CRS

Post-graduation in Data Science for Finance

Credit Risk Scoring

Group Project

Predicting Credit Default Risk:

A Comparative Analysis of Logistic Regression, Machine Learning, and Deep Learning Models

June 2025

Group 7:

Francisco Perestrello – 20241560

Gonçalo Gomes – 20211007

Nuno Vieira – 20241111

Petr Terletskiy – 20211580

NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

Abstract

This project aims to predict loan default probabilities and assess ongoing loans in unseen test data using a comprehensive consumer loan dataset of approximately 310,000 records with 28 variables, including loan ID, loan status, and numerical and categorical features. The methodology involved extensive exploratory data analysis, preprocessing steps such as outlier removal, missing value imputation, and feature engineering, followed by a dataset split into training and validation sets. Five models—logistic regression, random forest, CatBoost, gradient boosting, and neural networks—were developed and optimized using grid search with cross-validation, evaluated via F1 score, and AUC. Logistic regression achieved 0.77 accuracy, 0.59 F1 Score for the Default Class, and 0.82 AUC, while random forest reached 0.78 accuracy, 0.69 F1 Score for the Default Class, and 0.85 AUC. CatBoost and gradient boosting excelled with 0.94 accuracy, 0.91 F1 Score for the Default Class, and 0.99 AUC, outperforming neural networks (0.78 accuracy, 0.68 F1 Score for the Default Class, 0.85 AUC). Key findings underscore CatBoost and gradient boosting as the most effective due to their handling of complex patterns, though their complexity contrasts with the interpretability of logistic regression.

NOVA IMS Information Management School

Table of Contents

1. Introduction 1

2. Exploratory Data Analysis 2

3. Preprocessing & Feature Engineering 4

4. Modelling Results 6

5. Discussion 8

6. Conclusion 10

Bibliography 11

Annex 12

1. Introduction

Credit risk, the likelihood that a borrower will fail to repay a loan, is a critical concern in the financial industry, directly impacting lending decisions and institutional stability. It arises when borrowers default on their obligations, potentially leading to significant financial losses for lenders. Effective credit risk assessment is thus vital for banks and financial institutions to optimize loan approvals, minimize defaults, and ensure sustainable profitability. In today's data-driven landscape, advanced analytical techniques, such as machine learning, enable more accurate predictions of default probability, enhancing decision-making processes.

The objective of this project is to predict whether the ongoing loans will result in default using a comprehensive consumer loan dataset comprising approximately 310,000 rows and 28 variables, including a unique loan ID, the target variable (loan status), and a mix of numerical (e.g., loan amount, interest rate) and categorical (e.g., loan grade, home ownership) features. By leveraging a diverse set of modeling approaches—Logistic Regression, Random Forest, CatBoost, Gradient Boosting, and Neural Networks—we aim to develop robust predictive models and compare their performance to identify the most effective approach for credit risk assessment.

Hypothesis 1	ML models (Random Forest, CatBoost, Gradient Boosting) will outperform the traditional Logistic Regression due to their ability to capture complex, non-linear relationships in the data.
Hypothesis 2	Neural Networks will provide superior predictive power for large datasets but may require careful tuning to avoid overfitting.
Hypothesis 3	Key features, such as debt-to-income ratio and interest rate, will be among the most influential predictors of default across all models.

Table 1 – Hypotheses to be tested throughout the project

These hypotheses will be tested through rigorous statistical analysis, feature engineering, and model evaluation.

This analysis aims to bridge classroom theory with practical application by developing models that support more informed risk assessment and lending decisions. By evaluating the predictive power of each model and interpreting key drivers of default, our work seeks to provide actionable insights for financial institutions, contributing to more effective credit risk management strategies.

2. Exploratory Data Analysis

This section will cover the analysis of the consumer loan dataset by conducting a comprehensive Exploratory Data Analysis (EDA) of the entire dataset. This process included detailed analysis and visualization of the target variable, as well as all numerical features (both continuous and discrete) and categorical features. By performing this approach, we aimed to uncover patterns, relationships, and potential anomalies within the data, providing a foundational understanding for subsequent steps.

DATASET OVERVIEW

The dataset comprises a total of 310,704 rows and 28 columns, which was divided into a training set (233,028 rows, 28 columns) and validation set (77,676 rows, 28 columns). There is an ID feature, a target feature (*loan_status*), 10 categorical features, and 16 numerical features.

TARGET FEATURE

The target variable was initially represented with multiple categories (Figure 1) reflecting the status of loans, including 'Fully Paid', 'Charged Off', 'Late (31-120 days)', 'In Grace Period', 'Late (16-30 days)', and 'Default'. To align with the project requirement of transforming the response variable into a binary format for default prediction, we redefined *loan_status* into a binary classification: '0' for non-default and '1' for default. This transformation was implemented using a mapping where 'Fully Paid' and 'In Grace Period' were assigned as non-default status, while 'Late (16-30 days)', 'Late (31-120 days)', 'Charged Off', and 'Default' were assigned as default status (Figure 2).

DATASET SPLIT

To ensure no leakage from the validation set, the dataset split was performed immediately after transforming the target feature *loan_status* into a binary format. The split was executed using a stratification based on the target feature values to maintain the proportion of default and non-default instances across both sets.

All subsequent transformations and preprocessing steps were fit exclusively on the training set and then applied to the validation set, preventing the models from inadvertently learning from validation data and ensuring a robust evaluation process.

NUMERICAL FEATURES

The exploratory data analysis of the numerical features within the dataset revealed distinct patterns across both continuous and discrete variables, providing insights into their distributions and their relationship with loan defaults.

For continuous features, such as *loan_amnt*, *funded_amnt*, and *funded_amnt_inv*, the distributions were largely similar, though differences emerged in the largest outliers between defaults and non-defaults. The *int_rate* showed a narrower range for defaulted loans, while *installment* exhibited slightly more skewness for both defaulted and non-defaulted loans. Features like *annual_inc* and *dti* displayed high skewness for both groups with comparable shapes, whereas *revol_bal* appeared less skewed for defaulted loans, and *revol_util* indicated higher credit usage relative to available credit among defaults. The *out_prncp* was heavily concentrated around zero for both defaults and non-defaults, and *total_pymnt* was more right skewed for non-default loans, as anticipated. Visualizations for these distributions can be found in the Annex. Based on these observations, we decided to compute the skewness of the continuous features and consider log transformations for those that are highly skewed to normalize their

distributions. Additionally, a correlation matrix analysis was planned to identify and potentially remove features with high similarity to avoid redundancy.

For discrete numerical features, the analysis uncovered further insights into default behavior. The *delinq_2yrs* feature showed a steady increase in defaults with higher delinquency counts, though the histogram indicated that most observations had three or fewer delinquencies, suggesting that higher default rates at rare counts might have limited significance. This led to a decision to consider binning rare high values or removing outliers to enhance model stability. The *inq_last_6mths* feature exhibited a consistent rise in defaults with more inquiries, requiring no immediate transformation. For *open_acc*, defaults were slightly more prevalent at higher open account counts and when it was the client's only credit line, with the distribution peaking at low counts, prompting consideration of capping or binning extreme values (e.g., >30) to reduce skew. The *pub_rec* feature displayed erratic default patterns beyond 10 public records, attributable to most clients having none, leading to a consideration to remove rare high values or even exclude the feature due to low information value. Lastly, *total_acc* showed a slight decrease in defaults as total credit lines increased up to 80-90, where the distribution was most concentrated, with erratic behavior beyond due to low sample sizes.

In sum, in the preprocessing section, we will focus on applying log transformations to highly skewed continuous features, removing outliers where necessary, and eliminating redundant features based on correlation analysis to prepare the dataset for effective modeling.

CATEGORICAL FEATURES

The analysis of categorical features began with an examination of the *home_ownership* variable, where default rates across the main categories (MORTGAGE, RENT, OWN) showed relative consistency, suggesting limited standalone predictive power, while the infrequent 'ANY' category exhibited an anomalously low default rate likely due to insufficient sample size. To enhance its utility, we decided to derive two binary indicators: *owns_house* (0 for RENT and ANY, 1 for MORTGAGE and OWN) to capture ownership status, and *house_expenses* (0 for OWN and ANY, 1 for MORTGAGE and RENT) to reflect recurring housing obligations.

Then, the *verification_status* feature revealed a counterintuitive trend: borrowers marked as 'Not Verified' had a significantly lower default rate compared to those labeled 'Verified' or 'Source Verified', which are functionally similar. This suggests that loans requiring verification might implicitly carry higher risk. To improve interpretability and model performance, we merged 'Verified' and 'Source Verified' into a single category and created a binary *is_verified* feature (1 for Verified or Source Verified, 0 for Not Verified).

The *term* feature, a binary indicator of loan duration (36 or 60 months), showed modest differences in default rates, leading to a decision to encode it as 0 for 36 months and 1 for 60 months. The *issue_d* feature, capturing loan issuance date without the year, was found to lack temporal context and introduce noise, resulting in a decision to drop it. Further analysis focused on *grade* demonstrated a clear increasing trend in default rates from A to F (Figure 20), reinforcing its ordinal nature despite minor deviations likely due to small sample sizes, leading to an encoding decision where grades are mapped to integers (A=7, B=6, ..., F=2, G=1) to preserve this hierarchy. For *purpose*, most loans were linked to debt consolidation, naturally dominating defaulted loans as well, followed by credit card and home improvement loans (Figure 22). Rarer purposes were underrepresented, leading to a decision to retain only the top three categories and reclassify others as 'other' to reduce sparsity.

The *emp_length* feature, an ordinal indicator of employment duration, exhibits a concentration of loans among borrowers with over 10 years of experience. Missing values in *emp_length* frequently coincide with missing *emp_title*, which often implies unemployment. Accordingly, missing *emp_length* entries are imputed as 'Unemployed' when *emp_title* is also missing, and as 'Unknown' otherwise. Cleaned values are then mapped to integers as follows: Unknown = 0, Unemployed = 1, <1 year = 2, 1 year = 3, ..., 10 years = 12, 10+ years = 13. Although the default rate does not follow the ordinal progression of employment length, the feature remains classified as ordinal due to its inherent structure (Figure 23). A noticeable preference toward lending to individuals with stable employment is evident from the higher frequency of loans issued to borrowers with longer employment histories. In turn, *emp_title*, with its high number of unique free-text entries (Figure 24) and missing values coinciding with

emp_length gaps, offered limited standardized value, and its information can be largely captured by *annual_inc* and *emp_length*, leading to a decision to drop it. Similarly, *earliest_cr_line*, despite its potential to reflect credit history, suffered from inconsistent formatting, rendering it unusable and prompting its removal.

Lastly, the *addr_state* feature, representing U.S. states, was deemed suitable for transformation into the 9 U.S. Census Bureau divisions to reduce dimensionality and capture regional patterns effectively.

These insights will guide the preprocessing phase to implement the proposed transformations, encodings, and feature drops to optimize the dataset for modeling.

3. Preprocessing & Feature Engineering

After our initial exploration, we moved to preprocessing and feature engineering to get the data ready for modeling. For our numerical features, we checked and addressed inconsistencies such as duplicate IDs, identified and handled outliers, imputed any missing values, and normalized and scaled the variables. Our categorical features underwent a transformation process that included further feature engineering and encoding to convert them into a suitable format for modeling. Finally, we performed feature selection by conducting a thorough correlation analysis to identify the most relevant features for our models.

NUMERICAL FEATURES

The preprocessing of numerical features began with an assessment of data integrity, revealing no duplicate loan IDs across the dataset, ensuring each record's uniqueness.

To address outliers, the Interquartile Range (IQR) method with a threshold of 3 was used, incorporating log transformation for eligible continuous features to stabilize variance. This process identified and removed 5,438 outliers (2.33%) from the training set, and 1,795 outliers (2.31%) from the validation set using the train set's IQR, thereby enhancing the dataset's robustness for modeling.

Missing value analysis showed minimal gaps, with *dti*, *revol_util*, and *inq_last_6mths* having missing values in both training and validation sets. Imputation was performed using statistics derived solely from the training set—mean for continuous features, median for discrete, and mode for categorical—effectively eliminating all missing values in both sets and ensuring there is no leakage in between the training and the validation sets.

Feature engineering followed to enrich the dataset with new numerical variables designed to capture deeper insights into credit risk. The *dti_with_loan* feature was created by multiplying the debt-to-income ratio (*dti*) by the ratio of loan amount to annual income, offering a refined measure of debt management capacity. The *payment_to_income* ratio was computed as the installment divided by annual income, highlighting the financial strain from monthly payments. The *util_to_loan* feature combined revolving utilization (*revol_util*) and balance (*revol_bal*) with loan amount, assessing overall credit usage relative to the loan. Additionally, the *delinq_impact* score multiplied past delinquencies (*delinq_2yrs*) by the ratio of loan amount to total accounts (*total_acc*), reflecting the severity of historical credit issues. The *remaining_principal_ratio*, calculated as outstanding principal (*out_prncp*) divided by loan amount, provided insight into repayment progress, while the *payment_efficiency* metric, derived from total payments (*total_pymnt*) divided by the sum of loan amount and outstanding principal, evaluated payment effectiveness.

These engineered features aim to enhance the predictive power of the subsequent modeling phase by incorporating more nuanced financial indicators.

CATEGORICAL FEATURES

The preprocessing of categorical features started with transforming the *home_ownership* variable into two binary indicators to better reflect housing-related risk factors, where *owns_house* was set to 1 for 'MORTGAGE' and 'OWN' and 0 for 'RENT' and 'ANY', and *house_expenses* was set to 1 for 'MORTGAGE' and 'RENT' and 0 for 'OWN' and 'ANY', after which the original *home_ownership* column was dropped to streamline the dataset.

For *verification_status*, the categories 'Verified' and 'Source Verified' were consolidated into a single group, creating a binary *is_verified* feature with 1 for these verified statuses and 0 for 'Not Verified', followed by the removal of the original column to enhance model interpretability. The *term* feature, representing loan duration, was encoded as a binary variable with 0 for '36 months' and 1 for '60 months' to maintain its structural simplicity, while the ordinal *grade* feature was mapped to integers from 7 ('A') to 1 ('G') to preserve its risk hierarchy, aligning with observed default trends.

Further processing involved addressing less informative or noisy features, starting with the removal of *issue_d* due to its lack of year context, which rendered it unhelpful for temporal analysis. The *purpose* feature was refined by retaining only the top three categories—'debt_consolidation', 'credit_card', and 'home_improvement'—and reclassifying all other purposes as 'other' to reduce sparsity and improve generalization.

The *emp_length* feature underwent enhanced imputation, where missing values were set to 'Unemployed' if both *emp_length* and *emp_title* were missing, or 'Unknown' if only *emp_length* was missing, followed by mapping to integers from 0 ('Unknown') to 13 ('10+ years') to reflect its ordinal nature and employment stability. Consequently, the *emp_title* column, with its high variability and redundant information mostly covered by *annual_inc* and *emp_length*, was dropped, as was *earliest_cr_line* due to its inconsistent formatting that hindered meaningful extraction of credit history insights.

The final transformation targeted the *addr_state* feature, which was converted into the corresponding U.S. Census Bureau division (e.g., 'CT' to 'New England') using a predefined mapping to capture regional economic patterns, after which the original *addr_state* was removed to reduce dimensionality.

This comprehensive preprocessing and feature engineering approach enhanced the dataset's quality and relevance for predicting loan default probabilities.

FEATURE NORMALIZATION AND ENCODING

The normalization of numerical features began with addressing skewness, where log transformations were applied to handle heavy right-skewed continuous features (e.g.: *annual_inc* - Figure 8) to improve model performance. A skewness analysis on the training data identified features with skewness above 1.0 as highly skewed, leading to the transformation of nine features, while features with moderate or low skewness (skewness lower than 1.0) were left untransformed.

Subsequently, Min-Max Scaling was implemented to scale all numerical features, including the log-transformed ones, to a [0, 1] range, fitting the scaler on the training data and applying it to both training and validation sets to ensure compatibility with deep learning models, logistic regression, and other machine learning algorithms.

For categorical feature encoding, a one-hot encoding approach was adopted to transform variables such as *grade* and *addr_division* into a suitable format for modeling. The used one-hot encoder created binary columns for all but the last category of each feature to avoid multicollinearity, storing the used categories and dropping the original columns.

FEATURE SELECTION

The feature selection process focused on eliminating highly correlated numerical features (Figure 25) to enhance model efficiency and prevent multicollinearity. In this step, we identified pairs of features with absolute correlation coefficients exceeding a threshold of 0.8, subsequently calculating the mean absolute correlation of each feature with all others to determine which to retain, favoring the feature with lower average correlation to preserve the most unique information. As a result, six features were removed due to their high correlation: *funded_amnt*, *funded_amnt_inv*, *grade*, *loan_amnt*, *remaining_principal_ratio*, *util_to_loan*.

4. Modeling Results

This section details the outcomes of the modeling efforts aimed at predicting loan default probabilities, utilizing a consistent methodology across all approaches. Each model was optimized using grid search with five-fold cross-validation, focusing on the F1 score to ensure a balanced evaluation of precision and recall, complemented by the Area Under the Curve (AUC) to gauge discriminative power.

The process involved tuning hyperparameters specific to each algorithm, selecting the best configuration based on training data performance, and validating the results on a separate set with metrics including F1 score, AUC, and classification reports. Additionally, feature importance was visualized for the top 15 contributors in each model to highlight key predictors.

LOGISTIC REGRESSION

The logistic regression model leveraged a linear framework enhanced by 'L1' (Lasso) and 'L2' (Ridge) regularization to mitigate overfitting, with the grid search exploring penalty types and strengths via the 'C' parameter alongside solver options. The selected model, optimized for F1 score, provided a clear view of feature influence through coefficient analysis, revealing the most impactful variables in default prediction.

The best model parameters, determined through the grid search, included a 'C' value of 10, an 'L1' penalty, and the 'liblinear' solver, reflecting a strong preference with the Lasso penalty to enhance sparsity and the 'liblinear' solver for efficient convergence. The optimal model achieved a mean cross-validation F1 score of 0.5880 and a validation F1 score of 0.5867, indicating modest performance in balancing precision and recall. The validation AUC of 0.8213 further underscores the model's strong discriminative ability, while the classification report highlights a precision of 0.70 and recall of 0.51 for the default class (*loan_status* = 1), suggesting a moderate ability to identify defaults, tempered by a higher precision (0.80) and recall (0.90) for non-defaults (*loan_status* = 0).

The most impactful features, as shown in the feature importance plot (Figure 27), include *payment_to_income* and *pub_rec* with the highest coefficient magnitudes, indicating their significant influence on default probability, alongside *total_pymnt* and *payment_efficiency*, which collectively shape the model's predictive power for *loan_status*.

NEURAL NETWORKS

In order to use a Deep Learning model, we implemented a binary classification neural network. A key consideration in this setup was the treatment of validation data, given that, unlike the other machine learning models in our study (which were trained using K-Fold cross-validation), this model was trained using a single training-validation split.

To preserve a consistent and fair comparison with the other models, we decided to set aside the original validation set (defined at the beginning of the notebook) as a final test set, ensuring it remained completely untouched during model training and hyperparameter tuning.

Within the training data, we performed an additional train-validation split (using an 80-20 stratified split). This new validation subset was used exclusively to monitor model performance during training (e.g., for early stopping and class balancing), while the final original validation set would later serve for out-of-sample model evaluation.

Since the dataset is imbalanced, as is common in credit risk applications, we computed class weights and applied them during model training to penalize misclassification of the minority class more heavily.

The model architecture consists of a simple feedforward neural network:

- An input layer based on the number of features in the dataset;
- Two hidden layers with 64 neurons each and ReLU activations;
- A final output layer with a single neuron and sigmoid activation for binary classification.

We compiled the model using the 'Adam' optimizer and 'binary cross-entropy' loss, along with AUC and accuracy as evaluation metrics. The training process included early stopping, which monitors validation loss and halts training if no improvement is observed for a given number of consecutive epochs (set to 10), while restoring the best-performing model weights.

The model trained for 31 epochs, achieving a validation AUC of 0.8516 and a validation accuracy of 76.7%, as shown in the output log. These metrics demonstrate that the model captures a significant portion of the signal needed to distinguish between defaulting and non-defaulting loans. However, further evaluation is needed in the held-out validation set (untouched during training).

The classification report (Figure 28) shows that the neural network achieved an F1 score of 0.68 for the default class (loan_status = 1), with a recall of 0.73 and precision of 0.63, indicating the model is relatively effective at identifying true defaulters, though at the cost of some false positives. For the non-default class (loan_status = 0), the model achieved an F1 score of 0.83, with precision of 0.87 and recall of 0.80, suggesting strong ability to correctly identify non-defaulters. The overall accuracy of 78%, along with a macro average F1 score of 0.75, reflects a balanced performance across both classes. The confusion matrix further supports these findings, revealing 17,765 true positives and 6,426 false negatives for class 1, reinforcing the model's sensitivity in detecting defaults.

RANDOM FOREST

For the random forest, the approach incorporated balanced subsample weighting to address class imbalance, with the grid search tuning the number of trees, maximum depth, minimum samples per leaf, and maximum features. The resulting model offered robust predictions, with feature importance rankings shedding light on the most significant contributors to default risk.

The optimal configuration featured a maximum tree depth of 15, utilized the square root of features for each split, required a minimum of 5 samples per leaf, and included 100 estimators, striking a balance between depth and feature sampling to enhance predictive accuracy while mitigating overfitting. This setup yielded a mean cross-validation F1 score of 0.6894 and a validation F1 score of 0.6855, indicating strong precision-recall balance. The validation AUC of 0.8549 further highlights excellent discriminative power, with the classification report (Figure 30) showing a precision of 0.63 and recall of 0.76 for the default class (*loan_status* = 1), suggesting effective identification of defaults, complemented by a precision of 0.87 and recall of 0.79 for non-defaults (*loan_status* = 0).

The most impactful features, as shown in the importance plot (Figure 31), include *total_pymnt* and *payment_efficiency* with the highest scores, followed by *int_rate* and *out_prncp*, underscoring their critical role in shaping the model's *loan_status* predictions.

CATBOOST

The CatBoost model utilized automatic class weight balancing to handle imbalanced data, with the grid search adjusting iterations, learning rate, tree depth, and 'L2' leaf regularization. This configuration delivered strong predictive performance, with feature importance visualization underscoring the critical variables driving the outcomes.

The optimal parameters, set at a depth of 6, 200 iterations, an L2 leaf regularization of 3, and a learning rate of 0.2, reflect a well-tuned balance that maximizes model depth and learning speed while maintaining regularization, achieving a mean cross-validation F1 score of 0.9130 and a validation F1 score of 0.9107, marking it as one of the two best-performing models. The validation AUC of 0.9888 indicates exceptional discriminative ability, while the classification report (Figure 32) reveals a precision of 0.86 and recall of 0.96 for the default class (*loan_status* = 1), demonstrating excellent default detection, paired with a precision of 0.98 and recall of 0.93 for non-defaults (*loan_status* = 0).

The most impactful feature, as highlighted in the importance plot (Figure 33), is *int_rate* with the highest score, followed by *out_prncp* and *total_pymnt*, emphasizing their pivotal role in shaping the model's accurate *loan_status* predictions.

GRADIENT BOOSTING

The gradient boosting model was refined by exploring estimators, learning rate, maximum tree depth, and subsample ratio, ensuring a robust structure through the grid search process. The optimized model highlighted key features influencing default risk through its importance scores, demonstrating effective generalization across the validation set.

The best configuration, featuring a learning rate of 0.2, a maximum tree depth of 5, 100 estimators, and a subsample ratio of 0.8, struck an effective balance between learning speed and tree complexity, achieving a mean cross-validation F1 score of 0.9119 and a validation F1 score of 0.9099, positioning it as one of the two best-performing models alongside CatBoost. The validation AUC of 0.9867 reflects outstanding discriminative capability, while the classification report (Figure 34) indicates a precision of 0.90 and recall of 0.92 for the default class (*loan_status* = 1), showcasing strong default detection, alongside a precision of 0.96 and recall of 0.95 for non-defaults (*loan_status* = 0).

The most influential features, as depicted in the importance plot (Figure 35), include *int_rate* and *total_pymnt* with the highest scores, followed by *out_prncp* and *revol_bal*, once again underscoring their significant contribution to the model's accurate *loan_status* predictions.

5. Discussion

In this study, several classification models were developed to predict credit default risk, namely logistic regression, random forest, gradient boosting, CatBoost, and a deep learning neural network. The models were evaluated on a held-out validation set using metrics such as F1 score, AUC, precision, and recall. While all models demonstrated reasonable performance, CatBoost stood out with the highest validation F1 score (0.91) and AUC (0.99), making it the most effective model for this specific credit scoring problem.

The logistic regression model, although more interpretable, underperformed with a validation F1 score of 0.5867 and AUC of 0.82. It struggled particularly in recall (0.51) for the positive class (defaults), which limits its reliability in high-stakes decision environments where false negatives are costly. However, it remains a valuable baseline due to its transparency and ease of interpretation, especially when regulatory explainability is paramount. Its reliance on features like *payment_to_income* and *pub_rec* aligns with common credit risk factors, which builds stakeholder trust even if predictive power is modest.

The random forest model offered moderate improvement, reaching a validation F1 score of 0.6855 and AUC of 0.85. It provided better recall (0.76) while maintaining interpretability through feature importance rankings, with *total_pymnt* and *payment_effienciency* as top drivers. Compared to logistic regression, it offers a better trade-off between performance and interpretability, making it suitable for organizations looking for both robustness and a degree of explainability.

Gradient boosting and CatBoost both delivered strong predictive performance, but CatBoost outperformed across all metrics: F1 (0.91), AUC (0.99), and precision/recall balance. Notably, it achieved a high recall (0.96) for defaults while maintaining strong precision (0.86), suggesting the model can identify risky applicants without generating excessive false positives. Despite being more complex, CatBoost handles categorical data efficiently and demonstrated strong generalization with minimal overfitting. Its dominant reliance on *int_rate* and *out_prncp* as predictive features is also intuitively aligned with credit lending logic.

The deep learning model yielded reasonable results (F1 = 0.6643), outperforming logistic regression and closely matching random forest. However, it lagged behind both gradient boosting and CatBoost. Furthermore, while it offers flexibility in modeling non-linearities, it lacks transparency and explainability, making it harder to justify to credit risk officers or regulators. Its black-box nature also limits its use in risk-sensitive environments where interpretability is essential.

From a practical standpoint, CatBoost's superior performance makes it the most appropriate model for real-world credit risk applications. It can support automated accept/reject decisions, with its high recall minimizing the chance of overlooking high-risk borrowers. Furthermore, CatBoost could support a risk-based pricing strategy, where borrowers with higher predicted default probabilities receive higher interest rates, improving risk-adjusted returns. This enables lenders to personalize credit terms based on risk, aligning pricing with expected loss.

Despite these strengths, several limitations must be noted. First, the dataset presents class imbalance, which although addressed through class weights and threshold tuning, may still influence model calibration. Secondly, some features (eg., *int_rate*) dominate predictive power, which could signal information leakage or overly strong correlation with the target.

In future work, it would be valuable to explore ensemble methods that combine the strengths of multiple models and test robustness under different economic cycles to offer deeper insights into business implications.

Model	Accuracy	F1 (Class 1)	Val. AUC
Logistic Regression	0,77	0,59	0,82
Neural Network	0,78	0,68	0,85
Random Forest	0,78	0,69	0,85
CatBoost	0,94	0,91	0,99
Gradient Boosting	0.94	0,91	0,99

Table 2 – Comparative Performance Metrics of Credit Risk Prediction Models

6. Conclusion

This project addressed the critical challenge of predicting loan default probabilities, including evaluating ongoing loans in unseen test data, using a dataset of approximately 310,000 records with 28 variables, with the primary hypothesis that machine learning models (Random Forest, CatBoost, Gradient Boosting) would outperform traditional Logistic Regression due to their capacity to capture complex, non-linear relationships. The findings largely support this hypothesis, as CatBoost and Gradient Boosting achieved the highest performance with 0.94 accuracy, 0.91 F1 Score for the Default Class, and 0.99 AUC, significantly surpassing Logistic Regression's 0.77 accuracy, 0.59 F1 Score for the Default Class, and 0.82 AUC, reflecting their superior ability to model intricate credit risk patterns.

However, Hypothesis 2 was not fully validated, as Neural Networks lagged behind with 0.78 accuracy, 0.68 F1 Score for the Default Class, and 0.85 AUC, likely due to overfitting despite tuning efforts, highlighting the need for more robust regularization. Hypothesis 3 was strongly confirmed, with features like *int_rate* and debt-to-income-related metrics consistently ranking among the top predictors across models, underscoring their pivotal role in default risk assessment.

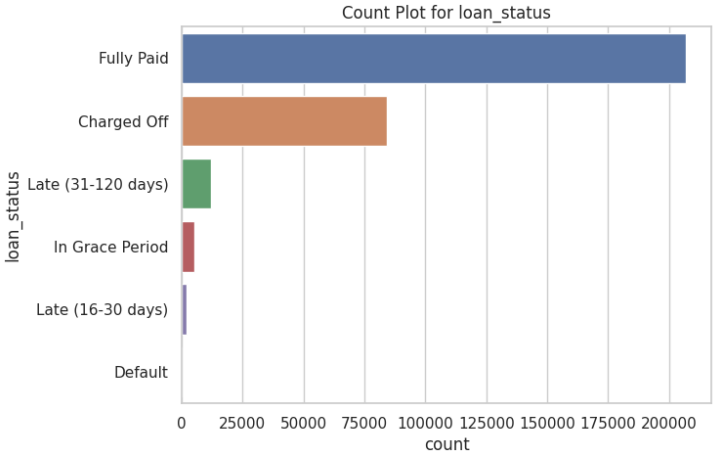
These insights enable enhanced accept/reject decisions and risk-based pricing strategies, though limitations such as data imbalance and training time suggest future work should focus on larger, balanced datasets and optimized computational approaches to refine predictive accuracy and practical applicability.

NOVA IMS Information Management School

Bibliography

Ashofteh, A. (2023). Big Data for Credit Risk Analysis: Efficient Machine Learning Models Using PySpark. In: Pilz, J., Melas, V.B. and Bathke, A., eds. *Statistical Modeling and Simulation for Experimental Design and Machine Learning Applications*. Springer, pp. 245–259.

Annex



loan_status	count
Fully Paid	207036
Charged Off	84180
Late (31-120 days)	12154
In Grace Period	5151
Late (16-30 days)	2162
Default	21

Figure 1 – Loan Status Distribution Pre-process

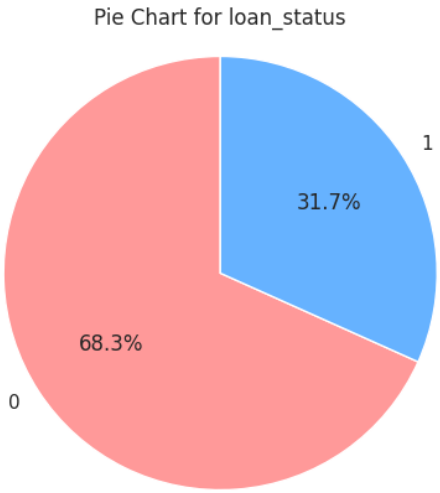


Figure 2 – Loan Status Distribution Post-process

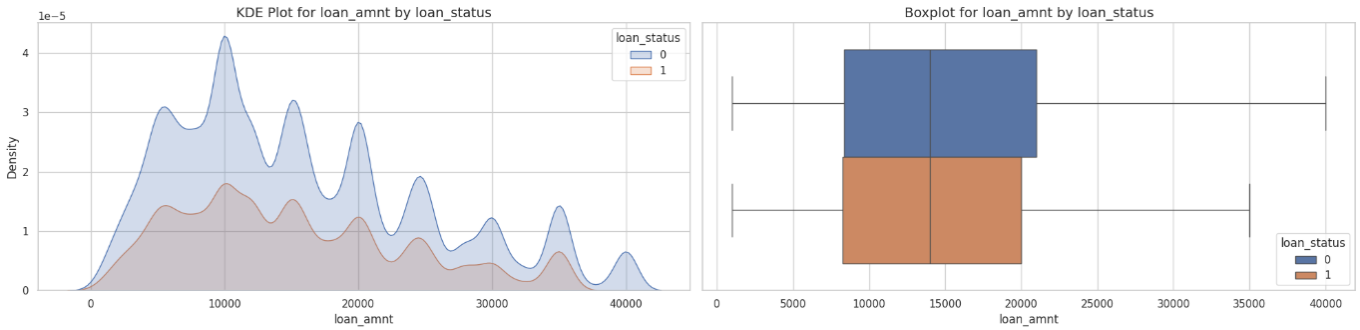


Figure 3 – KDE Plot and Boxplot for *loan_amnt* by Loan Status

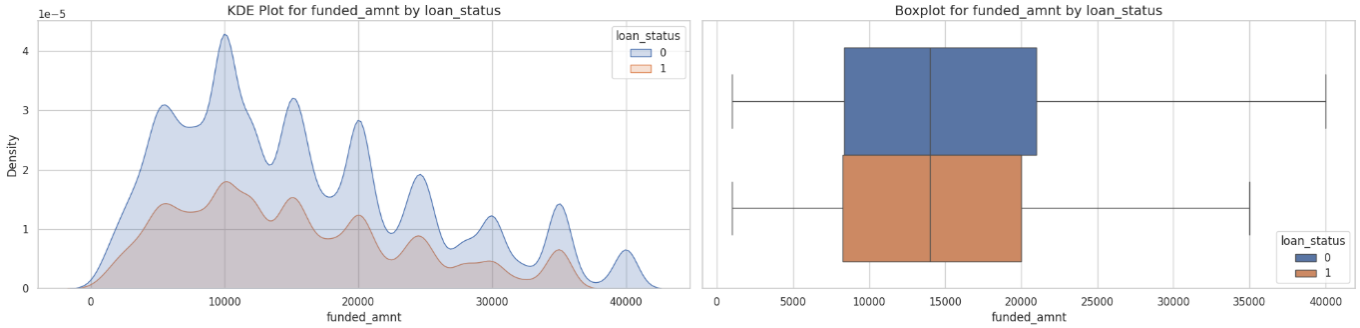


Figure 4 – KDE Plot and Boxplot for *funded_amnt* by Loan Status

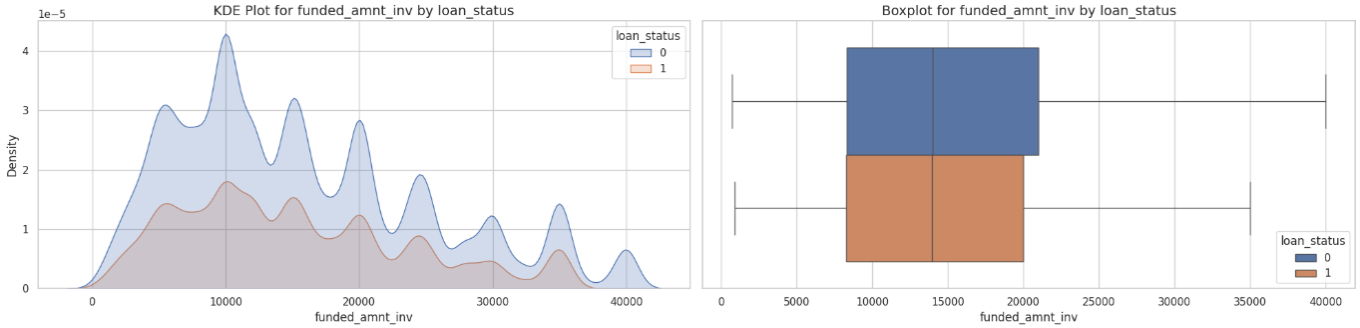


Figure 5 – KDE Plot and Boxplot for *funded_amnt_inv* by Loan Status

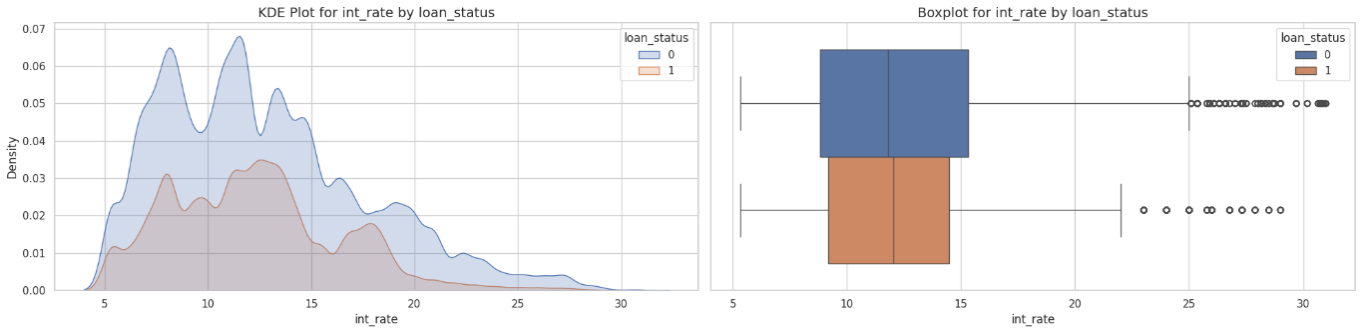


Figure 6 – KDE Plot and Boxplot for *int_rate* by Loan Status

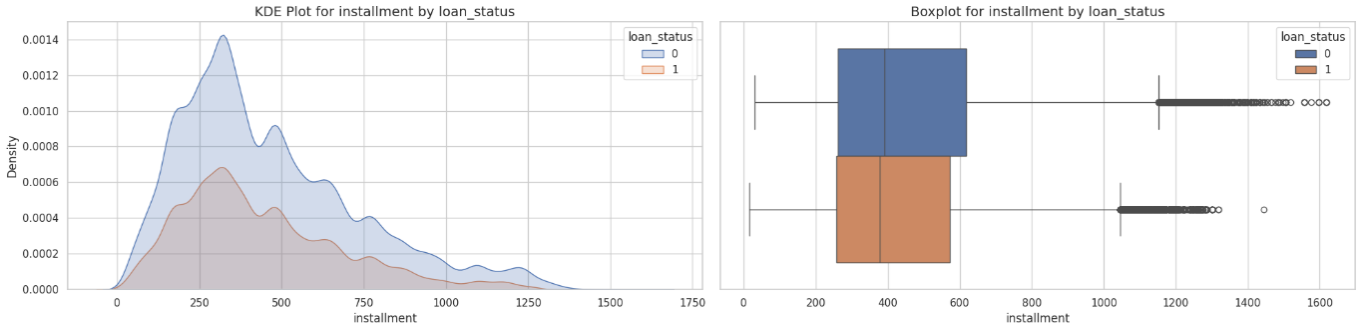


Figure 7 – KDE Plot and Boxplot for *installment* by Loan Status

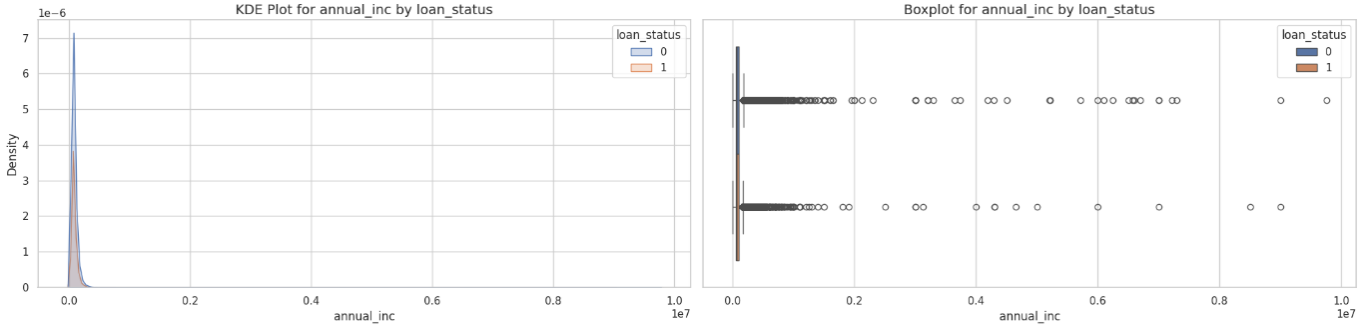


Figure 8 – KDE Plot and Boxplot for *annual_inc* by Loan Status

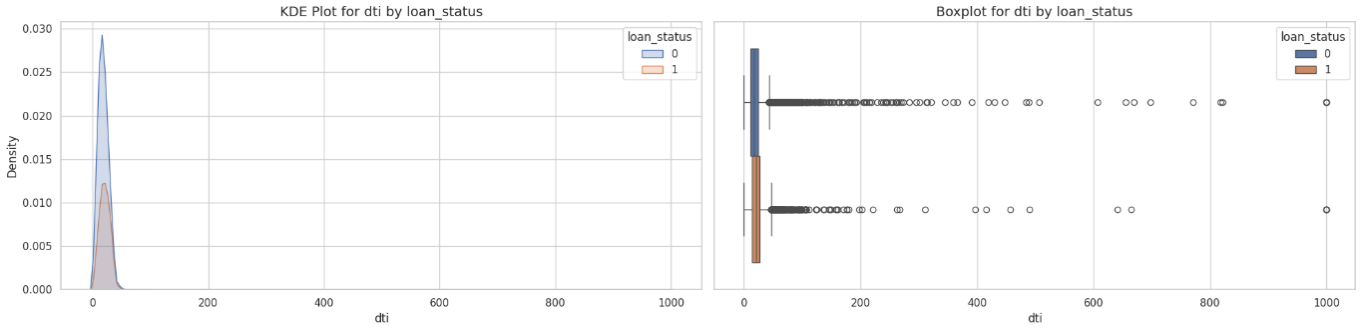


Figure 9 – KDE Plot and Boxplot for *dti* by Loan Status

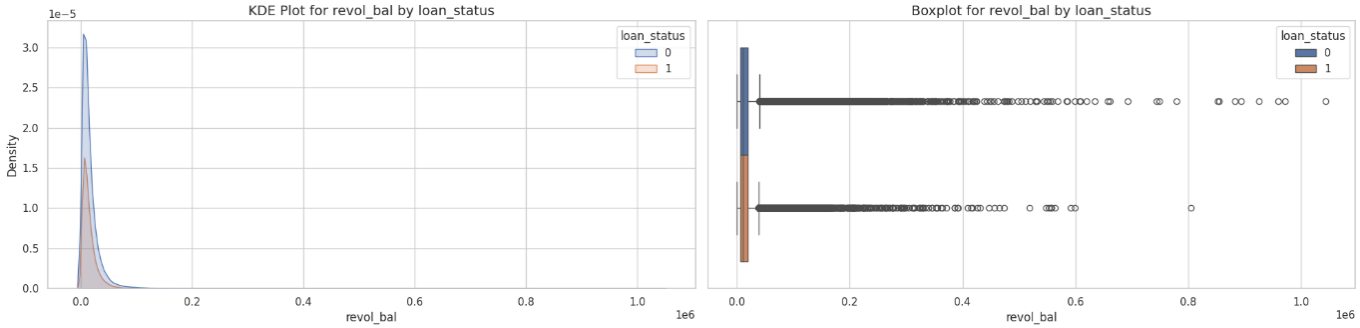


Figure 10 – KDE Plot and Boxplot for *revol_bal* by Loan Status

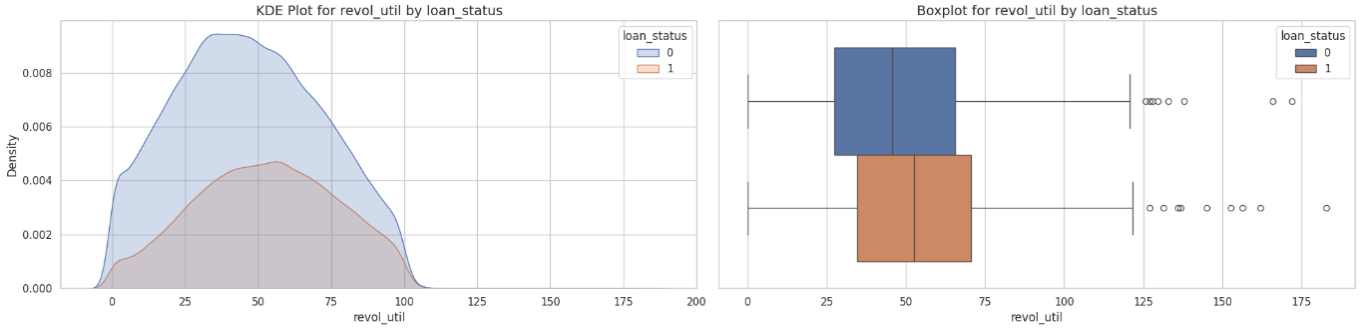


Figure 11 – KDE Plot and Boxplot for *revol_util* by Loan Status

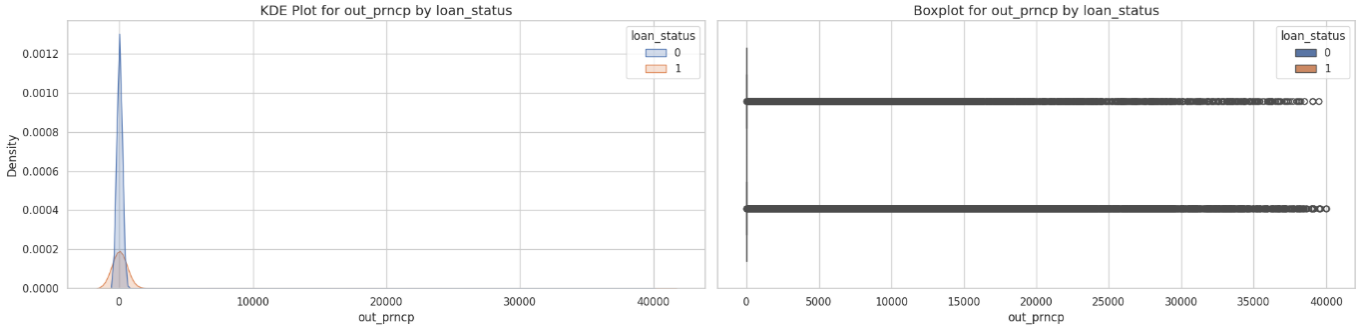


Figure 12 – KDE Plot and Boxplot for *out_prncp* by Loan Status

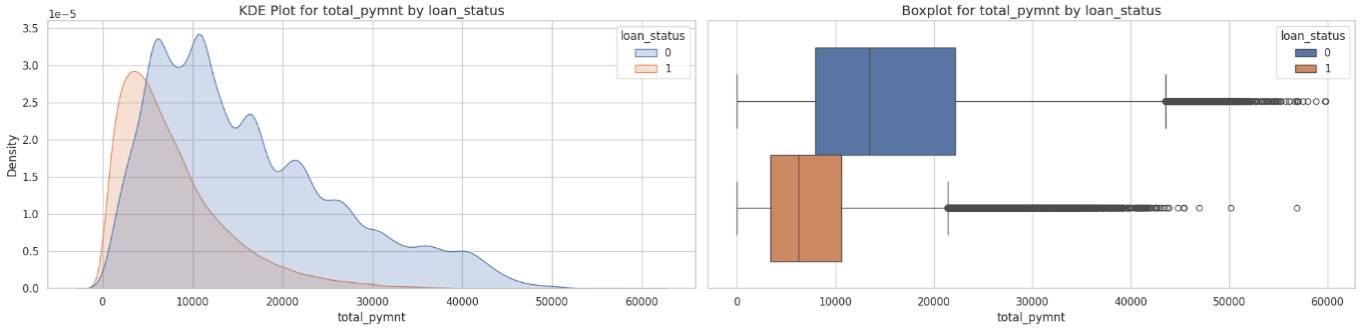


Figure 13 – KDE Plot and Boxplot for *total_pymnt* by Loan Status

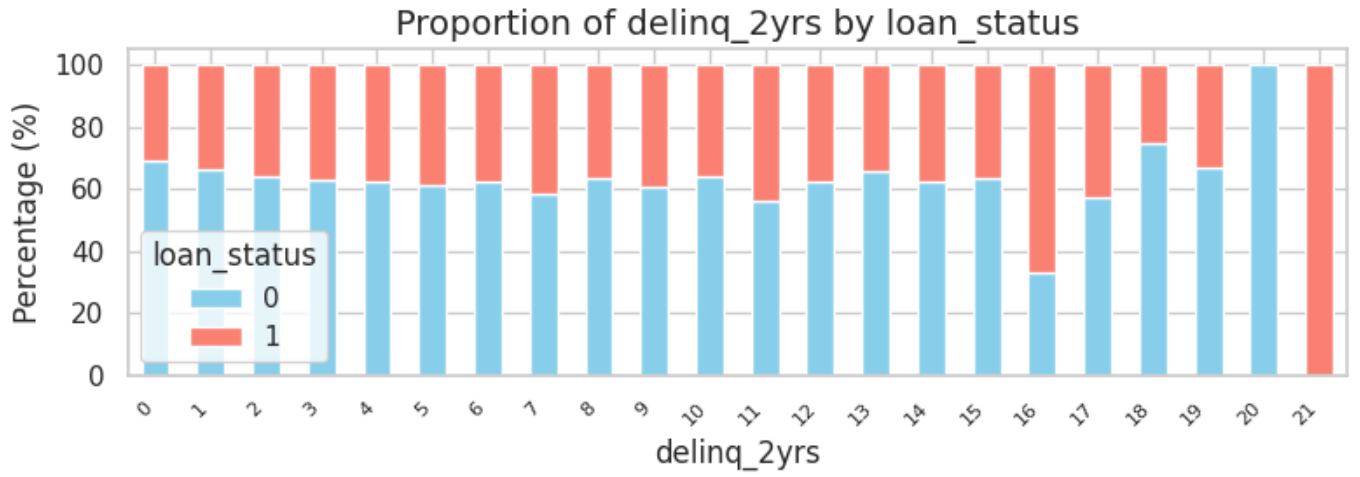


Figure 14 – Proportion of *delinq_2yrs* by Loan Status

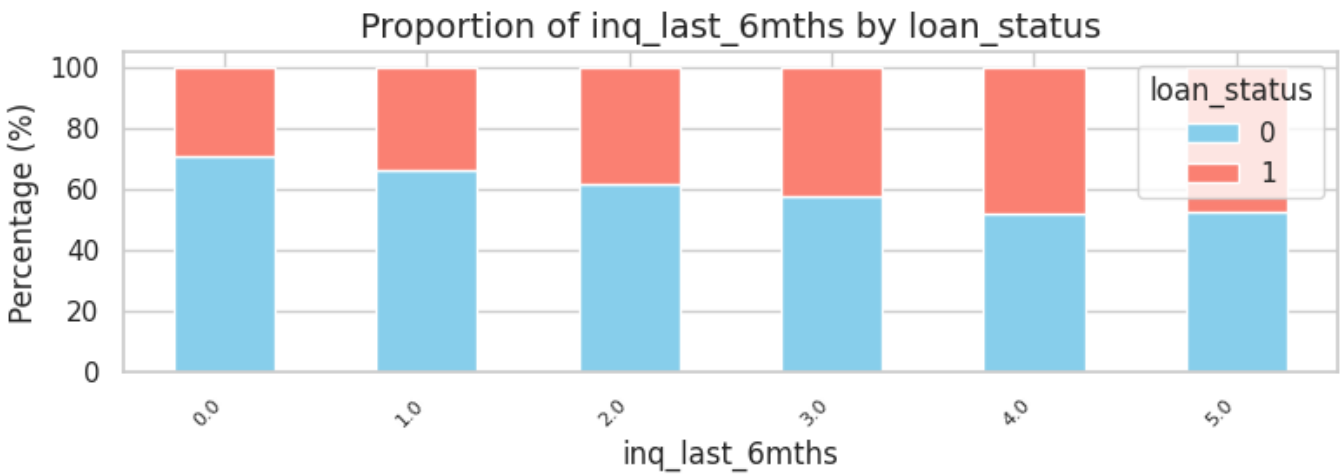


Figure 15 – Proportion of *inq_last_6mths* by Loan Status

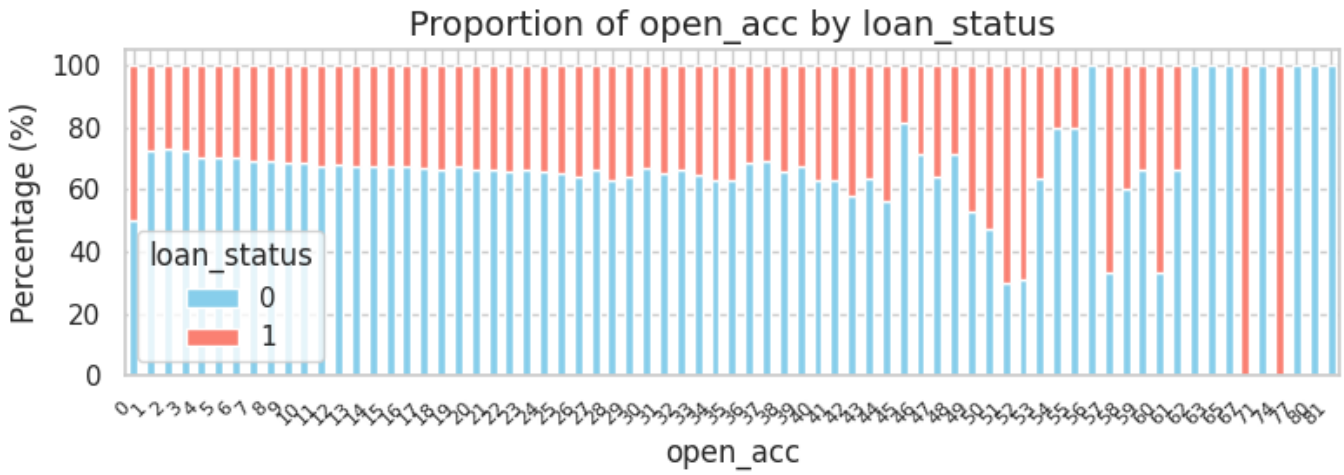


Figure 16 – Proportion of *open_acc* by Loan Status

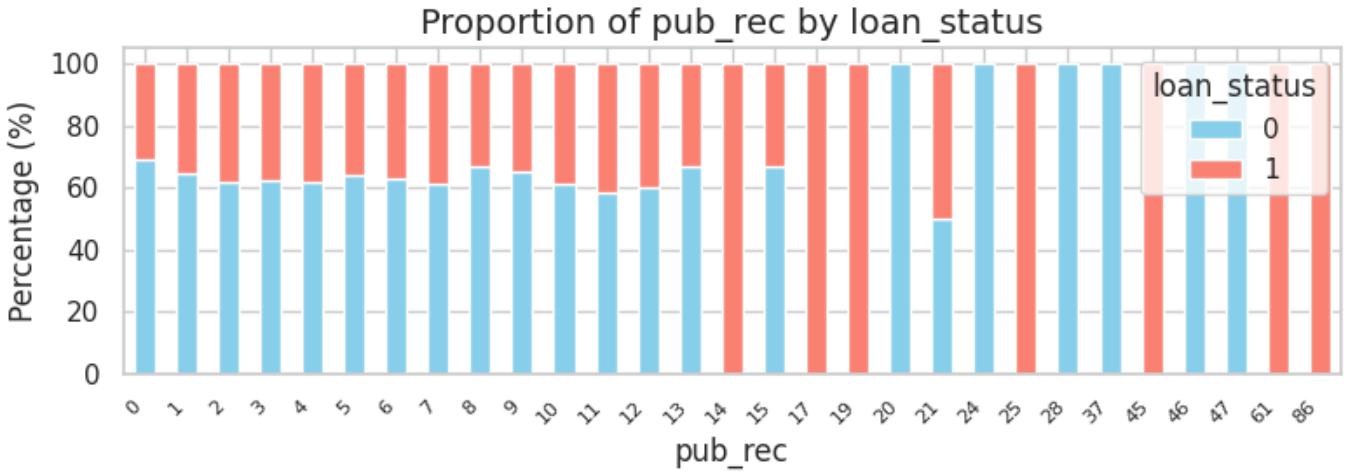


Figure 17 – Proportion of *pub_rec* by Loan Status

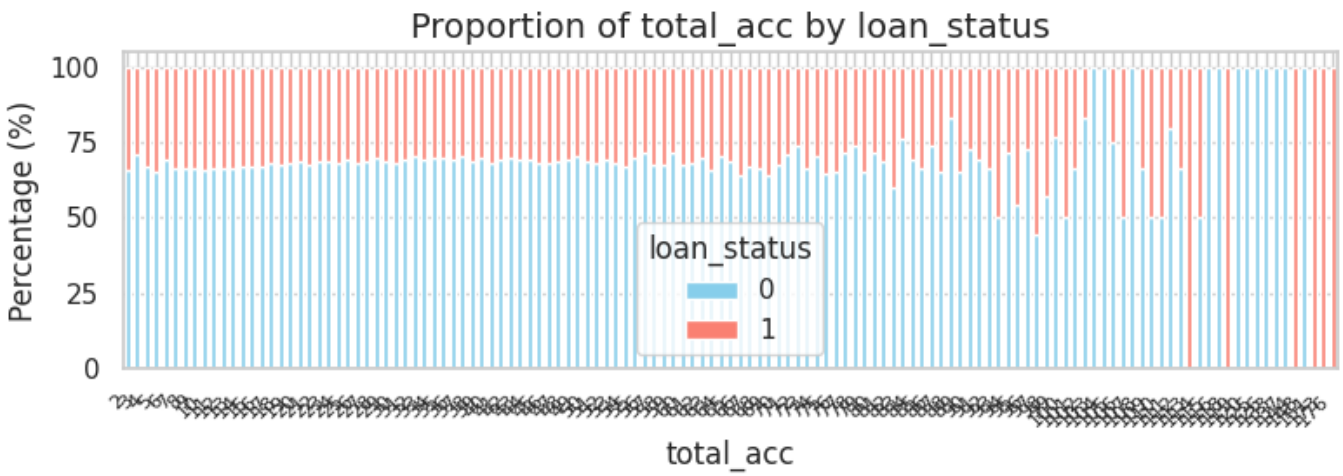


Figure 18 – Proportion of *delinq_2yrs* by Loan Status

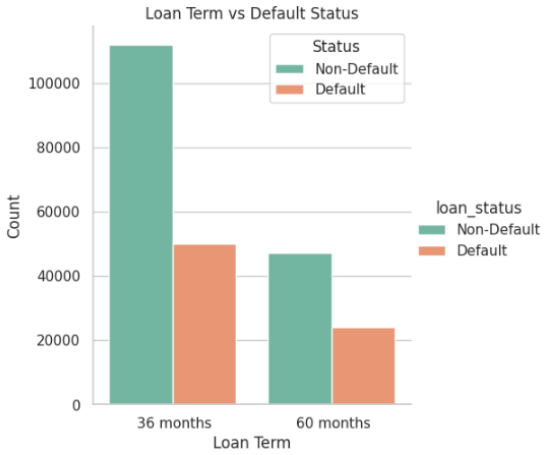


Figure 19 – Distribution of Loan Term by Loan Status

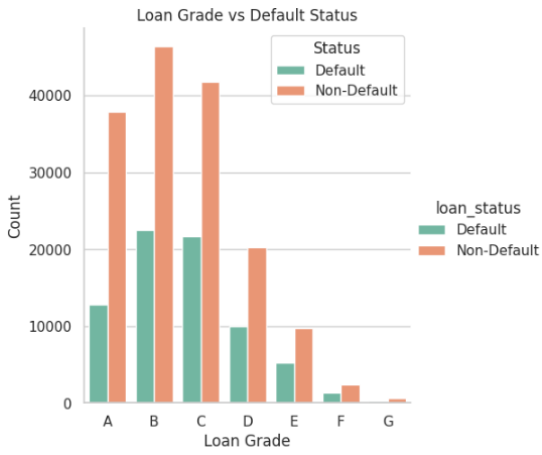


Figure 20 – Distribution of Loan Grade by Loan Status

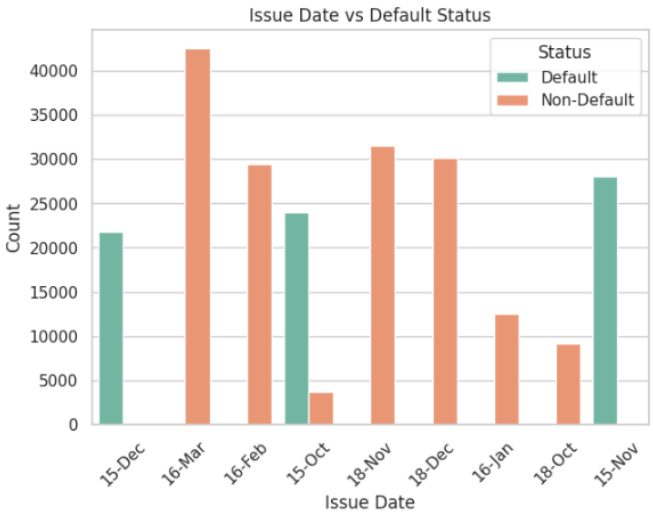


Figure 21 – Distribution of Issue Date by Loan Status

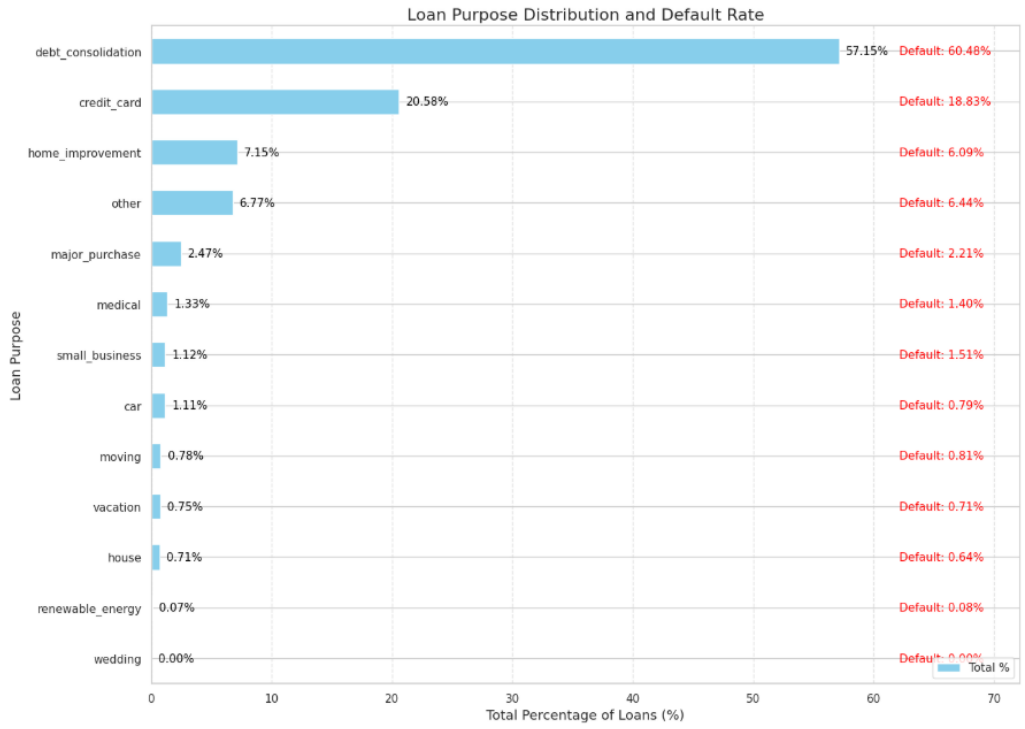


Figure 22 – Distribution of Loan Purpose and Default Rates

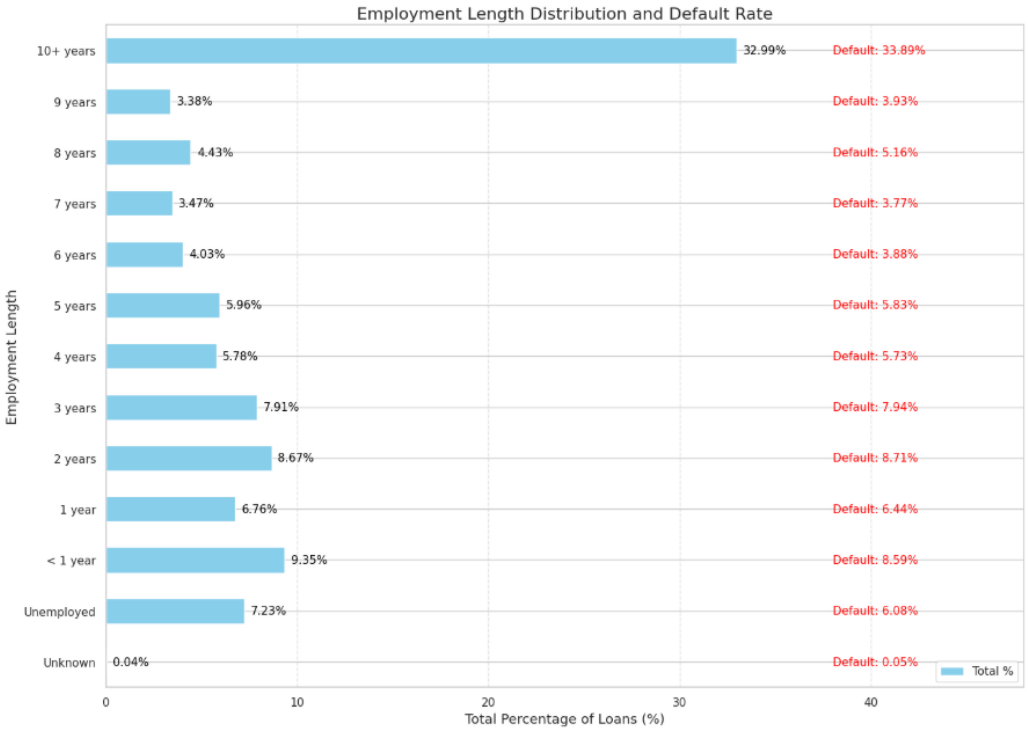


Figure 23 – Distribution of Employment Length and Default Rates

emp_title	Count	Percentage
NaN	22188	9.52
Teacher	4237	1.82
Manager	3699	1.59
Owner	2024	0.87
Registered Nurse	1825	0.78

Figure 24 – Employment Title Top5 Categories

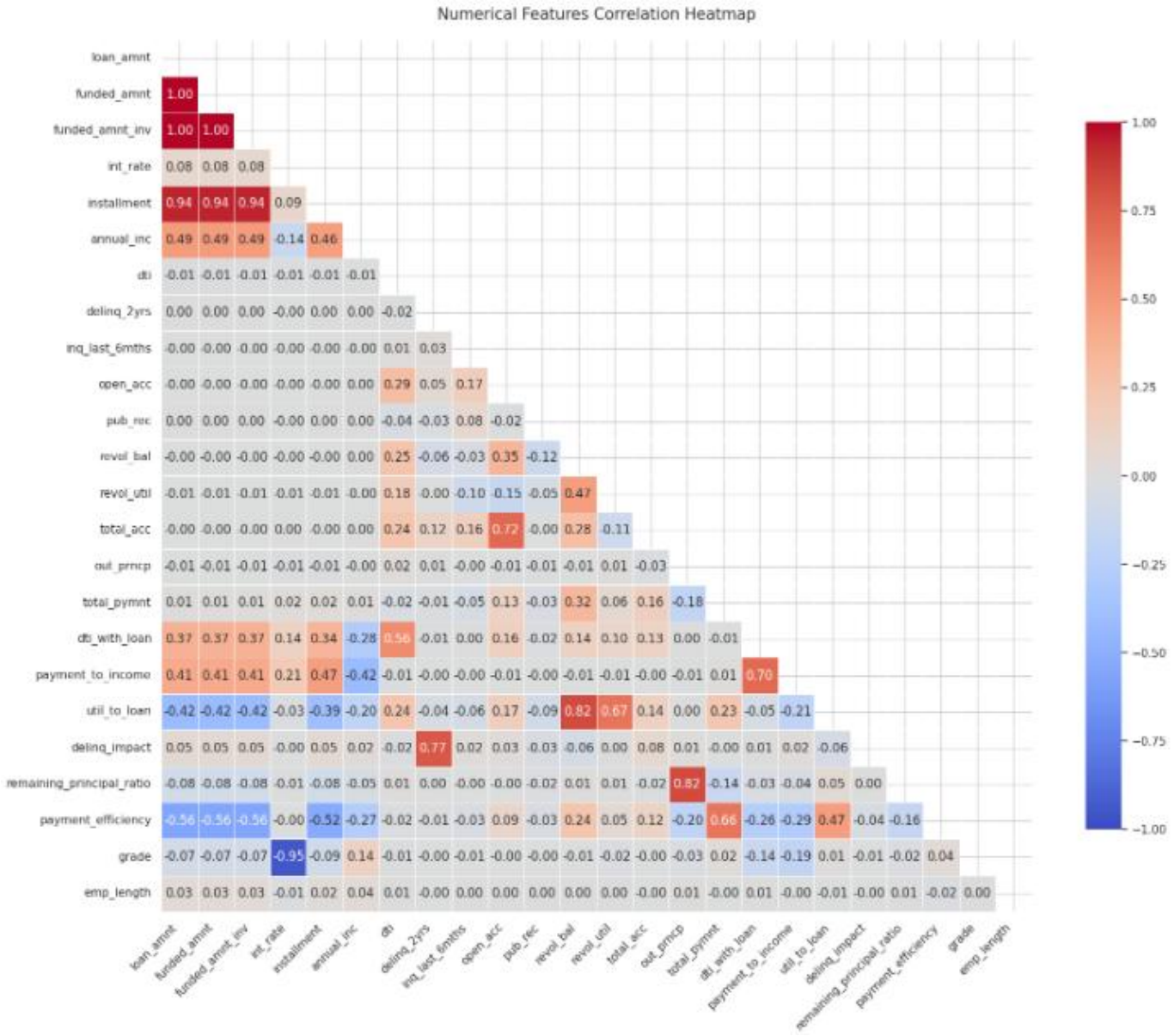


Figure 25 – Correlation Matrix

```
==================================================
Logistic Regression Results
==================================================
Best Hyperparameters: {'C': 10, 'penalty': 'l1', 'solver': 'liblinear'}
Mean CV F1: 0.5880
Validation F1: 0.5867
Validation AUC: 0.8213

Classification Report:
              precision    recall  f1-score   support

           0       0.80      0.90      0.84     51807
           1       0.70      0.51      0.59     24191

    accuracy                           0.77     75998
   macro avg       0.75      0.70      0.72     75998
weighted avg       0.76      0.77      0.76     75998

Logistic Regression Confusion Matrix:
[[46498  5309]
 [11946 12245]]
```

Figure 26 – Logistic Regression Results

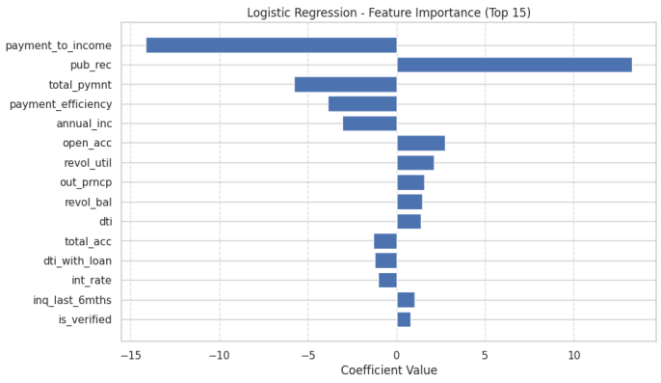


Figure 27 – Logistic Regression Feature Importance

```
F1 Score on validation set: 0.6770

Classification Report:
              precision    recall  f1-score   support

           0       0.87      0.80      0.83     51807
           1       0.63      0.73      0.68     24191

    accuracy                           0.78     75998
   macro avg       0.75      0.77      0.75     75998
weighted avg       0.79      0.78      0.78     75998


Confusion Matrix:
[[41282 10525]
 [ 6426 17765]]
```

Figure 28 – Neural Network Results

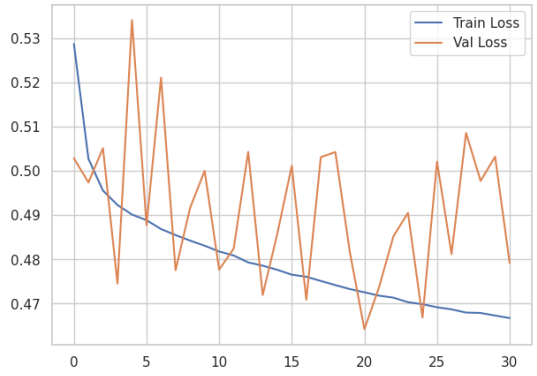


Figure 29 – Neural Network Train/Val Loss per Epoch

```
==================================================
Random Forest Results
==================================================
Best Hyperparameters: {'max_depth': 15, 'max_features': 'sqrt'
Mean CV F1: 0.6894
Validation F1: 0.6855
Validation AUC: 0.8549

Classification Report:
              precision    recall  f1-score   support

           0       0.87      0.79      0.83     51807
           1       0.63      0.76      0.69     24191

    accuracy                           0.78     75998
   macro avg       0.75      0.77      0.76     75998
weighted avg       0.80      0.78      0.78     75998

Random Forest Confusion Matrix:
[[46657  5150]
 [10278 13913]]
```

Figure 30 – Random Forest Results

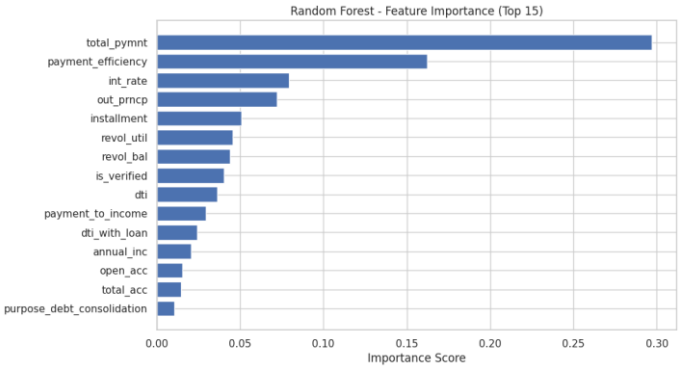


Figure 31 – Random Forest Feature Importance

```
==================================================
CatBoost Results:
==================================================
Best Hyperparameters: {'depth': 6, 'iterations': 200, 'l2_leaf_reg': 3
Mean CV F1: 0.9130
Validation F1: 0.9107
Validation AUC: 0.9888

Classification Report:
              precision    recall  f1-score   support

           0       0.98      0.93      0.95     51807
           1       0.86      0.96      0.91     24191

    accuracy                           0.94     75998
   macro avg       0.92      0.95      0.93     75998
weighted avg       0.94      0.94      0.94     75998

CatBoost Confusion Matrix:
[[49367  2440]
 [ 1774 22417]]
```

Figure 32 – CatBoost Results

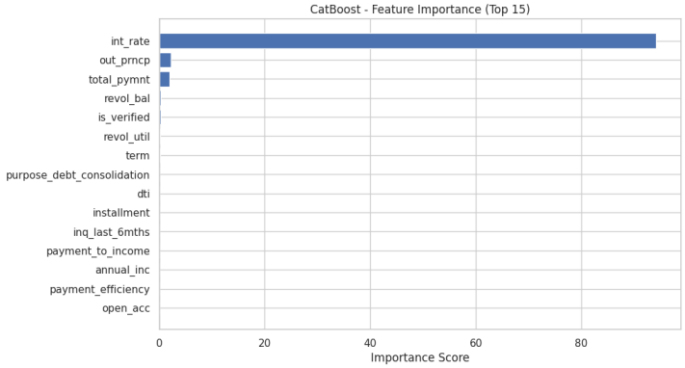


Figure 33 – CatBoost Feature Importance

```
==================================================
Gradient Boosting Results:
==================================================
Best Hyperparameters: {'learning_rate': 0.2, 'max_depth': 5
Mean CV F1: 0.9119
Validation F1: 0.9090
Validation AUC: 0.9867

Classification Report:
              precision    recall  f1-score   support

           0       0.96      0.95      0.96     51807
           1       0.90      0.92      0.91     24191

    accuracy                           0.94     75998
   macro avg       0.93      0.93      0.93     75998
weighted avg       0.94      0.94      0.94     75998

Gradient Boosting Confusion Matrix:
[[49390  2417]
 [ 1980 22211]]
```

Figure 34 – Gradient Boosting Results

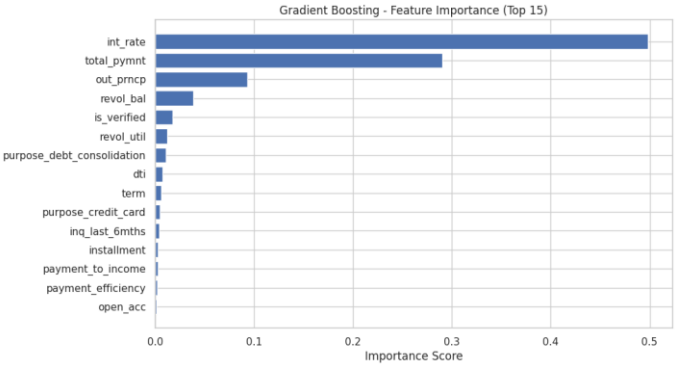


Figure 35 – Gradient Boosting Feature Importance