Home Credit Default Risk

Predicting how likely each applicant is of repaying a loan?

Agenda

- Business problem
- Approach
- Data / Data wrangling
- Exploratory data analysis
- Predictive modeling
- Conclusion
- Future Scope of work

Business Problem

Home Credit is an organization that serves the unbanked population with access to loans. Such individuals that do not have a built-up credit score have a challenging time securing loans from financial institutions.

Home Credit strives to broaden financial inclusion for the unbanked population by providing a positive and safe borrowing experience. In order to make sure this underserved population has a positive loan experience; Home Credit makes use of a variety of alternative data--including telco and transactional information--to predict their clients' repayment abilities

So, problem statement would be predicting how likely each applicant is of repaying a loan?

Approach : Uses supervised machine learning to classify one of two categories.

Supervised machine learning is a phenomenon where the ML learns from the data without anyone writing explicit code logic.

Target Labels

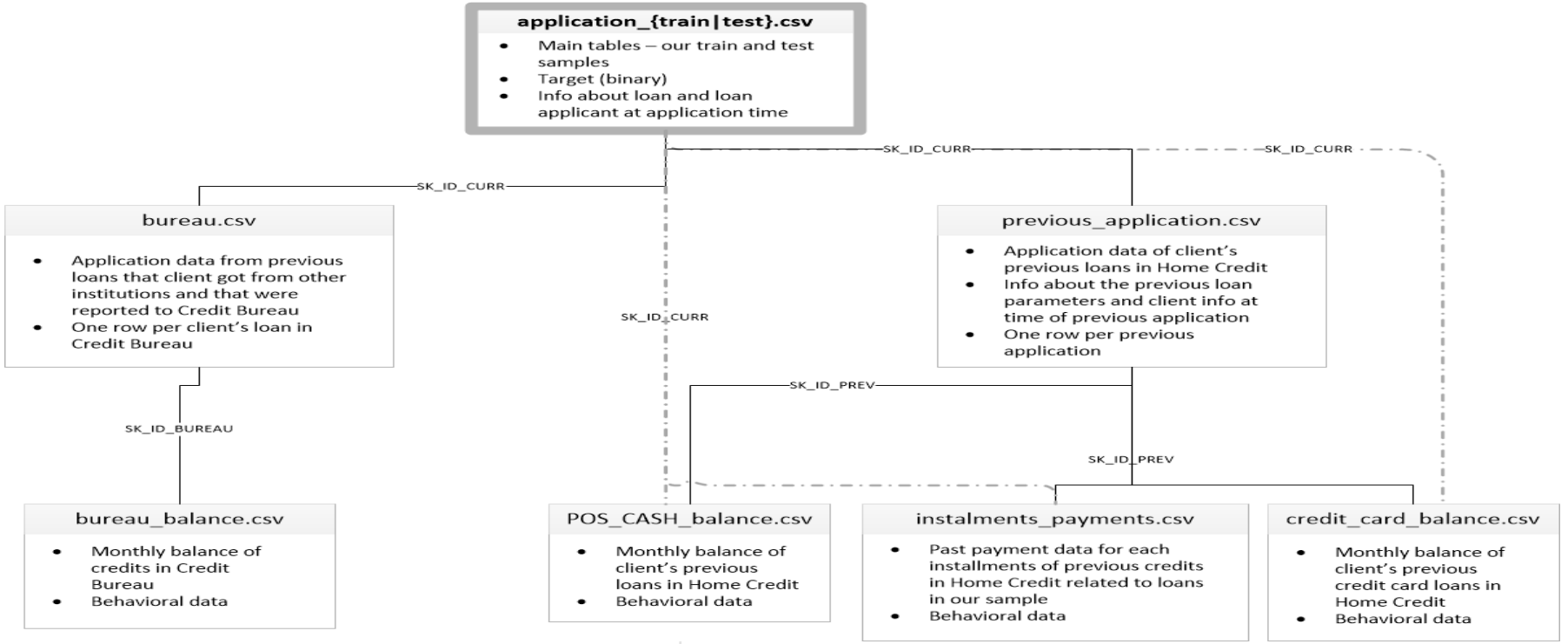
0

Target label of 0 indicates that there was no difficulty in repaying the loan on time.

1

Target label of 1 indicates that there was difficulty in repaying the loan on time.

Data Wrangling : Source - Kaggle



application_train.csv

- 307511 Records, 122 Columns
- Imbalanced Target Labels.
- Source for training machine Learning models.
- Target Labels :
- 0's – 282686, 1's - 24825

application_test.csv

- 48744 Records, 121 Columns
- No Target Labels.
- Source for testing the performance of machine Learning models.
- Target Labels :
- None – need to predict.

Data Wrangling : Outlier Handling & Encoding

Outlier Handling:

- Have not gone through complete data set columns, so looked at few important columns
- DOB has not any outliers
- Days Employed has outlier, it’s been handling after divided by -365 then > 0’s will be marked & masked as 0.
- Negative values being identified &removed

Categorical variables encoding Technique:

- OrdinalEncoder from sklearn being used for conversion

EDA – Resampling Techniques & Feature Selection

Resampling technique:

- Application train has imbalanced data
- Has more repaid history & lower non repaid record
- We need to down sample our datasets prior modeling
- We have applied sklearn.resample for down sampling our dataset

Feature Selection:

- We have more than 150 features in dataset, so we going to pick relevant features for predicting the loan repayment
- **Mutual Info Classifier & K best** has been used to identify the Best correlated feature which matches with TARGET variable based on fs score
- Top 20 features identified based on the technique

Predictive Modeling

Outcome of the model is expected to predicting how likely each applicant is of repaying a loan?

Expected Target Outcome: 0 or 1

0 – Not a defaulter, 1 – potential defaulter.

Performance Metrics used : ROC-AUC Score .

Models currently used : Logistic regression, Random forest, XGBoost, Naïve bayes, Model ensemble & Neural Network.

Predictive Modeling
Splitting Test & Train & Pre-Processing steps.

- Out of the main training dataset, a certain percentage is kept untrained to test the model's performance.
- Training set and validation set are split in following percentages: 66.66% : 33.33%.
- On the testing set, the target labels are hidden, until the performance is evaluated.

Preprocessing data set before creating ML modeling:

1. Imputer – Added SimpleImputer & strategy to calculate ''Median" for the set, transformed data set.
2. Scaler – Added MinMaxScaler for all feature value range from 0-1 for computation

Predictive Modeling – AUC ROC Performance Prior Hyper tuning

Model Name	Accuracy
LogisticRegression	*0.6658108026204868*
RandomForestClassifier	*0.6692317960672662*
NaïveBayesClassifier	*0.6500302040198895*
XGBClassifier	*0.6646329460239638*
Ensemble Stacked	*0.6756984867435408*
Deep Learning -Keras	*0.6600490746931655*

More Explanation in Notebook.

Predictive Modeling –Hyper tuning Techniques

XG Boost – Bayes Search CV

Used Bayes Search CV for Hyper tuning XG Boost

- ***Best ROC-AUC: 0.744***

- Best params:
OrderedDict([('colsample_bylevel', 0.801557907191101 4), ('colsample_bytree', 0.4436488945765141 3), ('gamma', 3.811128976537413e-05), ('learning_rate', 0.270039020618534 2), ('max_delta_step', 18), ('max_depth', 36), ('min_child_weight', 2), ('n_estimators', 83), ('reg_alpha', 1.505756025547201 8e-06), ('reg_lambda', 659), ('scale_pos_weight', 256), ('subsample', 0.883566582389917 7)])

Observation: After Hyper Tuning XGB ROC-AUC Score improved from 0.666 to 0.731

Conclusion

Bayesian Optimization was used to tune the XG Boost Classification models. Subsequently, a kfold cross validation with 8 splits was conducted to evaluate the validity of the models.

The performance improved with roc_auc of 0.731

This is a significant improvement on the early models that did not include as many variables and only relied on the mean groupings of past financial transactions.

It can therefore be concluded that the new variables significantly help in the identification of features that might make someone more likely to have challenges in repaying a loan.

Feature Scope of Work

- We made prediction with 10-15 % of features, if we make prediction with all features together, we may get better results.
- We used ordinal encoder for the categorical conversion for all algorithms, if we use specific encoder for specific algorithms, we may get better accuracy.
- In Neural Network keras model we may try with Embedded layer for better learning & accuracy improvement