Machine Learning Models for Loan Default Prediction

Kang Du, Zihao Fang, Zhennan Feng, Jingtong E

May 5, 2022

1 Introduction

Peer-to-peer (P2P) lending is a popular practice among individuals and small businesses to receive or grant funding to matched counterparties through online platforms.(Turiel, 2020) LendingClub was the world's peer-to-peer lending platform at one point.

Every user on LendingClub (LC) can be a lender or a borrower. For every proposed loan on LC, the user or potential lender can access relevant information such as the borrower's income, education level, homeownership, etc. The lender can then decide on whether and how much to lend to the counterparty. A user on LC can filter these P2P loans based on grade, a measure assigned by LC according to the borrower's financial status and other metrics.

One of the disadvantages of P2P lending is the counterparty credit risk or default risk. Rarely do P2P platforms provide any insurance mechanisms. Default risk is correlated with LC-assigned grades for each loan. Although default risk is somewhat encapsulated in the assigned grade by LC (A being the most secure), the grade is also influenced by the interest rate, among other factors.

The most profitable loans are often ranked below grade B. Therefore, it is crucial to select the loans that will be unlikely to default so P2P investors can select the most profitable loans with minimal default risk. Our project is designed to solve this problem by finding the best model by utilizing K-Nearest Neighbor (Peterson, 2009), Random Forest (Breiman, 2001), and Naïve Bayes (Irina, 2001) and evaluating model performance using accuracy and confusion matrices.

2 Data set and Cleaning

2.1 Data Set and feature extraction

We obtained our dataset through Kaggle, https://www.kaggle.com/datasets/wordsforthewise/ lending-club. This dataset contains the full LC data from the company's website. The original dataset contained two subsets, one with accepted loans and the other with rejected loans. We disregarded the rejected loans as the rejected loans never show up and could not have been invested by users on LC. The accepted dataset contains over 2 million records (N = 2260700).

2.2 Cleaning

2.2.1 Feature Engineering

Each entry contains 151 features. However, not all features are useful, and some are null for almost all entries. After manually examining the features, we identified 17 most relevant features and isolated our dataset based on the 17 features using pandas dataframe. The 17 features are: open_acc, funded_amnt, int_rate, term, verification status, annual_inc, dti, installment, application_type, emp_length, grade, sub_grade, home_ownership, loan_status, fico_range_high, fico_low, issue_d.

2.2.2 Addressing Issue Date and Term Data Types

The term feature is recorded as "36 months" or "60 months." In order to calculate the potential yearly profit, we took only the numbers and converted the data type to int. We converted the issue_d column to issue_yr as we only need the issue year to separate the training and testing sets.

2.2.3 Visual Inspection and Outlier Removal

We explored the distributions of the numeric columns through histograms and boxplots. Figure 1 is an example of our inspection for the funded_amnt feature. As shown in Figure 1, the funded amount feature peaks at $10000, with local maximums at $15000, $20000, $25000, $30000, $35000, and $40000. The same visual inspection is performed on open_acc, int_rate, annual_inc, dti, installment, and fico. We then removed the outliers from the dataset, reducing the total number of observations to 2152667.

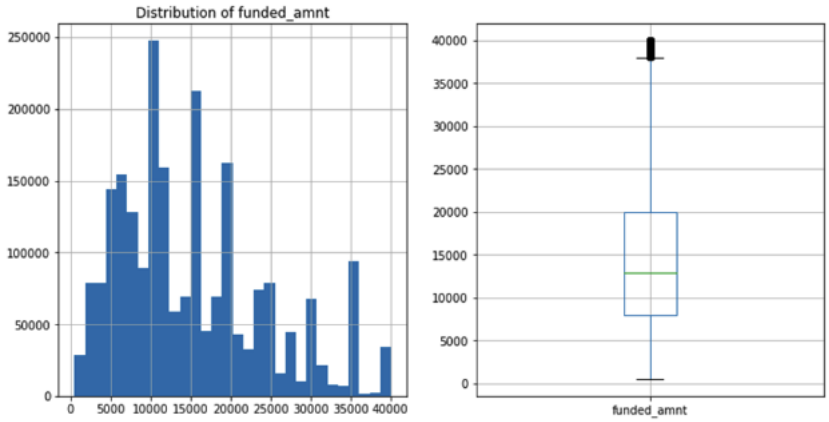

Figure 1: Distribution of funded_amnt

2.2.4 Loan Status Cleaning

There are also several levels to the loan status. For our analysis, we are only concerned with the fully paid loans and the charged-off loans. A loan charge-off means that the loan is in default. Other levels to a loan include current, meaning the loan is still ongoing, late, in grace period, or other. The current loans are dismissed as we cannot determine the end status of current loans (the loan IDs are not given). All other levels are also ignored as they are not definite.

After keeping only the fully paid and charged-off loans, we have 1036043 fully paid observations and 255751 charged off observations.

3 Models and Training

We used Random Forest, Naïve Bayes, and K-Nearest Neighbor models for training and testing. We also ran an up-sampled Random Forest model.

3.1 Training and Testing

Unlike normal datasets, the training and testing set must be split based on year. In our analysis, since we categorized data based on the year in 2.2.2, we grouped data from 2012 to 2016 as the training set (N = 1045866) and data from 2017 and 2018 as the testing set (N = 206921).

Before feeding the training data into our models, we performed one-hot encoding on all categorical variables and min-max scaled all numerical variables to make sure the models are not impacted by the scale of the numbers or the categorical variables.

3.2 Random Forest

After training the random forest model with 25 estimators and with Gini criterion. The model, trained with the scaled training set, returned an accuracy of 77.6%.

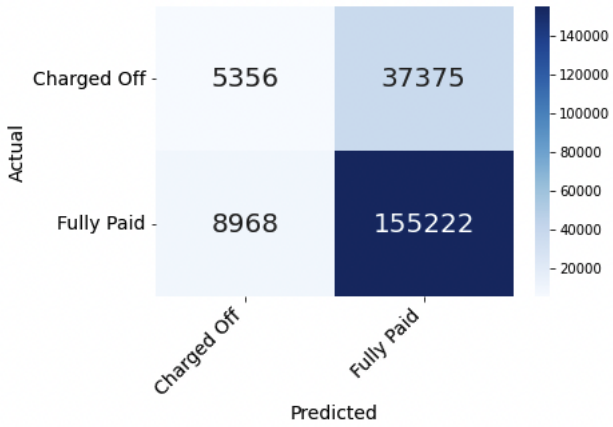

Figure 2: Confusion Matrix for Random Forest

3.3 Unsampled Random Forest

We suspected that the asymmetry in our training data was decreasing the model's accuracy. The original dataset had around 5 times fully paid loans as there were charge-off loans. To examine the correctness of this hypothesis, we upsampled the charged-off portion of our training data by resampling the charged-off portion (N = 838453) to equal the same as the fully paid portion (N = 838453). With a balanced training set, we ran a Random Forest model again, with 25 estimators and Gini criterion. The upsampled model returned an accuracy of 75.1%.

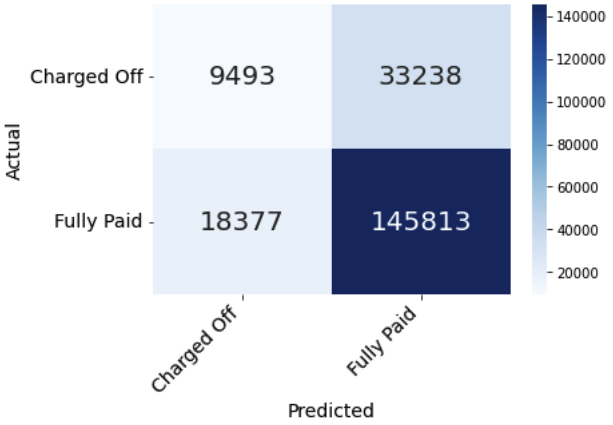

Figure 3: Confusion Matrix for Upsampled Random Forest

3.4 Naïve Bayes

After training the Naive Bayes model with the scaled training set, the model returned an accuracy of 76.4%.

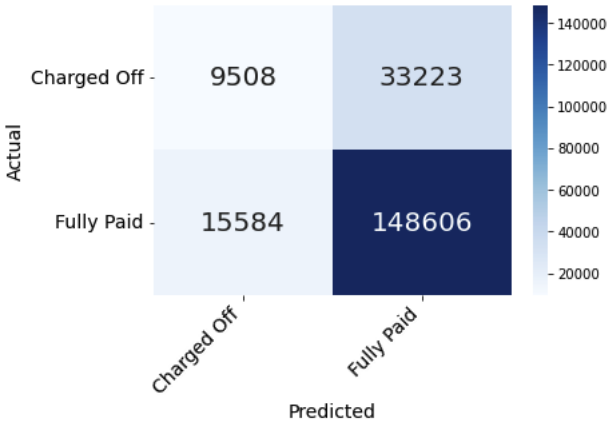

Figure 4: Confusion Matrix for Naive Bayes

3.5 K-Nearest Neighbor

After training the K Nearest Neighbor model with 9 neighbors and the scaled training set. The model returned an accuracy of 76.9%.

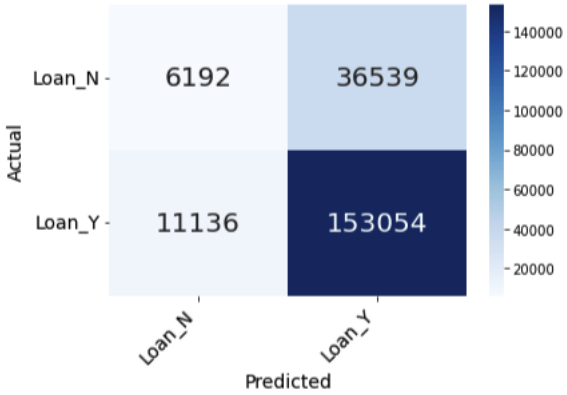

Figure 5: Confusion Matrix for K-Nearest Neighbor

4 Conclusion

Out of the 4 models we trained, the initial Random Forest had the highest accuracy (77.6%). The upsampled Random Forest model offered the lowest accuracy (75.1%). However, all four models have very similar accuracy scores. All models produced high true positives and low true negatives. We suspect that this phenomenon is caused by the nature of the problem. Predicting future loans based on current loans may not be highly accurate because of the different macroeconomic factors. Future work should explore more models such as SVM and include other independent variables to increase accuracy.

5 Acknowledgements

- The dataset was obtained through https://www.kaggle.com/datasets/wordsforthewise/lending-club
- his project was inspired by and was an extension of a project in COMP 488: Data Science in the Business World. However, the emphasis of this project was model performance and comparisons.
- Some code snippets were taken from class notebooks of COMP 488, developed by prof. Daniel Ringel.

References

[1] A. K. I. Hassan and A. Abraham, "Modeling consumer loan default prediction using ensemble neural networks," *2013 INTERNATIONAL CONFERENCE ON COMPUTING, ELECTRICAL AND ELECTRONIC ENGINEERING (ICCEEE)* 2013, pp. 719-724, doi: 10.1109/ICCEEE.2013.6634029.

[2] Turiel J.D. and Aste T.. "Peer-to-peer loan acceptance and default prediction with artificial intelligence," *The Royal Society* 2020, http://doi.org/10.1098/rsos.191649

[3] Leif E. Peterson, "K-nearest Neighbor", *Scholarpedia*, 4(2):1833.

[4] Breiman,L., "Random Forests", *Machine Learning* 45, 5-32 (2001), http://doi.org/10.1023/A:1010933404324

[5] Rish Irina, "An Imperial Study of the Naïve Bayes Classifier", *IJCAI 2001 Work Empir Methods Artificial Intelligence*, (3)