Analyzing and Predicting Loan Default Using Machine Learning Techniques

Qinjunjie Pu
Brown University, DSI

1 Introduction

A loan is the major source of income for the banking sector as well as the biggest source of financial risk for banks. Large portions of a bank's assets directly come from the interests earned on loans given. Though lending loans is quite beneficial for both the parties, the activity does carry great risks. These risks represent the inability of a borrower to pay back the loan by the designated time which was decided mutually by both the lender and the borrower and it is referred to as 'Credit Risk'.

In 2021, a study demonstrated that loan prediction exhibits high accuracy in forecasting the proportion of individuals who will not be late in repaying their loans, while the model displays some limitations in predicting the proportion of individuals who will be late in repaying their loans.[1] Another paper concluded that the choice of features and the algorithm are two major aspects when deciding whether to give an individual a loan or not.[2] For the dataset of this analysis, one study achieves a accuracy of 0.83 by logistic regression, 0.88 by random forest and 0.89 by XGBoost model.[3]

The purpose of this project is to develop a binary classification model to predicts loan defaults from a Loan Default Dataset from Kaggle.[4] It includes loan amount, interest rates, borrower credit scores, and other financial indicators. The target variable is binary (1 for default, 0 for non-default). This dataset is ideal for exploring classification models, feature engineering, and risk assessment strategies. It enables testing of various models like logistic regression, random forests, and XGBoost, along with interpretability techniques to derive insights.

2 Exploratory Data Analysis

The dataset comprises 17,972,550 observations and 34 columns. One of the columns is the target variable, which indicates the status of loan default. One column records the unique identification of each observation, and another records the year(All data is from the same year). From the remaining columns, 31 are considered as features. For the features, 19 are categorical, 2 are ordinal, and the remaining 10 columns are continuous

Figure 1 shows the distribution of the target variable. From the visualization, the imbalance of the target variable is not significant.

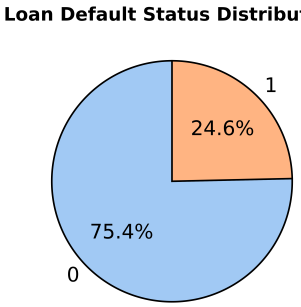
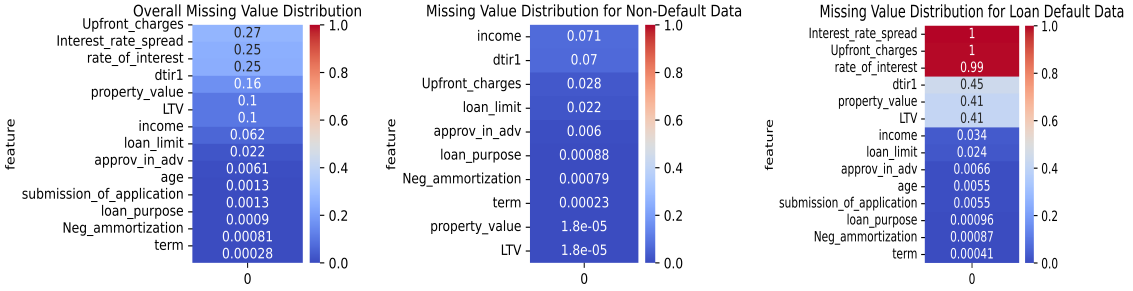


Figure 1: Loan Default Status Distribution

Figure 2 illustrates the missing value distribution across the overall dataset, default data, and non-default data respectively. Since 3 of features are almost missing for each observation with loan default status 1, these features are removed in the following analysis and model.



2.1: Overall Missing Value Distribution

2.2: Missing Value Distribution For Non-Default Data

2.3: Missing Value Distribution For Loan Default Data

Figure 2: Missing Value Distribution of Dataset

Figure 3 demonstrates a heatmap of correlation between each continuous feature and target variable, showing a moderate correlation between "loan_amount" and "property_value". The "credit_score" and "LTV" variables have weak correlations with other features, while Status seems to have minor correlations across the dataset.

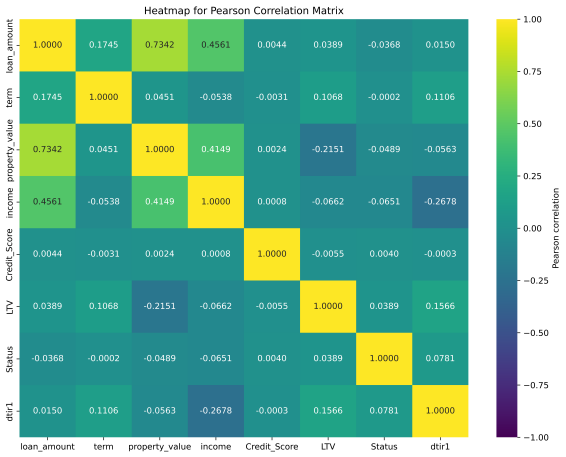
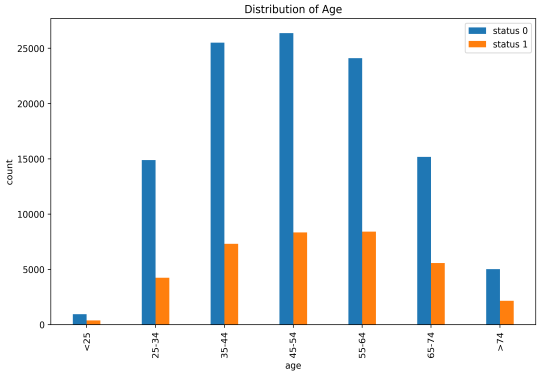
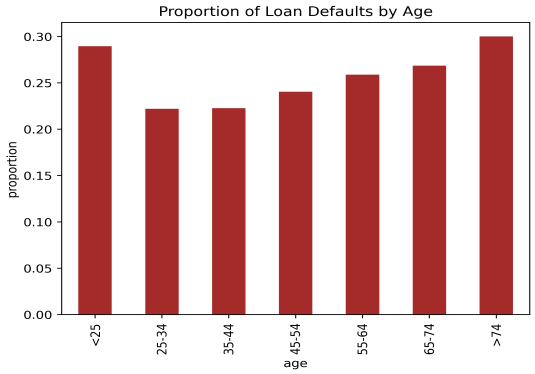


Figure 3: Correlation Across The Features And Target Variable

Figure 4 shows the distribution of different age groups in two status and their relative proportion,showing the very young and older people with higher probability to default.



4.1: Distribution of Loan Defaults by Age



4.2: Proportion of Loan Defaults by Age

Figure 4: Loan Defaults Analysis by Ages

The following graphs illustrate the distribution of loan amounts and LTV for the purpose of conducting a detailed analysis of the relevant features. Figure 5 indicates that higher loan amounts are more prevalent among borrowers with a default status. Conversely, figure 6 demonstrates that a higher LTV is more common among non-default borrowers.

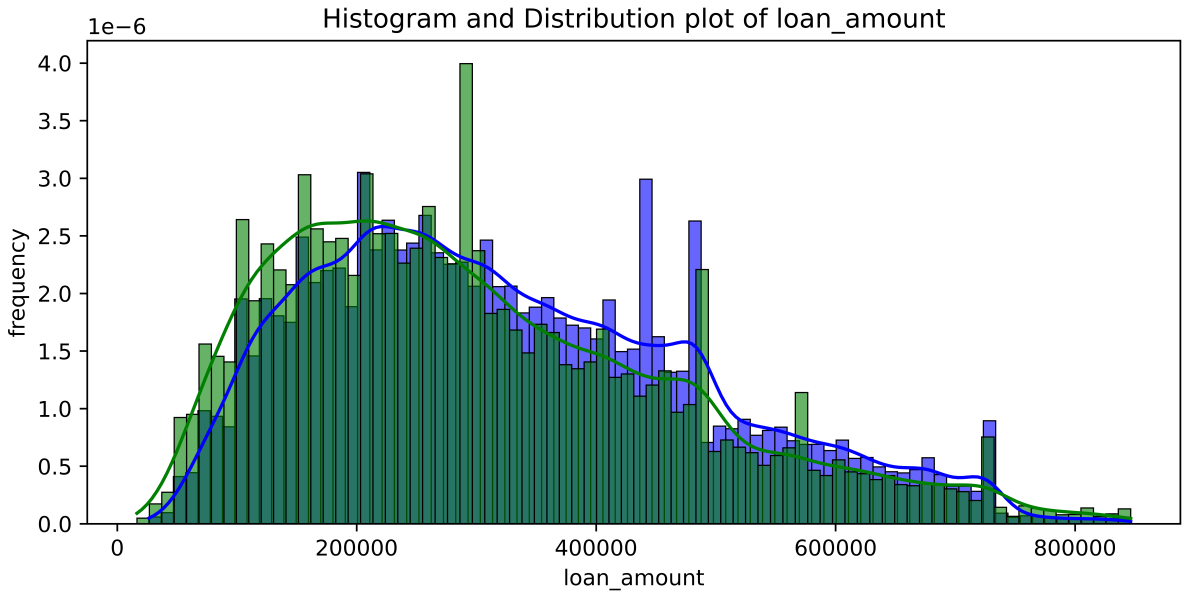


Figure 5: Distribution of Loan Amount

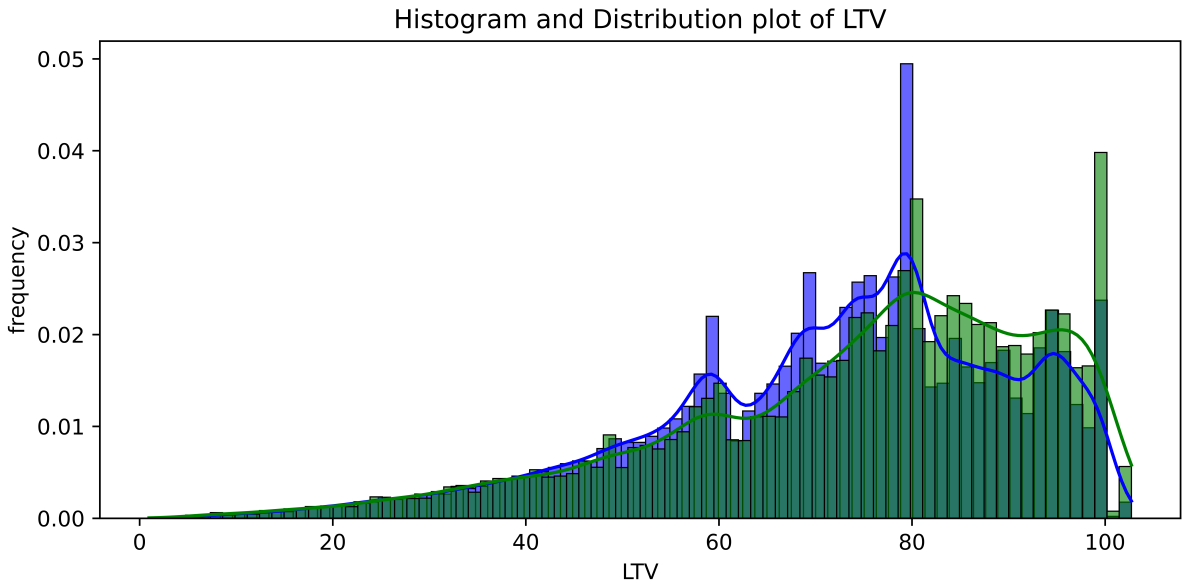


Figure 6: Distribution of LTV

3 Methods

3.1 Splitting

Since the data contains more than one hundred thousand observations with missing values in both continuous and categorical features, 20000 rows of the dataset are sampled to reduce the computational cost of model training.The data is first split into 80% for training and validation and 20% for testing by a Stratifiedsplit method. The training data undergoes further division using StratifiedKFold method with 4 folds, ensuring each fold maintains the same proportion of target classes.

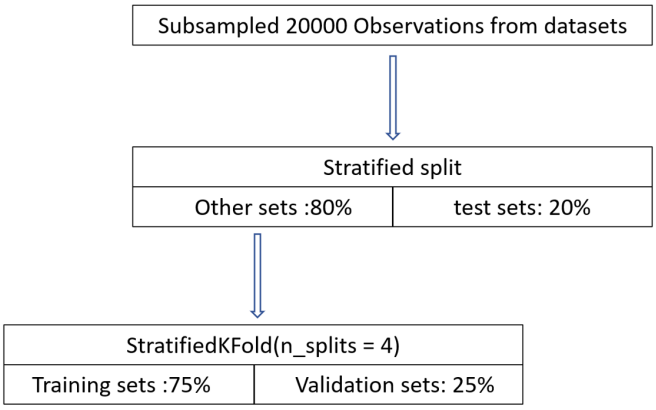


Figure 7: Process of Splits

3.2 Preprocessing

In the case of categorical features, an one-hot encoding is employed for those with a minimum of three distinct values, whereas a binary label encoding is utilized for features with exact two unique values. For ordinal features, an ordinal encoding is employed for the purpose of ordering the various values. In instances where a value is absent, the term “missing”is applied for both categorical and ordinal features. Following the encoding process, a standard scalar is utilized for all features to scale the data. It should

be noted that in this preprocessing stage, missing values associated with continuous features are not imputed. Additionally, a reduced feature method will be implemented in the models at a later stage, with the exception of XGBoost.After the preprocessing, there are 53 features in the dataset.

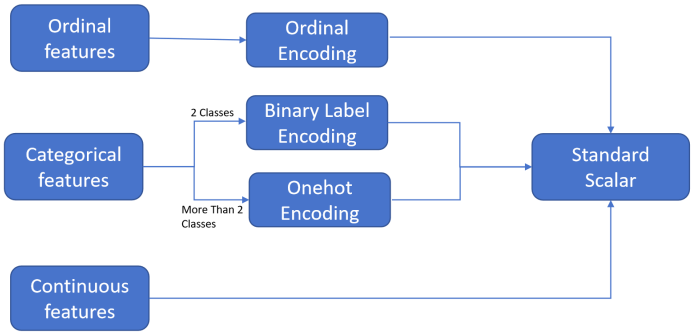


Figure 8: Process of Preprocessing

3.3 Models

Five models (XGBoost, Random Forest, K-nearest Neighbors, Support Vector Machine, and Logistic Regression) were trained in this project. For the XGBoost model, GridSearchCV was directly employed for hyperparameter tuning. For the remaining models, reduced-features methods were utilized to train multiple models for the purpose of handling missing values in continuous features.Table 1 illustrates the hyperparameters that were optimized during the training phase.

ML Model	**Hyperparameter**	**Values**
XGBoost Classifier	max_depth	3, 10, 50, 100
	learning_rate	0.01, 0.1, 1
	n_estimators	100, 200, 500
Random Forest Classifier	max_depth	5, 10, 30
	max_features	0.3, 0.6, 1
	n_estimators	1, 3, 10, 30, 100
KNeighbors Classifier	n_neighbors	3, 5, 10, 30
	weights	uniform, distance
Support Vector Classifier	gamma	0.001, 0.01, 0.1, 1, 10, 100, 1000
	C	0.01, 0.1, 1, 10, 100
Logistic Regression	penalty	none, l1, l2
	C	0.1, 1, 10

Table 1: Hyperparameters and their values for various ML models.

To simplify the training process, the metric used for model evaluation was the accuracy of the prediction for loan default status. However, since the number of default cases is smaller than the non-default cases, accuracy alone may not fully capture the model's performance. Therefore, the metrics like F1-score, Recall and Precision were also considered in the assessment. Each model was trained and tested in 5 different random states,which helps to address potential issues associated with data splitting and non-deterministic models.It also offers a more comprehensive performance evaluation, as different random states can reveal potential weaknesses or biases. By averaging results, it provides a more reliable assessment of the model's ability to generalize, reducing the impact of any single data split or random variation in performance.

4 Results

After averaging the accuracy scores and computing their variances, it is observed that the XGBoost Classifier achieved the highest accuracy score, with 0.891. By reduced-features methods, the Logistic Regression model performed an accuracy score of 0.867 higher than the previous model.[3] The KNeighbors Classifier was with the poorest performance, with an accuracy score of 0.858. Notably, all models outperformed the baseline accuracy score of 0.761.

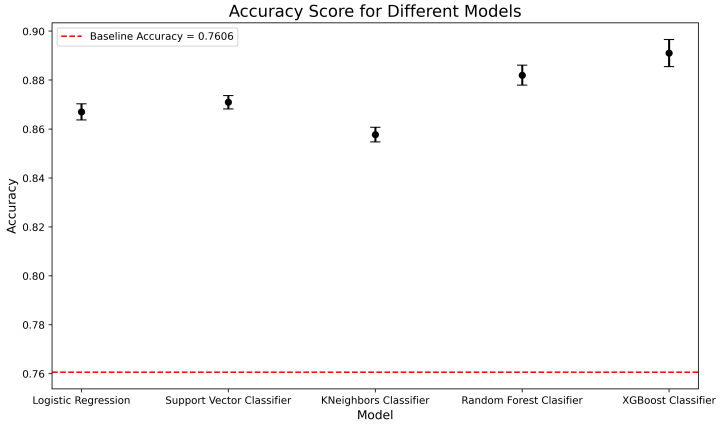


Figure 9: Accuracy Score for Different Models

Although the overall accuracy of the model is close to 0.9, its ability to accurately identify loan defaults is still unsatisfactory. From the following table and graphs, all models exhibit poor recall, similar to the findings in previous articles.[1, 3] Methods such resampling training data,using weighted classes and changing evaluation metrics like f-1.5 scores and recall were also attemped for model training, but these ways did not significantly improve the performance of algorithms. This result indicates that there is still considerable potential for advancement in feature engineering, feature selection, and model selection for loan default problems.

ML Model	**Accuracy**	**F1 Score**	**Recall**	**Precision**
XGBoost Classifier	0.891	0.724	0.603	0.908
Random Forest Classifier	0.882	0.707	0.579	0.910
KNeighbors Classifier	0.858	0.604	0.441	0.960
Support Vector Classifier	0.871	0.648	0.496	0.936
Logistic Regression	0.867	0.649	0.498	0.928

Table 2: Test Scores of the Algorithms

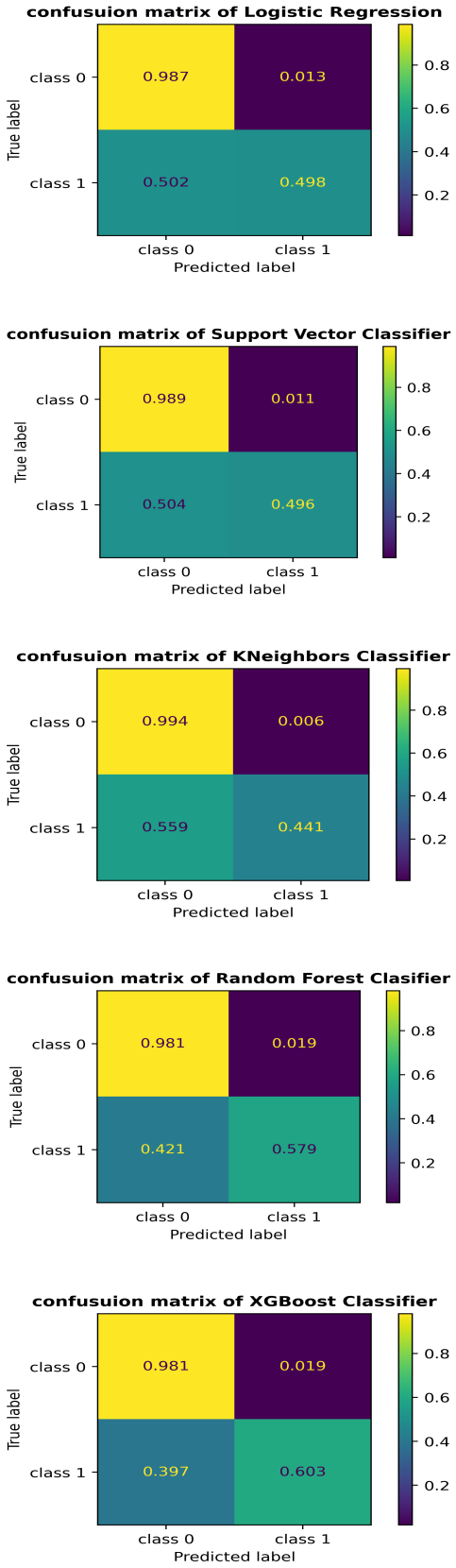


Figure 10: Confusion Matrices of Different Models

Since the XGBoost Classifier performed best among 5 algorithms, the analysis of feature importance

was focused on this algorithm. The following analysis is based on the XGBoost Classifier with the highest test score among 5 random states.

Several methods were applied to figure out which features are the most significant in the model. Firstly, as shown in Figure 11, permutation feature importance was used to assess the impact of each feature by shuffling them and observing the change in the model's performance. In addition, feature importance scores from the XGBoost model were examined. Figures 12 and 13 display the feature importance based on weight and total gain, respectively. And Figure 14 presents the SHAP global feature importance for the XGBoost model. Based on these graphs, features like "LTV", "property value", "dirt1" and "income" appear most frequently as the most important features in the model.

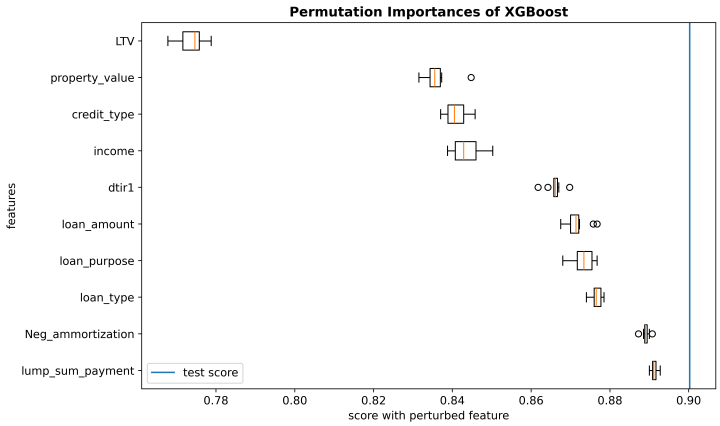


Figure 11: Accuracy Score for Different Models

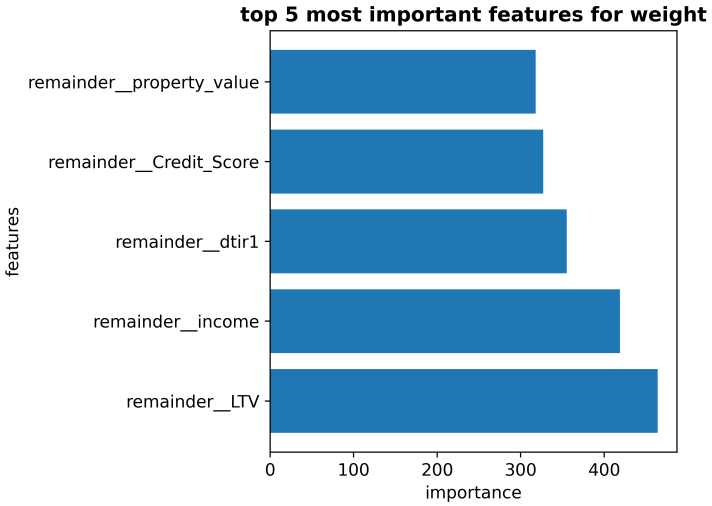


Figure 12: Global Feature importance by Weight

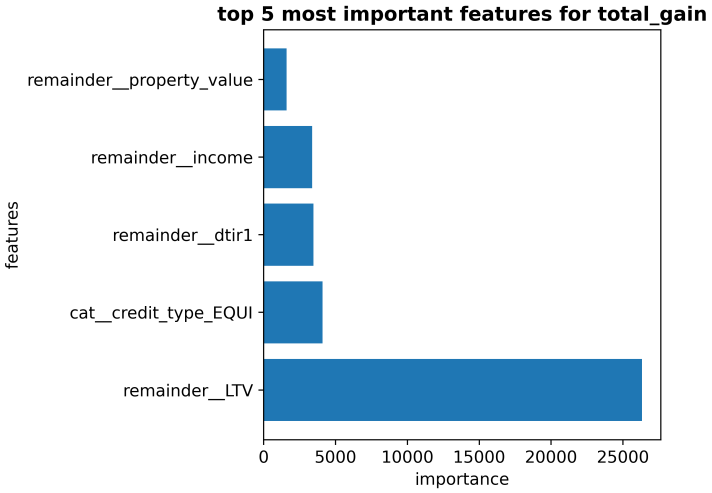


Figure 13: Global Feature importance by Total Gain

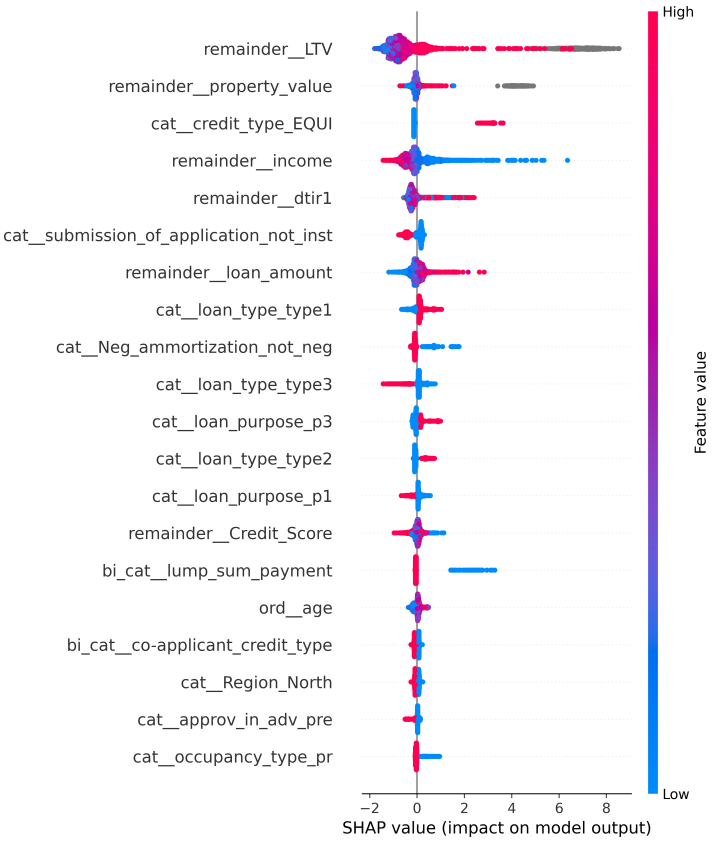


Figure 14: Global Feature importance by SHAP value

For local importance, SHAP value were studied to figure out the influences of each features.Figure 15 demonstrates the predicted probability of class 1 of two examples and the influences of those important features. Features such as “loan_purpose” and “LTV” have the most significant impact on the final prediction for the examples.

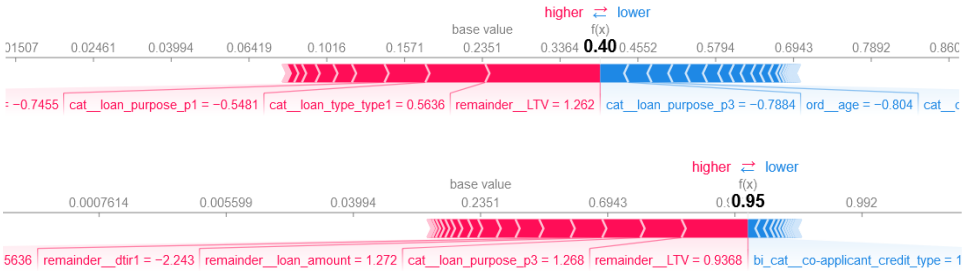


Figure 15: Two Examples for SHAP Local Feature Fmportance

5 Outlook

In the experiment, the XGBoost classifier achieved the highest accuracy (0.891) and F1 score (0.724). This model correctly identified almost all observations with class 0 and captured about 3/5 of the observations with class 1. To improve the performance for class 1, methods such as resampling the training data, using weighted classes, and adjusting the evaluation metrics (e.g., F1.5 score and recall) were attempted. These approaches yielded similar results: one additional class 1 sample was correctly predicted, but at the cost of approximately five additional false positive predictions. Despite the improvements in class 1 prediction, the trade-off with false positives highlights the challenge of balancing precision and recall. To address this, further fine-tuning of the hyperparameters and exploration of more advanced techniques such as deep learning may be a necessary choice.

Also, feature engineering and collecting more features is a potential to build a better model, for example, interaction terms between features could improve the model's ability to distinguish between classes.

References

[1] Mehul Madaan et al. "Loan default prediction using decision trees and random forest: A comparative study". In: *IOP conference series: materials science and engineering*. Vol. 1022. 1. IOP Publishing. 2021, p. 012042.

[2] Peter Martey Addo, Dominique Guegan, and Bertrand Hassani. "Credit risk analysis using machine and deep learning models". In: *Risks* 6.2 (2018), p. 38.

[3] *Loan Default Prediction: A Comprehensive Approach*. URL: `https://www.kaggle.com/code/amosrr/loan-default-prediction-a-comprehensive-approach#5.4-Cross-Validation-of-the-Model`.

[4] *Loan Default dataset*. URL: `https://www.kaggle.com/datasets/yasserh/loan-default-dataset/data`.

GitHub Repository

https://github.com/Bail111/Loan_Default-Analysis.git