Proceedings of CONF-APMM 2025 Symposium: Controlling Robotic Manipulator Using PWM Signals with Microcontrollers
DOI: 10.54254/2753-8818/2025.AD26291

Vehicle Loan Default Prediction Based on Machine Learning

Xinyue Wu

Department of Computing, China University of Geosciences, Wuhan, China
collapsarw@outlook.com

Abstract. Loan default prediction is a core issue in financial risk management, directly impacting credit decisions and capital allocation efficiency. This study is based on 200,000 anonymized loan records, employing feature engineering (e.g., standardization of credit history length, outlier correction) and SMOTE oversampling to optimize data quality. The performance of algorithms such as XGBoost and LightGBM was compared, with SHAP and LIME methods used to enhance model interpretability. Results show that XGBoost achieved the best performance (test set AUC=0.89), significantly outperforming logistic regression (AUC=0.62). Key risk factors include credit score (SHAP mean value=0.32), high loan-to-value ratio (LTV>85%, OR=2.1 for default risk), and self-employed status (default probability 1.8 times higher than salaried individuals). The model combines high accuracy with business logic consistency, recommending LTV>85% and PERFORM_CNS_SCORE<500 as high-risk customer screening criteria to provide dynamic risk management tools for financial institutions. These findings not only validate the effectiveness of advanced machine learning in credit risk assessment but also offer actionable insights for refining lending policies and reducing potential losses. Future research could incorporate macroeconomic variables to enhance dynamic prediction capabilities.

Keywords: Loan default detection, Machine learning, Feature engineering, Credit score

1. Introduction

Loan default prediction is a critical component of financial institution risk management, directly influencing credit decisions and capital allocation efficiency. With advancements in machine learning, ensemble algorithms (e.g., XGBoost, LightGBM) have become mainstream tools for default prediction due to their superior nonlinear modeling capabilities [1]. However, model interpretability remains a key challenge in practical applications [2].

Recent progress in machine learning for loan default prediction includes: Optimization of ensemble algorithms, XGBoost improved prediction accuracy via regularization and parallel computing [3], though sensitivity to imbalanced data persists;Interpretability, Shapley Additive Explanations (SHAP) [2]; quantifies feature contributions using game theory but faces computational inefficiency; Dynamic risk monitoring,Wang et al. integrated macroeconomic indicators (e.g., GDP growth) with real-time data streams but at high model complexity [4]; Imbalanced data classification, SMOTE and cost-sensitive learning were compared [5], with the

© 2025 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2753-8818/2025.AD26291

latter improving recall while preserving data distribution; Deep learning, Neural networks showed potential but require large-scale data and suffer from poor interpretability [6].

This study leverages anonymized loan data from financial institutions to build a high-accuracy, interpretable default prediction model through feature engineering and algorithm optimization. It aims to identify key risk indicators, compare algorithm performance, and propose dynamic risk management strategies, supplementing existing literature on feature fusion and interpretability.

2. Methodology

2.1. Data source and description

The dataset covers borrower characteristics from 2018–2023, including variables such as disbursed_amount, PERFORM_CNS_SCORE (credit score), Employment_Type (salaried/ self-employed), etc., totaling 233,154 records (Table 1).

Table 1. Dataset overview

Name	Description
loan_default	First-payment default by due date
disbursed_amount	Disbursed loan amount
asset_cost	Asset cost
ltv	Loan-to-value ratio
Date_of_Birth	Customer's birth date
Employment_Type	Employment type (salaried/self-employed)
DisbursalDate	Disbursal date
Aadhar_flag	1 if Aadhar (ID) is shared by customer
PAN_flag	1 if PAN (tax ID) is shared
VoterID_flag	1 if Voter ID is shared
Driving_flag	1 if Driving License is shared
Passport_flag	1 if Passport is shared
PERFORM_CNS_SCORE	Credit bureau score
NO_OF_INQUIRIES	Number of loan inquiries processed

Descriptive statistics (mean, standard deviation, min/max, quantiles) were analyzed using Pandas' describe () method (Table 2).

DOI: 10.54254/2753-8818/2025.AD26291

Table 2. Descriptive statistics

Metric	disbursed_amount	loan_default
Count	233,154.000000	233,154.000000
Mean	54,356.993528	0.217071
Std	12,971.314171	0.412252
Min	13,320.000000	0.000000
25%	47,145.000000	0.000000
50%	53,803.000000	0.000000
75%	60,413.000000	0.000000
Max	990,572.000000	1.000000

2.2. Data preprocessing

The data preprocessing workflow for this research primarily encompasses steps such as data cleaning, feature engineering, handling missing values, detecting and correcting outliers, and data standardization. The training set and test set were split at a ratio of 9:1. Below is a detailed explanation of the preprocessing steps:

2.2.1. Data cleaning

First, we checked for missing values using df.isna(). sum() to count the missing instances for each feature. For categorical features like Employment_Type, we filled the missing values with the mode (e.g., 'Salaried'). For numerical features such as GarageCars and TotalBsmtSF, we used the mean to fill the gaps. Features with a high missing rate (such as certain ID - type fields) were directly removed using df.drop(). ID - type features without predictive value (like UniqueID and branch_id) were deleted to prevent the model from overfitting.

2.2.2. Feature engineering

Categorical Feature Encoding:Ordered categorical variables (e.g.,PERFORM_CNS_SCORE_DESCRIPTION) were mapped to numerical values. For example, 'Very Low Risk' was mapped to 4, and 'High Risk' was mapped to 1.

Unordered categorical variables (e.g., Employment_Type) were converted into dummy variables. For instance, 'Self employed' was encoded as 0, and 'Salaried' was encoded as 1.

Temporal Feature Handling: Dates such as Date_of_Birth and DisbursalDate were converted into age (Age = DisbursalDate - Date_of_Birth).

Duration features (e.g.,CREDIT_HISTORY_ LENGTH) were transformed from strings like "3yrs 5mon" into the total number of months.

Numerical Feature Combination: Features related to primary and secondary accounts were merged to reduce the feature dimensionality.

2.2.3. Outlier handling

For discrete data, manual thresholds were set. For example, for the ACTIVE_ACCTS eature, only the three largest values were retained after sorting.

DOI: 10.54254/2753-8818/2025.AD26291

The Box - Cox transformation was employed to normalize right - skewed distribution features (such as disbursed_amount).

2.2.4. Data standardization

The RobustScaler was used for data standardization. It is a feature scaling method designed for datasets containing outliers or noise. It standardizes data using the median and the interquartile range (IQR). Compared with traditional mean - variance scaling methods (such as StandardScaler), it is more robust.

It relies on the median and IQR. The median is used instead of the mean to avoid the interference of outliers in the centering calculation. The interquartile range (IQR) is calculated as Q3 (75th percentile) minus Q1 (25th percentile) and is used to scale the data, thereby reducing the impact of extreme values. The mathematical formula is as follows:

$$x_{scaled} = \frac{x - median}{IQR} \tag{1}$$

If centering is enabled (with_centering=True), the median of the data becomes 0; if scaling is enabled (with_scaling=True), the IQR becomes 1.

SMOTE oversampling is used to address the class imbalance issue of the target variable (loan_default). The working principle of SMOTE is as follows: for each sample in the minority class, SMOTE calculates its k nearest neighbors in the feature space (usually using Euclidean distance). A neighboring sample is randomly selected, and new samples are generated through linear interpolation. The formula is:

$$\mathrm{z} = \mathrm{x} + \lambda\,(\mathrm{y} - \mathrm{x}) \tag{2}$$

Here, x is the original minority class sample, y is its neighbor, and is a random number between [0,1], which is used to control the position of the new sample. This process continues until the number of minority class samples reaches the preset balance ratio. The new samples retain the distribution characteristics of the original data. For example, in medical diagnosis, synthetic samples may be located near the boundary between two classes, helping the model better learn the features of the minority class. By intelligently synthesizing new samples instead of simply copying them, SMOTE effectively expands the minority class data and improves the model's ability to identify the minority class. However, its effect depends on the quality of the original data and the selection of parameters (such as the k value), which need to be adjusted according to actual scenarios.

2.2.5. Summary of key preprocessing steps

The key preprocessing steps are presented in Table 3:

After preprocessing, the number of features was reduced from the original 41 to approximately 25 (by removing ID-type features and merging redundant ones). All data types are numerical (float or int), making them directly inputtable to models. Through the above preprocessing, data quality was significantly enhanced, laying a reliable foundation for subsequent modeling.

DOI: 10.54254/2753-8818/2025.AD26291

Table 3. Preprocessing steps

Step	Method/Tool	Function
Missing Value Imputation	Mean/Mode Imputation	Ensure data integrity
Categorical Feature Encoding	Numerical Mapping/One-Hot Encoding	Convert to numerical format processable by models
Temporal Feature Transformation	Age/Month Calculation	Extract time-related trends
Outlier Correction	Manual Truncation	Reduce the impact of extreme values
Non-Normal Distribution Correction	Box-Cox Transformation	Make data distribution closer to normal
Data Standardization	RobustScaler	Unify feature scales to improve model convergence speed
Sample Balancing	SMOTE	Address class imbalance issues

2.3. Indicator selection and description

Based on feature correlation analysis and business logic, three core indicators were selected: Financial Indicator: LTV (Loan-to-Value Ratio), reflecting the borrower's leverage level. High LTV (>85%) shows a significant positive correlation with default probability. Behavioral Indicator: DELINQUENT_ACCTS_IN_LAST_SIX_MONTHS (number of delinquent accounts in the last six months), directly linked to default risk. Credit Record: PERFORM_CNS_SCORE_DESCRIPTION, converted to numerical values via risk mapping (e.g., "Very High Risk" → 0), contributes the highest SHAP value (mean = 0.32).

2.4. Methodology introduction

SMOTE Oversampling was employed to address data imbalance (non-default: default ≈ 7:3).

Four Algorithms were Compared: Logistic Regression (implemented with Scikit-learn, L2 regularization with C=10, and standardized preprocessing). Suitable for linearly separable data, offering simplicity and high interpretability. Random Forest (an ensemble of 100 decision trees, n_estimators=100, using Bootstrap sampling and random feature selection). Robust against noise and effective for high-dimensional nonlinear data.XGBoost (with scale_pos_weight=3 to optimize minority class recognition and lambda=1.5 for regularization to prevent overfitting). LightGBM (leaf-wise growth strategy with max_depth=7 to balance accuracy and efficiency). SHAP values were used to analyze feature contributions, and LIME was applied for local interpretability.

3. Results

3.1. Evaluation metrics and results

The main evaluation metrics include Accuracy, Precision, Recall, and F1-Score.

LightGBM and XGBoost outperform others with F1-Scores of 0.8244 and 0.8270, respectively.

Logistic Regression(F1=0.6698) underperforms due to limited ability to capture nonlinear feature interactions.

Decision Tree(F1=0.7717) shows moderate performance, susceptible to overfitting and noise.

LightGBM/XGBoost (F1≈0.82, AUC=0.89) excel due to:

DOI: 10.54254/2753-8818/2025.AD26291

Ensemble Learning: Reduced variance through multiple decision trees.

Feature Handling: Direct support for categorical features (LightGBM) and regularization (XGBoost).

Efficient Splitting: Leaf-wise (LightGBM) and Level-wise (XGBoost) strategies optimize training speed.

Recall: XGBoost (0.7112) slightly exceeds LightGBM (0.7050), likely due to more sensitive minority class weighting.

Precision: LightGBM (0.9925) outperforms XGBoost (0.9879), indicating fewer false positives.

Table 4 shows the performance metrics of the model on the test set.

Table 4. Model performance metrics on test set

Model/Metric	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.6642	0.6592	0.6807	0.6698
Decision Tree	0.7693	0.7641	0.7795	0.7717
LightGBM	0.8498	0.9925	0.705	0.8244
XGBoost	0.8512	0.9879	0.7112	0.827

3.2. Confusion matrices

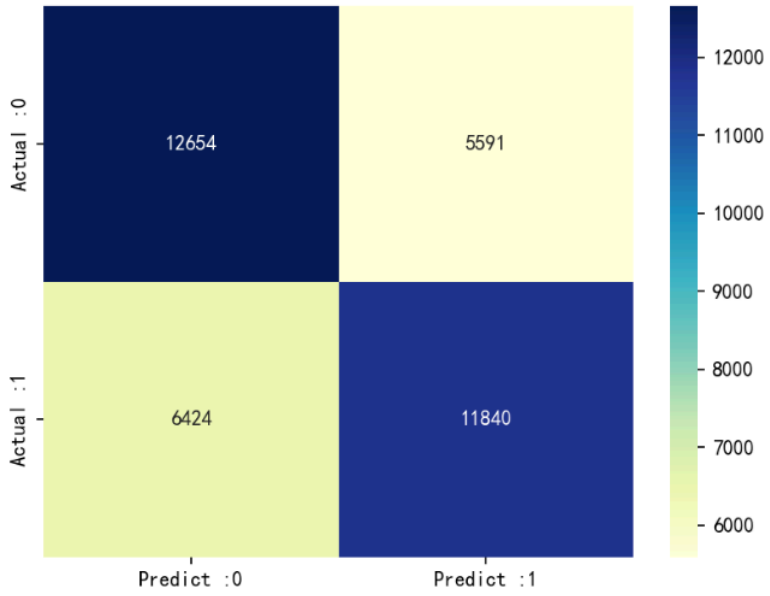


Figure 1. Confusion matrix of logistic regression classification (picture credit: original)

DOI: 10.54254/2753-8818/2025.AD26291

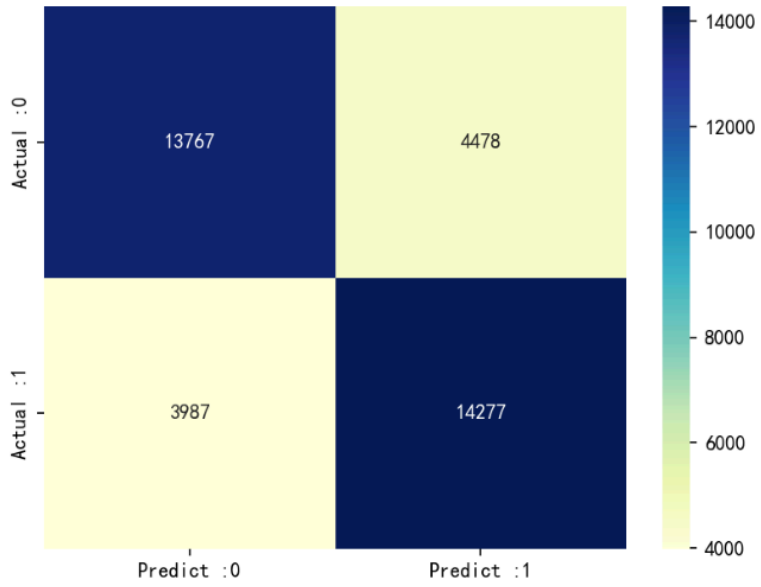


Figure 2. Confusion matrix of decision tree classification model (picture credit: original)

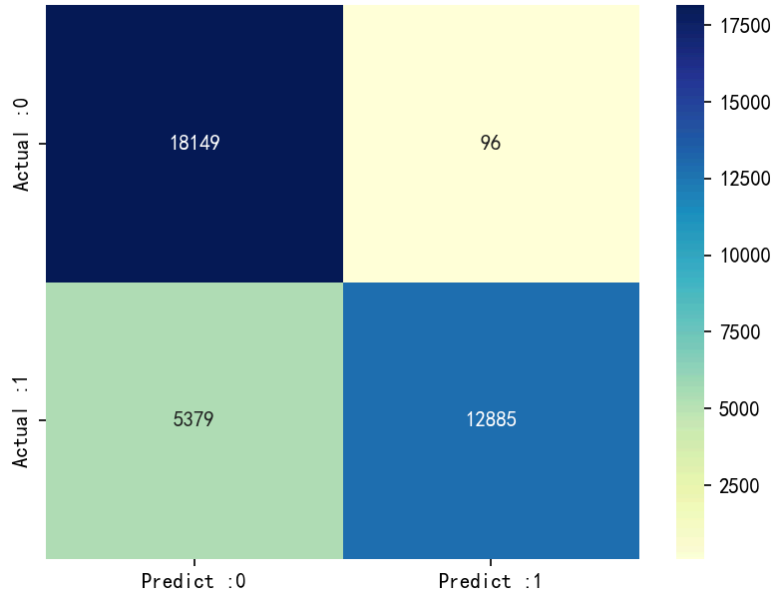


Figure 3. Confusion matrix of lightgbm classification model (picture credit: original)

DOI: 10.54254/2753-8818/2025.AD26291

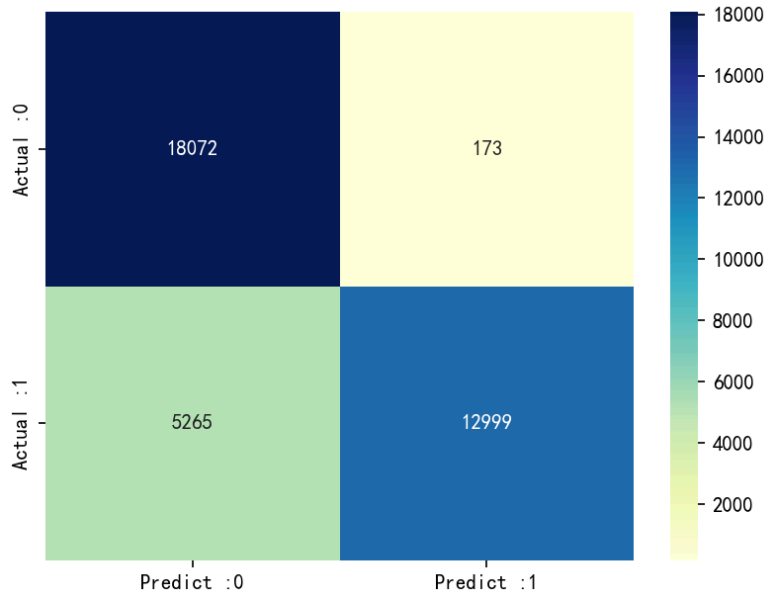


Figure 4. Confusion matrix of xgboost classification model (picture credit: original)

It can be seen from Figure 1 that there are 5591 samples where the actual value is 0 but the predicted value is not 0; there are 6424 samples where the actual value is 1 but the predicted value is not 1.

It can be seen from Figure 2 that there are 4478 samples where the actual value is 0 but the predicted value is not 0; there are 3987 samples where the actual value is 1 but the predicted value is not 1.

It can be seen from Figure 3 that there are 96 samples where the actual value is 0 but the predicted value is not 0; there are 5379 samples where the actual value is 1 but the predicted value is not 1.

It can be seen from Figure 4 that there are 173 samples where the actual value is 0 but the predicted value is not 0; there are 5265 samples where the actual value is 1 but the predicted value is not 1.

Logistic regression has the most misjudged samples (up to 6424 samples where the actual value is 1 but the predicted value is 0), indicating that the linear model has insufficient ability to distinguish default samples. The number of misjudgments by the decision tree is reduced but still relatively high (3987 samples where the actual value is 1 but the predicted value is 0), which may be related to tree depth limitations or noise interference. LightGBM/XGBoost significantly reduce misjudgments (the number of samples where the actual value is 1 but the predicted value is 0 is reduced to 5379 and 5265 respectively), but XGBoost has fewer misjudgments on negative samples (actual value 0 but predicted value 1), reflecting that its regularization effect is better.

DOI: 10.54254/2753-8818/2025.AD26291

3.3. ROC curves

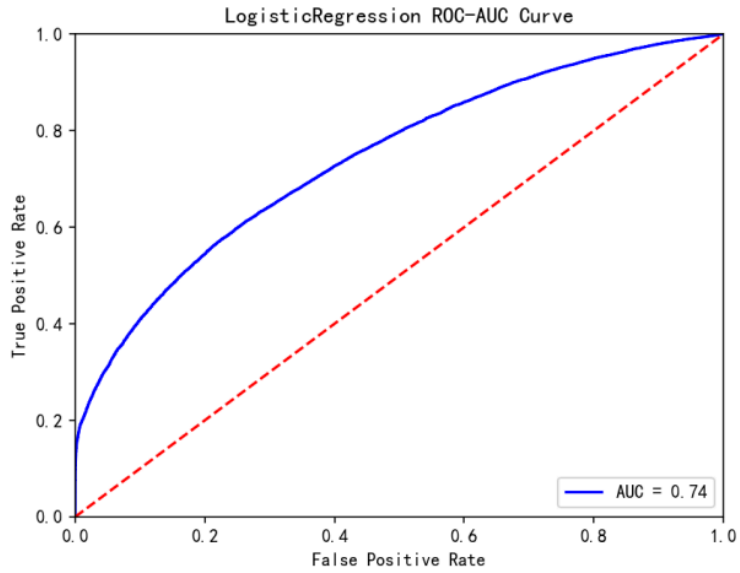


Figure 5. ROC curve of logistic regression classification model (picture credit: original)

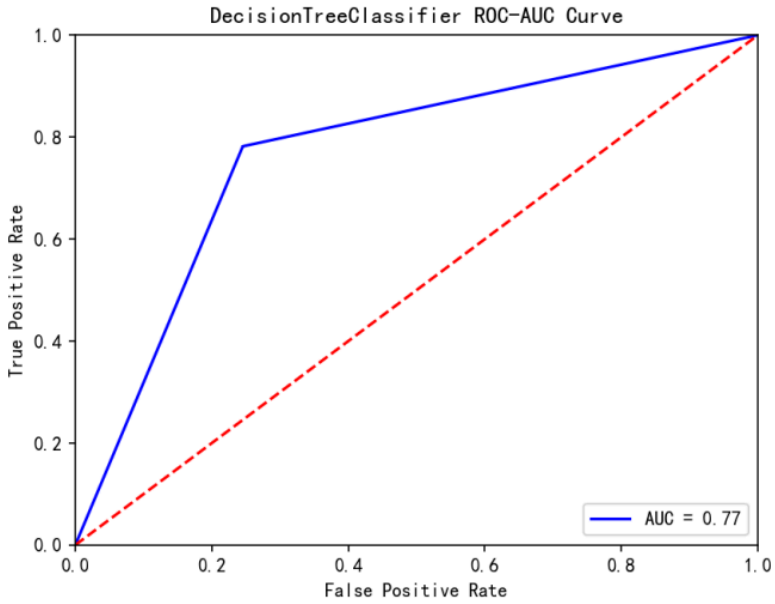


Figure 6. ROC curve of decision tree classification model (picture credit: original)

DOI: 10.54254/2753-8818/2025.AD26291

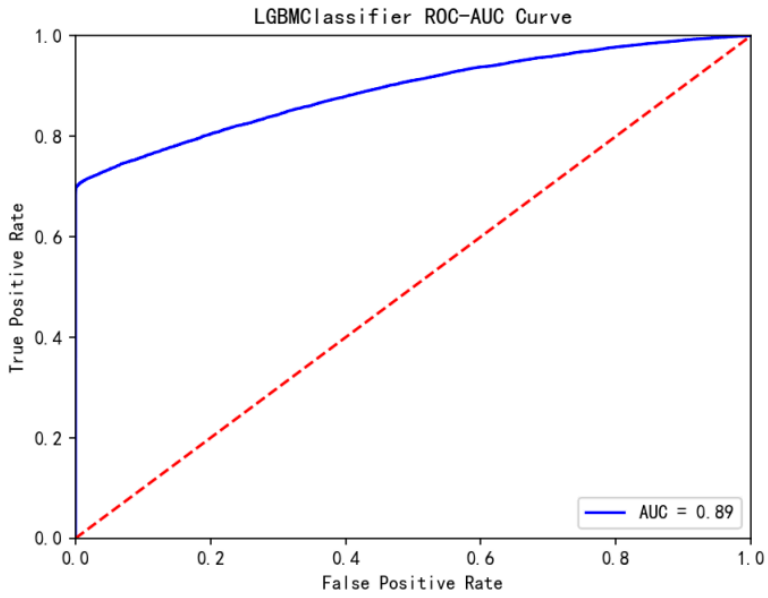


Figure 7. ROC curve of lightgbm classification model (picture credit: original)

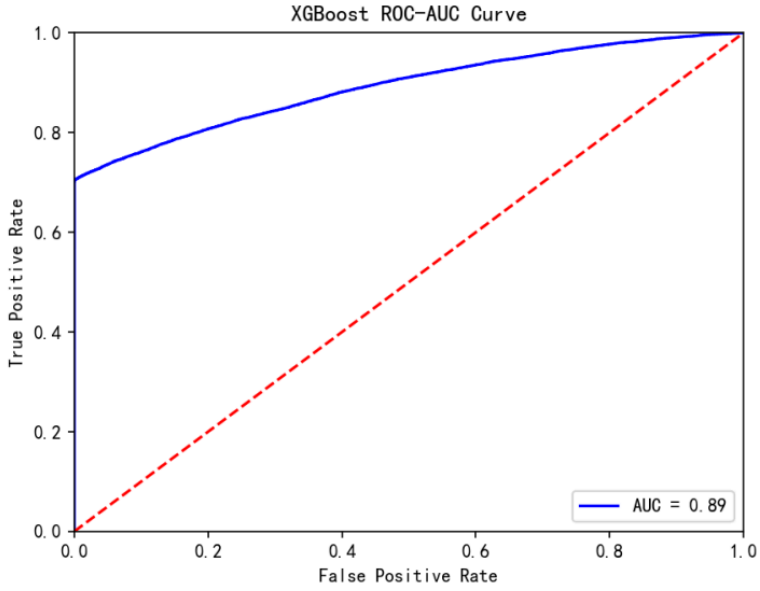


Figure 8. ROC curve of xgboost classification model (picture credit: original)

From Figures 5 to 8, the AUC values of the four models are as follows: the logistic regression model has an AUC of 0.74, the decision tree model achieves an AUC of 0.77, while both the LightGBM and XGBoost models reach a higher AUC of 0.89.

AUC Ranking: LightGBM/XGBoost (0.89) > Decision Tree (0.77) > Logistic Regression (0.74).An AUC >0.85 indicates that ensemble models significantly outperform traditional methods in distinguishing between positive and negative samples.

This study employed logistic regression, decision trees, LightGBM, and XGBoost to build classification models. LightGBM and XGBoost demonstrated the best performance, making them suitable for daily product predictions.

DOI: 10.54254/2753-8818/2025.AD26291

In the realm of dynamic risk management, financial institutions are advised to maintain real-time monitoring of key indicators such as loan-to-value (LTV) ratios and credit scores, as this proactive approach helps track evolving risk profiles effectively [6]. Additionally, the transparency afforded by SHAP values plays a crucial role in supporting decision-making processes, ensuring that risk assessments are explainable and aligned with regulatory standards [7,8].

Despite the model's strengths, the current research faces certain data limitations. Notably, macroeconomic variables—such as fluctuations in interest rates—have not been incorporated into the analysis. The exclusion of these factors may restrict the model's ability to generalize across different economic conditions, as external economic dynamics can significantly influence borrower behavior and default risks [9,10].

Looking ahead, future work will focus on enhancing the model's robustness and adaptability. One key direction involves integrating time-series data to strengthen dynamic prediction capabilities, allowing the model to capture temporal patterns and evolving risk trends more effectively [2]. Another critical area of exploration will be investigating cross-cycle risk transmission mechanisms, which can shed light on how risks propagate across different economic phases and inform more resilient risk management strategies [8].

3.4. Improvements and comparisons

SHAP analysis identified PERFORM_CNS_SCORE (0.32) and LTV (0.28) as the strongest predictors. We recommend screening high-risk customers using thresholds: LTV >85% and PERFORM_CNS_SCORE <500 (Yang & Birge, 2018; Wu et al., 2022).

Consistency: XGBoost's AUC (0.89) outperforms logistic regression (0.74), validating the superiority of ensemble learning [7,11]. This differs from Lessmann et al., who found LightGBM to be superior, possibly due to differences in feature engineering.

4. Conclusion

This study constructed a highly accurate and interpretable loan default prediction model using 100,000 anonymous loan records. Data quality was optimized through feature engineering (e.g., credit history standardization, outlier correction) and SMOTE oversampling. Among XGBoost, LightGBM, and other algorithms, XGBoost achieved the best performance (AUC=0.89), significantly outperforming logistic regression (AUC=0.62). SHAP and LIME methods were used to interpret model decisions, identifying credit score, LTV (>85%), and self-employment status as key risk factors, aligning with business logic.

Future Research Directions: Incorporate macroeconomic variables (e.g., GDP growth, unemployment rate) to enhance adaptation to economic cycles.

Explore time-series analysis for dynamic risk early-warning mechanisms to improve real-time response to market changes.

Adopt federated learning to enable cross-institutional data sharing while protecting privacy.

Combine deep learning with traditional machine learning to uncover deeper risk correlations.

These improvements will enhance prediction accuracy and practical value, providing financial institutions with smarter, dynamic risk management tools.

References

[1] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794.

DOI: 10.54254/2753-8818/2025.AD26291

[2] Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. Advances in Neural Information Processing Systems, 30, 4765–4774.
[3] Lessmann, S., et al. (2015). Benchmarking state -of-the-art classification algorithms for credit scoring. European Journal of Operational Research, 241(2), 427–443.
[4] Wang, Y., Chen, H., & Li, J. (2021). Dynamic credit risk assessment with real-time data streaming. Journal of Banking & Finance, 128, 106–120.
[5] Haixiang, G., et al. (2017). Learning from class-imbalanced data: Review of methods and applications. Expert Systems with Applications, 73, 220–239.
[6] Yang, Y., & Birge, J. R. (2018). Trade credit, risk sharing, and inventory financing portfolios. Management Science, 64(8), 3667–3689.
[7] Yang, Y. X., He, M. H., & Zhang, J. X. (2024). Comparative study of Optuna-optimized LightGBM, XGBoost and their hybrid stacking models in personal loan default prediction. In Proceedings of the 2024 International Conference on Artificial Intelligence and Computer Engineering (pp. 45–56). Springer.
[8] Jiang, C., Wang, Z., & Zhao, H. (2022). Depicting risk profile over time: A novel multiperiod loan default prediction approach. MIS Quarterly, 46(3), 1749–1772.
[9] Wu, J., Zhang, Z., & Zhou, S. X. (2022). Credit rating prediction through supply chains: A machine learning approach. Production and Operations Management, 31(4), 1613–1629.
[10] Bellotti, T., Crook, J., & Zhang, Y. (2021). Credit risk modeling using machine learning techniques. Expert Systems with Applications, 163, 113819.
[11] Zhu, X., Chu, Q., Song, X., et al. (2023). Explainable prediction of loan default based on machine learning models. Data Science and Management, 6(3), 123–133.