Predicting Mortgage Approvals From Government Data

Hwang Ching, April 2019

Executive Summary

This report presents an analysis of data concerning mortgage approvals from government data. The analysis is based on 500,000 observations of application data, each containing specific characteristics of applicant relative background.

After exploring the data, finally a stacked model to predict whether the application will be approved or not from the data we got.

After performing the analysis, it can be conclued the following conclusions:
While many factors can help indicate the accept probabilty, and significant features found in this analysis were: ‘ loan_amount’, ‘applicant_income’, ‘minority_population_pct’, ‘tract_to_msa_md_income_pct’ ,’ loan_purpose’, ’preapproval', ‘applicant_race’, ‘county_code', and ‘lender’.

Data Exploration

The final goal of this competition is to predict whether a mortgage application was accepted or denied according to the given dataset. The number of Training dataset and testing dataset were 500,000 rows respectively. There are 21 variables in this dataset . We can divide the features to two parts : categorical type and continuous type.

categorical type:

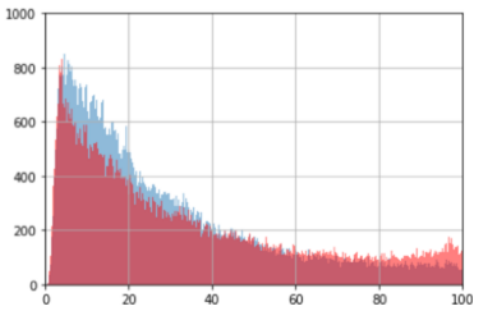
- msa_md
- state_code
- county_code
- lender
- loan_type
- property_type
- loan_purpose
- occupancy
- preapproval
- applicant_sex
- applicant_race
- applicant_ethnicity

continuous type:

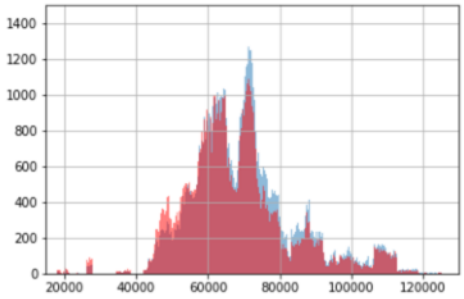
- loan_amount - Size of the requested loan in thousands of dollars
- applicant_income
- population
- minority_population_pct
- ffiecmedian_family_income

• tract_to_msa_md_income_pct
• number_of_owner-occupied_units
• number_of_1_to_4_family_units

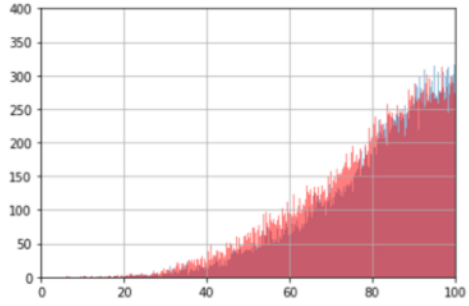
First, I focus on the continuous features. Each histogram for continuous feature shows below.It shows the distribution bases on whether the mortgage application was accepted or not for each features.



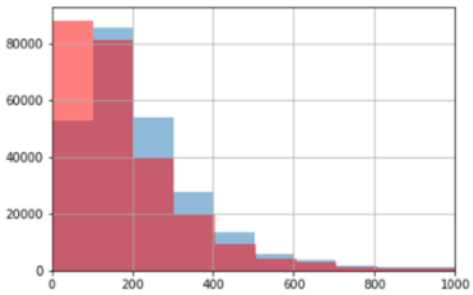
minority_population_pct



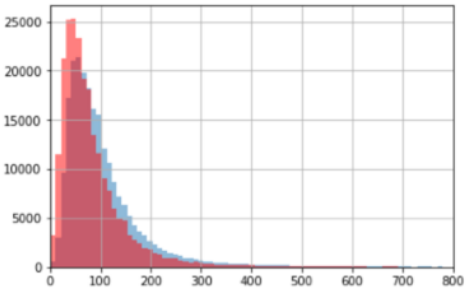
ffiecmedian_family_income



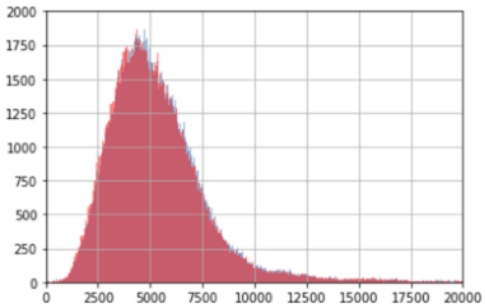
tract_to_msa_md_income_pct



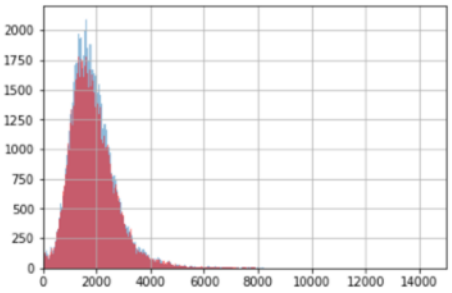
loan_amount



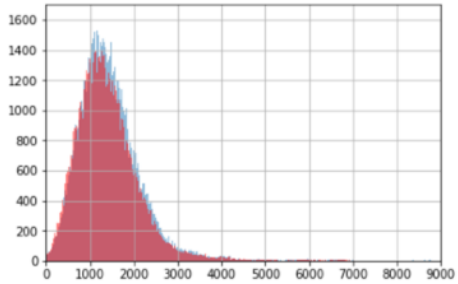
applicant_income



population



number_of_1_to_4_family_units



number_of_owner-occupied_units

In the histogram, red represents the distribution of the application which was accepted, on the other hand, blue represents those was not accepted. However, it can be found that there are significant difference between two colors in the features like “ loan_amount“, ”applicant_income”, “minority_population_pct”,and ”tract_to_msa_md_income_pct”.

After having explored the relationship between mortgage permission stauts and numeric features, an attempt was made to discern any apparent relationship between categorical feature values and mortgage permission stauts.

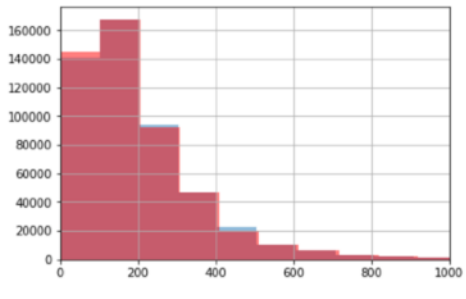
However, The bar charts show some clear different between the mortgage was accepted or not for different categorical features. we can observe the different between the mortgage was accepted or not.For example: features like 'loan_purpose','preapproval', 'applicant_race', 'county_code', and 'lender' have more obvious distinct distribution.

After looking into features, I choose ' loan_amount', 'applicant_income', 'minority_population_pct', 'tract_to_msa_md_income_pct' ,' loan_purpose', 'preapproval', 'applicant_race', 'county_code', and 'lender' as the key features during this analysis.

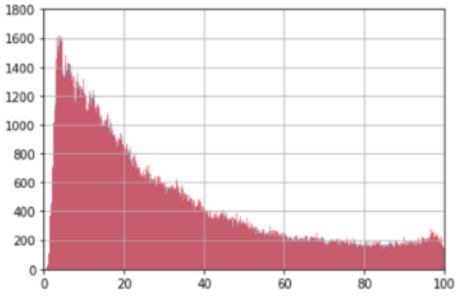
Feauture Engineering

Train Data and Test Data

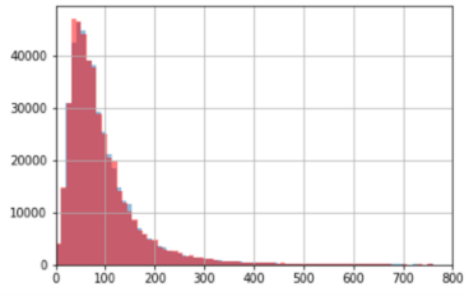
The different distribution between the train data and test data may have significant influrence on predicting data. Therefore,we should take a look into those key features before doing feautre engineering. In the following histogram, we can find out there aren't obvious difference between them.



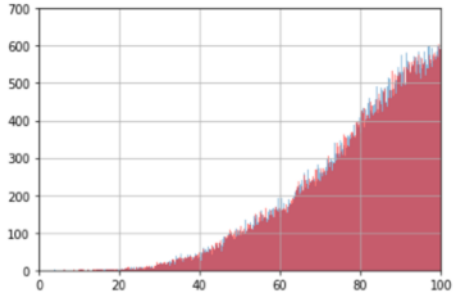
loan_amount



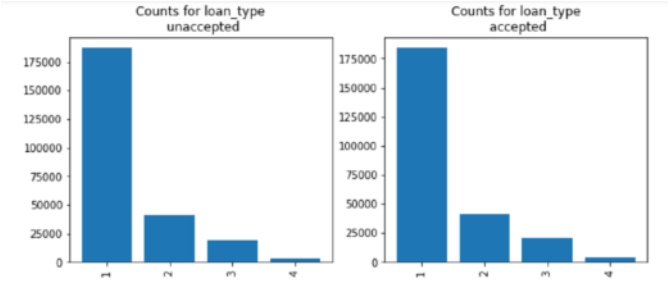
minority_population_pct



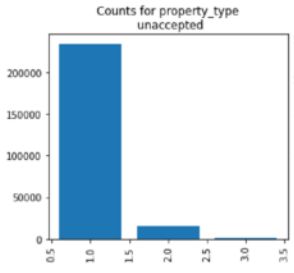
applicant_income

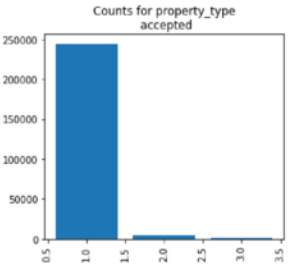


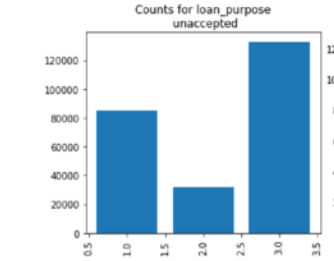
tract_to_msa_md_income_pct

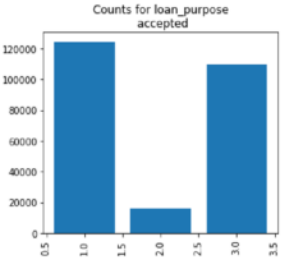

Counts for loan_type
unaccepted

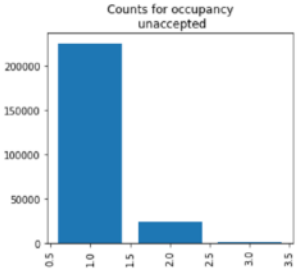
Counts for loan_type
accepted

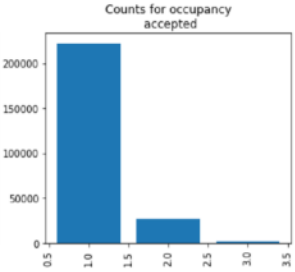

Counts for property_type
unaccepted

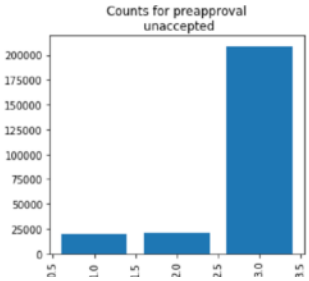

Counts for property_type
accepted

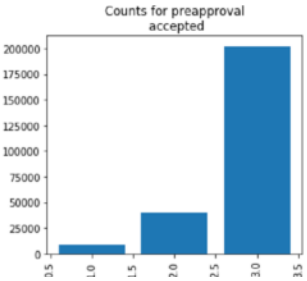

Counts for loan_purpose
unaccepted

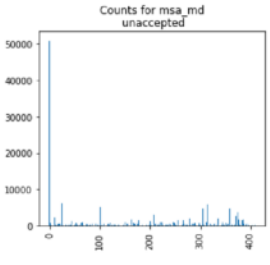

Counts for loan_purpose
accepted

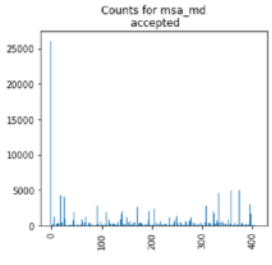

Counts for occupancy
unaccepted

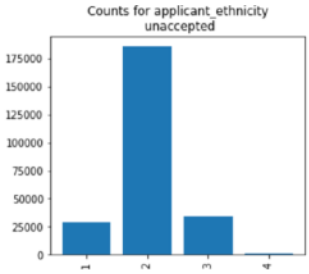

Counts for occupancy
accepted

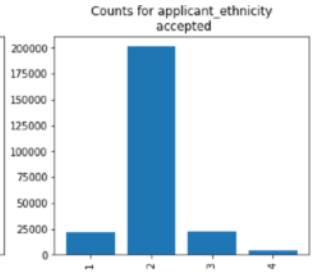

Counts for preapproval
unaccepted

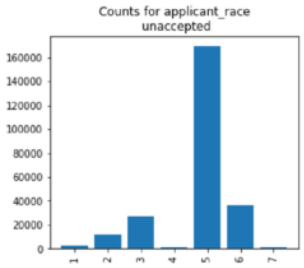

Counts for preapproval
accepted

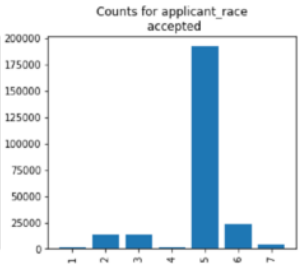

Counts for msa_md
unaccepted


Counts for msa_md
accepted


Counts for applicant_ethnicity
unaccepted

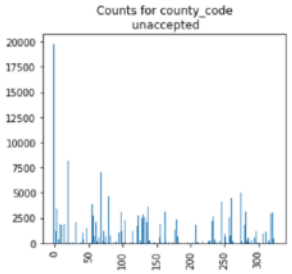

Counts for applicant_ethnicity
accepted

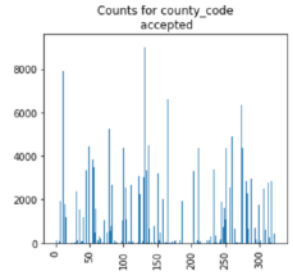

Counts for applicant_race
unaccepted

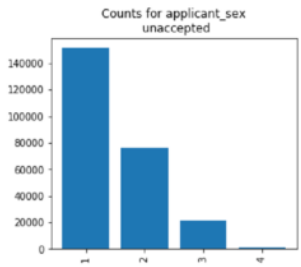

Counts for applicant_race
accepted

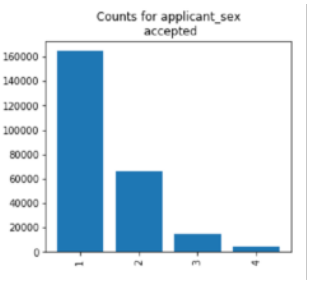

Counts for state_code
unaccepted

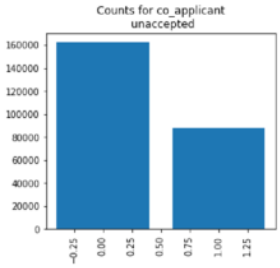

Counts for state_code
accepted

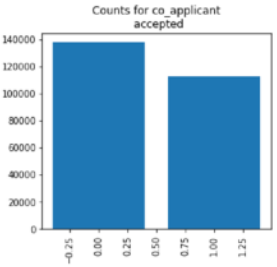

Counts for county_code
unaccepted


Counts for county_code
accepted

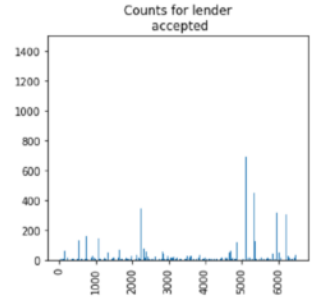

Counts for applicant_sex
unaccepted


Counts for applicant_sex
accepted


Counts for co_applicant
unaccepted


Counts for co_applicant
accepted


Counts for lender
unaccepted

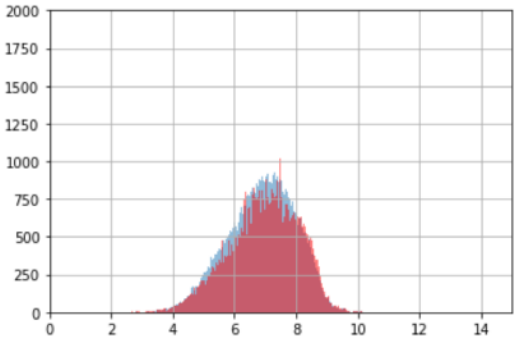

Counts for lender
accepted

Deal With Missing Values

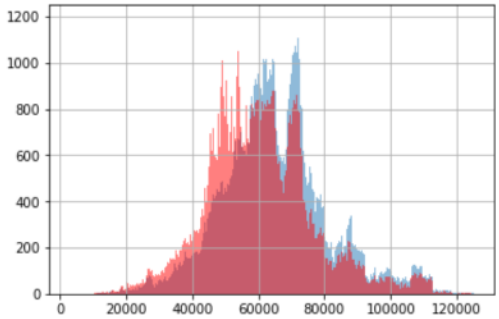
Secondly, when working with data, you will often encounter missing values in data-set. How you deal with them can be crucial for your analysis and the conclusion you will draw. In this analysis I used a non-parametric algorithm called k-nearest-neighbors (KNN) to replace missing values. KNN is an algorithm that is useful for matching a point with its closest k neighbors in a multi-dimensional space. It can be used for data that are continuous, discrete, ordinal and categorical which makes it particularly useful for dealing with all kind of missing data. The missing value can be approximated by the values of the points that are closest to it, based on other variables.

Variables Transformation

Applying statistical measures across this data set may not give desired result. Data transformation comes to our aid in such situations.In those key features I chose, some of them had too much values , for example, there are over 300 different county code and over 5000 thousand lender number. Thus , I did some transformation for those features. I caculated accept probability of each county and lender. After that, I just turn the county code (and lender) to a number between 0 and 1. Therefore, normalization or scaling refers to bringing all the columns into same range. In this competitoin , I use Min-Max normalization to keep the columns in the same scale. Besides , I also did some calculation to get ‘tract median family income’ (ffiecmedian_family_income multiplied by tract_to_msa_md_income_pct)and ‘minority population’ (minority_population_pct multiplied by population). In the next session, you can see the improvement after feature engineering.According those histogram I mention before, We can find out some of them are right-skewed. The log transformation can be used to make highly skewed distributions less skewed. This can be valuable both for making patterns in the data more interpretable and for helping to discover the correlation.



minority population_log



tract median family income_log

Modeling

The main goal of this competition is to predict to mortgage application will be accepted or not, so it is a binary classification which will with two possible outcomes. The following algorithm are what I tried to apply into my prediction model.

1.Logistic Regression : it is designed for this purpose (classification), and is most useful for understanding the influence of several independent variables on a single outcome variable.

2.K-Nearest Neighbours: it is simple to implement, robust to noisy training data, and effective if training data is large.

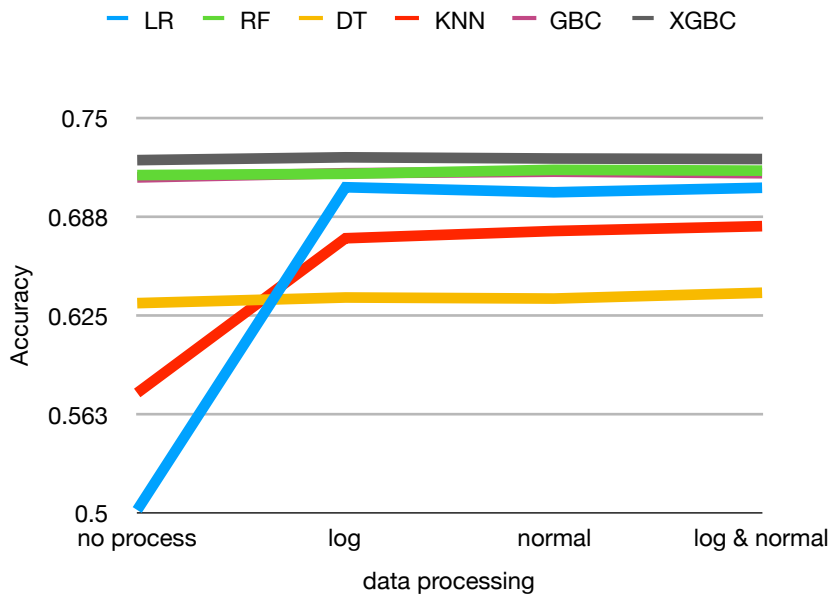
3.Decision Tree: it is simple to understand and visualise, requires little data preparation, and can handle both numerical and categorical data.

4.Random Forest: reduction in over-fitting and random forest classifier is more accurate than decision trees in most cases.

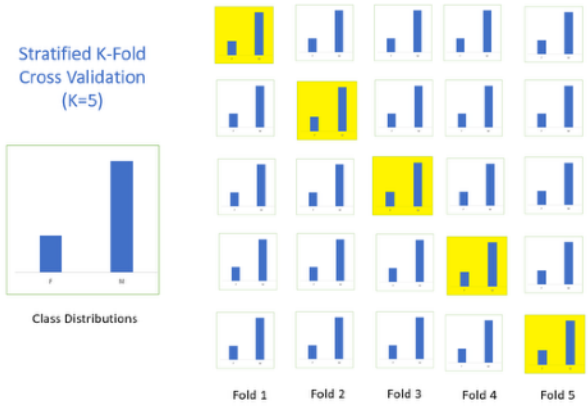
5.Gradient Boosting: basically GBM can be used to solve almost all objective function that we can write gradient out. This including things like ranking and poission regression, which Random Forest is harder to achieve.

6.XGBoosting: it is easily interpretable, relatively fast to construct, and it can naturally deal with both continuous and categorical data.

During all submissions, I tested six model in different stages: data without specific process ; with log-transformation ; with normalization but no log-transformation; with normalization and log-transformation. The whole process in following line chart :



Besides, overfitting is a normal problem you may faced when handling the data prediction. it means the prediction of an analysis that corresponds too closely or exactly to a particular set of data, and may therefore fail to fit other data. To solve this problem, I used 5-fold Cross-Validation to improve my model performance.



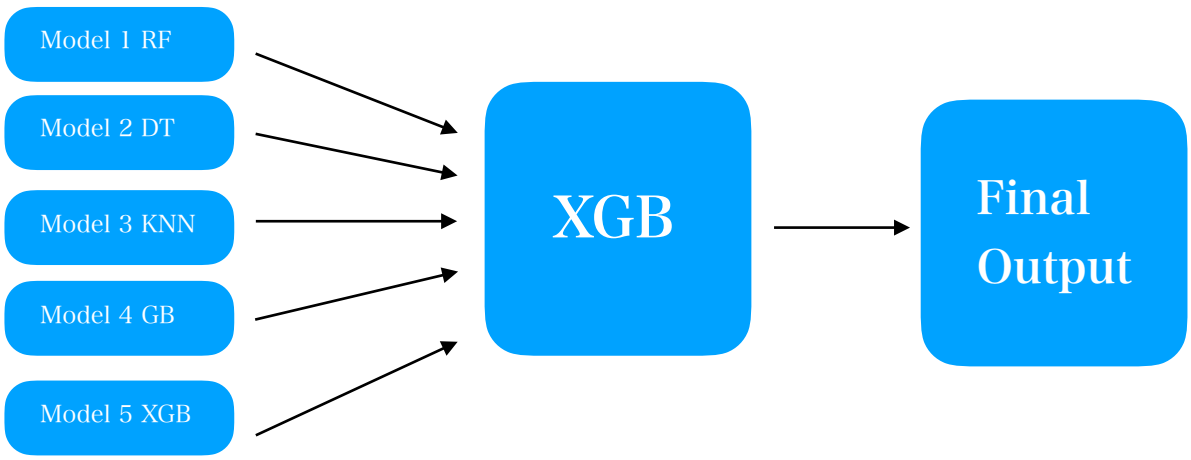
source:https://s3-ap-south-1.amazonaws.com/av-blog-media/wp-content/uploads/2015/11/skfold.png

In the following table, you can find out the cross-validation score in each model.

5-fold Cross-Validation score				
model	**no process**	**log**	**normal**	**log & normal**
LogisticRegression	0.500228	0.705464	0.701530	0.706876
RandomForestClassifier	0.713072	0.716322	0.716784	0.716372
DecisionTreeClassifier	0.635852	0.637178	0.637124	0.636794
KNeighborsClassifier	0.578375	0.674888	0.678516	0.681658
GradientBoostingClassifier	0.714114	0.715426	0.715594	0.715594
XGBClassifier	0.723540	0.724838	0.725168	0.725168

As you can see, XGBClassifier is the model which got highest accuracy score and cross-validation score.

Beside using single model, I also try to ensemble (also called stacking) five different model to combine information from multiple predictive models to generate a new model. It will outperform each of the individual models due its smoothing nature and ability to highlight each base model where it performs best and discredit each base model where it performs poorly. As a result, I combined RandomForest, DecisionTree, KNeighbors, GradientBoosting and XGB five models and use their output as the input of second layer model to predict the final prediction. Finally , I got the best submission score 0.7257 (best ranking 8th).



Conclusion

This analysis has shown that the accept probability can be confidently predicted from its characteristics. In particular, ‘ loan_amount’, ‘applicant_income’, ‘minority_population_pct’, ‘tract_to_msa_md_income_pct’ ,’ loan_purpose’, ’preapproval', ‘applicant_race’, ‘county_code', and ‘lender’ have a great effect on the prediction. Besides, the feature engineering also played an important role in the analysis, log-transformation and normalization can enhanced the accuracy during the prediction. However, if I want to get better use in model selection, I should get deeper understanding in each model. Using model stacking will outperform each of the individual models due its smoothing nature and ability to highlight each base model where it performs best and discredit each base model where it performs poorly. It also can make performance better.