Predicting Loan Defaults: A Machine Learning Approach

Ashwin Purushothama Dhas
CF969 - Big-Data for Computational Finance
Assignment 2
2320993
ap23710@essex.ac.uk

I. Introduction

The goal of this study is to predict loan defaults using data from a peer-to-peer lending platform. The dataset includes various features related to the borrowers and their loan applications. To achieve this, several machine learning models were employed, including Linear Regression, Ridge Regression, Lasso Regression, Random Forest, and Neural Network. This report documents the preprocessing steps, model evaluations, and findings from the analysis.

II. Data Preparation

A. *Load Data*

Three CSV files were provided:

- train.csv: Training dataset
- test.csv: Testing dataset
- varDescription.csv: Description of the variables

The dataset was loaded into Python for further processing.

B. *Drop Redundant Variables*

The columns **id** and **member_id** were dropped as they are unique identifiers and do not contribute to the predictive power of the models.

C. *Encoding Categorical Variables*

Label encoding was used to convert categorical variables into numerical values. However, some columns with object data types could not be encoded directly due to inconsistencies or unexpected values. These columns **(grade, emp_length, home_ownership, application_type)** values were handled manually by creating a map of encoded values.

D. *Handling Missing Values*

Missing values were handled using **mean**, **mode**, and **median** imputation based on the skewness of the data. This ensured that the imputation method was appropriate for the distribution of each feature.

E. *Descriptive Statistics*

Descriptive statistics were explored to understand the distribution and central tendency of the features. Key insights included:

- The mean loan amount was approximately $15,058 with a standard deviation of $9,177.
- Interest rates ranged from 5.31% to 30.99%, with a mean of 13.09%.
- The average debt-to-income ratio (DTI) was 18.8%.
- The majority of borrowers had no delinquencies in the past two years, with a mean value of 0.31.
- The average annual income of borrowers was approximately $78,208, indicating a wide range of income levels among borrowers.
- Significant skewness was observed in features like annual income, total payment amounts, and revolving balance, requiring careful handling during imputation and modeling.

F. *Correlation Analysis*

A correlation matrix was plotted to identify highly correlated features. Features with high correlation (correlation coefficient ¿ 0.8) were removed to prevent multicollinearity, which can adversely affect model performance.

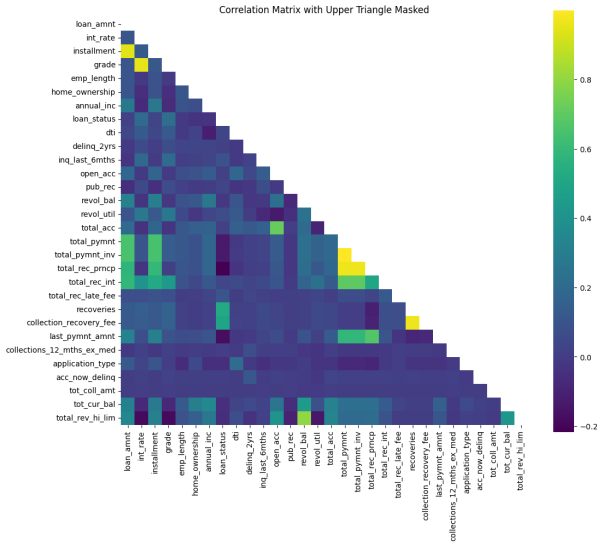

Fig. 1. Correlation Matrix

- **Best 10 features:** `recoveries`, `collection_recovery_fee`, `grade`, `total_rec_prncp`, `int_rate`, `last_pymnt_amnt`, `total_pymnt`, `total_pymnt_inv`, `total_rec_late_fee`, `inq_last_6mths`
- **Worst 10 features:** `tot_coll_amt`, `collections_12_mths_ex_med`, `acc_now_delinq`, `emp_length`, `total_acc`, `delinq_2yrs`, `open_acc`, `revol_bal`, `loan_amnt`, `installment`

Some notable feature correlations included:

- `total_pymnt_inv` and `total_pymnt` had a very high correlation of 0.999, indicating that they essentially provide the same information.
- `total_pymnt` and `total_rec_prncp` also showed high correlation at 0.967.
- `total_pymnt_inv` and `total_rec_prncp` had a correlation of 0.966.
- `recoveries` and `collection_recovery_fee` had a correlation of 0.966.
- `int_rate` and `grade` were highly correlated at 0.953.
- `loan_amnt` and `installment` had a correlation of 0.945.

G. *Data Balancing Using SMOTE*

The target variable loan_status was imbalanced, with fewer instances of 'Charged Off'. **SMOTE (Synthetic Minority Oversampling Technique)** was applied to balance the classes. The distribution before and after applying SMOTE is shown below:

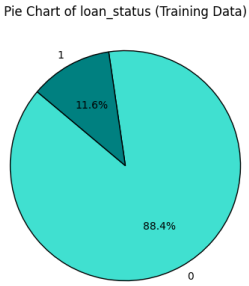

Fig. 2. Before SMOTE

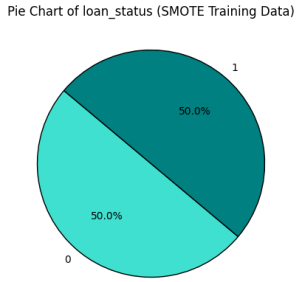

Fig. 3. After SMOTE

H. *Handling Outliers*

Outliers were identified and handled using the Interquartile Range (IQR) method. For each feature, the first quartile (Q1) and the third quartile (Q3) were calculated to determine the IQR. Data points falling below Q1 - 1.5IQR or above Q3 + 1.5IQR were considered outliers. These outliers were then capped to the respective lower and upper limits to minimize their impact on the model. This process helped reduce the skewness of the data, making the distributions more symmetric. For example, below figures shows the results of handling the outliers for `revol_util` feature.

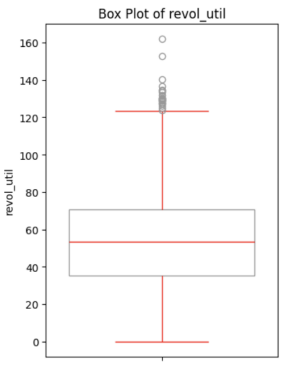

Fig. 4. Before Handling Outliers

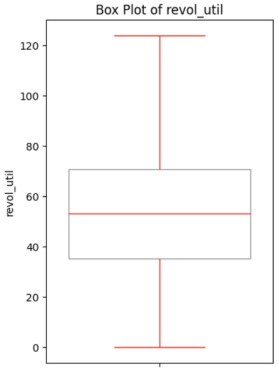

Fig. 5. After Handling Outliers

III. Model Evaluation

Several classification algorithms are employed to build predictive models, including:

1) **Linear Regression**
2) **Ridge Regression**
3) **Lasso Regression**
4) **Random Forest**
5) **Neural Network Model**

A. *Linear Regression*

The Linear Regression model was fitted using all predictors.

- Mean Squared Error for training data: 0.1087
- Mean Squared Error for testing data: 1.5803

TABLE I
CLASSIFICATION REPORT FOR TRAINING DATA

Class	Precision	Recall	F1-score	Support
0	0.83	0.94	0.88	199,905
1	0.93	0.81	0.87	199,905
Accuracy	**0.87**	**0.87**	**0.87**	**399,810**
Macro avg	0.88	0.87	0.87	399,810
Weighted avg	0.88	0.87	0.87	399,810

TABLE II
CLASSIFICATION REPORT FOR TESTING DATA

Class	Precision	Recall	F1-score	Support
0	0.96	0.87	0.91	199,782
1	0.42	0.73	0.54	26,285
Accuracy	**0.85**	**0.85**	**0.85**	**226,067**
Macro avg	0.69	0.80	0.72	226,067
Weighted avg	0.90	0.85	0.87	226,067

B. *Ridge Regression*

The Ridge Regression model was fitted with hyperparameter tuning for λ (ranging from 0.01 to 3). **The best λ was found to be 0.01**.

- Best alpha (lambda): 0.01
- Mean Squared Error for the best model on training data: 0.1087
- Mean Squared Error for the best model on testing data: 1.5803

TABLE III
CLASSIFICATION REPORT FOR TRAINING DATA

Class	Precision	Recall	F1-score	Support
0	0.83	0.94	0.88	199,905
1	0.93	0.81	0.87	199,905
Accuracy	0.87			
Macro avg	0.88	0.87	0.87	399,810
Weighted avg	0.88	0.87	0.87	399,810

TABLE IV
CLASSIFICATION REPORT FOR TESTING DATA

Class	Precision	Recall	F1-score	Support
0	0.96	0.87	0.91	199,782
1	0.42	0.73	0.54	26,285
Accuracy	0.85			
Macro avg	0.69	0.80	0.72	226,067
Weighted avg	0.90	0.85	0.87	226,067

C. *Lasso Regression*

The Lasso Regression model was fitted with hyperparameter tuning for λ (ranging from 0.01 to 3). **The best λ was found to be 0.01**.

- Best alpha (lambda): 0.01
- Mean Squared Error for the best model on training data: 0.1115
- Mean Squared Error for the best model on testing data: 1.7596

TABLE V
CLASSIFICATION REPORT FOR TRAINING DATA

Class	Precision	Recall	F1-score	Support
0	0.81	0.95	0.88	199,905
1	0.94	0.78	0.85	199,905
Accuracy	0.87			
Macro avg	0.88	0.87	0.87	399,810
Weighted avg	0.88	0.87	0.87	399,810

TABLE VI
CLASSIFICATION REPORT FOR TESTING DATA

Class	Precision	Recall	F1-score	Support
0	0.96	0.90	0.93	199,782
1	0.47	0.70	0.56	26,285
Accuracy	0.87			
Macro avg	0.71	0.80	0.74	226,067
Weighted avg	0.90	0.87	0.88	226,067

D. *Random Forest*

The Random Forest model was fitted and optimized.

- Mean Squared Error for the Random Forest model on training data: 0.0
- Mean Squared Error for the Random Forest model on testing data: 0.0379

TABLE VII
CLASSIFICATION REPORT FOR TRAINING DATA

Class	Precision	Recall	F1-score	Support
0	1.00	1.00	1.00	199,905
1	1.00	1.00	1.00	199,905
Accuracy	1.00			
Macro avg	1.00	1.00	1.00	399,810
Weighted avg	1.00	1.00	1.00	399,810

TABLE VIII
CLASSIFICATION REPORT FOR TESTING DATA

Class	Precision	Recall	F1-score	Support
0	0.96	1.00	0.98	199,782
1	0.99	0.68	0.81	26,285
Accuracy	0.96			
Macro avg	0.97	0.84	0.89	226,067
Weighted avg	0.96	0.96	0.96	226,067

Best hyperparameters: 'bootstrap': True, 'ccp_alpha': 0.0, 'class_weight': None, 'criterion': 'gini', 'max_depth': None, 'max_features': 'sqrt', 'max_leaf_nodes': None, 'max_samples': None, 'min_impurity_decrease': 0.0, 'min_samples_leaf': 1, 'min_samples_split': 2, 'min_weight_fraction_leaf': 0.0, 'n_estimators': 100, 'n_jobs': None, 'oob_score': False, 'random_state': 42, 'verbose': 0, 'warm_start': False

The importance of variables in predicting loan default using the Random Forest model is determined by the model's ability to split the data effectively at each node. This is quantified by a metric known as "feature importance," which measures how much each feature reduces the impurity in the dataset.

- **Recoveries:** This variable had the highest importance score of 0.547. It significantly reduced the impurity at many nodes across the decision trees in the forest, indicating that it plays a critical role in distinguishing between defaults and non-defaults.
- **Last Payment Amount:** With an importance score of 0.097, this variable was the second most important. It helped the model make accurate splits by indicating the borrower's payment behavior, which is highly indicative of their likelihood to default.
- **Inquiries in Last 6 Months:** This variable had an importance score of 0.078. It reflects recent credit-seeking behavior, which can be a strong predictor of financial distress and potential default.
- **Interest Rate:** The interest rate had an importance score of 0.053. Higher interest rates can be associated with higher risk loans, making this feature important in predicting defaults.
- **Total Payment:** With an importance score of 0.039, the total amount paid by the borrower was also significant. This variable reflects the borrower's repayment capacity and history.

E. Neural Network

A Neural Network model was selected and optimized. The chosen architecture was a multi-layer perceptron with three hidden layers. The chosen neural network model was selected due to its ability to capture complex, non-linear relationships in the data, which traditional linear models may fail to identify. The multi-layer perceptron architecture with multiple hidden layers allows for a deep learning approach, enhancing the model's capacity to learn intricate patterns from the features. Additionally, the use of regularization techniques like L2 regularization helps prevent overfitting, ensuring the model generalizes well to unseen data. Early stopping during training further optimizes performance by halting training when no significant improvement is observed, balancing accuracy and training efficiency.

The accuracy for the model was:

- Training Data Accuracy: 0.9182
- Testing Data Accuracy: 0.9542

IV. MODEL EVALUATION

A. Significant Predictors

The significant predictors identified by the models are as follows:

1) **post charge off gross recovery**
2) **post charge off collection fee**
3) **Grade associated with the loan**
4) **Principal received to date**
5) **Interest rate of the loan the applicant received**

6) **last payment amount**
7) **Payments received to date for total amount funded**
8) **Payments received to date for portion of total amount funded by investors**
9) **Late fees received to date**
10) **Inquiries into the applicant's credit during the last 6 months**

B. Model Comparison

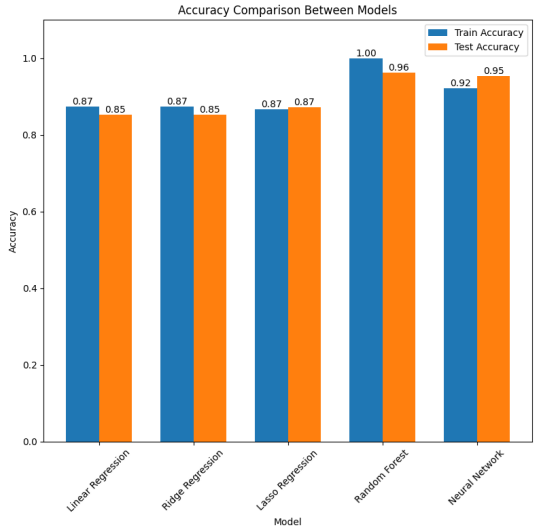

Fig. 6. Model Accuracies

- The Linear Regression model showed decent performance with an accuracy of **0.8748** on training data and **0.8529** on testing data. However, the model's performance on class 1 (Charged Off) was less satisfactory, with a precision of **0.42** and a recall of **0.73** on the testing data.
- Ridge Regression yielded similar results to Linear Regression with an accuracy of **0.8748** on training data and **0.8529** on testing data. The performance metrics for class 1 were the same as those for Linear Regression, indicating no significant improvement.
- Lasso Regression performed slightly better than Linear and Ridge Regression with an accuracy of **0.8720** on testing data. It had better precision **(0.47)** and recall **(0.70)** for class 1, making it more reliable for predicting defaults.
- The Random Forest model outperformed all other models with an accuracy of **1.0** on training data and **0.9621** on testing data. It showed excellent precision **(0.99)** and reasonable recall **(0.68)** for class 1, indicating a strong predictive power for defaults. The model did not overfit despite its high performance on training data.
- The Neural Network model showed strong performance with an accuracy of **0.9182** on training data and **0.9542** on testing data. It maintained good precision **(0.92)** but had a lower recall **(0.67)** for class 1 compared to Random Forest. Despite this, it was a robust model overall.

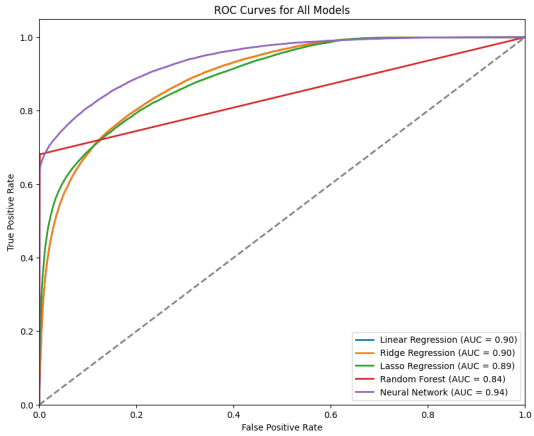

Fig. 7. ROC curves for all models

C. Discussion

Among the models evaluated, the **Random Forest** model demonstrated the best predictive power for predicting loan defaults, with the highest accuracy and well-balanced precision and recall. The Neural Network model also performed well but had slightly lower recall for class 1 compared to Random Forest. Therefore, the Random Forest model is identified as the best model for predicting loan defaults from the given data.

CONCLUSION

To conclude, the proposed methodology aims to contribute significantly to the field of financial risk management by providing an effective tool for early detection and intervention of loan defaults. By accurately predicting the likelihood of loan default based on selected attributes, financial institutions can identify at-risk borrowers and implement targeted strategies to mitigate risk, thereby enhancing the stability and profitability of lending operations.

REFERENCES

[1] R. G. Brown and D. E. Mues, "An experimental comparison of classification algorithms for imbalanced credit scoring data sets," Expert Systems with Applications, vol. 39, no. 3, pp. 3446–3453, 2012.

[2] I. Yeh and C. Lien, "The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients," Expert Systems with Applications, vol. 36, no. 2, pp. 2473–2480, 2009.

[3] J. L. Thomas, "A survey of credit and behavioural scoring: Forecasting financial risk of lending to consumers," International Journal of Forecasting, vol. 16, no. 2, pp. 149–172, 2000.

[4] S. Lessmann, B. Baesens, H. V. Seow, and L. C. Thomas, "Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research," European Journal of Operational Research, vol. 247, no. 1, pp. 124–136, 2015.

[5] J. B. Heaton, N. G. Polson, and J. H. Witte, "Deep learning for finance: Deep portfolios," Applied Stochastic Models in Business and Industry, vol. 33, no. 1, pp. 3–12, 2017.