has more work experience are more likely to repay the loans.
5. There are only two types of loans, cash loan and revolving loans and the number of people who requested cash loans are a lot than revolving loans.
6. People who take loans in large amounts are likely to repay the loan than people who take small loan amounts.

4 Results

It is clear see that the LGBM model outperforms the Neural net model based on the number of misclassified points and the false positive rate (0.095 vs 0.201). For the given classification problem, it is crucial that loans aren't handed out to candidates who won't be able to repay them. Hence, the LBGM model is a clear choice for Home Credit organization.

```
The percentage of misclassified points : 20.081362441883964
[[76177 22887]
 [16850 81966]]
```

Figure 4: Neural Net Confusion Matrix

```
The percentage of misclassified points : 5.263796240145546
[[98056  1008]
 [ 9408 89408]]
```

Figure 5: LGBM Confusion Matrix

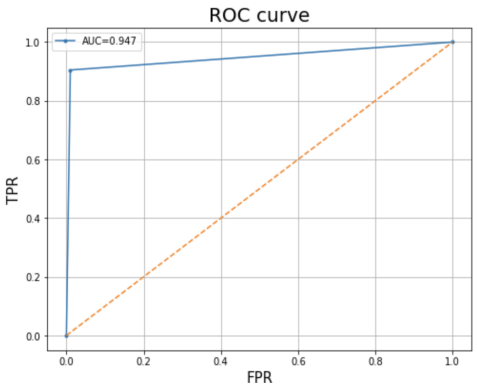


Figure 6: ROC curve for LGBM model

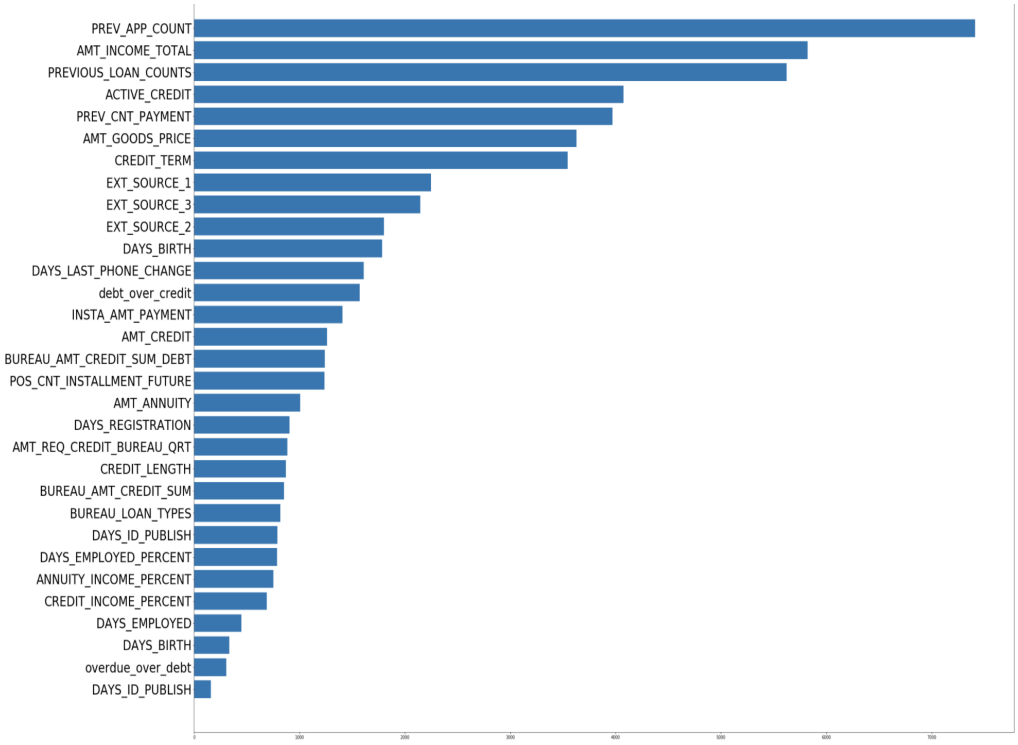


Figure 7: Importance of the different Features Engineered

5 CONCLUSION

We have come up with a model which better assess candidates with as minimal possible errors and compare the performance of a Light Gradient Boosting Machine to that of a neural network, which model is efficiently able to capture patterns in a highly structured financial data and draw conclusions as to which kind of model is better for problems like these and why.

As a part of the project we learned the importance of domain knowledge. We initially skipped past the feature engineering step and used just the raw features because we didn't have much idea about financial terms and got poor results. We later took a step back to understand the data set and features better and got much better results. Another improvement that we did in the end was trying to fit the model to a simple logistic regression model which provided very similar results to the LGBM model which stands a testament to the fact that model complexity does not always guarantee the best results. Also balancing the target variables proved to be crucial in minimizing the false positives.

References

[1] https://www.kaggle.com/c/home-credit-default-risk/data

[2] M. Sudhakar, and C.V.K. Reddy, "Two Step Credit Risk Assessment Model For Retail Bank Loan Applications Using Decision Tree Data Mining Technique", International Journal of Advanced Research in Computer Engineering Technology (IJARCET), vol. 5(3), pp. 705-718, 2016.

[3] J. H. Aboobyda, and M.A. Tarig, "Developing Prediction Model Of Loan Risk In Banks Using Data Mining", Machine Learning and Applications: An International Journal (MLAIJ), vol. 3(1), pp. 1–9, 2016.

[4] K. Kavitha, "Clustering Loan Applicants based on Risk Percentage using K-Means Clustering Techniques", International Journal of Advanced Research in Computer Science and Software Engineering, vol. 6(2), pp. 162–166, 2016.

[5] A.B. Hussain, and F.K.E. Shorouq, "Credit risk assessment model for Jordanian commercial banks: Neuralscoring approach", Review of Development Finance, Elsevier, vol. 4, pp. 20–28, 2014.

[6] https://nycdatascience.com/blog/student-works/prediction-of-credit-default-risk/

[7] https://www.sciencedirect.com/science/article/abs/pii/037842669290016S

[8] https://www.researchgate.net/publication/309626126-Credit-Risk-Analysis-and-Prediction-Modelling-of-Bank-Loans-Using-R

[9] https://medium.com/@praveenkotha/home-credit-default-risk-end-to-end-machine-learning-project

[10] https://medium.com/@himanshuxd/activation-functions-sigmoid-relu-leaky-relu-and-softmax-basics-for-neural-networks-and-deep-8d9c70eed91e