DS-GA 1001 Final Project

Credit Default Risks from Loans

Ziyi Xie (zx1153), Bella Lyu(hl4229), Xuhua Luo(xl3583), Binfeng Xu(bx2010)

Introduction to Data Science, New York University

December 3, 2020

Business Understanding

With the development of society, the demands of people for using money in many different fields is rapidly increasing. Most of the time, the size of the opportunity is extremely large since almost everyone needs money and a good credit can bring them money in a short period to solve the urgent problems. Sometimes, people have large loans since they are going to start a business, marriage, buy a car or a house, which means they have to borrow money from banks or other companies. At the same time, not everyone, such as students, low wage workers, people who are unemployed or bankrupt, and so on, owns a good credit score when they need to borrow. First, we need to understand what the default risk is. The simple indicator of risk is: Expected Loss = Default Probability × Loss Severity. A higher default risk usually comes with a higher interest rate.[2] That is saying that people who do not have enough credit score need to pay a much higher interest rate since lenders are having risik to get their money back. What's more, when people cannot repay the loan, the loss of lenders is immeasurable. Therefore, in our project, we would like to help lessen the severity of a loss for the lenders and also help unbanked customers to have a positive and safe borrowing experience.

In addition, in 2020, due to the large-scale impact of the coronavirus pandemic, it brings profound pressure on the United States's economy. In Fitch Ratings, the US default forecasts rates of term loan(LL) and high-yield(HY) bonds are 17%-20% and 15%-18% in 2020-2022, whereas the default forecasts rates in 2008-2010 is 15% and 22% for term loan(LL) and high-yield(HY) bonds, respectively.[1] As we know the period of 2008 to 2010 is a great recession and it is easy to find out the default risk in 2020-2022 is much higher than the normal period. Today, people are more likely to have huge economic pressure since they borrow more money than usual and they could not afford to repay. In order to lower the default risk from loans, there are several ways for reference: 1. Thoroughly check a new customer's credit record. 2. Use that first sale to start building the customer relationship. 3. Establish credit limits. 4. Make sure the credit term of your sales agreements are clear. 5. Use credit risk insurance. 6. Use factoring. 7. Develop a standard process for handling overdue accounts.[3]

In this scenario, we want to build a model using machine learning to predict if the applicant is capable of repaying the loan and then the lenders can only borrow money from applicants who meet the requirements. Therefore, data of personal information such as family status, health, education, occupation, will be the irreplaceable elements in anticipating people's ability to repay loans. We consider this as a binary classification problem and make predictions if the applicants obtain the ability to loan by applying different models. In order to achieve an ideal prediction, we need to collect historical data of which kinds of people are identified as default or non-default and then predict the capability of applicants paying their loans before approval.

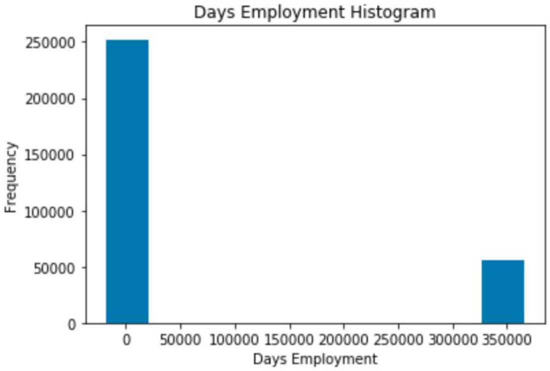
Our solution can be useful on several levels. Obviously, from the economic perspective, the lenders, especially banks, are the greatest beneficiaries. They can use it to help their decision making. The burden of human resources and the loss of money from applicants who do not repay the loan can be reduced. From the social aspect, our analysis would give the public a robust understanding of the factors that are related to the ability of repayment. It can inspire researchers and socialists for further development.

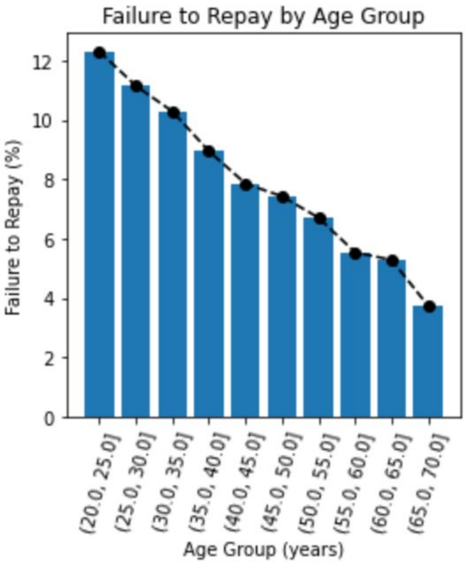
Data Understanding

The dataset used is from Home Credit Default Risk in Kaggle to help Home Credit predict whether or not a client will repay a loan or have difficulty. The dataset includes 7 different sources of data: 1) *application_train* (307511 observations and 122 variables) with binary target variable (0: the loan was repaid; 1: the loan was not repaid) / *application_test* (48744 observations and 121 variables) with no target variable. 2) *bureau*: data about client's previous credits from other financial institutions 3) *bureau_balance* :monthly data about the previous credits in bureau. 4) *previous_application*: previous applications for loans at Home Credit of clients who have loans in the application data. 5) *POS_CASH_BALANCE*: monthly data about previous point of sale or cash loans clients have had with Home Credit. 6) *credit_card_balance*: monthly data about previous credit cards clients have had with Home Credit. 7) *installments_payment:* payment history for previous loans at Home Credit. In this project, we mainly use *application_train* and *application_train* (Other data is for understanding and reference)

From the distribution of the target variable in the training set, we find there are 282, 686 zeros (non-default) and 24, 825 ones (default), which indicates a class imbalance problem. We find some anomalies in data when doing EDA. For example, the maximum value of 'DAYS_EMPLOYED' is 365243, which is unreasonably more than 1000 years. When we compare anomalous and non-anomalous data, we find that the anomalies have a lower rate of default. In terms of correlations between each variable and the target, we use Pearson correlation coefficient and find that the most positive correlation is DAYS_BIRTH and the most negative correlation is REGION_POPULAION_RELATIVE.

Furthermore, we look deeper into each variable, especially DAYS_BIRTH, the one has the most correlation. To be more intuitive, we divide DAYS_BIRTH by 365 to make it "age of the client". After we separate the data into smaller groups based on age, from the bar plot of age group (years) vs failure to repay (%), we find that younger clients have more chances to not repay the loan. This is an obvious observation which can alert banks to pay more attention to those younger applicants when they apply for loans or take some actions to make sure they can repay on time.





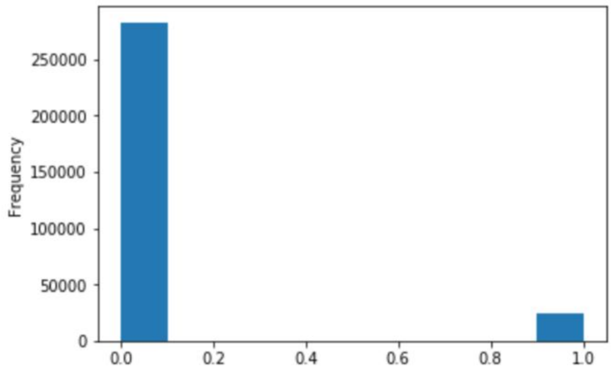
Also to better understand the data, we analyze those categorical variables and find some detailed information about the loan situation of applicants. 90.5% loans are Cash loans which were taken by applicants. Unaccompanied clients have the highest proportion of the total clients when applying for the applications. In terms of the purpose of loan, owning realty takes up 69.4% (89% for house/apartment) while owning cars only takes up 34%. In addition, more than half of the applicants are working people (laborers are the most), as well as married people. 71 % applicants have secondary and 24.3 % have higher education.

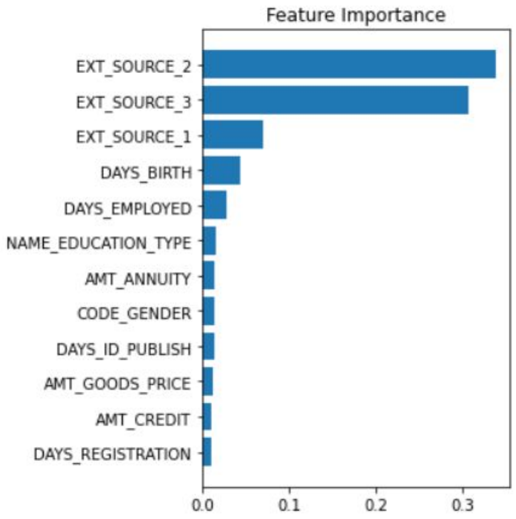
Although we can find some important information and features which seem highly related to whether an applicant could repay the loan or not, for more accurate prediction, we need to do feature engineering and select useful features through feature importance.

Data Preparation

We find there are 67 out of 122 columns that have missing values. For those missing values, we replace them with -999, as a special token to the model. Categorical columns also need to be encoded for processing. Therefore, if the column is categorical, we use numerical encoding to transform them into floats. To deal with the anomalies in 'DAYS_EMPLOYED', we change the anomalous values into NaN (which is replaced with -999 as well), and create a boolean column to indicate the anomaly.

Our target variable is already given which is 'TARGET'(distribution plot attached below). It is a binary variable (0: the loan was repaid; 1: the loan was not repaid). As mentioned in the data understanding section, from the distribution of the target variable in the training set, we find there are 282, 686 zeros (non-default) and 24, 825 ones (default), which indicates a class imbalance problem.

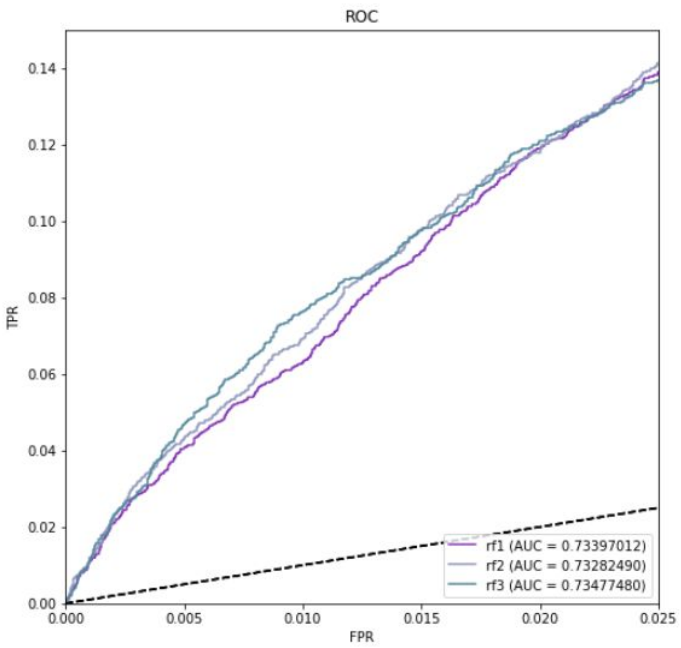




To understand more about what features are important, we feed the training data into a random forest classifier and obtain the feature importance. The plot here is the feature importance where the value exceeds 0.01. Notice how the three external source data are actually the most important features. In the data set, it is not well documented where those three features come from; they are said to be 'normalized score from external data source'. Other than the top 3, we find age, worked days, education type are the next three most important features, which makes sense. Ideally we want to feed all the features into the model and let it decide which one it wants to use, but we may risk issues such as having to use more computation power, and overfitting. The solution is to use dimension reduction methods, for example, only keep features where the importance is greater than 0.01. More will be discussed in the model part.

Until now, we are only using the root of the data tree: 'application_{train|test}.csv', it is natural to think that we should try to extract features from the rest of the data files. So we will extract information from all the csv files that is connected to the root of the tree. The left tree has two bureau files. Those two files contain information of previous loan information of customers who have loan history at other institutions reported to the bureau. The files on the right tree contain information about the customers' previous application at home credit, the related instalment payments, and cash and credit card balances. To add more features to the dataset, we treat categorical data and numerical data differently. For categorical data, we do a count for each category and calculate their sum and mean (normalized count) as features. For numerical data, we use their count, mean, max, min and sum as features. We do another feature importance calculation after we added the new features, where we have the number of total features increases from 122 to 901. Some of the newly engineered features do rank higher than the original ones in terms of their feature importance, such as 'credit_card_balance_CBT_DRAWINGS_ATM_CURRENT_mean' and 'previous_application_NAME_CONTRACT_STATUS_Refused_count_norm', indicating the usefulness of the new features. Now we have a total of 901 features that will be considered as potential parameters for our model of the next stage.

Next, we add some financial domain knowledge features to the existing 901 features. After some financial research, we find ratio quantities to be a very important indicator in a lot of the cases. Thus we add features ‘CREDIT_INCOME_PERCENT’ and ‘ANNUITY_INCOME_PERCENT’, which are the percentages of credit amount and loan annuity in the total income of the applicant. In addition, since the annuity is due by month, we create a feature called ‘CREDIT_TERM’ which counts the length of payment in months. Also, it might be invalid to only consider the days of employment without considering its relationship with the applicant’s age, so we add a feature called ‘DAYS_EMPLOYED_PERCENT’ which counts the ratio of the length of employment to the applicant’s age.



After all features have been well constructed into our dataset, we do a preliminary analysis on the effectiveness of our feature engineering. We want to compare three sets of features. The first set is the original application_{train|test}.csv, which contains 122 features. The second set is after we add features from the data on the left and right tree, with a total of 901 features. The third set is after we add four more domain knowledge features, with a total of 905 features. The train data is splitted into train and validation set in a 80/20 fashion. Then we use random forest classifiers to fit the train data, call the models rf1, rf2,rf3, and use the validation data to plot the ROC curve. The result is shown here above, and the plot is zoomed in near the initial phases of the curves. We only care about the initial phase of the curve because from the distribution of the training data, we only have approximately 8% defaults, and this 8% threshold is at the beginning of the ROC curve. We observe that rf3 has the best AUC value, and surprisingly rf2 has lower AUC value than rf1 even though rf2 is using more features. This is not bothering when we realise that we don't really care about what happens later on the curve, but AUC takes account of the whole curve. That being said, on the zoomed in part we find that for a majority of the curve rf3>rf2>rf1 in terms of performance. This shows the effectiveness of ordinary feature engineering and domain knowledge feature engineering to some degree. The 905 features we used to fit rf3 are then feeded into the model in the next stage.

Modeling & Evaluation

Given the processed data, there are many choices of machine learning models to be chosen from. Intuitively, the biggest challenge is that the training set alone contains 307511 samples and 905 features, resulting in the initial size of dataframe over 1.6G. Considering this, models requiring increased dimensional representation such as Supporting Vector Machines will be too expensive computationally. Therefore we propose following reasonable models for initiation:

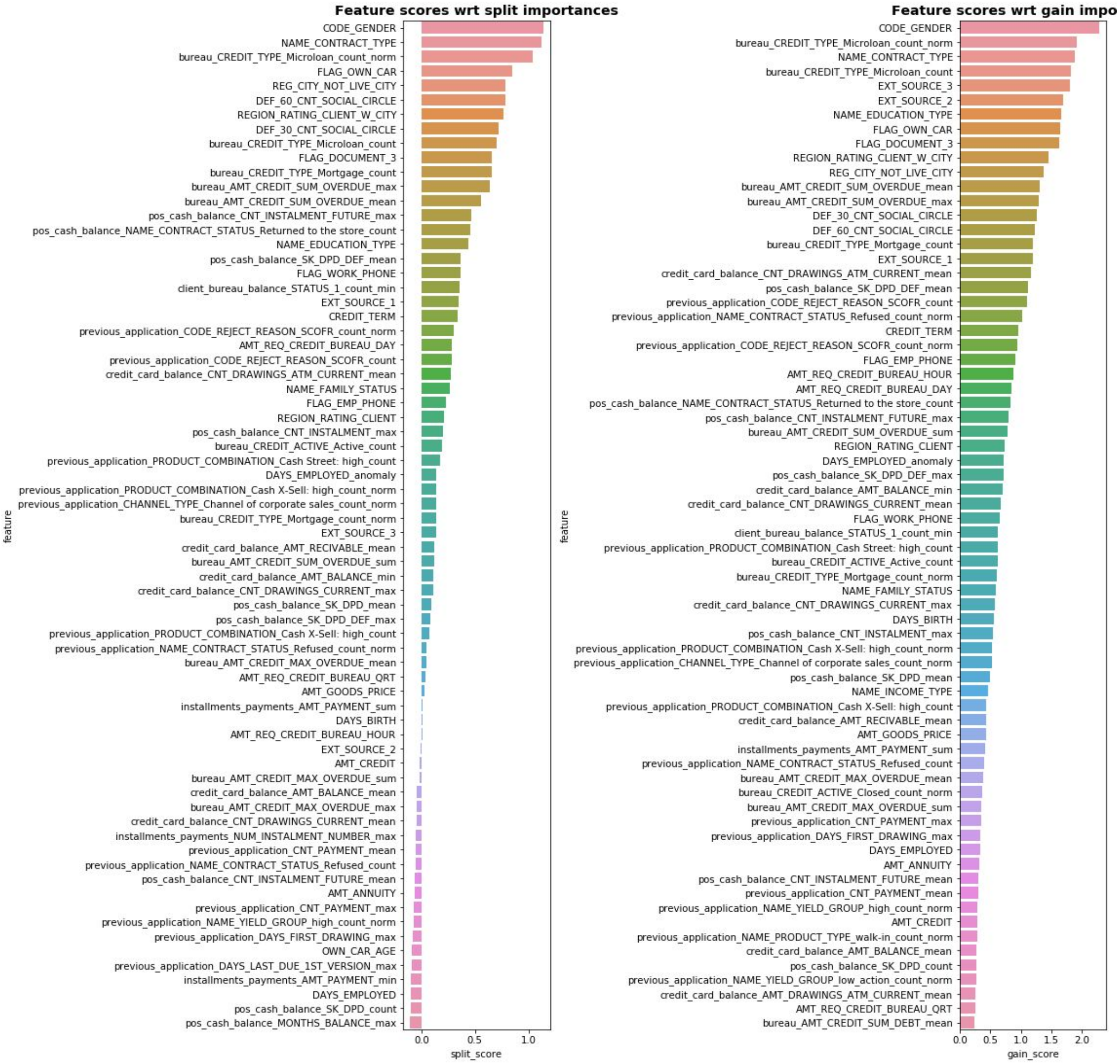
- Naive Bayes. Naive Bayes classification is purely based on observed posterior probability of training data in classes. The advantage is that Naive Bayes makes no assumption of the data distribution before fitting, and that it is every efficient because of its deterministic nature. However, Naive Bayes usually lack model capacity and are weak on inference especially for data not appeared in the training set.
- Logistic Regression. It is intuitive and familiar to most, making the explainability high. The model is easy to implement because it does not require hyperparameter tuning. However,

Logistic Regression makes strong assumptions of the data linearity, and performs weakly for nonlinear data (which is the case in most real world datasets).

- Random Forest. This algorithm ensembles the prediction of multiple decision trees, thus not likely to overfit on single batches of the data. The model is strong in capturing deep relationships between features and generalizing well to new data. The cons are that features must be hand designed and that the black-box nature makes Random Forest hard to be explained.
- Gradient Boosting Decision Trees. Instead of building independent trees as Random Forest, GBDT builds weak learners ensemble along the way. The pros of GBDT include its strong capacity in fitting varied distributions and the invariance to outliers and imbalance classes. However, the model contains many hyperparameters to finetune and is hard to be explained and visualized.

Before building the listed models, we choose kFold Cross Validation as our validation strategy to examine and compare their performance. We split the training data randomly into 5 folds and fit-valid models 5 times based on the split. We decide not to consider time-series splitting because there is no evident timestamp among samples. We also think stratified split is unnecessary so as to simulate the imbalance nature of target in the real world.

For each of the listed algorithms above, we build a corresponding baseline model with non-optimized hyperparameters. And we validate the Area Under ROC Curve based on the 5 fold Cross Validation as defined previously, and then average out across folds. AUC is a good metric because it is not biased on the size of test data. The same train_valid split config is passed to each model so that we can compare the auc scores equally.



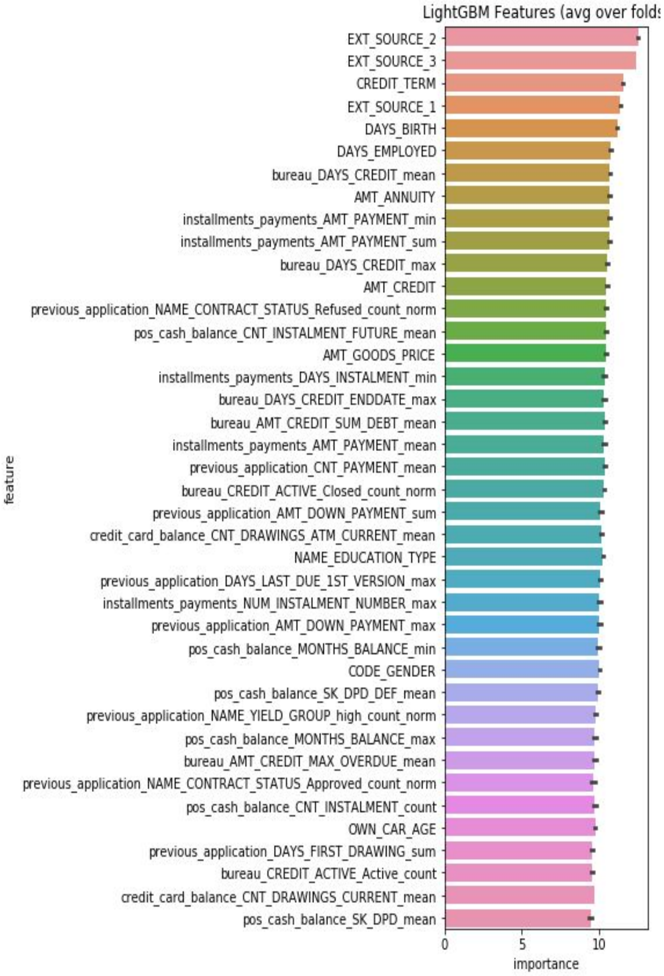
The last noticeable issue is that since we have 906 features, the training cycle is extremely expensive, and data points are harder to be classified in high dimensional spaces. Therefore we implement Permutation Importance[4] to select features before building the baseline models. We randomly shuffle the target and run GBDT multiple times to get the average gain-split importance of features, calling null importance. These importance are reduced from the true feature importance without target shuffle to purify the commitment of each feature in gains and splits, disregarding the structural and correlational effects of other features.

Finally we build the baseline models with selected thresholded null-importance features (of size 97). The auc scores for each model is listed below in the table. It can be observed that the Gradient Boosting Decision Trees model is significantly distinguishable from others. Therefore we will use the GBDT model as our baseline.

	Cross Validation AUC	Config Notes
Naive Bayes	0.548	-
Logistic Regression	0.504	Add small l2 regularization
Random Forest	0.651	Specified small max_depth
Gradient Boosting Decision Trees	0.784	Specified small max_depth; Add small l2 regularization

Until now we have generated a decent baseline of stable validation AUC 0.784, which is already close to the local Cross Validation score (0.806) of the 1st ranking solution on the Kaggle competition. There are many ways to improve this model on the baseline. While handcrafting more meaningful features should give huge boosts, such procedure is very time consuming and not necessary for this project. Instead, we will focus on hyperparameter tuning. We use Bayesian Optimization with Optuna[5] which learns a Gaussian Process model on a proposed range of hyperparameter settings, and makes search moves based on the predicted expectation of improvements. We propose a wide range for each hyperparameter commonly used for GBDT, including *num_leaves* for model capacity, *l1/l2* term for regularization, *min_child_samples, features_fraction, bagging_fraction* for robustness and preventing overfitting. Using the best set of tuned hyperparameters, the validation accuracy using the same configuration of 5 fold Cross Validation increases to 0.789.

So far we have built and optimized a decent model ready to be deployed. But before we move on to the next step, it is important to draw some conclusions and inference based on current progress. While training the GBDT model with optimized hyperparameters, we calculated a new set of feature importance. The top five features of highest importance are EXT_SOURCE_1/2/3, CREDIT_TERM and DAYS_BIRTH.



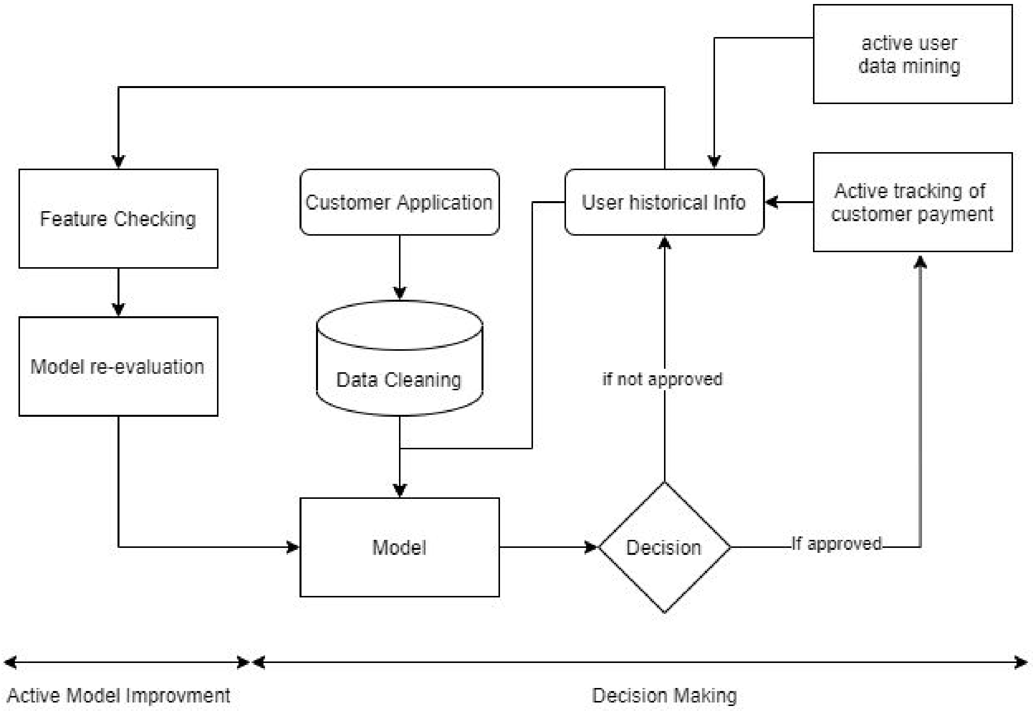
EXT_SOURCE represents fraud level scores evaluated from some third party institution. They are all recognized to be important by our model. Therefore, even though we do not know the exact institutions giving these evaluations due to data privacy, our model gives credit to these external evaluations.

CREDIT_TERM is a feature we create during feature engineering. It is the ratio of appliant's annuity loan amount and total credit amount, measuring length of payment in months. Therefore, it is important for the credit bank to check how many months left for the applicant to pay his loans.

DAYS_BIRTH is also an important feature for our model. It is less intuitive and a correlation check of linearity between DAYS_BIRTH and the target yields a correlation coefficient = 0.078, which is trivial. Even though there is no universal conclusion whether bigger ages inferring higher payment ability, we can still conclude that age plays an important role in prediction.

At this point, our model can serve as a good module that predicts how capable a specific applicant of repaying his loan before the credit is processed. The bank or business institutes can refer to this model at the time a new applicant comes up and take into consideration our model outcome before making decisions. A successful deployment of this model is expected to save man labor costs and lower the probability that loans are not paid. The model can be retrained and further fine-tuned with by adding in more training data and features.

Deployment



Here is a flow diagram for the deployment of our model. The application from the customer plus historical information for this customer are converted into readable data, and is cleaned before being fed into the model. The model then makes a decision of whether to approve or disapprove the loan request. If the application is approved, then the system will record all the action the customer takes about this loan into historical info; If the application is not approved, then this denial data will be automatically recorded into the historical Info as well. Once in a while, the user's historical info is taken into a ‘Active

Model Improvement' cycle, where the running model is dissected to see if it needs any improvements on choosing effective models, and to see if the currerrent model still performs as expected on the evaluation data. This concludes the model deployment phase. In addition, the system should also actively be collecting user information from all sources to improve the number of features accessible, which will help improve the model.

There might be some issues we may encounter during the model deployment process. The data fed into the current model are all historical data representing user behaviors from the past. It is possible that validation accuracy keeps falling years later with the current model. Therefore it is crucial to keep collecting new credit and loan records and retrain the model frequently to keep it up to date.

In the loan process, the problem of racial bias exists. A recent study in Northwestern University indicates that Black and Latino mortgage applicants are more likely to be declined than White. Although they might be successfully offered with a loan, in most cases, the loan comes with a high interest rate[6]. Data from the US Federal Reserve shows that more than 50% companies owned by the Black were turned down for loans, which rate is twice as high as the White owners.[7] This discrimination creates inequality and it reduce the credibility of the data. To reduce the risk that our model will suffer criticism from being racially biased, we should test it before employment to see if there is significant discrepancy between its treatments towards majority ethnicity groups and minority ethnicity groups. This evaluation result should then be used to modify the model to achieve equality in its treatment among various customers.

Another possible issue the model may face during deployment is those non-transparent issues, which will not be detectable before its deployment. To resolve this issue, banks should transition to our new model from the existing model step by step instead of one complete transition. In this way, we will be able to observe if the new model is actually performing better or worse than the old one. We will also be able to observe underlying issues with the model on a small scale, preventing catastrophic failure of the entire loan system. Another way of dealing with this is to test the model on real applications for sometime and evaluate its performance before completely switching to it.

Reference

1. Fitch Ratings: Credit Ratings & Analysis For Financial Markets, www.fitchratings.com/research/corporate-finance/pandemic-to-keep-us-loan-hy-default-rates-elevated-through-2022-05-10-2020.
2. “Default Risk - Overview, Assessment, and Key Factors.” Corporate Finance Institute, corporatefinanceinstitute.com/resources/knowledge/credit/default-risk/.
3. About the Author: Mélanie CarterKnowledge Partnerships Lead, et al. “7 Ways to Manage Credit Risk and Safeguard Your Global Trade Growth.” Trade Ready, 1 Dec. 2020, www.tradeready.ca/2014/trade-takeaways/7-ways-manage-credit-risks-safeguard-global-trade-growth/.
4. André Altmann, Laura Toloşi, Oliver Sander, Thomas Lengauer, Permutation importance: a corrected feature importance measure, Bioinformatics, Volume 26, Issue 10, 15 May 2010, Pages 1340–1347, doi.org/10.1093/bioinformatics/btq134
5. Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta,and Masanori Koyama. 2019. Optuna: A Next-generation Hyperparameter Optimization Framework. In KDD.
6. Nance-Nash, Sheryl. “Racial Bias in Mortgage Lending Is Very Real, but There Are Steps You Can Take to Secure a Loan When the Odds Are Stacked against You.” Business Insider, Business Insider, 9 June 2020, www.businessinsider.com/personal-finance/how-to-get-a-mortgage-racial-bias-in-lending-2020-6.
7. “Black-Owned Firms Are Twice as Likely to Be Rejected for Loans. Is This Discrimination? | Gene Marks.” The Guardian, Guardian News and Media, 16 Jan. 2020, www.theguardian.com/business/2020/jan/16/black-owned-firms-are-twice-as-likely-to-be-rejected-for-loans-is-this-discrimination.

Appendix

All the team members participate in a brainstorm of project ideas and structures. Everyone works in a joint effort and discusses frequently.

- Business Understanding: Xuhua, Bella
- Data Understanding/Preparation: Bella, Ziyi
- Model: Binfeng
- Deployment: Ziyi, Xuhua