Loan Default Prediction Using Lending Club Data

Yunfei (Cynthia) Xing
Data Science Initiative, Brown University
GitHub [4]

1. Introduction

1.1 Motivation

Loan default occurs when a borrower fails to repay a loan according to the terms agreed upon. This is a critical financial concern that not only adversely affects the borrower's credit rating, but also imposes significant consequences on the lending financial institution, including revenue loss, increased operational costs, and potential reputation damage.

This project aims to employ machine learning techniques to predict loan defaults based on an analysis of the loan's specifications and the borrower's financials. The model will benefit lenders by providing a data-driven tool to optimize their risk-reward balance, and borrowers by enabling proactive loan management and better loan terms negotiation. Furthermore, by reducing the incidence of unexpected loan defaults, this model contributes to a more stable economic environment and overall economic health.

1.2 Dataset

Lending Club is the first peer-to-peer lender to register its offerings as securities with the SEC, and to offer loan trading on a secondary market.[1] The dataset, sourced from Kaggle, is an aggregation of Lending Club's loan records from 2007 to 2017, originally compiled from the Lending Club website.[3] It includes a comprehensive array of borrower details, credit history, loan specifications, financial data, and other miscellaneous information. The full dataset consists of an extensive 1,646,801 entries across 150 features. It is independently and identically distributed and contains some missing values.

1.3 Previous Work

Previous research efforts have focused on classifying loan defaults using Lending Club data, but these studies covered different time periods, involved varying numbers of data points and features. For instance, one study, which analyzed 264,796 data points across 27 features, found that the XGB achieved a test accuracy of 89.60%, while Random Forest reached 88.94% accuracy.[1] Another study, which sampled 11,000 data points from a larger pool of 814,986, and used 29 features, reported achieving a test accuracy of 87.04% with KNN and a significantly higher 94.81% accuracy with Random Forest.[2]

2. Exploratory Data Analysis

During EDA, data cleaning and feature selection were conducted to ensure that only relevant features would be fed into the ML Pipeline.

2.1 Target Variable

The target variable in the original dataset is ‘loan_status’, with 9 different categories covering different statuses of ongoing and completed loans. For this project, the focus was narrowed to completed loans specifically categorized as ‘Fully Paid’, ‘Charged Off’ or ‘'Default’ to form a binary classification problem.

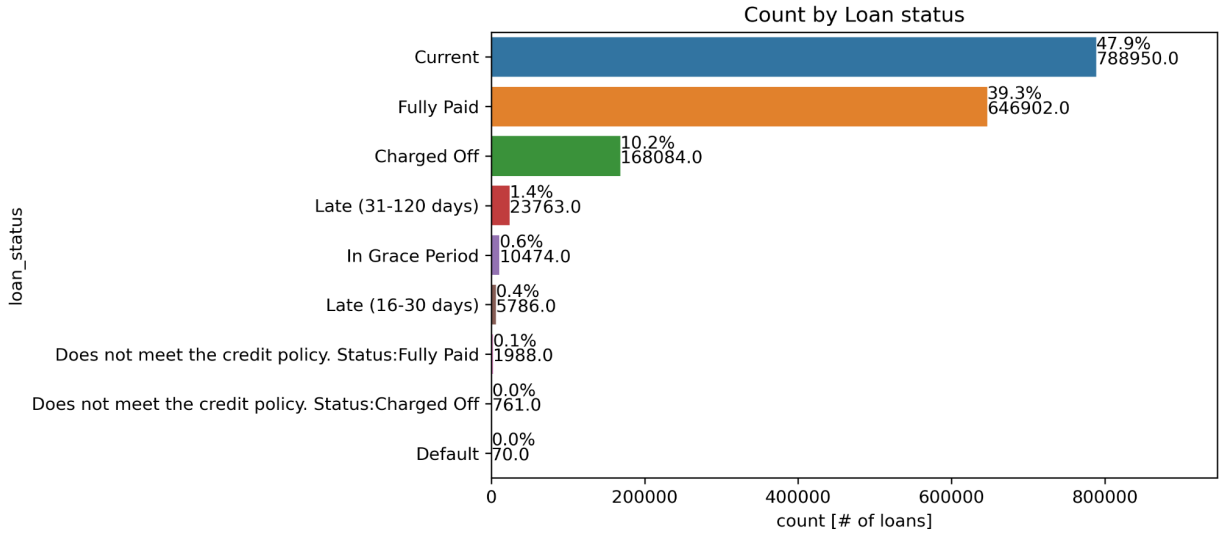

Fig 1. Loan Status Distribution in Full Dataset

Figure 2 illustrates the outcome of this step, where 'Fully Paid' loans constitute roughly 80% and 'Defaulted / Charged Off' loans 20%, creating an unbalanced classification scenario. The target variable is binary-encoded: 1 signifies 'Charged Off' or 'Default' status, and 0 denotes 'Fully Paid' loans.

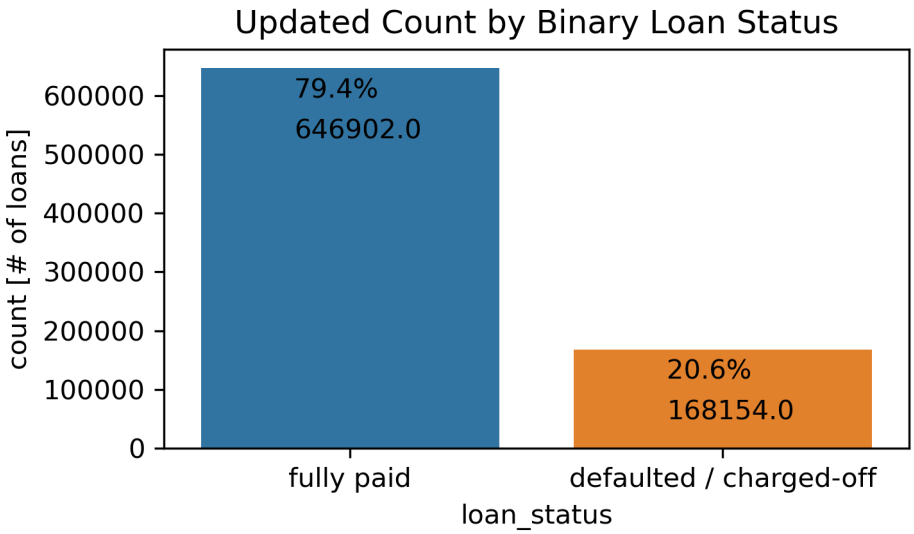

Fig 2. Binary Loan Status Distribution in Full Dataset

2.2 Feature Selection

To enhance model performance and interpretability, feature selection was performed adhering to a predefined set of criteria established during EDA.

First, features with greater than 20% missing values are removed to ensure data quality, retaining 60% of the dataset's features with minimal missing data. The removal mainly affected features under common categories and co-borrower-related features, but the dataset's I.I.D. nature ensures this does not affect the classification task.

Next, features not accessible to borrowers at the time of making investment decisions were excluded to align the model with practical decision-making processes. Some examples of retained features are fico_score, annual_income, loan_amount, term, interest_rate.

Finally, features that show a low correlation with the target variable ‘defaulted’, or are highly correlated with another feature are omitted.

After this process, 37 features were selected from the original 150. Following further data cleaning and reformatting, such as dropping features being used as an identification metric of the loans, the cleaned dataset retains 32 features.

2.3 Missing Values

After data cleaning, approximately 9% of the dataset rows contain missing values. To allow model selection across various ML techniques, such as Logistic Regression, KNN, Random Forest, etc., these rows were omitted. Given the extensive size of the dataset, the remaining data

is sufficient to capture the dataset's nuances. Hence, our final cleaned dataset contains 739,391 rows with 32 features.

2.4 EDA & Feature Analysis

Descriptive statistics were obtained using .describe(), which provided insights into the shape of the dataset's distribution, including minimum and maximum values. The .info() method was utilized to assess data types and the presence of missing values. Duplicate records were checked using .duplicates(). Finally, the distribution of the target variable within the cleaned dataset is presented in Figure 3.

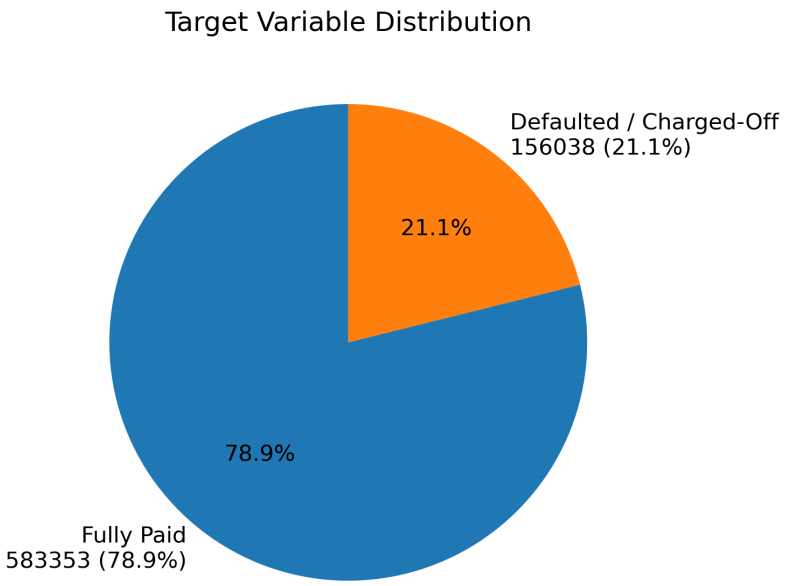

Fig 3. Loan Status Distribution in Cleaned Dataset

EDA was performed on every categorical and continuous variable. Selected interesting findings are discussed in the following paragraphs.

As shown in Figure 4 below, loan default rates differ by home ownership status, with renters and mortgage holders showing lower rates, potentially due to credit maintenance and financial constraints. Surprisingly, non-homeowners have lower default rates, challenging the assumption that home ownership equates to financial stability, as homeowners have higher default rates in the dataset.

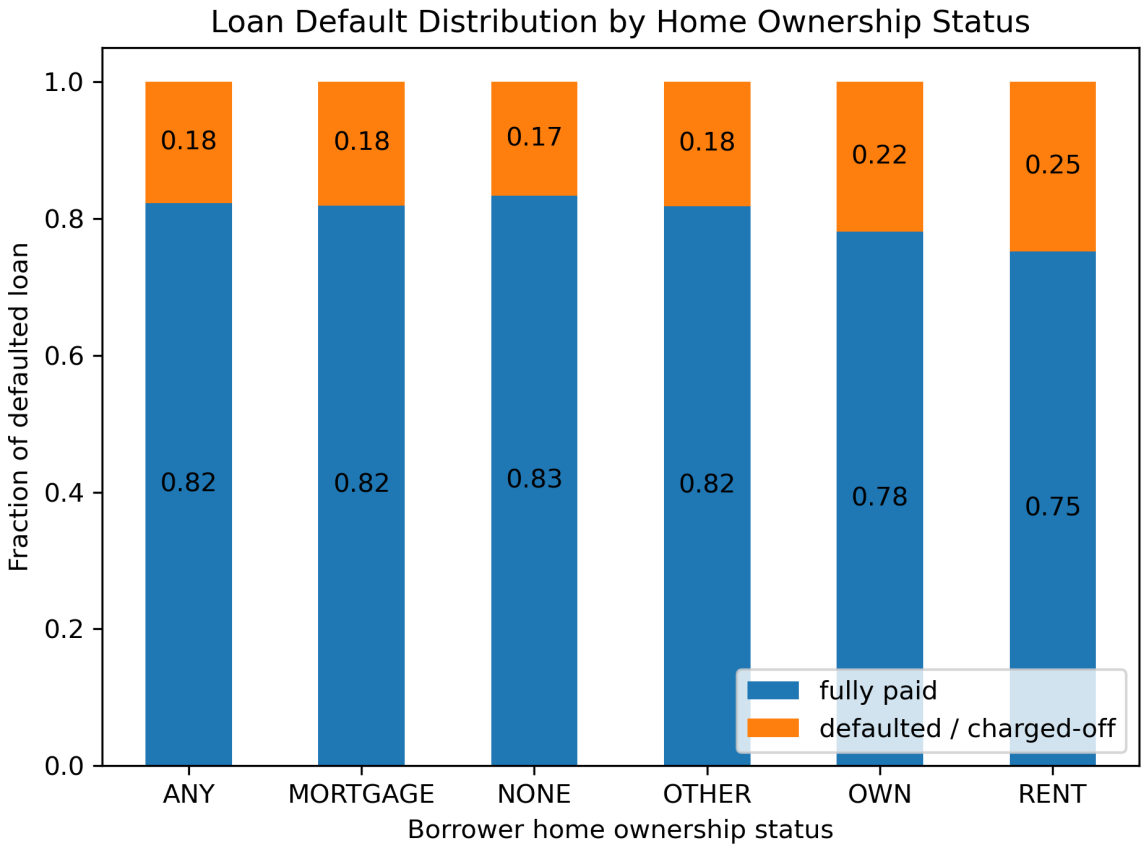

Fig 4. Loan Default Distribution by Home Ownership Status

As shown in Figure 5 below, loan default rate fluctuates with the borrower's income verification status by Lending Club. Interestingly, unverified incomes correspond to lower default rates than verified incomes, questioning the efficacy of the verification process in predicting default risk and implying other influencing factors.

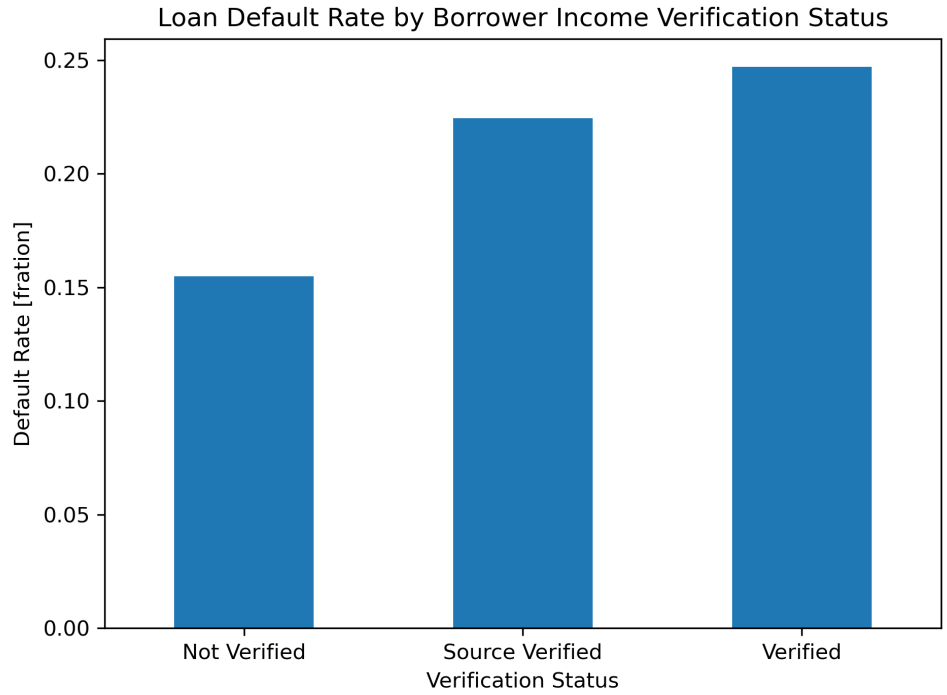

Fig 5. Loan Default Rate by Borrower Income Verification Status

As shown in Figure 6 below, when looking at the distribution of borrowers' debt-to-income (DTI) ratios, there is a noticeable skew towards lower DTI ratios among those who have fully paid off their loans, suggesting that these borrowers have a more manageable balance between income and debt. On the other hand, a higher DTI ratio indicates a higher proportion of income going towards debt repayment, which may signal potential financial strain.

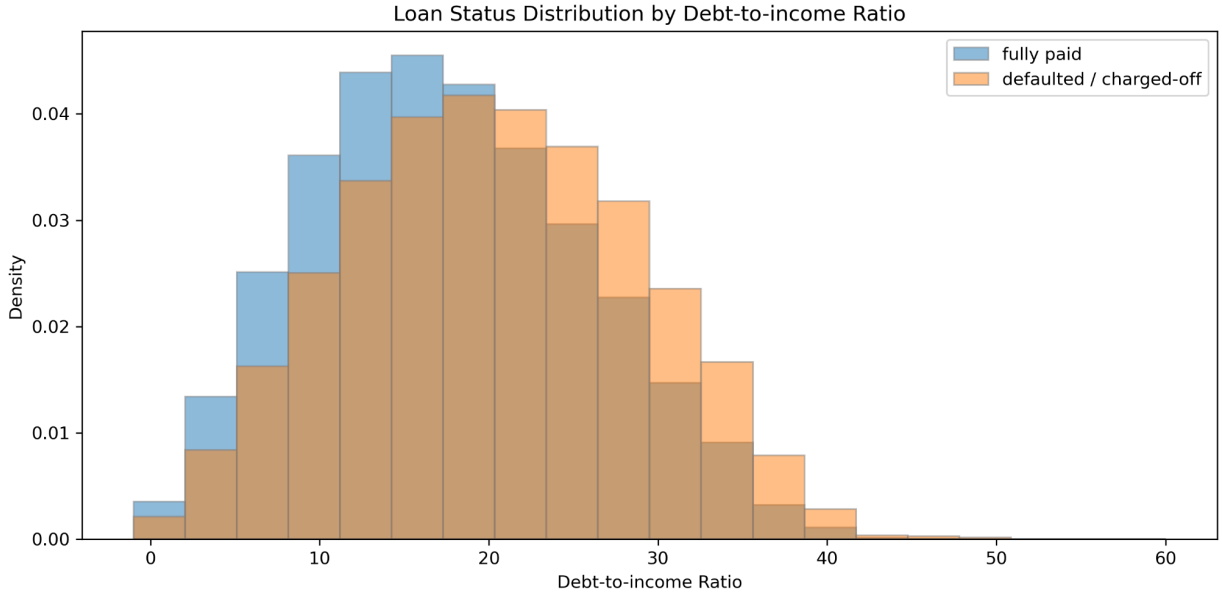

Fig 6. Loan Status Distribution by Debt-to-income Ratio

Figure 7 illustrates the relationship between loan interest rates and default status, indicating that defaulted loans are associated with substantially higher interest rates. Interest rates for loans vary significantly, with a range from as low as 5.3% to as high as 31%. This could imply that higher interest rates are a contributing factor to the likelihood of loan default.

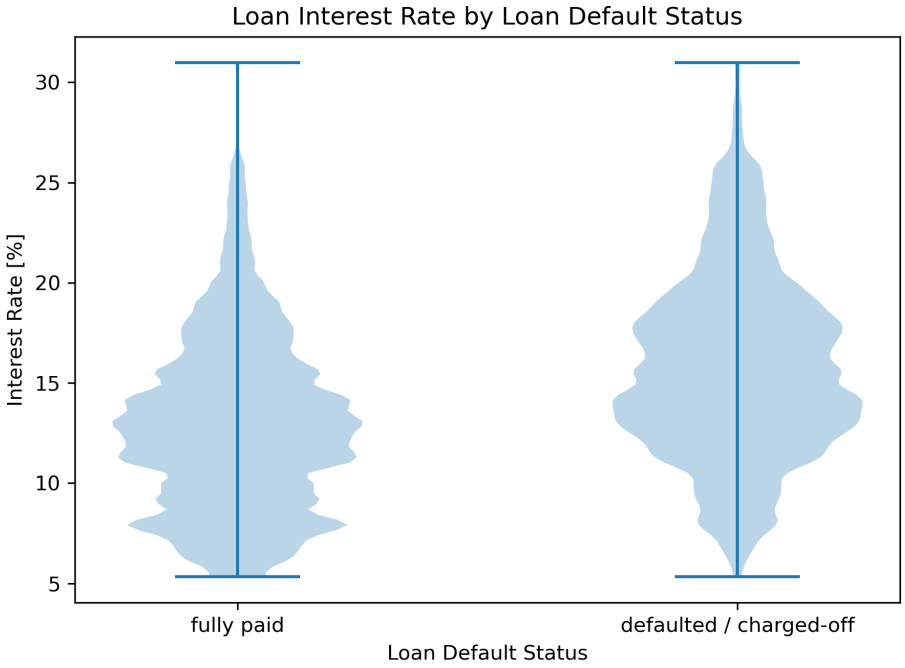

Fig 7. Loan Interest Rate by Loan Default Status

3. Methodology

3.1 Splitting

Since the dataset is I.I.D. and comprise over 740,000 records, it will be split using train_test_split method from sklearn into 98% training and 1% each for validation and testing. To address the target variable's imbalance and mitigate overfitting, StratifiedKFold will be applied for cross-validation, providing a balance between bias and variance in test error rate estimates. However, to adhere to the specified 98-1-1 split while also utilizing StratifiedKFold, a 'CustomSplit' class was created, allowing a custom splitting ratio and a configurable number of splits in CV.

3.2 Preprocessing

Four preprocessors were used based on the different data types of the features. Categorical features are processed using OneHotEncoder, while ordinal features are handled by OrdinalEncoder. Continuous variables are scaled using MinMaxScaler for those with a defined

range and StandardScaler for all other continuous variables. The preprocessing process resulted in 103 features.

3.3 ML & CV Pipeline

Due to computing power limitations, a subset of 200,000 data points will be used to run through five iterations with different random states, and only three folds are performed during cross-validation. GridSearchCV is applied within the pipeline for hyperparameter tuning and model validation, with accuracy as the scoring metric. The best-performing model will then be refitted to the entire dataset to compute the final test scores. Figure 8 illustrates the pipeline for one such iteration.

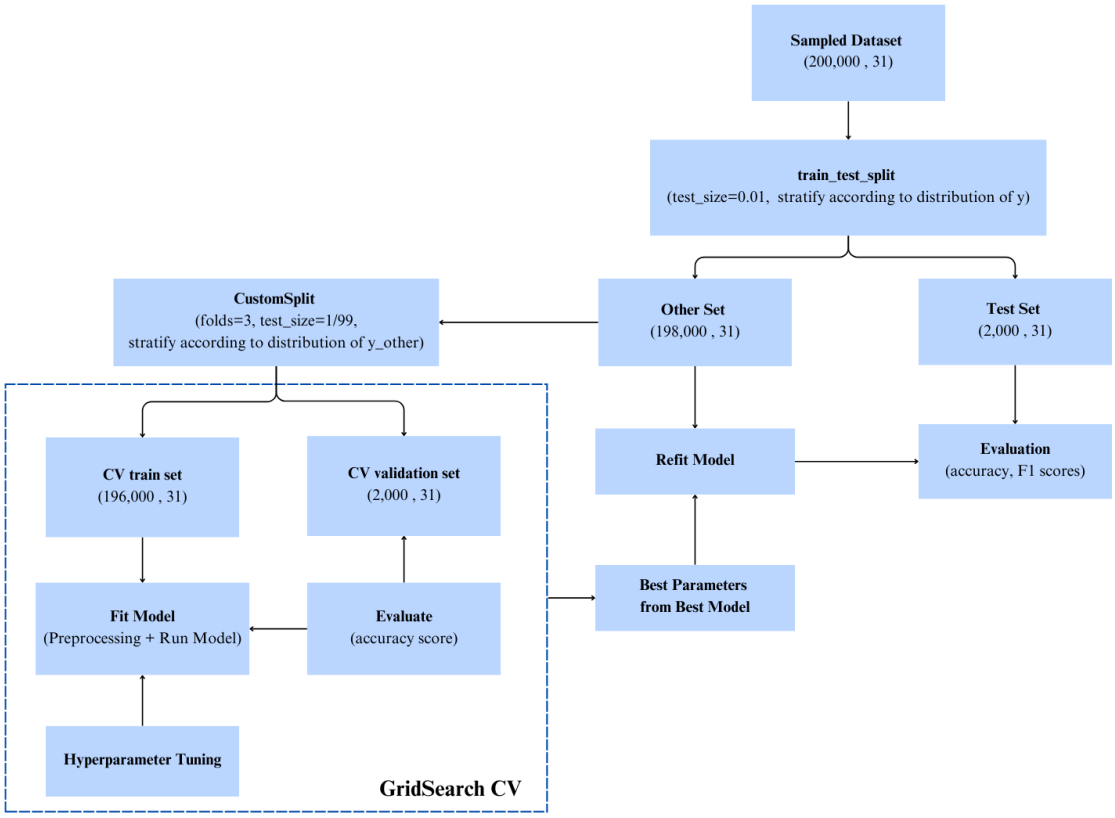

Fig 8. ML and CV Pipeline

Four machine learning algorithms are used in this analysis: Logistic Regression, XGBoost Classifier, Random Forest Classifier, and KNN Classifier. For each algorithm, several hyperparameters were tuned, as shown in Table 1.

Logistic Regression	C: [0.0001, 0.001, 0.01, 0.1, 1]
XGBoost Classifier	reg_alpha: [1e-1, 1e0, 1e1], reg_lambda: [1e-1, 1e0, 1e1],

	max_depth: [5, 10, 20]
Random Forest Classifier	max_depth: [1, 3, 10, 30, 100], max_features: [0.25, 0.5, 0.75, 1.0]
KNN Classifier	n_neighbors: [10, 20, 30, 100], weights: [‘uniform’, ‘distance’]

Table 1. Tuned hyperparameter values for each ML algorithm

4. Results

4.1 CV Results

The baseline accuracy of the model, determined by predicting the majority class for the target variable, stands at 78.90%. All models showed a high accuracy score outperforming the baseline, and out of the models, XGBoost Classifier showed the best accuracy, outperforming baseline by 202 standard deviations. This is shown in both Table 2 and Figure 9.

The noticeably lower accuracy of the KNN model compared to the other three models could be attributed to its challenges in high-dimensional spaces, as is the case with the 103 features present here. Additionally, KNN's performance can be significantly affected by noise and imbalances in the dataset, which also applies here, given the imbalance in the target variable.

Model	Mean Accuracy Score	Std Accuracy Score	Number of stds above Baseline Accuracy
Logistic Regression	0.9932	0.001806	113
XGBoost	0.9968	0.001030	202
Random Forest	0.9956	0.001356	152
KNN	0.8736	0.007317	12

Table 2. Mean Performance for each ML algorithm

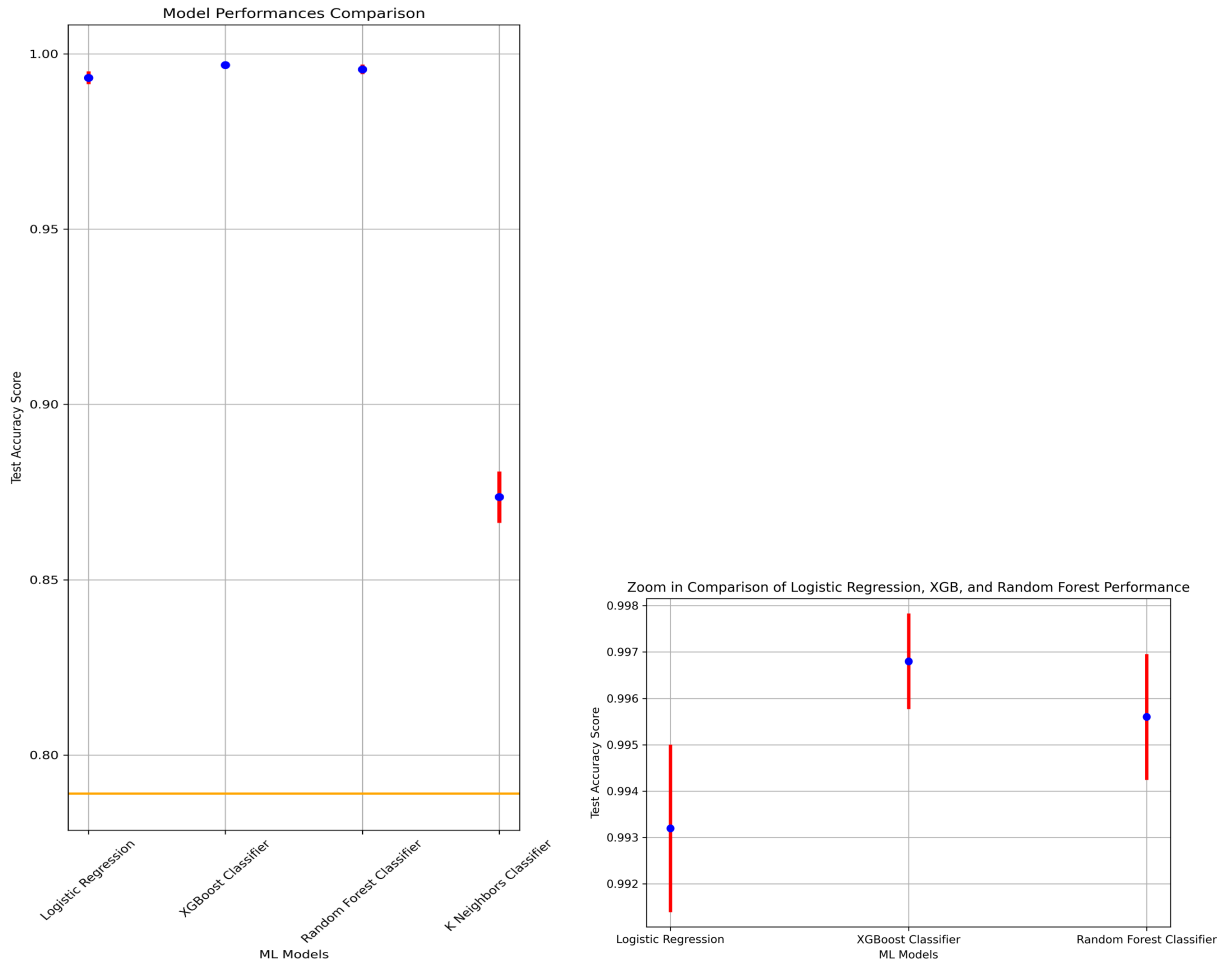

Fig 9. Performance of ML Algorithms

4.2 Best Model Results

Table 3 shows the validation and test scores achieved with the best model. The confusion matrix in Figure 10, with a threshold set at 0.2 to counteract the dataset's imbalance, reveals only 25 misclassifications out of 7,394 data points, demonstrating the model's robust predictive capability. This threshold choice aims to minimize false positives and negatives while maintaining a balanced consideration of accuracy, precision, recall, and F1 scores.

	Accuracy	Precision	Recall	F1 score
Validation Set	0.9981	0.9981	0.9929	0.9955
Test Set	0.9966	0.9974	0.9865	0.9919

Table 3. Performance for Best Model

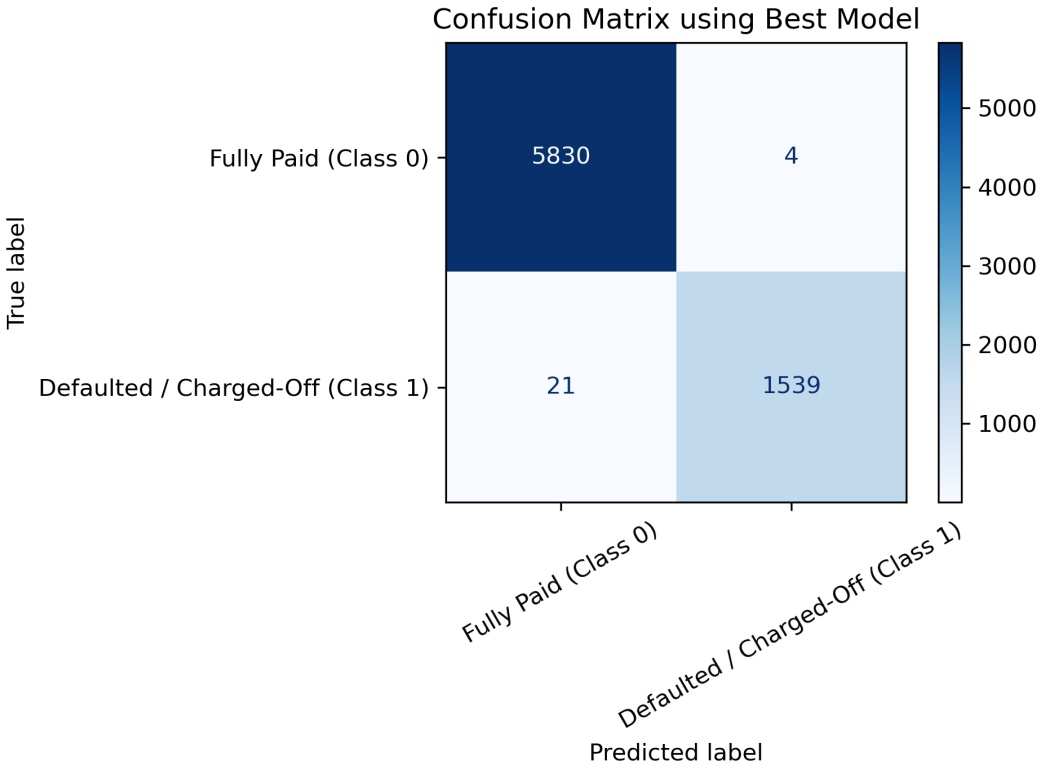

Fig 10. Confusion Matrix for Best Model

4.3 Feature Importance

To understand the key drivers of the model's predictions, three global feature importance metrics were employed: Permutation Importance, XGBoost weight and total_gain metrics, and SHAP values. 'Total_payment' emerged as the most impactful feature, with 'last_payment', 'installment', 'loan_amount', and 'total_received_interest' also consistently ranking high in importance. On the other hand, features like 'address_state', 'application_type', and 'loan_purpose' were among the least influential. This suggests that characteristics of the loan itself might be more critical in predicting loan defaults than borrowers' financial profiles.

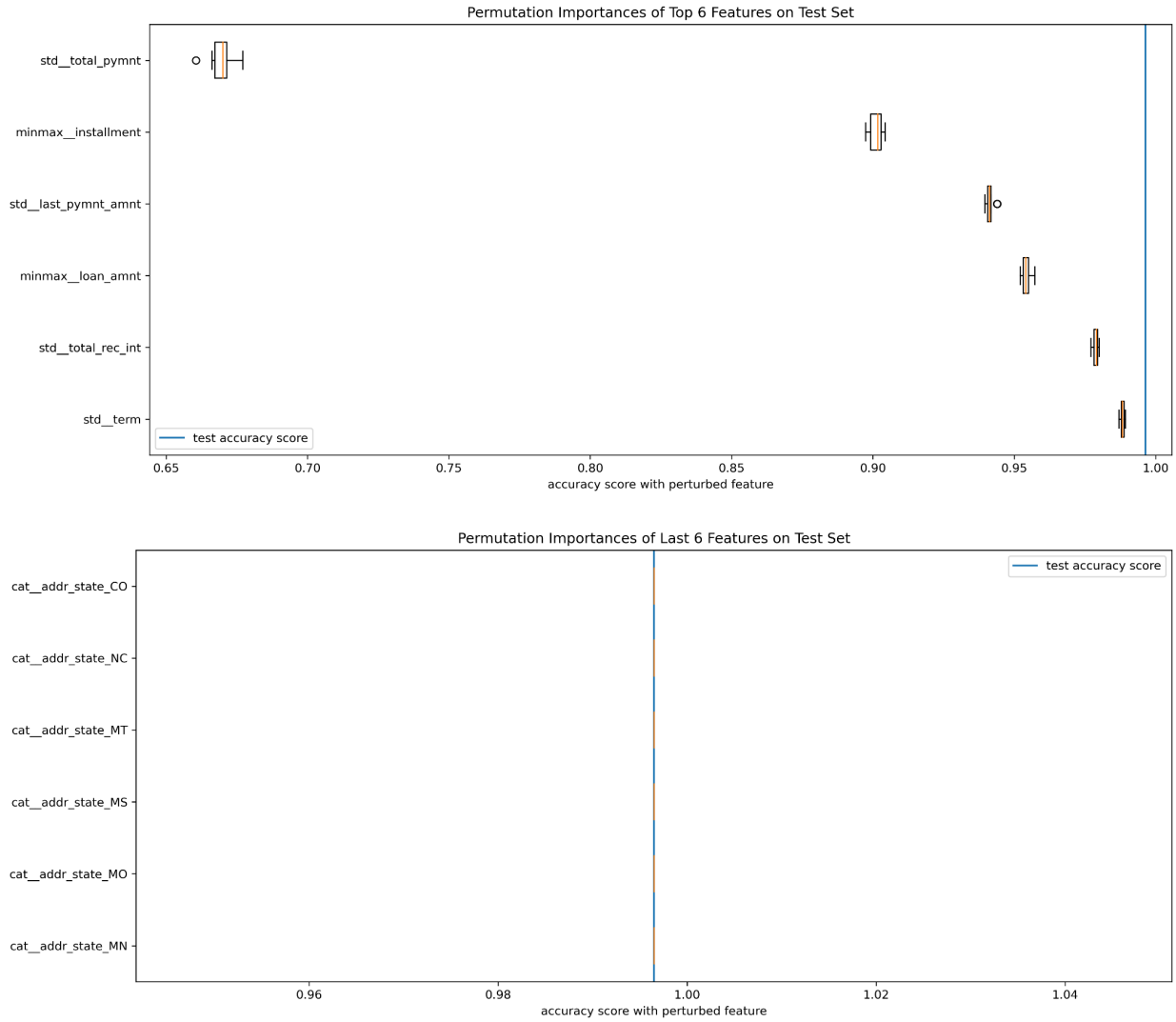

Fig 11. Global Feature Importance - Permutation Importance

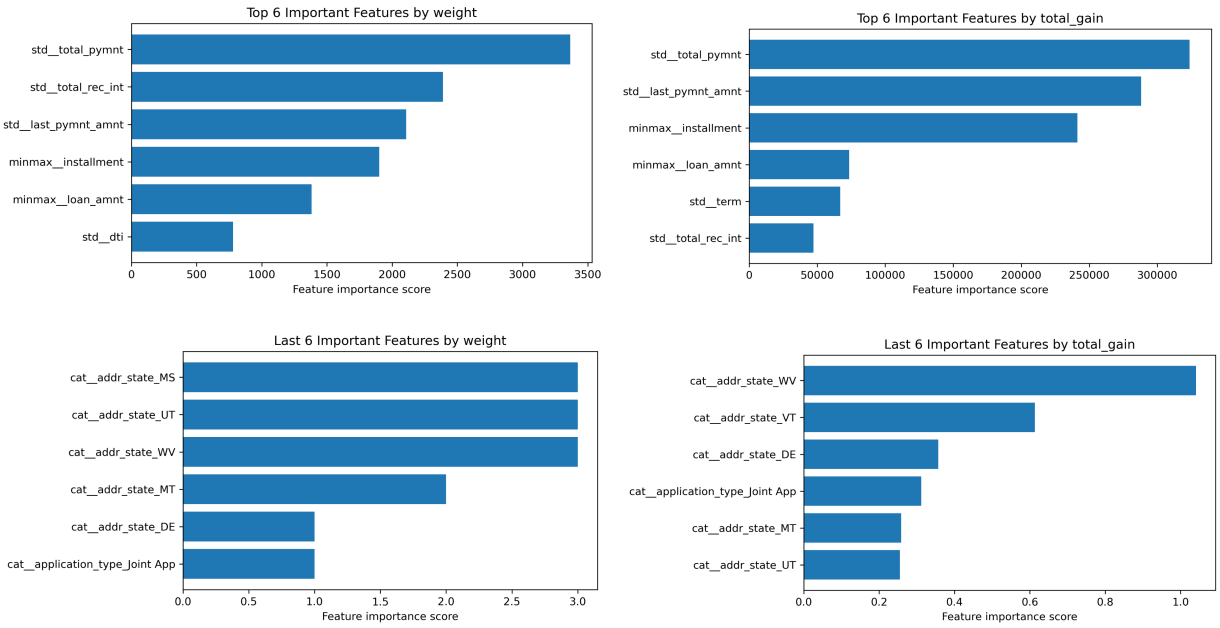

Fig 12. Global Feature Importance - XGBoost metrics

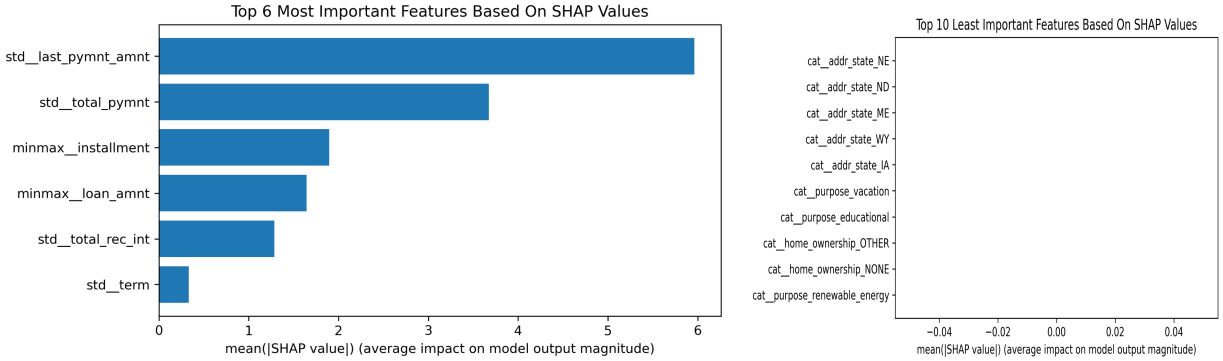

Fig 13. Global Feature Importance - SHAP Value

Local Feature Importance using SHAP values corroborated the significance of features like 'total_payment', 'last_payment_amount', and 'total_received_interest', which were also highlighted in global feature importance. However, for individual data points, specific features could hold relatively higher importance in driving predictions. For instance, loan purpose category 'home_improvement' was a key determinant for point 520, and 'bank_card_open_to_buy' significantly impacted prediction for point 777. This variation underscores the complexity of factors influencing loan default risk at an individual level.

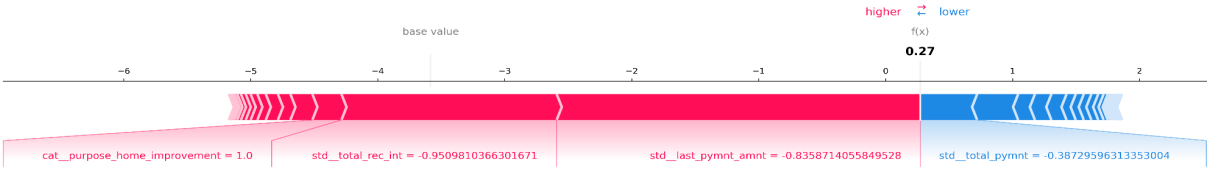

Fig 14. Local Feature Importance - SHAP Value at index 520

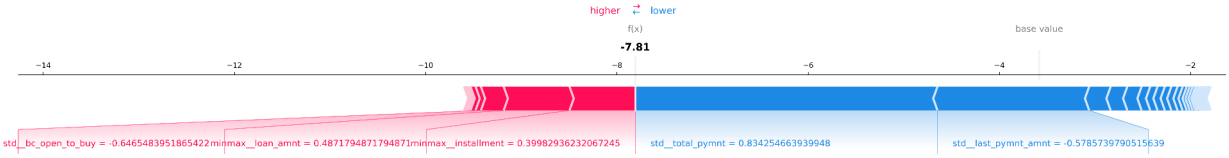

Fig 15. Local Feature Importance - SHAP Value at index 777

5. Outlook

The model outperforms prior work due to a larger dataset that enables detailed pattern recognition without overfitting. Key to its success are well-chosen features and algorithms like XGBoost, which uses early stopping, and Random Forest, which handles class imbalances effectively. Adjusting KNN's class weights further mitigates majority class bias.

Future enhancements include increasing cross-validation folds for robust model assessment and refining hyperparameter tuning for greater generalizability. Exploring more ML and non-ML models, such as SVM and neural networks will be crucial, as errors in predicting loan defaults carry substantial financial implications. Misclassifying potentially default loans as fully paid, or vice versa, carries significant economic consequences for both the borrowers and financial institutions involved. Achieving minimal false positives and negatives is therefore a priority.

6. Reference

[1]Sayah, Fares. "Lending Club Loan Defaulters Prediction." Kaggle, www.kaggle.com/code/faressayah/lending-club-loan-defaulters-prediction.

[2] Sathyanarayan, T. "Case Study 2 - Loan Default Probability." GitHub repository, GitHub, www.github.com/tatsath/fin-ml/blob/master/Chapter%206%20-%20Sup.%20Learning%20-%20Classification%20models/CaseStudy2%20-%20Loan%20Default%20Probability/LoanDefaultProbability.ipynb.

[3] "Lending Club Loans Data." Kaggle, https://www.kaggle.com/datasets/mlfinancebook/lending-club-loans-data.

[4] Github Repository - https://github.com/yxing12/loan-default-predictor