Proceedings of CONF-SEML 2025 Symposium: Machine Learning Theory and Applications
DOI: 10.54254/2755-2721/158/2025.TJ23486

Evaluating Machine Learning Models for Personal Credit Risk: A Comparison of Logistic Regression, Random Forest, and XGBoost

Feifan Zhao

School of Computer Science, Xi'an Polytechnic University, Xi'an, China
42209210302@stu.xpu.edu.cn

Abstract: Traditional methods for predicting personal credit risk have historically lacked accuracy and comprehensiveness, failing to effectively analyze a large number of nonlinear user characteristics. Large-scale modeling algorithms have advanced quickly in recent years, and more and more financial institutions are adopting them to forecast the likelihood of personal credit default. However, challenges persist in understanding the distinct features and applicability of different models. In this research, a publicly accessible personal credit dataset from LendingClub covering the years 2007–2010 is empirically analyzed using three well-known algorithms: Logistic Regression (LR), Random Forest (RF), and eXtreme Gradient Boosting (XGBoost). The Synthetic Minority Over-sampling Technique (SMOTE) is used to generate synthetic samples for the minority class to tackle the notable class imbalance in the dataset. The performance of these models is evaluated and contrasted by four pivotal criteria and Receiver Operating Characteristic (ROC) curves. The outcomes illustrate that XGBoost outperforms the other models across all evaluated metrics. Based on these findings, this paper recommends XGBoost as the preferred algorithm for personal credit risk prediction.

Keywords: Personal credit risk, Logistic Regression, Random Forest, XGBoost, SMOTE

1. Introduction

In the early 20th century, people accumulated a certain level of wealth and were no longer merely concerned with meeting basic needs. Instead, they pursued higher aspirations, such as exceeding their immediate means through credit consumption. This demand led to the development of personal credit loan strategies. In 1946, John Biggins, a banker at Franklin National Bank, first proposed this concept. He designed a system that allowed bank customers to make purchases at local merchants, with the bank covering payments and billing customers monthly. This system is widely considered one of the earliest bank-issued credit cards, though restricted to local merchants and requiring bank-mediated transactions [1].

In 1958, Bank of America in California issued BankAmericard, the prototype of modern personal credit cards. This first widely used revolving credit card permitted purchases within preset credit limits and offered installment payment options. In 1976, BankAmericard was rebranded as Visa, now one of the largest global credit card networks [2].

However, following the 2008 U.S. subprime mortgage crisis, global unemployment rates surged, triggering economic downturns and sharp increases in personal credit defaults. Subsequently, the

DOI: 10.54254/2755-2721/158/2025.TJ23486

COVID-19 pandemic emerging in late 2019 caused widespread bankruptcies among individuals due to national epidemic control measures, exposing financial institutions to substantial credit default risks.

Prior to the adoption of machine learning methods, financial institutions relied on multiple credit evaluation approaches. The FICO score (300-850 range), a primary U.S. credit assessment system, evaluates multidimensional individual financial factors including credit history, debt levels, and repayment records to determine loan terms. Financial ratios like the Debt-to-Income Ratio (DTI) and Credit Utilization Ratio (CUR) remain prevailing indicators, with elevated values signaling default risks.

With advancements in machine learning and large model algorithm optimization, conventional credit evaluation systems have become obsolete. Modern personal credit assessments require analyzing complex feature interactions beyond traditional capabilities of methods. Simple predictive formulas cannot meet contemporary accuracy standards for credit risk assessment. Compared to mainstream machine learning algorithms, traditional default prediction methods demonstrate significantly lower efficiency and accuracy [3].

A variety of machine learning models, such as Random Forest (RF), eXtreme Gradient Boosting (XGBoost), Gradient Boosting Decision Tree (GBDT), Logistic Regression (LR), and Deep Learning, are commonly applied for predicting personal credit risk. Compared to conventional statistical techniques, research has indicated that machine learning significantly increases the precision of credit risk forecasts [4]. These machine learning models have demonstrated superior predictive power in assessing credit risk, effectively handling complex datasets and capturing intricate patterns that traditional methods might overlook [5]. However, the effectiveness of a single model is often limited, and in special cases, it becomes necessary to explore internal algorithms for potential model fusion and improvement. Guo developed an assessment index system and used the analytic hierarchy method to improve classification accuracy in his credit risk control technique, which was based on a weighted RF algorithm [6]. Liu proposed two improved GBDT models that address the limitations of traditional GBDT, such as limited feature diversity [7]. Ren innovatively applied GBDT to simulate the mass balance of Asian alpine glaciers, removing dependence on predefined physical laws and high computational cost [8]. In the paper investigated by Machado and Karray, optimal prediction was achieved using k-Means and DBSCAN for unsupervised learning, and AdaBoost, RF, decision trees, SVM, and neural networks for supervised learning [9].

In the data analysis of financial institutions like investment banks and stock exchanges, the utilization of machine learning for credit risk assessment has grown in prominence. However, most data analysts struggle to efficiently analyze the massive customer datasets at their disposal and select the most suitable and user-friendly model. The purpose of this investigation is to evaluate and contrast the performance of three prominent classical modeling algorithms LR, RF, and XGBoost in the domain of predicting credit risk.

2. Dataset and models

2.1. Dataset

This research takes advantage of the Kaggle dataset, which is openly accessible and includes actual loan data gathered from LendingClub from 2007 to 2010. The dataset comprises 9,536 rows and 14 columns. The first column, credit.policy, sets as the target variable, indicating whether a borrower meets LendingClub's credit underwriting criteria (1 represents non-default, 0 represents default). The remaining 13 columns are feature variables. Table 1 provides detailed descriptions of each feature.

DOI: 10.54254/2755-2721/158/2025.TJ23486

Table 1: Dataset attributes table

Columns	Description
credit. policy	Whether the borrower meets LendingClub's credit criteria
purpose	Loan purpose category
int. rate	Loan interest rate
installment	Monthly payment amount if loan is approved
log.annual.inc	Natural log of reported annual income
dti	Debt-to-income ratio
FICO	FICO credit score
days.with.cr.line	Total days since first credit line opened
revol.bal	Current balance on revolving credit accounts
revol.util	percentage of available revolving credit being used
inq.last.6mths	Recent 6-month credit inquiries count
delinq.2yrs	Past 2 years' 30+ day delinquencies
pub.rec	Number of negative public records
not.fully.paid	Unpaid loan status indicator (1 = defaulted)

2.2. Feature engineering

Missing values and outliers are analyzed using heat-map detection and box plot analysis, respectively. No missing values or significant outliers were identified. To ensure compatibility with the three models, categorical variables in the purpose column must be encoded into numerical formats. Nowadays, these are tremendous types of encoding algorithms applied in different models. According to numerous studies, One-Hot Encoding is one of the most effective approaches for LR to eliminate ordinal bias from categorical values. For the past relative paper, Label Encoding for tree-based models (RF and XGBoost), which efficiently handles categorical variables without increasing dimensionality [10]. The purpose column contains 7 categories, and Label Encoding preserves computational efficiency while avoiding high-dimensional sparse matrices.

2.3. Models

2.3.1. Logistic Regression

LR is a Generalized Linear Model, specifically designed to address binary classification problems. In the context of credit risk prediction, its objective is to construct a probabilistic model based on historical data with the purpose of predicting the likelihood of a borrower defaulting or the capacity of refunding regularly. LR is essentially a probabilistic classifier, with its output being the probability of the event occurring $P(y = 1|\boldsymbol{x})$, which is then mapped to class labels through a threshold (typically 0.5 for binary classification problems).

LR employs the Sigmoid function to convert the continuous values of the linear combination to the interval (0,1), supposing a linear relationship between the features and the target variable. The mathematical form is:

$$P(y = 1|\boldsymbol{x}) = \sigma(\boldsymbol{w}^T\boldsymbol{x} + b) = \frac{1}{1+e^{-(\boldsymbol{w}^T\boldsymbol{x}+b)}} \quad (1)$$

In this formula, $\boldsymbol{x}$ represents the input feature vector; $\boldsymbol{w}$denotes the model weight parameters, reflecting the strength and direction of influence of each feature on default risk. $\boldsymbol{w}$ is the bias term, adjusting the intercept of the decision boundary; $(\boldsymbol{w}^T\boldsymbol{x} + b)$ is the Sigmoid function, ensuring the output value lies within (0,1).

DOI: 10.54254/2755-2721/158/2025.TJ23486

LR employs the log-likelihood function as the optimization objective. However, it typically uses the Negative Log-Likelihood to minimize error. The mathematical expression is

$$J(\mathbf{w}, b) = -\sum_{i=1}^{N} [y_i \log p_i + (1 - y_i) \log (1 - p_i)] \tag{2}$$

In this formula, $p_i = \sigma(\mathbf{w}^\top \mathbf{x}_i + b)$,$y_i$ is the true label.

The loss function is essentially the cross-entropy loss, which measures the divergence between the true distribution y and the predicted distribution p.

The parameters $\mathbf{w}$ and $\mathbf{b}$ in LR are optimized via gradient descent. The updated rules are:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta \frac{\partial J}{\partial \mathbf{w}} \tag{3}$$

$$\mathbf{b}^{(t+1)} = \mathbf{b}^{(t)} - \eta \frac{\partial J}{\partial \mathbf{b}} \tag{4}$$

In these two iterative functions,η is step size.

By deriving the partial derivatives of the loss function $J(\mathbf{w}, b)$

$$\frac{\partial J}{\partial \mathbf{w}} = \sum_{i=1}^{N} (p_i - y_i)\mathbf{x}_i \tag{5}$$

$$\frac{\partial J}{\partial b} = \sum_{i=1}^{N} (p_i - y_i) \tag{6}$$

the iterative update functions are obtained:

$$\mathrm{w}^{(t+1)} = \mathrm{w}^{(t)} - \eta \sum_{i=1}^{N} (p_i - y_i)\mathrm{x}_i \tag{7}$$

$$b^{(t+1)} = b^{(t)} - \eta \sum_{i=1}^{N} (p_i - y_i) \tag{8}$$

2.3.2. Random Forest

RF is an ensemble learning technique that builds several decision trees and aggregates their output to increase generalization ability. The model leverages Bootstrap Sampling and Feature Subset Selection to introduce randomness during tree construction, thereby reducing overfitting risk and enhancing prediction accuracy. Each of the several decision trees that make up an RF was trained using a distinct subset of the training dataset.

Given a training dataset:

$$D = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N} \tag{9}$$

Here, $\mathbf{x}_i$ is the feature vector of the i-th sample, and y_i is the target variable.

The prediction output of the RF is the majority voting result of all decision trees:

$$\hat{y} = \arg \max_{c} \sum_{t=1}^{T} 1(f_t(\mathbf{x}) = c) \tag{10}$$

where $f_t(\mathbf{x})$ is the prediction of the t-th tree, T is total trees count, c is class label, and $1(f_t(\mathbf{x}) = c)$ is the indicator function.

The training process primarily involves Bootstrap Sampling, where N samples are randomly drawn with replacement from the original dataset to train each decision tree. Feature Subset Selection happens during the construction of each tree, $m \ll d$ features are randomly selected from the total d features as candidates as each node splitting, and the optimal feature is chosen for splitting. The splitting process is typically optimized using the Gini index in classification problems:

$$G(D) = 1 - \sum_{c} p_c^2 \tag{11}$$

Here p_c denotes the probability of class **c** in the dataset.

DOI: 10.54254/2755-2721/158/2025.TJ23486

The training of RF can be viewed as an iterative optimization process, where the construction of each decision tree follows a recursive splitting procedure. The iterative equation is:

$$D_t^{(j+1)} = D_t^{(j)} - \arg\ \max_{\theta} \Delta(D_t, \theta) \tag{12}$$

Here θ represents the optimal split feature and split point and $\Delta(D_t, \theta)$ denotes the splitting gain of the current node.

The entire forest training process converges to a stable ensemble, where each tree f_t approximately satisfies:

$$\mathbb{E}[f_t(\mathbf{x})] \approx f^*(\mathbf{x}) \tag{13}$$

Here $f^*(\mathbf{x})$ is the theoretically optimal decision function.

In this research, credit risk analysis is a classical binary classification problem, and the cross-entropy loss function is employed:

$$L(y, \hat{y}) = -\sum_{i=1}^{N} \sum_{c} 1(y_i = c)\log\ p_c \tag{14}$$

2.3.3. eXtreme Gradient Boosting

XGBoost is a machine learning model improved from GBDT. Compared to traditional GBDT, XGBoost enhances model performance and convergence speed by optimizing the objective function, incorporating regularization terms, column sampling, and parallel computation and so on.

XGBoost is an additive model that constructs T decision trees in a stepwise optimization manner to minimize the objective function. Its mathematical formulation is:

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} f_k(\mathbf{x}_i) \tag{15}$$

where $\hat{y}_i^{(t)}$ represents the sample $\mathbf{x}_i$'s expected value after t-th iteration, f_k denotes k-th decision tree, and t is the current iteration step.

XGBoost approximates the true target value y_i by learning a sequence of decision trees f_k:

$$\hat{y}_i = \sum_{k=1}^{T} f_k(\mathbf{x}_i) \tag{16}$$

Each tree f_k is selected from a function space , defined as:

$$f_k(\mathbf{x}) = w_{q(\mathbf{x})} \tag{17}$$

where $q(\mathbf{x})$ represents the structure of the decision tree, mapping the sample $\mathbf{x}$ to a leaf node index, and $w_{q(\mathbf{x})}$ is the weight of the corresponding leaf node.

In order to optimize the model, XGBoost minimizes the goal function, which is made up of two parts:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{N} l(y_i, \hat{y}_i^{(t)}) + \sum_{k=1}^{t} \Omega(f_k) \tag{18}$$

The term $l(y_i, \hat{y}_i)$ represents the logarithmic loss function, its complete expression is:

$$l(y_i, \hat{y}_i) = -[y_i\log\ \sigma(\hat{y}_i) + (1 - y_i)\log\,(1 - \sigma(\hat{y}_i))] \tag{19}$$

where $\sigma(\hat{y}_i) = \frac{1}{1+e^{-\hat{y}_i}}$ is the sigmoid activation function.

The regularization term $\Omega(f_k)$ controls model complexity:

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{20}$$

DOI: 10.54254/2755-2721/158/2025.TJ23486

Here, γT represents penalty item of the quantity of leaf nodes in order to prevent overfitting, and $\lambda \sum w_j^2$ constrains the leaf node weights.

XGBoost greatly increases optimization efficiency by approximating the loss function via the second-order Taylor expansion. The expanded loss function at the t-th iteration is:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{N} \left[l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i) \right] + \Omega(f_t) \tag{21}$$

First-order derivative (gradient):

$$g_i = \frac{\partial l(y_i, \hat{y}_i)}{\partial \hat{y}_i} \tag{22}$$

Second-order derivative (Hessian):

$$h_i = \frac{\partial^2 l(y_i, \hat{y}_i)}{\partial \hat{y}_i^2} \tag{23}$$

Suppose the new tree f_t has J leaf nodes, and the score of each leaf node j is w_j. Following that, the goal function can be redefined as:

$$\mathcal{L}^{(t)} = \sum_{j=1}^{J} \left[G_j w_j + \frac{1}{2} H_j w_j^2 \right] + \gamma J + \frac{1}{2} \lambda \sum_{j=1}^{J} w_j^2 \tag{24}$$

In this function, $G_j = \sum_{i \in I_j} g_i$ is the sum of gradients for leaf node j; $H_j = \sum_{i \in I_j} h_i$ is the sum of second derivatives (Hessians) for leaf node j.

By taking the derivative of w_j and setting it to zero, the optimal leaf node weight is derived:

$$w_j^* = -\frac{G_j}{H_j + \lambda} \tag{25}$$

The most effective splitting criterion for the tree is obtained by substituting this into the goal function:

$$\mathcal{L}_{\text{split}} = \frac{1}{2} \left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma \tag{26}$$

If this value is greater than zero, the split is executed; otherwise, splitting terminates, and the iteration concludes.

3. Experimental set-up

3.1. Positive and negative example

The dataset exhibits severe class imbalance with 7,710 positive instances versus 1,868 negative instances. The three most prevalent strategies for addressing this imbalance include under-sampling, over-sampling, and Synthetic Minority Over-sampling Technique (SMOTE) [11].

The simple method of traditional under-sampling involves decreasing the size of the dominant class at random to equal that of the less frequent class. While it may reduces computational costs and training time, this strategy risks discarding critical information and limiting the model's learning capacity. Sometimes it is unacceptable, especially in small datasets. Classic oversampling may contribute to bias in small datasets due to overfitting and degraded generalization performance, leading to higher prediction errors. Hence, this research adopts the SMOTE algorithm.

SMOTE is essentially an oversampling strategy that synthesizes new minority-class samples rather than replicating existing ones. To be more specific, a minority-class sample x_i is selected randomly and a neighboring sample x_{NN} which is stochastically chosen from its k-nearest neighbors. A synthetic sample x_{new} is generated using the following function [12].

DOI: 10.54254/2755-2721/158/2025.TJ23486

$$x_{\text{new}} = x_i + \lambda \cdot (x_{NN} - x_i) \tag{27}$$

Here λ is a random number in (0,1).

3.2. Standardization

Standardization is applied during preprocessing to scale features to a uniform range, enhancing model stability and convergence speed, particularly for gradient descent-based algorithms like LR. Standardized features (mean = 0, standard deviation = 1) reduce gradient descent step sizes and accelerate training. In this dataset, features exhibit vastly different scales such as *int.rate* < 0.5 and *revol.bal* ranging from hundreds to hundreds of thousands. Without standardization, large-scale features could dominate training and lead to skewing predictions.

4. Result

4.1. Evaluation

Table 2: Performance comparison of three models

Models/Metrics	accuracy	precision	recall	F1
LR	0.8476	0.9439	0.8619	0.9010
RF	0.9755	0.9876	0.9818	0.9847
XGBoost	0.9849	0.9896	0.9916	0.9906

The comparative analysis of model performance in Table 2 reveals XGBoost's consistent dominance over LR and RF across all metrics. LR, though interpretable, exhibits significant limitations (accuracy: 0.8476; recall: 0.8619), reflecting its inability to model non-linear feature interactions critical for imbalanced credit risk prediction. XGBoost's superiority stems from its regularization mechanisms, second-order optimization for efficient convergence, and capacity to capture complex feature interdependencies—advantages that align with the high-dimensional, non-linear nature of credit datasets. These findings underscore XGBoost's suitability for financial risk assessment, balancing predictive power and generalization, while lower recall of LR highlights its inadequacy in identifying high-risk borrowers.

Next, this research concentrates on the ROC curves of the three models, shown in the following Figure 1:

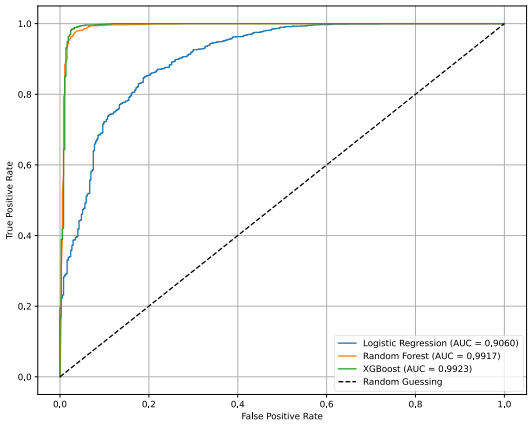

Figure 1: ROC Curve comparison of three models (picture credit: original)

DOI: 10.54254/2755-2721/158/2025.TJ23486

Figure 1 presents ROC curves comparing the three models, with Area Under the Curve (AUC) values indicating discriminatory power. XGBoost (AUC = 0.9923) slightly outperforms RF (AUC = 0.9917), underscoring its robustness in distinguishing defaulters from non-defaulters. LR (AUC = 0.9060), while acceptable, significantly underperforms the ensemble methods, highlighting the superiority of tree-based models in capturing complex, non-linear patterns. The close proximity of XGBoost and RF curves reflects their high effectiveness, with the marginal edge of XGBoost attributed to its iterative error correction and feature weighting.

Overall, these findings validate the effectiveness of ensemble learning approaches in financial risk prediction, suggesting that institutions seeking to optimize risk classification should prioritize advanced tree-based models over traditional linear approaches.

4.2. Feature importance analysis

Permutation Importance serves in this research to measure the significance of each attribute. By examining the decline in model prediction capacity when the values of a feature are randomly shuffled, this model-agnostic technique assesses the relevance of a feature. The core principle is that if a feature is critical to the model's prediction, shuffling its values disrupts the actual relationship between the feature and the target variable, leading to a significant deterioration in evaluation metrics[13]. Conversely, if a feature is negligible, shuffling its values results in minimal changes to the performance of the model [14].

The following Figures 2, 3, and 4 display the feature importance bar charts for the three models.

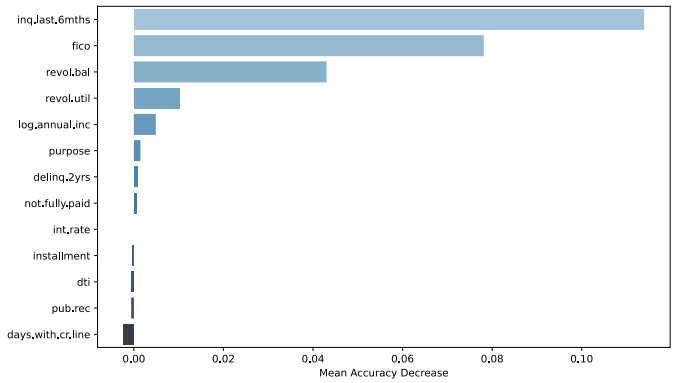

Figure 2: Logistic Regression permutation importance (picture credit: original)

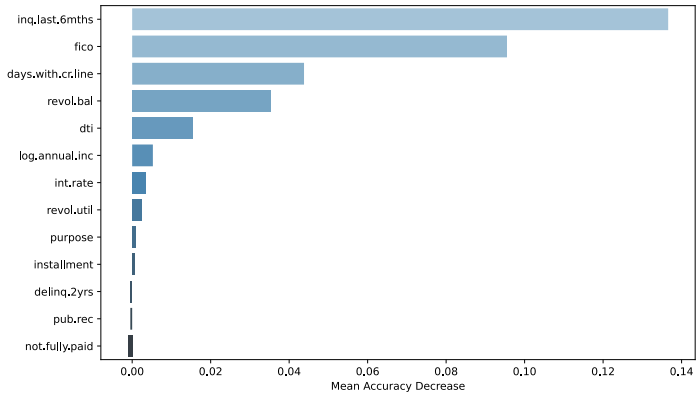

Figure 3: Random Forest permutation importance (picture credit: original)

DOI: 10.54254/2755-2721/158/2025.TJ23486

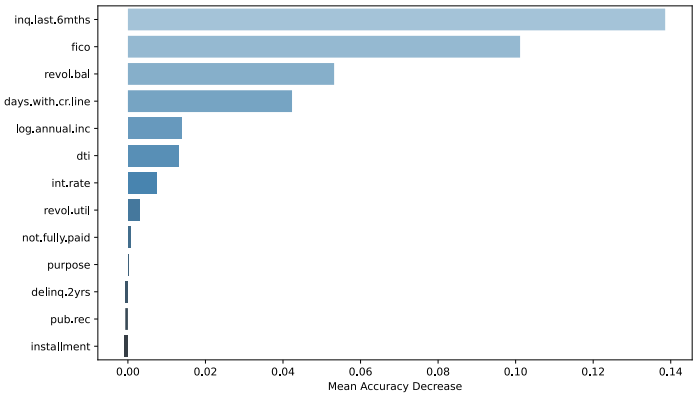

Figure 4: XGBoost permutation importance (picture credit: original)

The three figures reveal distinct patterns in how each model prioritizes features for credit risk prediction. LR emphasizes linearly separable features such as inq.last.6mths and fico. However, its limited ability to capture non-linear relationships results in lower recall and overall performance. In contrast, RF and XGBoost, as tree-based ensemble methods, demonstrate greater flexibility in identifying complex interactions, prioritizing additional features like days.with.cr.line and dti. XGBoost further enhances feature importance by incorporating regularization, highlighting revol.bal as a critical indicator of default risk. Tree-based models achieve superior predictive accuracy, making them ideal for minimizing financial risk in credit risk analysis.

5. Conclusion

With the rapid expansion of financial services and the increasing demand for accurate credit risk evaluation, leveraging advanced machine learning techniques has become pivotal for effective decision-making in lending institutions. In the context of evaluating personal credit risk, this paper performs a thorough comparative examination of three popular classical machine learning algorithms. The SMOTE technique is effectively applied to synthesize new minority-class samples to eliminate the intrinsic class imbalance in actual financial data.

Experiments on a sizable LendingClub dataset show that XGBoost generally performs better than LR and RF in a variety of assessment measures. XGBoost is especially appropriate for the intricate process of evaluating personal credit due to its capacity in coping with high-dimensional data and simulating nonlinear relationships. Moreover, it exhibits strong robustness and generalizability under different sampling conditions.

This research endeavors to juxtapose the efficacy of three distinct models within the domain of forecasting the likelihood of credit default, ultimately aiming to identify the most optimal solution. In the future, further experiments may explore hybrid models that integrate the strengths of multiple algorithms, potentially achieving improved predictive performance through ensemble approaches.

In conclusion, this paper underscores the practical superiority of XGBoost in handling nonlinear, imbalanced financial datasets and offers valuable insights into building more reliable and intelligent credit assessment models in real-world financial applications.

References

[1] *Li, S. (2024). Machine Learning in Credit Risk Forecasting — A Survey on Credit Risk Exposure. Accounting and Finance Research, 13(2), 107-107.*

[2] *Mousaab, E. G., Chaabita, R., & Idamia, S. (2024). Machine Learning dans l'évaluation du risque crédit : revue systématique. Zenodo.*

DOI: 10.54254/2755-2721/158/2025.TJ23486

[3] *Ahmed, F., Nizam, K., Sajid, Z., Qamar, S., & Ahsan. (2024). Striking a balance: Evaluating credit risk with traditional and machine learning models. Bulletin of Business and Economics, 13(3), 30–35.*

[4] *Suhadolnik, N., Ueyama, J., & Da Silva, S. (2023). Machine Learning for Enhanced Credit Risk Assessment: An Empirical Approach. Journal of Risk and Financial Management, 16(12), 496.*

[5] *Shi, S., Tse, R., Luo, W., D'Addona, S., & Pau, G. (2022). Machine learning-driven credit risk: a systemic review. Neural Computing and Applications, 34(17), 14327-14339.*

[6] *Yangyudongnanxin, G. (2021). Financial credit risk control strategy based on weighted random forest algorithm. Scientific Programming, 2021, 1–9.*

[7] *Liu, W., Fan, H., & Xia, M. (2022). Credit scoring based on tree-enhanced gradient boosting decision trees. Expert Systems with Applications, 189, 116034.*

[8] *Ren, W., Zhu, Z., Wang, Y., Su, J., Zeng, R., Zheng, D., & Li, X. (2024). Comparison of Machine Learning Models in Simulating Glacier Mass Balance: Insights from Maritime and Continental Glaciers in High Mountain Asia. Remote Sensing, 16(6), 956.*

[9] *Machado, M. R., & Karray, S. (2022). Assessing credit risk of commercial customers using hybrid machine learning algorithms. Expert Systems with Applications, 200, 116889.*

[10] *Pargent, F., Pfisterer, F., Thomas, J., & Bischl, B. (2022). Regularized target encoding outperforms traditional methods in supervised machine learning with high cardinality features. Computational Statistics, 37(5), 2671–2692.*

[11] *Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, 16, 321–357.*

[12] *Elreedy, D., Atiya, A. F., & Kamalov, F. (2024). A theoretical distribution analysis of synthetic minority oversampling technique (SMOTE) for imbalanced learning. Machine Learning, 113(7), 4903–4923.*

[13] *Fisher, A., Rudin, C., & Dominici, F. (2019). All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. Journal of Machine Learning Research, 20(177), 1-81.*

[14] *Molnar, C., König, G., Bischl, B., & Casalicchio, G. (2024). Model-agnostic feature importance and effects with dependent features: a conditional subgroup approach. Data Mining and Knowledge Discovery, 38(5), 2903-2941.*