TO: Hui Chen

FROM: BA-1 Team

DATE: September 1, 2021

SUBJECT: Team Project Report

Developing Process

In order to maximize the expected profit from the mortgage lending business of banks, we develop a fully automated mortgage lending system which will decide whether to approve a mortgage application. We tested with a variety of models, including logistic regression (with raw input, standardized input, and standardized input after feature selection), KNN, decision tree, random forest and neural network. After fitting the data, we find the best threshold to the given data by plotting the net-profit-increase – threshold graph and finding the value to make the curve reach its maximum. For each model we also printed out the classification report and compared the plots of the error-rate – threshold graph and net-profit-increase – threshold graph.

Result analysis

The best model that we came up for our team project is the logistic regression model trained with data that are standardized The overall accuracy is able to reach 0.94. The model does fairly well on classifying the not default loans, with the f1-score reaching up to 0.97. However, the classification on the default loans wasn't very good (f1-score = 0.11, precision=1.00, recall=0.06), mainly due to the class imbalance nature of the data. (We did try with the) It is interesting that the threshold of the model should be set to a relatively low value (=0.02) to gain a net profit of 3731784.0 on the test dataset. This is because according to the equation of calculating the net profit

$$\text{Net Profit} = \sum \{0.5\% \times \text{ Not default loan balance } + 20\% \times \text{ Default loan balance}\},$$

it is really costly to have a default loan balance accepted (a type-II error) so the model would try to limit on type-II error rate, and become really conservative on accepting loans.

Data used

Some macro variables and borrower characteristics was used, we’ve left out some characteristics that takes on values that are either fixed, missing too much, or simply auxiliary to our work. All the files provided in the Test.zip were used. The final data sets where split into three files “data_state”, “data_msa” and “data_nmsa” according to the value of “cd_msa” and other msa traits and are treated separately, but they basically yield the same result. So our work will be mainly demonstrated in “train_state.ipynb”

- Features: ‘source’, ‘Quarter_orig’, ‘orig_rt’ , ‘orig_amt’, ‘oltv’ , ‘ocltv’ , ‘dti’, ‘cscore_b’, ‘mi_pct’, ‘num_bo’, ‘num_unit’, ‘fthb_flg’, ‘FRM30_rate’, ‘treasury_3mon_rate’, ‘weekly_income’, ‘foreclosure’, ‘prepaid_cnt’, ‘unemployment_rate’.
- Label: ‘delinquent30’

Some considerations

- The time series nature: we used time series validation on the training dataset to tune for our model, and our feature selection was based on that. We've considered implementing augment Dickey-Fuller test and Engle-Granger test.
- The reason why we used net profit increase to plot the performance of the model is because the loss of misclassifying a default loan is far greater than the profit of classifying a non-default loan, thus a single error can cause the net profit to go negative. So instead, using net profit increase value will make the graph always stay on the positive side.
- We have discussed about using the net profit equation as the loss function, but eventually we've came to an agreement that we should view it as a classification problem, because for such a problem, trying to maximize a certain value will probably over fit the train data. The model only gets trained to know what loans went default and what don't, and will not absorb other features. Also, the net profit equation is a discontinuous function, cannot be taken to simple regression. So we eventually think that this should be a weighted classification problem.
- Oversampling can solve the class imbalance problem, but it makes the accuracy go down to 0.88 and doesn't do well on generating a high net profit.

Appendix

- The reason why we included decision tree and random forest is to cope with the fact that the loss of obtaining False Positive and False Negative is different. So these models are designed to deal with the loss sensitive characteristic of this problem. But unfortunately the result was not better than the logistic regression method.