Proceedings of ICFTBA 2026 Symposium: Strategic Management & Business Analytics: Data-Driven Performance Models
DOI: 10.54254/2754-1169/2026.LD36396

Random Forest vs. XGBoost for Peer-to-Peer Loan Default Prediction: Evidence from LendingClub

Ziwei Feng

International Business College, South China Normal University, Foshan, China
20253638034@m.scnu.edu.cn

Abstract. This study compares Random Forest (RF) and Extreme Gradient Boosting (XGBoost) for predicting loan defaults on 25,000 LendingClub loans from 2015–2016, using 14 financial features and 5-fold cross-validation. Both models perform similarly: RF reaches an AUC-ROC of 0.717, XGBoost reaches 0.715. RF does better on precision (0.572 vs. 0.504), while XGBoost leads on recall (0.164 vs. 0.116) and F1-score (0.247 vs. 0.193). The gap is small enough that the choice should rest on operational priorities rather than accuracy: Random Forest for interpretability, XGBoost for recall.

Keywords: peer-to-peer lending, default prediction, Random Forest, XGBoost, credit risk

1. Introduction

Peer-to-peer lending has grown rapidly. Platforms such as LendingClub and Prosper now originate billions of dollars in loans annually, operating outside the conventional banking system [1]. Where a traditional bank relies on loan officers, physical branches, and long customer histories, a P2P platform works with a digital application and a credit bureau pull. This structure reduces costs but widens the information asymmetry between borrower and lender [2, 3]. During 2015–2016 alone, LendingClub originated more than $18 billion in loans—a volume that makes the accuracy of automated credit screening a pressing concern for both investors and regulators.

Default prediction is central to the platform's viability. A missed default translates directly into a loss for the investor funding that loan. Enough such losses, and investors withdraw, threatening the platform's existence [4]. Machine learning methods offer advantages over traditional logistic regression because they capture nonlinear relationships that linear models miss [5]. Among these methods, two ensemble tree models—Random Forest and XGBoost—consistently rank among the top performers in credit scoring benchmarks [6, 7]. Random Forest [8] builds a collection of trees, each trained on a different bootstrap sample, and averages their predictions. XGBoost [9] takes a sequential approach: each new tree is trained to correct the errors of the previous ones. Both are effective. Which performs better on real P2P lending data, and by what margin, remains an open question. Existing comparison studies have limitations. Many test a single algorithm [10], use small or synthetic datasets [11], or report only accuracy and AUC-ROC while omitting precision and recall [12]. Because defaults account for less than 20% of loans, accuracy alone is an unreliable metric [12]. Furthermore, the most commonly used benchmarks are European credit datasets [5] or Taiwanese credit card records [13]—markets whose regulatory environments and borrower profiles

DOI: 10.54254/2754-1169/2026.LD36396

differ from U.S. P2P lending. LendingClub's public database contains millions of loans with clean, standardized outcomes, yet it has rarely been used for head-to-head model comparisons that report the full set of metrics relevant to lending decisions.This paper evaluates RF and XGBoost side by side on 25,000 LendingClub loans, reporting AUC-ROC, F1-score, precision, and recall. Feature importance rankings are also compared.

2. Methodology

2.1. Data sources

The data is drawn from LendingClub's public loan database. Starting from 2,260,701 loans spanning 2007 to 2018, we restrict the sample to loans issued between January 2015 and December 2016 whose final status is either "Fully Paid" or "Charged Off." This leaves 668,640 loans. We chose the 2015–2016 window for two reasons. It begins after LendingClub's December 2014 IPO, by which point the company's underwriting had stabilized under public-market scrutiny. It ends well before the COVID-19 pandemic, whose stimulus checks and forbearance programs altered default behavior in ways that would complicate any comparison of model performance. A random sample of 25,000 loans is drawn from this window; the default rate in the sample is 21.5%, in line with LendingClub's own published charge-off figures for the period.

2.2. Data processing

Fourteen continuous financial features are retained, covering loan characteristics (loan amount, interest rate, term), credit history (delinquencies, credit inquiries, open credit lines, total credit lines, revolving balance, revolving utilization, public records), borrower demographics (annual income, employment length, debt-to-income ratio), and risk grade (LendingClub-assigned grade, A=1 to G=7). Home ownership and loan purpose variables were tested but excluded because sparse binary columns degraded performance. Continuous variables are kept in original scale since tree models are invariant to monotonic transformations. Missing values are limited. Employment length (2.1% missing) and revolving utilization (0.3% missing) are median-imputed. Debt-to-income ratio, delinquencies, credit inquiries, and public records have missing rates below 0.1% and are filled with zero. Interest rate strings are converted to numeric values. The 14 features were selected from the original 151 columns by excluding variables requiring proprietary knowledge (e.g., internal sub-grade scores) and those with over 50% missing values or near-zero variance. Among the retained features, interest rate and loan grade are notably collinear ($r \approx 0.85$), as both reflect LendingClub's risk assessment. Tree-based models handle this by selecting the single best feature at each split, treating the correlated pair as substitutes across different trees. High importance for both does not imply additive predictive value, but rather reflects their interchangeable use across bootstrap samples.

2.3. Models

Random Forest: 200 trees, max depth 5, min samples per split 10, max features sqrt. Deeper trees (depth 10–20) reduced AUC in preliminary tests. XGBoost: 200 rounds, learning rate 0.1, max depth 3, subsample 0.8, column subsample 0.8. Depths of 6–12 underperformed depth 3.

Both implemented in Python (scikit-learn 1.3, xgboost 2.0). No extensive tuning was performed. Logistic regression, neural networks, and support vector machines were not included. Prior

DOI: 10.54254/2754-1169/2026.LD36396

benchmarking studies indicate that tree-based ensembles consistently outperform logistic regression on structured credit data [5, 6], while neural networks and SVMs require substantially more tuning and offer marginal gains on datasets of this scale [7].

The two models differ fundamentally in how they handle error. Random Forest reduces variance by averaging uncorrelated trees, each trained on a different bootstrap sample. XGBoost reduces bias by fitting each new tree to the residuals of the ensemble. In credit scoring, where default signals are weak and class imbalance is severe, it is not obvious which strategy dominates. RF's averaging may be more robust to noisy features; XGBoost's sequential correction may better capture subtle interactions between borrower characteristics. This empirical question motivates the comparison.

2.4. Evaluation strategy

Stratified 5-fold cross-validation with four metrics: AUC-ROC, discriminative capacity across thresholds; precision, reliability of default predictions; recall, coverage of actual defaults; F1-score, harmonic mean of precision and recall. The precision-recall trade-off matters in practice. Platforms fearing losses may prioritize recall; those optimizing capital deployment may prefer precision [13].

3. Results

3.1. Model performance comparison

Table 1 reports cross-validation results. The two models perform similarly in discriminative power.

Table 1. Model performance

Model	AUC-ROC	F1	Precision	Recall
Random Forest	0.717	0.193	0.572	0.116
XGBoost	0.715	0.247	0.504	0.164

Note: Means across 5 folds. AUC-ROC SD = 0.005 (RF), 0.002 (XGBoost).

Interest rate alone reaches an AUC of 0.705, indicating that the platform's risk-based pricing already captures a substantial portion of predictable default variance. Remaining features contribute roughly 0.01–0.02 additional AUC. The precision-recall trade-off is clear: RF produces fewer false alarms, XGBoost catches more defaults.

The 0.002 AUC gap is well within one standard deviation. On this dataset, switching algorithms offers no meaningful accuracy gain. Platform choice should rest on operational factors. RF suits environments requiring interpretability: out-of-bag importance scores are stable and easier to communicate to regulators [14]. XGBoost suits environments where catching defaults is paramount —its recall advantage translates to roughly 48 more defaults caught per 1,000, at the cost of more good borrowers incorrectly flagged. The AUC-ROC of approximately 0.72 is lower than the 0.80+ reported in some credit scoring studies [6, 13], but comparable to results obtained specifically on P2P data. Guo et al. [11] reported AUC values of 0.68–0.74 across models on LendingClub data; Byanjankar et al. [10] found neural network AUC of 0.71 on a similar sample. The gap between P2P performance and traditional credit scoring reflects the absence of relationship history and in-person assessment that banks rely on. In P2P lending, models work with thinner information, so moderate AUC values are expected rather than indicative of model weakness. It is worth noting that all metrics in Table 1 use the default 0.5 classification threshold. In practice, P2P platforms can shift

DOI: 10.54254/2754-1169/2026.LD36396

this threshold to favor recall or precision depending on risk appetite, without retraining the model. The reported values thus represent a neutral baseline rather than optimized operating points.

The observed precision-recall trade-off aligns with the structural properties of the two algorithms. Random Forest averages predictions across 200 independent trees, each trained on a bootstrap sample. This averaging suppresses false positives—a tree that overfit to noise in one bootstrap sample is diluted by other trees that did not—leading to higher precision. XGBoost's sequential correction, by contrast, focuses each new tree on misclassified observations from previous rounds. When defaults are rare (21.5% of the sample), these misclassified observations disproportionately include defaulters, progressively improving recall at the cost of also fitting noise around individual default cases, which lowers precision. For platforms, this means RF is preferable when the cost of falsely flagging a good borrower is high; XGBoost is preferable when missing a default is the dominant concern.

3.2. Feature importance analysis

Table 2 shows top-10 features. Both models converge on interest rate, grade, debt-to-income ratio, and loan amount as core predictors.Interest rate shows up first or third in both models. That fits: it is LendingClub's own price of risk, and higher rates are meant to offset higher odds of default. Debt-to-income ratio is the standard affordability check—how much a borrower owes each month relative to what they earn. Loan grade rolls up a bundle of credit factors into a single letter score that the platform itself assigns, so in practice it is as close to a ready-made risk label as the dataset offers. Home ownership and loan purpose appear only in XGBoost's top 10, which hints that these categorical variables contain nonlinear structure that XGBoost's step-by-step fitting picks up but Random Forest's independent trees tend to wash out.

The ranking differences between the two models deserve a closer look. Revolving balance drops from third place in Random Forest to ninth in XGBoost. Term jumps the other way, from tenth in Random Forest to second in XGBoost. These swings come from how each model measures importance. Random Forest scores a feature by how much it reduces Gini impurity at each split, averaged across all trees. A feature that delivers a strong split on its own—even if another feature carries much of the same information—will rank high. XGBoost uses a gain-based measure: how much a feature lowers the loss, times how many observations pass through that split, summed across all trees. On that metric, term shoots up because it keeps chipping away at prediction error across a large share of nodes. Revolving balance slides because interest rate and grade do much of the same work earlier in the sequence, leaving less residual error for revolving balance to explain.

That home ownership and loan purpose appear in XGBoost's top 10 but not Random Forest's points to a known difference between bagging and boosting. Boosting can amplify faint but real signals that bagging's averaging tends to cancel out. A variable like home ownership carries almost no predictive power on its own—renters and homeowners default at broadly similar rates in this sample. But when a boosted model has already accounted for income and loan size, the residual patterns linked to home ownership start to show through. Whether that extra signal holds up in other time periods or is just fitting noise in this particular window is not something these data can settle.

Table 2. Feature importance rankings

Feature	Single-Feature AUC	RF Rank	XGBoost Rank
Interest Rate	0.705	1	1

DOI: 10.54254/2754-1169/2026.LD36396

Table 2. (continued)

Loan Grade	0.695	7	3
Debt-to-Income Ratio	0.578	2	8
Revolving Utilization	0.562	4	-
Term	0.551	10	2
Annual Income	0.538	5	-
Loan Amount	0.531	6	7
Revolving Balance	0.527	3	9
Open Credit Lines	0.518	9	9
Total Credit Lines	0.512	8	-

indicates the feature did not rank in the model's top 10 by gain-based importance.

3.3. Limitations

Several limitations apply. First, the analysis uses a single platform and time window (LendingClub, 2015–2016); generalizability to other markets, regions, or recessionary periods is untested. Second, we did not run an exhaustive hyperparameter search. The depth settings came from a quick scan—testing a handful of values and picking the one that gave the highest AUC. A full grid search might squeeze out another point or two, and it might shift the ranking between the two models. Third, default is treated as a yes/no outcome. In practice, a loan that misses six payments and a loan that misses two are not the same thing. Partial charge-offs matter for estimating how much a platform actually loses, and a binary label lumps them together. Fourth, we dropped categorical variables because their one-hot encodings hurt cross-validation performance. That choice may have been too blunt—alternative encodings like target encoding or embedding layers could retain whatever signal those variables carry without the sparsity that one-hot creates.

4. Conclusion

On 25,000 LendingClub loans and 14 financial features, Random Forest and XGBoost are a dead heat on discrimination: AUC of 0.717 against 0.715. RF wins on precision (0.572 vs. 0.504). XGBoost wins on recall (0.164 vs. 0.116) and F1 (0.247 vs. 0.193). Both models gravitate toward the same three features—interest rate, loan grade, debt-to-income ratio—and interest rate by itself already reaches an AUC of 0.705. The takeaway is that switching from one algorithm to the other will not move the needle on overall accuracy. The choice matters for how the model errs. A platform that wants to minimize false alarms should lean toward Random Forest. A platform that wants to catch as many defaults as possible, even at the cost of flagging some good borrowers along the way, should lean toward XGBoost. Future work needs to test whether these results hold when the economy turns—the 2015–2016 window was a quiet one, and a recession would stress both models in ways these data cannot show. Adding alternative data—bank transaction logs, social network signals—might push performance past the ceiling that interest rate alone sets. Extending the setup to predict not just whether a loan defaults, but how much is lost when it does, would get closer to what a platform actually cares about. And as regulators ask harder questions about model decisions,

DOI: 10.54254/2754-1169/2026.LD36396

running SHAP explanations on both models side by side would tell platforms not just which model works, but which one they can explain.

References

[1] Morse, A. (2015). Peer-to-peer crowdfunding: Information and the potential for disruption in consumer lending. *Annual Review of Financial Economics, 7*, 463-482.
[2] Serrano-Cinca, C., Gutiérrez-Nieto, B., & López-Palacios, L. (2015). Determinants of default in P2P lending. *PLoS ONE, 10*(10), e0139427.
[3] Lin, M., Prabhala, N. R., & Viswanathan, S. (2013). Judging borrowers by the company they keep. *Management Science, 59*(1), 17-35.
[4] Emekter, R., Tu, Y., Jirasakuldech, B., & Lu, M. (2015). Evaluating credit risk and loan performance in online peer-to-peer lending. *Applied Economics, 47*(1), 54-70.
[5] Lessmann, S., Baesens, B., Seow, H. V., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research, 247*(1), 124-136.
[6] Malekipirbazari, M., & Aksakalli, V. (2015). Risk assessment in social lending via random forests. *Expert Systems with Applications, 42*(10), 4621-4631.
[7] Xia, Y., Liu, C., Li, Y., & Liu, N. (2017). A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring. *Expert Systems with Applications, 78*, 225-241.
[8] Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5-32.
[9] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD*, San Francisco, pp. 785-794.
[10] Byanjankar, A., Heikkilä, M., & Mezei, J. (2015). Predicting credit risk in peer-to-peer lending. *IEEE SSCI*, Cape Town, pp. 719-725.
[11] Guo, Y., Zhou, W., Luo, C., Liu, C., & Xiong, H. (2016). Instance-based credit risk assessment for investment decisions in P2P lending. *European Journal of Operational Research, 249*(2), 417-426.
[12] He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE TKDE, 21*(9), 1263-1284.
[13] Tsai, C. F., & Chen, M. L. (2010). Credit rating by hybrid machine learning techniques. *Applied Soft Computing, 10*(2), 374-380.
[14] Bracke, P., Datta, A., Jung, C., & Sen, S. (2019). Machine learning explainability in finance. *Bank of England Working Paper, No. 816*.