MBS Delinquency Prediction using Logistic Regression and Machine Learning Models

By Yi-Hsuan Fan, Leyi Hu,
Chenqi Wu, Sike Yang

Professor Chris Kelliher and Eugene Sorets
MF772 Credit Risk
December 9th, 2023

Abstract

This paper investigates the application of logistic regression and machine learning in predicting loan delinquency. The study encompasses a thorough exploration of delinquency in the loan domain, detailing data cleaning and preprocessing techniques, and culminates in the implementation of models like logistic regression, support vector machine, neural network, and random forest for predictive analysis.

1. Introduction

1.1 Background

Loan delinquency remains a critical challenge in the financial sector, demanding advanced predictive models for risk management. This study investigates the applicability and performance of logistic regression, support vector machines, neural networks, and random forests in addressing this issue.

1.2 Objective

The primary objective is to compare and analyze the effectiveness of different predictive models in identifying and predicting loan delinquency.

2. Data Cleaning and Processing

2.1 Dataset Description

The dataset utilized in this project is sourced from Freddie Mac, a prominent mortgage financing entity in the United States. This publicly available Standard Single-Family Loan-Level Dataset comprises two distinct files. One file contains historical loan origination data, while the other encompasses monthly performance data for each loan mentioned in the origination data file. These two files were merged to create a comprehensive dataset covering the period from 1999 to 2020. For further details and comprehensive descriptions of the data, additional information is accessible on Freddie Mac's official websites.

2.2 Data Cleaning

The dataset undergoes cleaning to address unknown values, outliers, and inconsistencies.
After carefully reviewing the summary of the data frame and the descriptions of the columns, we manually removed irrelevant features and date information. Data preprocessing techniques are then applied to ensure the integrity of the input data.

2.3 Data Processing

To facilitate the modeling process, we converted categorical data into numerical form using label encoding and dummy variables. This step is crucial as many machine learning algorithms require numerical inputs. Furthermore, for certain numerical features such as Credit Score, MIP, DTI, and LTV, we created categorical ranges to replace the original columns, simplifying and enhancing their interpretability.

2.4 Feature Engineering

Moving on to feature selection and multicollinearity assessment, we employed Principal Component Analysis (PCA) to explore the variance explained by principal components. Simultaneously, we utilized SelectKBest with Mutual Information to identify the most relevant features. Based on mutual information scores, we selected the top six features deemed crucial for predicting loan delinquency, which are OrigUPB, OrigInterestRate, PropertyState, CreditRange, DTI_Range, LTV_Range, EverDelinquent, respectively.

To address potential multicollinearity issues, we conducted a variance inflation factor (VIF) analysis and plotted the heat map. Features with high VIF values, indicative of strong correlations with other features, were carefully considered. A subset of relevant features was then chosen and formed a new dataset for the following predictive modeling process.

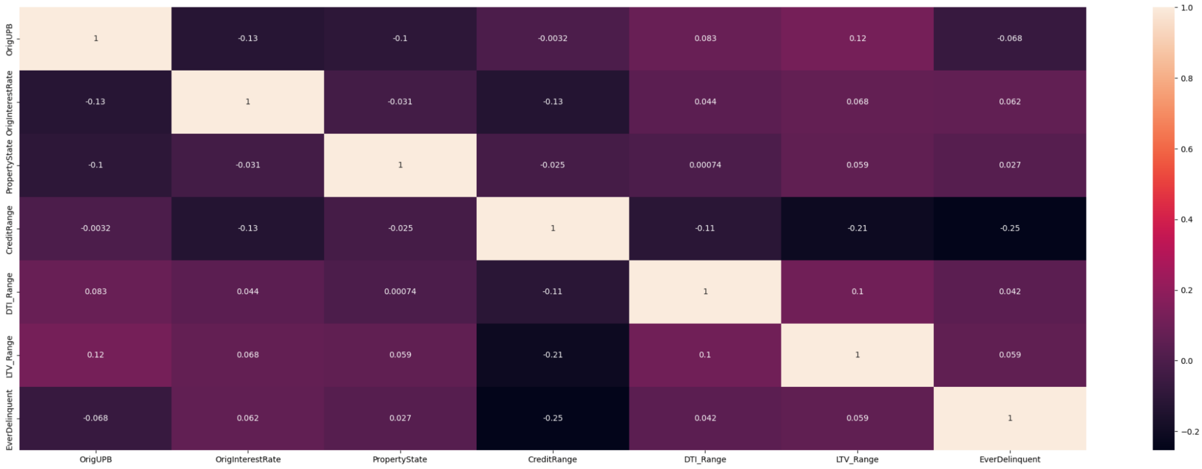

Figure1: heat map for the selected feature.

2.5 Data Splitting and Scaling

Initially, the dataset was split into features (X) and the target variable ('EverDelinquent'). The `train_test_split` function was employed to create distinct training and testing sets, allocating 70% of the data for training and 30% for testing, while ensuring reproducibility through the use of a random seed.

3. Predictive Models

3.1 Logistic Regression

3.1.1 Scaled Logistic Regression Model

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable for its efficiency. In the context of MBS delinquency prediction, our dependent variable is binary—either delinquency or non-delinquency. The logistic function outputs a value between 0 and 1, which can be interpreted as the probability of a particular class or event occurring. The odds of the event are modeled as a linear combination of the predictor variables, which are then transformed using the logistic function.

In this study, the logistic regression model was employed to estimate the probabilities of MBS delinquency. By scaling the input variables, we aimed to ensure that each feature's coefficient estimates would be based on the same scale, allowing for a fair comparison of the relevance of each feature. The scaling process was particularly vital for our model as it included features with differing scales of measurement, which could have skewed the model's interpretation of the data had they not been standardized.

The baseline scaled logistic regression model yielded an accuracy of 75.26%. The high accuracy suggests a good overall classification rate; however, the classification report indicated a substantial imbalance between the recall for the non-delinquent (0.75) and delinquent (0.58) classes, with particularly low precision and F1-score for predicting delinquency which is only 0.01. This underscored the challenge of imbalanced classes in predictive modeling.

3.1.2 Scaled and Balanced Logistic Regression Model

Recognizing the imbalance of dependent variables of the dataset, a second logistic regression model was specified with class weights set to 'balanced'. This adjustment was designed to counteract the imbalance by assigning a higher weight to the minority class and a lower weight to the majority class, effectively penalizing the misclassification of the minority class more heavily. This approach aimed to improve the model's sensitivity to the delinquent class, which is critical in the financial domain where the cost of false negatives is substantial.

The scaled and balanced model demonstrated a lower overall accuracy of 60.29%. Despite this, the balanced approach significantly improved the recall for the delinquent class (0.62) and the F1 score of the delinquent class(0.44), pointing to a model more adept at identifying potential delinquencies. This improvement in recall and F1 score is a crucial aspect of the model's utility in practical financial applications where failing to detect delinquency could entail higher risk and cost than false positives. The decrease in accuracy with the balanced model is a typical result when the model's focus shifts towards the minority class. The improvement in recall for the delinquent class suggests that the balanced model may be more suitable for practical applications despite the lower overall accuracy.

3.2 Support Vector Machines (SVM)

Support Vector Machines are a set of supervised learning methods used for classification, regression, and outlier detection. The advantages of SVM are evident in its effectiveness in high-dimensional spaces and its versatility manifested through various kernel functions.

The SVM model was applied to the dataset using different kernel functions: linear, polynomial, and radial basis function (RBF). Notably, the radial basis function (RBF) kernel was chosen based on its superior performance during the testing phase.

The selection of RBF was twofold: first, it demonstrated the best predictive performance among the kernels evaluated, and second, the mathematical properties of the RBF kernel align closely with the characteristics of our dataset.By mapping the data into a higher-dimensional space, the RBF kernel can handle the case where the relation between class labels and attributes is nonlinear.

The grid search process, an exhaustive search over a specified parameter space, was employed to optimize the SVM's hyperparameters, including the penalty parameter C and the γ parameter of the RBF kernel. This method ensures that the most suitable hyperparameters are selected to minimize the generalization error.

The SVM model, equipped with the RBF kernel, attained an accuracy of 75.20%. However, the F1-score stood at 0, a stark indication that, despite the model's overall accuracy, it struggles with the precise identification of true positives and true negatives. This metric is particularly crucial in the context of delinquency prediction, where the cost of false negatives (failing to identify a potential delinquency) and false positives (wrongly identifying a case as delinquent) can be substantial.

3.3 Light Gradient Boosting Machine (LGBM)

Despite SVM's known capabilities, it underperformed on the given dataset, suggesting that the algorithm's conventional advantages may not align well with the data's attributes. Conversely, LGBM's design to handle large datasets and work efficiently with categorical features suggests a potential superiority in this scenario.

The LGBM model underwent rigorous parameter tuning, including the adjustment of the learning rate, the number of leaves, the depth of each tree, the minimum number of samples per leaf and other parameters pertinent to boosting performance. The tuning process aimed to maximize the model's efficiency, especially considering the dataset's imbalance.

The LGBM model yielded a slightly improved accuracy of 75.48% and an F1-score of 0.05. This marginal enhancement over SVM suggests that while LGBM is more suited to the dataset, the feature set might still be insufficient in capturing all the relevant information for accurate predictions.

The comparison indicates that LGBM may be slightly more suited to the dataset than SVM, but neither algorithm performs optimally. The results suggest that the feature set, despite being high-dimensional, lacks the necessary depth to allow the algorithms to fully discern the underlying patterns within the data.

3.4 Neural Network

We utilized three neural network models for a binary classification task, predicting the likelihood of delinquency. The three variations include a normal model, a scaled model, and a scaled and rebalanced model using the Synthetic Minority Over-sampling Technique (SMOTE).

3.4.1 Standard Neural Network Model:

The first model is a standard neural network trained on the original dataset without any preprocessing. It consists of an input layer with 32 neurons, with a rectified linear unit (ReLU) activation function, followed by a hidden layer containing 16 neurons, also activated by ReLU, and an output layer with a sigmoid activation function. The model is compiled using the Adam optimizer and binary cross-entropy loss, and training for 100 epochs and a batch size of 16. To obtain predictions, the model was utilized to predict outcomes on the test set, generating probabilities thresholded at 0.5 to convert them into binary predictions. Then, the model achieves an accuracy of approximately 75.21% on the test set.

3.4.2 Scaled Neural Network Model:

In the second approach, the input features are scaled using StandardScaler before training the neural network. The model architecture remains the same as the standard model. The evaluation results show an accuracy of 75.22% on the test set, indicating that scaling did not significantly impact the model's performance.

3.4.3 Scaled and Rebalanced Neural Network Model:

The third model incorporates both feature scaling and rebalancing using SMOTE to address class imbalance. The dataset is resampled before scaling and training the neural network. The model is then evaluated on the original test set. The evaluation results show an accuracy of 59.45%. While the accuracy has decreased compared to the unscaled models, it is important to note that the model's ability to predict the minority class (EverDelinquent = 1) has improved, as evidenced by the increase in recall and f1 score for the positive class.

In conclusion, the trade-off between accuracy and sensitivity to the minority class is a common consideration in imbalanced datasets, However, accuracy may not adequately reflect the model's effectiveness; instead, recall and F1 score provide more nuanced insights into the model's performance, particularly in scenarios where correctly identifying the minority class (EverDelinquent = 1) is of primary importance. Therefore, based on our imbalanced data, the scaled and rebalanced adjustments indeed provided more precise prediction.

3.5 Random Forest

The Random Forest algorithm was selected for its proficiency in handling imbalanced datasets and scalability to accommodate large volumes of data. The Random Forest algorithm operates on the principle of constructing many decision trees at training time and outputting the class, the mode of the classes (classification) or mean prediction (regression) of the individual trees. This ensemble learning method is particularly effective in managing imbalanced datasets due to its inherent random sampling of features and bootstrap aggregating, or bagging, which reduces variance without increasing bias. By combining diverse trees that individually overfit to different aspects, the Random Forest corrects for the overfitting of individual trees, providing robust generalization capabilities on large datasets.

Before model training, data preprocessing steps included standardization of features to ensure uniformity and mitigate the influence of outliers. Parameter tuning was performed using

GridSearchCV, which optimizes the model parameters through cross-validation and grid search over a parameter space. The tuning was directed towards the *n_estimators* parameter, which determines the number of trees in the forest, and the *max_features* parameter, which defines the maximum number of features to consider when looking for the best split. The effectiveness of the hyperparameter tuning was evaluated based on the F1 score, a harmonic mean of precision and recall, which is particularly useful in the context of imbalanced datasets where a balance between precision and recall is desired. After the extensive search, the optimal set of parameters was determined to be {'max_features': 'auto,' 'n_estimators': 400}.

Additionally, two approaches were evaluated to address the class imbalance: the first involved utilizing the balanced class weighting option within the Random Forest model, and the second employed Synthetic Minority Over-sampling Technique (SMOTE) for oversampling the minority class. The performance of the base and oversampled models was compared using the Receiver Operating Characteristic (ROC) curve, which plots the actual positive rate against the false positive rate at various threshold settings. The Area Under the Curve (AUC) metric was used to quantify model performance, with the base model achieving an AUC of 0.5322 and the oversample model demonstrating an improved AUC of 0.5410, suggesting that the oversampling technique may enhance the model's ability to classify minority class instances correctly.

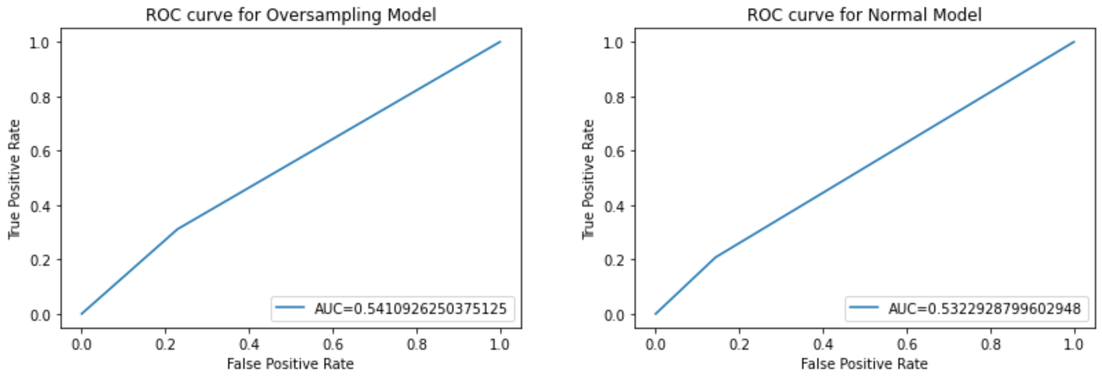

Figure 2 & 3 ROC curve for Oversampling and Normal models

4. Oversampling Approach

Recognizing the suboptimal predictive performance of our initial models, a critical review of the dataset and preprocessing techniques was undertaken. The data exhibited a significant class imbalance, with approximately 20% of the instances being classified as delinquent.

To rectify this, we implemented an oversampling technique to balance the classes in the training data. Oversampling involves augmenting the dataset by increasing the frequency of the underrepresented class to create a more balanced distribution. This approach is particularly suited to problems where the cost of missing a positive instance is high, as in delinquency prediction.

The models were then retrained on this balanced dataset, resulting in a substantial improvement in the F1-score for both SVM and LGBM models, as illustrated in the subsequent chart.

	Linear	SVM	LGBM	NN	RF
Delinquency f1-score	0.01	0	0.05	0.05	0.25
Updated Delinquency f1-score	0.44	0.44	0.32	0.43	0.31
Overall Accuracy	0.75	0.70	0.75	0.75	0.70
Updated Overall Accuracy	0.60	0.60	0.72	0.59	0.66

Table1 performance metrics for models

The enhancement in the F1-score underscores the models' improved competency in flagging potential delinquency.

However, this improvement did bring about a trade-off with a slight decline in overall accuracy. In the context of delinquency prediction, this trade-off is acceptable and often expected, as the heightened ability to detect true delinquent cases is critical. It reflects a strategic decision to prioritize the detection of delinquency over the model's overall accuracy rate, ensuring that fewer instances of delinquency go unnoticed.

5. Conclusion

We regard the SVM and Logistics Regression as the optimal model because they have achieved the highest updated F1-score post-data rebalancing, indicating a superior balance of precision and recall compared to other models. This metric is critical in the context of delinquency prediction, as it implies a solid ability to identify actual cases of delinquency while minimizing false detections. A high F1 score is particularly valuable in scenarios where the cost of false negatives is significant, such as in financial risk assessment, where failing to predict delinquency can lead to substantial losses.

To further enhance models' predictive power, several strategies could be employed:

1. Advanced Feature Engineering: Developing more complex and informative features

could help capture the nuances of delinquent behavior more accurately.

2. Class Weight Adjustment: Modifying the class weights can help the SVM focus on the delinquent class, which is usually underrepresented in the dataset.

3. Hyperparameter Optimization: Employing advanced techniques such as Bayesian optimization could lead to a more effective search for the optimal hyperparameters, potentially improving the model's performance.

These enhancements aim to refine the models' ability to generalize unseen data and further improve the precision-recall balance, making it even more effective for predicting delinquency.

Reference:

1. *Nikhil Arvind Jagannathan and Qiulei (Leo) Bao, Machine Learning–Based Systematic Investing in Agency Mortgage-Backed Securities*
2. *Sitzia, Luca and Baccaglini, Roberto and Malacchia, Vittorio and Cozzi, Federico, A Neural Network Approach for the Estimation of Mortgage Prepayment Rates (June 3, 2021). Available at SSRN: https://ssrn.com/abstract=4179429 or http://dx.doi.org/10.2139/ssrn.4179429*
3. *Han, Chang. Loan Repayment Prediction Using Machine Learning Algorithms. University of California, Los Angeles, 2019.*