Journal of Business Vol. 17, N°1
pp. 93-130
eISSN 2078-9424
doi: 10.21678/jb.2026.2798

 Received on June 8, 2025
Accepted on September 15, 2025

Credit Risk Analysis in Microfinance Using Machine Learning and a Conceptual Proposal for Integrating AI Agents

Diego Arriola León
Pontificia Universidad Católica del Perú
diegoaleon25@gmail.com

Mohsen Ghodrat
University Canada West
mohsen.ghodrat@ucanwest.ca

Abstract. This study presents an integrated approach to early default detection in microfinance by combining machine learning with historical payment-behavior analysis through the analysis of over 50,000 microcredits granted in Huancayo, Peru (2019–2021). The study focuses on microenterprises and informal entrepreneurs as segments with limited credit histories. The target variable is defined as default when accumulated days in arrears D exceed 25% of the contractual term $T=f\times q$ (payment frequency f times the number of installments q), i.e., $D/T>0.25$; operationally, this is equivalent to the average delay per installment D/q exceeding $0.25\times f$. Four families of supervised models were evaluated (GBM, BART, RF, and SVM). The positive class is set to "1," corresponding to default defined by $D/T>0.25$. Consequently, all predictions are interpreted as Pr(Y=1): the probability of default. We chose the decision threshold by maximizing F1 on the validation set and then applying it, without retraining, to the test set (including an out-of-time evaluation). Performance was reported using PR-AUC, ROC-AUC, F1, precision/recall, balanced accuracy, and G-Mean, as well as KS, Brier, and calibration assessment. The results indicate moderate differences across models: BART attains the highest balanced accuracy with a good sensitivity–specificity trade-off, while GBM exhibits consistent performance. RF and SVM are competitive with small gaps. Given the class imbalance, PR-AUC and F1 provide more informative signals than accuracy. Overall, the models enable risk segmentation and operational decision support. Integration with AI agents is proposed as a conceptual, future framework and was not implemented in this study.

Keywords: Microfinance, credit risk, payment default, small businesses, machine learning, predictive modeling, artificial intelligence agents.

Análisis del riesgo crediticio en microfinanzas mediante aprendizaje automático y una propuesta conceptual para la integración de agentes de IA

Resumen. Este estudio presenta un enfoque integrado para la detección temprana del incumplimiento en microfinanzas, que combina aprendizaje automático con el análisis del comportamiento histórico de pagos, a partir del estudio de más de 50 000 microcréditos otorgados en Huancayo, Perú, durante el período 2019–2021. El análisis se centra en microempresas y emprendedores informales, segmentos caracterizados por historiales crediticios limitados. La variable objetivo se define como incumplimiento cuando los días acumulados de atraso D superan el 25 % del plazo contractual $T = f \times q$ (frecuencia de pago f multiplicada por el número de cuotas q), es decir, cuando $D/T > 0{,}25$; operacionalmente, esto equivale a que el atraso promedio por cuota D/q exceda $0{,}25 \times f$. Se evaluaron cuatro familias de modelos supervisados (GBM, BART, RF y SVM). La clase positiva se definió como "1", correspondiente al incumplimiento según el criterio $D/T > 0{,}25$. En consecuencia, todas las predicciones se interpretan como Pr(Y = 1), es decir, la probabilidad de incumplimiento. El umbral de decisión se seleccionó maximizando la métrica F1 en el conjunto de validación y luego se aplicó, sin reentrenamiento, al conjunto de prueba (incluida una evaluación fuera de muestra temporal). El desempeño se reportó mediante PR-AUC, ROC-AUC, F1, precisión/recall, exactitud balanceada y G-Mean, así como los estadísticos KS, Brier y evaluaciones de calibración. Los resultados muestran diferencias moderadas entre los modelos: BART alcanza la mayor exactitud balanceada, con un buen equilibrio entre sensibilidad y especificidad, mientras que GBM presenta un desempeño consistente. RF y SVM resultan competitivos, con brechas pequeñas respecto a los modelos líderes. Dado el desbalance de clases, las métricas PR-AUC y F1 proporcionan señales más informativas que la exactitud simple. En conjunto, los modelos permiten la segmentación del riesgo y el apoyo a la toma de decisiones operativas. Finalmente, se propone la integración con agentes de inteligencia artificial como un marco conceptual futuro, el cual no fue implementado en este estudio.

Palabras clave: Microfinanzas; riesgo crediticio; incumplimiento de pago; pequeñas empresas; aprendizaje automático; modelamiento predictivo; agentes de inteligencia artificial.

Introduction

In the Peruvian context, microfinance institutions (MFIs) play a fundamental role in expanding access to financial services for small entrepreneurs and micro-scale businesses—such as convenience stores, market stalls, and family-run enterprises. This segment, largely composed of clients without formal credit histories or access to traditional banking, poses a particular challenge for credit risk assessment due to limited structured information and the absence of collateral.

Early identification of potential payment delays is crucial to preserving the health of microcredit portfolios and reducing losses from defaults. This study addresses that need using historical data from an MFI located in Huancayo, Peru, with over 50,000 microcredit operations recorded between 2019 and 2021. We defined a risk indicator as the ratio of accumulated days in arrears to the contractual loan term. Borrowers are flagged as in default when this ratio exceeds 25%—that is, when $D/T>0.25$, where D is total days in arrears and $T=f\times q$ is the contractual term (payment frequency in days f times number of installments q). This threshold separates minor, short-lived delays from more persistent delinquency, thereby improving analytical accuracy.

Based on this database, we developed and compare four machine-learning models—gradient boosting machine (GBM), Bayesian additive regression trees (BART), random forest (RF), and support vector machines (SVM)—to predict the probability of default at loan approval. We selected these algorithms to represent major supervised-learning families: boosting (GBM), Bayesian ensembles (BART), bagging (RF), and maximum-margin classifiers (SVM), which allowed us to compare methods with different learning principles and identify those most suitable for the microfinance context.

We evaluated model performance using a comprehensive set of robust classification and calibration metrics, including accuracy, Cohen's kappa, sensitivity, specificity, precision, recall, F1-score, geometric mean (GM), balanced accuracy, and McNemar's test, thus accounting for the natural class imbalance in the target variable. In addition, we assessed discrimination through ROC–AUC and precision–recall AUC curves, while the Kolmogorov–Smirnov (KS) statistic, Brier score, Brier skill score, and calibration plots provided further insights into the probability estimates. A distinctive feature of this study is its focus on clients from microenterprises and small-scale businesses that are typically underserved by larger financial institutions, providing evidence to inform predictive tools tailored to these portfolios.

The following sections present the conceptual framework and methodology, describe the database, detail the models, discuss the results, and outline future research directions, including a conceptual (and as yet unimplemented) framework for integrating AI agents to enhance monitoring and decision-making in credit assessment.

Theoretical Framework

Credit risk in microfinance refers to the probability that a borrower will partially or fully fail to meet contractual payment obligations, causing delays or losses for the institution. This challenge is especially acute where clients are small informal entrepreneurs—corner stores, market stalls, and family-run businesses—who lack formal credit histories or collateral. Microfinance institutions (MFIs) therefore need predictive tools that enable early identification of borrowers with high default propensity, optimizing resource allocation and supporting financial inclusion.

Operational Definition of Delinquency (Single Criterion Used Throughout)

Let D denote the total days in arrears accumulated by the borrower and $T = f \times q$ the contractual term (days), where f is the installment frequency in days (daily = 1, weekly = 7, bi-weekly = 14, monthly =30) and q the number of installments. We define the binary outcome $Y \in \{0,1\}$ as

$$Y = \begin{cases} 1\ (\textit{delinquent})\ \textit{if}\ D/T > 0.25, \\ 0\ (\textit{non-delinquent}) \end{cases} \qquad (1)$$

Equivalently, the average delay per installment D/q exceeds $0.25f$.

Numerical example. For a monthly loan ($f = 30$) with $q = 12 \Rightarrow T = 360$, if $D = 95$ then $D/T = 95/360 = 0.264 > 0.25 \Rightarrow Y = 1$.

Positive class and probabilities: The positive class is Y=1 (delinquency), and all models estimate Pr (Y=1).

In microfinance practice, exceeding one quarter of the contractual term is considered a material delinquency event, separating short-lived delays from persistent arrears. Normalizing by T ensures comparability across daily, weekly, and monthly schedules and prevents the penalization of temporary timing mismatches. As a robustness check, a sensitivity sweep over $20\% \leq D/T \leq 30\%$ preserved the relative ranking of models and did not alter substantive conclusions, supporting the choice of 25% as an operational and comparable rule.

Why Machine Learning for Credit Risk?

Compared with classical parametric approaches (e.g., logistic regression), modern ML methods can capture non-linearities and high-order interactions without explicit specification (Hastie, Tibshirani, & Friedman, 2009). They can also handle heterogeneous predictors and complex dependencies typical of credit-risk data (Khandani, Kim, & Lo, 2010) and often deliver stronger performance under class imbalance when paired with appropriate metrics and validation (He & Garcia, 2009; Sokolova & Lapalme, 2009). Large-scale benchmarking in credit scoring shows that tree ensembles and SVMs are competitive with, or superior to, logistic regression across many datasets (Lessmann, Baesens, Seow, & Thomas, 2015; Brown & Mues, 2012). We therefore compared four representative ML families while holding the predictor set constant across models to isolate algorithmic effects.

Overview of Modeling Choices

We employed four representative supervised-learning families—SVM (maximum-margin), RF (bagging of trees), GBM (boosted trees), and BART (Bayesian tree ensemble)—to cover complementary inductive biases (margins, variance reduction, sequential error correction, and probabilistic inference). All models re trained on the same predictor set produced by the unified preprocessing pipeline and output Pr(Y=1) (delinquency). Hyperparameters follow standard defaults with modest tuning, and the decision threshold was selected on the validation set by maximizing F1, then applied unchanged to the test sets. This common setup allowed us to attribute performance differences to the algorithms rather than to features or thresholds.

Support Vector Machines (SVM)

The SVM model was intended to find an optimal hyperplane that separates the default and non-default classes by maximizing the margin between them.

Its objective is to determine the parameters (w and b) that satisfy:

$$\min_{w,b} \frac{1}{2}||w||^2, \text{ subject to: } y_i(w \cdot x_i + b) \geq 1 \text{ for all } i \in \{1, 2, ..., N\} \quad (2)$$

Where:

- i: The index of the training simples.
- $w \in \mathbb{R}^d$: The weight vector defining the separating hyperplane.
- $b \in \mathbb{R}$: The bias term of the hyperplane.
- $y_i \in Y$: The class label for the i-th training sample. In binary classification $Y = \{-1, +1\}$.

When classes are not linearly separable SVM employs kernel functions, such as the radial basis function (RBF), which projects the data into a higher-dimensional space to enable non-linear separation (Huang, Chen, & Wang, 2007).

Random Forest (RF)

RF is an ensemble method based on the construction of multiple decision trees trained on subsets of data and randomly selected features (Malekipirbazari & Aksakalli, 2015).

The final prediction for a client was obtained through majority voting among all trees in the forest:

$$H(x) = \text{mode}\{h_b(x)\}, \quad \text{for } b = \{1,2, ..., B\} \qquad (3)$$

Where:

- $h_b(x)$: The predicted class label for input x by the b-th decision tree in the ensemble.
- B: Total number of decision trees in the random forest.

This strategy reduces variance and the risk of overfitting, which proves effective in datasets with noise and high dimensionality.

Gradient Boosting Machine (GBM)

The GBM (Friedman, 2001; Chen & Guestrin, 2016) builds decision trees sequentially, each correcting the residual errors of the previous ones. The prediction function is updated as:

$$F_m(x) = F_{m-1}(x) + \nu \cdot h_m(x) \qquad (4)$$

where:

- $F_m(x)$: The accumulated model at iteration .
- ν: The learning rate.
- $h_m(x)$: The decision tree fitted at iteration .

This method minimizes a loss function (e.g., log-loss) and has been shown to perform well in credit scoring due to its ability to model complex interactions among variables.

Bayesian Additive Regression Trees (BART)

BART (Rinaldo et al., 2018) is a non-parametric Bayesian approach that represents the prediction function as a sum of multiple shallow regression trees:

$$y = g_1(x) + g_2(x) + ... + g_m(x) + \varepsilon \qquad (5)$$

where:

- $g_j(x)$: The prediction given by the j-th regression tree (partial contribution of tree j).
- m: Total number of regression trees in the ensemble.
- ε: Random error term capturing residual noise not explained by the trees.

Each tree contributes partially to the final outcome, and the model uses Markov Chain Monte Carlo (MCMC) methods to sample from the posterior space and estimate predictive distributions. Unlike GBM and RF, BART incorporates uncertainty into predictions, providing probabilistic intervals that are useful for financial decision-making.

Data Preparation and Train/Validation/Test Split

To prevent overfitting and assess the generalization capability of the models, we divided the dataset into a 70/15/15 stratified split (train/validation/test)—maintaining the class proportions—as well as an out-of-time test slice (Nhung & Simoni, 2021). Preprocessing followed an R recipes pipeline fitted only on the training set and applied to the validation/test. This included removal of identifiers, elimination of zero-variance predictors, one-hot encoding of categorical variables, and standardization of numeric predictors.

- Data quality checks. Observations with implausible or erroneous values (e.g., negative loan amounts, impossible dates, corrupted records) were removed prior to modeling.
- Missing values. Numerical variables were imputed using the median, while categorical variables were imputed with the mode. This conservative approach ensured that imputation did not distort variable distributions.
- Outliers. Extreme outliers were winsorized at the 1st and 99th percentiles to limit their impact on model estimation. This procedure allowed us to preserve the overall distribution while reducing the undue influence of extreme values.
- Variable preparation. Categorical predictors were encoded using one-hot schemes, numerical variables were standardized, and highly redundant or low-variability predictors were excluded to improve model parsimony.

Decision threshold. Models output probabilities of default P(Y=1). We selected the cut-off on the validation set by maximizing the F1-score, and then applied it without retraining to the test sets (including the out-of-time evaluation).

Cut-off Point Optimization

The models produced probabilities Pr (Y=1) rather than hard labels. Instead of assuming a fixed 0.5 cut-off, we selected the threshold on the validation set by maximizing the F1-score (harmonic mean of precision and recall), which is appropriate under class imbalance. Specifically, for each model we swept a fine grid of thresholds in [0.001, 0.999], compute precision, recall, sensitivity, specificity and F1 on validation, and choose the threshold with the highest F1. We then applied the selected cut-off unchanged (no retraining) to the test set, including the out-of-time slice.

For reporting, threshold-independent metrics (ROC-AUC, PR-AUC) were computed on test using the raw probabilities, while threshold-dependent metrics (F1, precision/recall, balanced accuracy, G-Mean, and Cohen's κ) were computed on test at the chosen validation cut-off. This procedure prioritizes the identification of high-risk cases while controlling for the trade-off between false positives and false negatives in a manner consistent with model selection.

Model Evaluation Indicators

Evaluating the performance of classification models in credit-risk problems requires metrics that go beyond mere accuracy, especially when the classes are imbalanced (i.e., when delinquency cases are far fewer than non-delinquency cases). In this study, we used three indicators widely recommended in the specialized literature—Cohen's kappa, geometric mean (GM), and F1-score (Lessmann et al., 2015; Malekipirbazari & Aksakalli, 2015; Huang et al., 2007). In addition, we reported threshold-independent metrics—ROC-AUC and PR-AUC—to assess ranking quality, as well as threshold-dependent metrics computed at the validation-selected cut-off: precision/recall (and F1), balanced accuracy, G-Mean, and Cohen's κ. For diagnostic analysis we also included the Kolmogorov–Smirnov (KS) statistic, calibration curves, and the Brier score.

Precision (PPV)

Precision measures the proportion of predicted positives that are truly positive, emphasizing the cost of false alarms in operational settings.

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

where:

- TP = true positives.
- FP = false positives.

Specificity (TNR)

Specificity measures the proportion of actual negatives correctly identified, reflecting control over false positives.

$$Specificity = \frac{TN}{TN+FP} \quad (7)$$

where:

- **TN** = true negatives.
- **FP** = false positives.

Balanced Accuracy

Balanced accuracy averages sensitivity and specificity, mitigating the bias of plain accuracy under class imbalance.

$$Balanced\ Accuracy = \frac{Sensitivity+Specificity}{2} \quad (8)$$

where:

- **Sensitivity** =TP / (TP+FN)
- **Specificity** =TN/(TN+FP)

Cohen's Kappa (κ)

The Kappa metric evaluates the degree of agreement between the model's predictions and the actual observations, with the result adjusted to account for agreement occurring by chance (Cohen, 1960; Viera & Garrett, 2005).

The formula is expressed as:

$$\kappa = (p_0 - p_e) / (1 - p_e) \quad (9)$$

where:

- p_0: is the observed agreement proportion and is defined as:

$$p_0 = (TP + TN) / (TP + TN + FP + FN) \quad (10)$$

- p_e: is the expected agreement proportion by chance and is defined as:

$$p_e = [(TP + FP)(TP + FN) + (FN + TN)(FP + TN)] / (TP + TN + FP + FN)^2 \quad (11)$$

The values of κ close to 1 indicate strong agreement between predictions and observations, while values close to 0 suggest performance similar to random chance. Moreover, we used the components of the confusion matrix TP (true positives), TN (true negatives), FP (false positives), and FN (false

negatives) to evaluate the performance of classification models with the following definitions:

TP is the number of cases correctly classified as positive by the model: e.g., clients with late payments correctly identified as delinquent. TN is the number of cases correctly classified as negative by the model: e.g., clients without late payments correctly identified as non-delinquent. FP is the number of cases incorrectly classified as positive when they are actually negative: e.g., clients without late payments that the model incorrectly classifies as delinquent. FN is the number of cases incorrectly classified as negative when they are actually positive: e.g., clients with late payments that the model incorrectly classifies as "non-delinquent."

Geometric Mean (GM)

The Geometric Mean is used to evaluate the balance between sensitivity and specificity of a model, and is particularly useful in problems where the classes are imbalanced (Sokolova & Lapalme, 2009).

$$\mathrm{GM} = \sqrt{(\,\mathrm{Sensitivity} \times \mathrm{Specificity})} \qquad (12)$$

Where:

- Sensitivity (Recall) = TP / (TP + FN)
- Specificity = TN / (TN + FP)

A high GM value indicates that the model can correctly detect both delinquent clients (positives) and non-delinquent clients (negatives), thus avoiding bias toward the majority class.

F1-score

The F1-score is the harmonic mean between precision and recall, making it a key metric for evaluating the balance between false positives and false negatives (Powers, 2011).

$$\mathrm{F1} = 2 \times (\mathrm{Precision} \times \mathrm{Recall}) \,/\, (\mathrm{Precision} + \mathrm{Recall}) \qquad (13)$$

where:

- Precision = TP / (TP + FP)
- Recall (Sensitivity) = TP / (TP + FN)

This indicator is particularly relevant in credit risk contexts, where misclassifying a client as "non-delinquent" when in reality they present a high probability of default can lead to significant financial losses for the institution.

ROC-AUC

The ROC-AUC is the area under the receiver operating characteristic curve, which plots true positive rate versus false positive rate across all thresholds; it summarizes ranking quality independently of any single cutoff.

$$ROC - AUC = \int_0^1 TPR(FPR)\, d(FPR),\ FPR = \frac{FP}{FP+TN} \quad (14)$$

where:

- **TPR** = Sensitivity.
- **FPR** = false positive rate.

PR-AUC

The PR-AUC is the area under the precision–recall curve across thresholds; it is often more informative than ROC-AUC with imbalanced classes because it focuses on the quality among predicted positives.

$$PR - AUC = \int_0^1 Precision(Recall)\, d(Recall) \quad (15)$$

where:

- **Precision** =TP/(TP+FP).
- **Recall** =TP/(TP+FN).

Kolmogorov–Smirnov (KS)

KS quantifies the maximum separation between the score distributions of positives and negatives; higher values denote better global discrimination.

$$KS = \max_t |F_1(t) - F_0(t)| \quad (16)$$

where:

- $F_1(t)$ = CDF of predicted scores for $Y = 1$.
- $F_0(t)$ = CDF of predicted scores for $Y = 0$.

Brier Score

The Brier score is the mean squared error of predicted probabilities and combines calibration and discrimination. The lower the values, the better.

$$Brier = \frac{1}{n}\sum_{i=1}^{n}(p_i - y_i)^2 \quad (17)$$

where:

- p_i = predicted probability of delinquency for observation iii.
- $y_i \in \{0,1\}$ = observed outcome (1 = delinquent).
- n = number of observations.

Calibration Curves

Calibration curves compare, by probability bins (e.g., deciles), the mean predicted probability to the observed event rate. An ideally calibrated model lies on the 45° line.

$$\text{For bin } b\text{: } \bar{p}_b \text{ vs. } \bar{y}_b \qquad (18)$$

where:

- $\bar{p}_b$ = average predicted probability in bin b.
- $\bar{y}_b$ = observed delinquency rate in bin b.

Transparency, Scope, and Limitations

- **Environment and versions**. We ran the analyses in the R software environment (version 4.5.1) with the kernlab, ranger, gbm, bartMachine, recipes, caret, pROC, and PRROC packages
- **Data split.** 70/15/15 stratified split (train/valid/test) preserving class proportions, with an out-of-time slice (Huancayo, 2019–2021).
- **Target and positive class.** Binary outcome with positive class = "1" (delinquency); all models output Pr(Y=1).
- **Preprocessing (leakage-safe).** Recipes fitted only on train and applied to validation/test: removal of identifiers, step_zv (zero-variance predictors), one-hot encoding of categoricals, and standardization of numeric predictors.
- **Data quality.** Records with implausible/erroneous values (e.g., negative amounts, impossible dates, corrupted entries) were removed; no winsorization or outlier clipping was applied.
- **Threshold policy.** The decision threshold was chosen on validation by maximizing F1 over a fine grid [0.001,0.999] and applied without retraining to the test sets (including the out-of-time slice).
- **Models and key settings.**
  - ✓ SVM (kernlab): type="C-svc," kernel="rbfdot" C=1, prob.model=TRUE.
  - ✓ Random Forest (ranger): num.trees=500, mtry≈√p, min.node.size=5, probability=TRUE, importance="impurity".
  - ✓ GBM (gbm): n.trees=100, interaction.depth=3, shrinkage=0.05, n.minobsinnode=10, cv.folds=5, distribution="bernoulli".
  - ✓ BART (bartMachine): num_trees=50, use_missing_data=FALSE, serialize=FALSE, mem_cache_for_speed=FALSE.
- **Seeds / reproducibility.** Random seeds fixed for splitting and training (e.g., set.seed(42) for the split; set.seed(1234) for model training).

- **Scope and limits.** Findings come from a single MFI; external validity may be limited and metric levels are moderate under class imbalance. Results are useful but not definitive.

Agentic Artificial Intelligence Framework in Microfinance Risk Analysis

The implementation of agentic workflows in microfinance risk analysis represents a paradigm shift from static, periodic assessments to dynamic, continuous monitoring systems. Unlike traditional machine learning approaches that rely on batch processing and predetermined intervals, the proposed agentic framework enables real-time adaptation to the volatile conditions characteristic of microenterprise environments. This continuous processing capability is particularly crucial for informal businesses that experience rapid changes in cash flow, seasonal variations, and market disruptions that traditional scoring models often fail to capture promptly. The proposed agentic AI framework consists of six specialized agents that operate collaboratively to enhance microfinance risk assessment and decision-making processes. Each agent performs specific functions while contributing to the overall objective of improving credit risk management in underserved markets.

The data collection and preprocessing agent (DCP Agent) serves as the foundation of the framework by continuously gathering, validating, and preparing multi-source data for risk analysis. This agent autonomously collects information from payment systems, external credit bureaus, and macroeconomic indicators, addressing the challenge of limited formal credit history in microenterprise populations. The agent implements real-time data ingestion protocols, performs comprehensive data validation and anomaly detection, conducts feature engineering specific to informal business characteristics, and integrates alternative data sources such as utility payment records. By ensuring continuous, high-quality data flow that captures the dynamic nature of microenterprise operations, this agent enables more accurate risk predictions and supports the framework's ability to serve populations traditionally excluded from formal banking systems.

The market intelligence and context agent (MIC Agent) operates in parallel with the DCP Agent to monitor external factors that significantly impact borrower risk profiles and overall market conditions. This agent tracks macroeconomic indicators, local market dynamics, and other contextual variables that influence microenterprise performance and default risk. The agent performs continuous economic indicator monitoring and trend analysis, assesses local market conditions, and monitors regulatory changes that may affect borrowing populations. Through this comprehensive

environmental monitoring, the agent provides contextual understanding that enhances risk models by incorporating external factors often overlooked in traditional credit scoring approaches, thereby improving the accuracy of risk assessments in volatile emerging market environments.

The risk assessment and scoring agent (RAS Agent) represents the core analytical component of the framework, dynamically evaluating and updating credit risk scores using the ensemble machine learning models identified in this study. This agent maintains and operates the trained ML models including the GBM, BART, RF, and SVM, continuously updating risk scores as new data becomes available from other agents. The agent implements model versioning and A/B testing protocols to ensure optimal performance, conducts real-time risk score calculation and updates, monitors model performance and detects concept drift, manages ensemble model weights and optimization, and adjusts decision thresholds based on portfolio performance metrics. The agent's contribution to the overall framework lies in providing dynamic, accurate risk assessments that adapt to changing borrower circumstances and market conditions, representing a significant improvement over static scoring approaches traditionally used in microfinance.

The payment monitoring and early warning agent (PMW Agent) specializes in tracking payment behaviors and identifying subtle patterns that precede default events through advanced pattern recognition techniques. This agent continuously monitors payment streams, analyzing payment timing, amounts, frequency variations, and correlations with external economic factors. The agent performs real-time payment behavior analysis, generates early warning signals based on sophisticated payment pattern recognition, identifies seasonal and cyclical patterns specific to microenterprise operations, and integrates payment data with external economic indicators to improve predictive accuracy. By enabling proactive intervention before defaults occur, this agent reduces potential losses while supporting borrower financial health through early identification of distress signals, contributing significantly to the framework's preventive approach to risk management.

The decision support and recommendation agent (DSR Agent) serves as the integration point between complex machine learning outputs and practical business decisions by generating actionable recommendations for loan officers and automated decision-making systems. This agent synthesizes outputs from all other agents to provide clear, explainable recommendations for credit decisions and intervention strategies while considering institutional business rules, regulatory constraints, and policy requirements. The agent generates credit approval and rejection recommendations with associated

confidence intervals, recommends intervention strategies including loan restructuring and additional borrower support, and produces explainable AI outputs to ensure regulatory compliance and institutional transparency. Through this comprehensive decision support capability, the agent bridges the gap between complex ML outputs and practical business applications, ensuring appropriate human oversight while improving decision consistency and processing speed.

The client engagement and support agent (CES Agent) enhances borrower engagement through proactive, personalized communication strategies based on individual risk profiles and payment behaviors. This agent focuses on maintaining adaptive communication with borrowers by leveraging behavioral insights and machine learning outputs to deliver targeted reminders, financial guidance, and support messages through multiple channels. The agent implements automated, behavior-triggered communication protocols such as pre-due date reminders and post-missed payment alerts and provides personalized financial guidance tailored to client risk tiers. This agent directly supports the dual mission of microfinance institutions by improving repayment rates through timely client-friendly interventions, reducing default rates via increased borrower awareness and support, and enhancing trust and transparency between lenders and borrowers, which is particularly important in informal economic contexts where traditional banking relationships may be limited or non-existent.

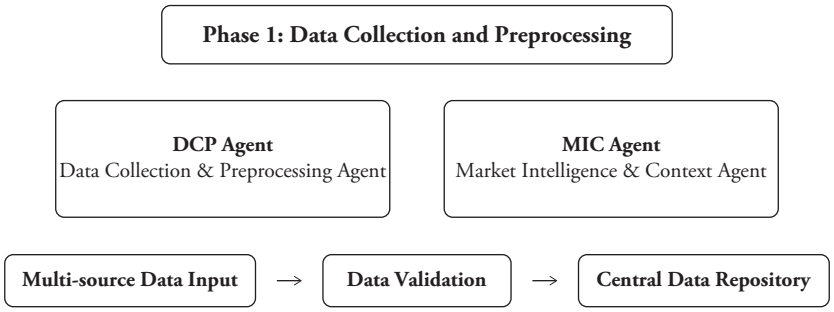
Agent Interaction Workflow

The agent interaction workflow operates through three distinct phases that ensure continuous, coordinated processing of microfinance risk assessment tasks. This systematic approach enables real-time risk evaluation while maintaining the adaptability necessary for dynamic microenterprise environments.

Phase 1. Continuous data processing initiates the workflow through parallel operation of the DCP Agent and the MIC Agent. The DCP Agent continuously ingests data from multiple sources including payment systems, external credit bureaus, and macroeconomic indicators, implementing comprehensive data validation protocols and quality assessment procedures. Simultaneously, the MIC Agent monitors external contextual factors such as macroeconomic indicators, local market conditions, and regulatory changes that may impact borrower risk profiles. Both agents feed their processed outputs into a centralized data repository equipped with quality indicators and metadata tags that facilitate subsequent processing phases. This parallel processing architecture ensures that both internal borrower-specific data

and external market intelligence are continuously updated and available for risk assessment procedures.

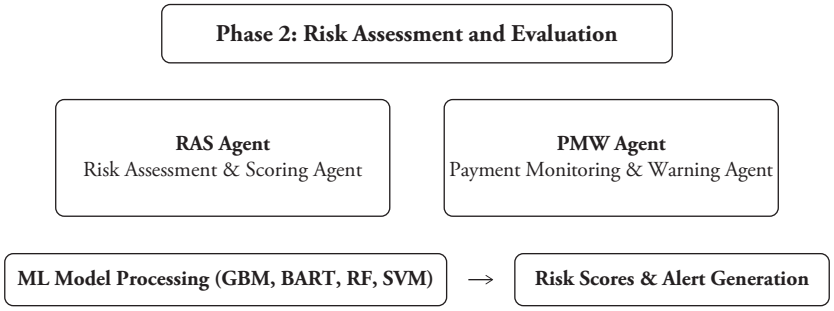
Figure 1
Multi-source Data Ingestion and Preprocessing Workflow



Source: Compiled by authors.

Phase 2. Risk assessment and evaluation leverages the validated data from Phase 1 through coordinated operation of RAS Agent and the PMW Agent. The RAS Agent processes new and updated data through the ensemble machine learning models, including the GBM, BART, RF, and SVM, generating dynamic risk scores and confidence intervals for individual borrowers. Concurrently, the PMW Agent analyzes payment patterns and behavioral indicators to identify early warning signals that may precede default events, employing pattern recognition algorithms to detect changes in payment timing, frequency, and amounts. The outputs from both agents are synthesized to produce comprehensive risk assessments that combine predictive modeling results with behavioral analysis, generating risk scores and alert notifications that inform subsequent decision-making processes.

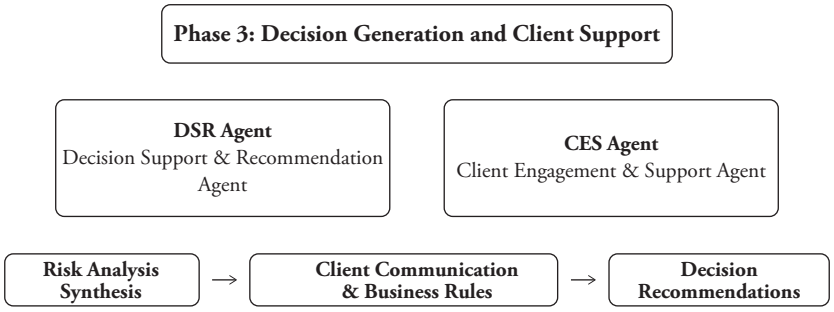
Figure 2
Risk Scoring Through Ensemble Models and Payment Monitoring



Source: Compiled by authors.

Phase 3. Decision generation and client support represents the framework's operational decision-making phase, coordinating two agents to generate actionable recommendations and implement client engagement strategies. The DSR Agent synthesizes risk assessments from Phase 2 with institutional business rules, regulatory requirements, and credit decision objectives to generate credit approval and rejection recommendations with confidence intervals and intervention strategies. Simultaneously, the CES Agent initiates personalized communication protocols based on individual risk profiles and payment behaviors, implementing behavior-triggered messaging, financial guidance delivery, and repayment support strategies through multiple communication channels. Human loan officers or automated decision systems then act based on the integrated outputs from risk assessment and client engagement activities, ensuring appropriate oversight while leveraging the framework's analytical capabilities.

Figure 3
Risk-based Recommendations and Personalized Client Communication



Source: Compiled by authors.

Methodology

The following section details the steps taken in the development of this study, describing the methodological process applied for the construction, evaluation, and validation of the predictive models. This procedure encompassed the initial data exploration and identification of relevant variables, followed by preprocessing, algorithm training, parameter optimization, and final performance evaluation, with the objective of selecting the most suitable model for predicting delays in microcredit payments.

Variable Description

The dataset analyzed contains approximately 50,000 records of microcredits granted by a financial institution located in the province of Huancayo, Peru,

during the period 2019–2021. Each record corresponds to a loan granted to a client, primarily small businesses and self-employed ventures (such as corner stores, neighborhood shops, and low-capital enterprises), allowing the identification of repayment behavior patterns in a segment generally underserved by large-scale financial institutions.

Table 1 outlines the original set of variables provided by the financial institution, which include the sociodemographic, contractual, and financial characteristics of both clients and loans. These variables are classified as "numerical," "categorical," and "date." The subsequent parts of the methodology section describe the criteria applied for data cleaning, transformation, and exclusion of certain variables in order to build the final set of predictors used in the models.

In particular, the dependent variable is defined as a binary indicator of **delinquency** (Delay), constructed from the ratio of the borrower's accumulated overdue days to the contractual term of the loan. Formally:

$$Delayi = \begin{cases} 1\ , if \frac{D_i}{T_i} > 0.25 \\ 0\ , if \frac{D_i}{T_i} \leq 0.25 \end{cases} \qquad (19)$$

where D_i represents the total overdue days of client i, and T_i is the contractual term in days, calculated as the number of installments multiplied by the repayment frequency. For example, a loan with 12 monthly installments (T= 360 days) and 120 days in arrears would be classified as "Delay" since 120/360=0.33>0.

Table 1
Description of Variables Used in the Study

Variable	Tipo	Description
Item	Numeric	Unique identifier of the record
Address	Categorical	Area or zone of residence of the client
Disbursed Capital	Numeric	Amount of the loan granted to the client
Disbursement Date	Date	Exact date on which the loan was disbursed
Disbursement Month	Numeric	Month extracted from the disbursement date (1–12)
Number of Installments	Numeric	Total number of installments agreed for loan repayment
Payment Frequency	Categorical	Loan repayment periodicity (daily, weekly, biweekly, monthly)
Term	Numeric	Duration of the loan in months (1 daily, 7 weekly, 14 biweekly, 30 monthly)
Credit Type	Categorical	Classification of the loan (consumer, mortgage, SME, etc.).

Product	Categorical	Specific type of financial product contracted
Credit Score	Categorical	Client's credit rating (Excellent, Very Good, Good, Fair, Low, or No Score)
Financial Information	Categorical	Whether detailed financial information is available (Yes/No)
Guarantor	Categorical	Whether the loan has a guarantor (Yes/No)
Average Delay	Numeric	Average number of days of delayed payment per installment
Total Delay	Numeric	Total number of overdue days accumulated across all installments
Outstanding Capital Balance	Numeric	Amount of capital pending repayment at the time of analysis
Delinquency (Target Variable)	Categorical (binary)	Indicates whether a client is considered in delay (1 = delinquent, 0 = non-delinquent). Defined as delinquency when cumulative days in arrears exceed 25% of the contractual term

Source: Compiled by authors.

Exploratory Analysis and Variable Selection

Before building the predictive models, we conducted an exploratory data analysis (EDA) with the objective of evaluating the distribution of the variables and their relationship with the dependent variable (Delay), and identifying the main factors contributing to credit risk. This process included:

- Reviewing and handling missing values and extreme outliers. Missing values were addressed through simple imputation (median for numerical variables and mode for categorical variables), while extreme outliers were winsorized at the 1st and 99th percentiles to reduce their undue influence.
- Assessing the distribution of numerical variables (disbursed capital, term, days in arrears, outstanding balance).
- Analyzing the frequencies of categorical variables (type of credit, credit score, presence of guarantor).
- Constructing correlation matrices and conducting independence tests to identify redundancy and relevance among predictors.

The initial analysis revealed that certain predictors exhibited very low variability (for example, administrative products with few records) or high redundancy relative to others. These variables were excluded from the final modeling stage to improve parsimony.

The relative importance analysis showed that payment frequency (45.1%), number of installments (22.5%), and the "no score" category in the credit rating (20.7%) were the most influential predictors of delinquency,

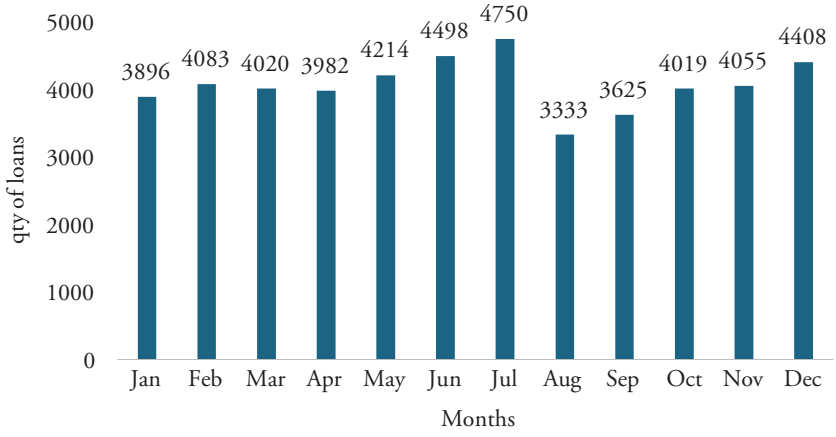
highlighting the relevance of credit history availability as a key risk factor. A complete ranking of variable importance is reported in Table X to ensure transparency regarding the relative contribution of each predictor.

Furthermore, we observed temporal trends related to loan placements and amounts disbursed. For example, the number of loans granted peaked in July (4,750 loans) and then declined significantly in August and September, possibly reflecting seasonal or market-related factors. The total disbursed amount followed a similar pattern, reaching S/ 3,477 thousand in July, before stabilizing at lower levels toward the end of the year. These patterns suggest the existence of credit placement cycles that may influence delinquency behavior.

We discarded the variables without a significant predictive contribution as well as those presenting multicollinearity. During this stage we used the target variable of delinquency (Delay), formally defined in the previous section, as the reference criterion for model training and validation.

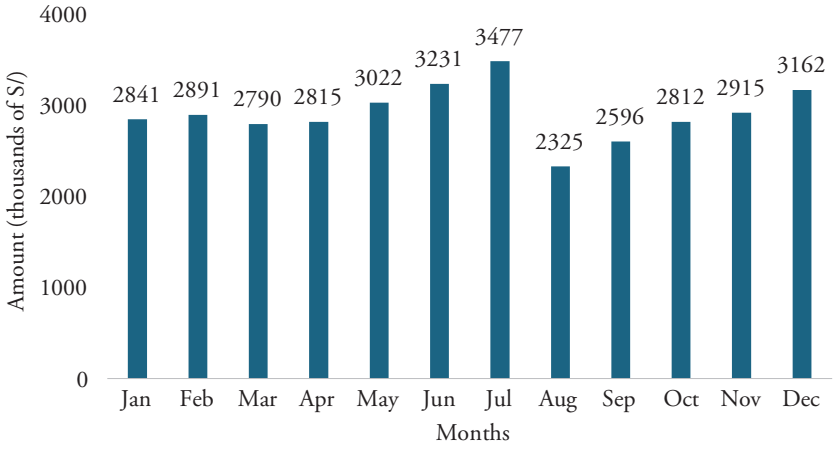
By the end of the cleaning and selection process, we obtained a reduced set of predictor variables that we used consistently across the four models evaluated (BART, GBM, RF, and SVM). This set, presented in Table 2, constitutes the basis for ensuring comparability in the results.

Figure 4
Loans Disbursed Per Month



Source: Compiled by authors.

Figure 5
Amount Disbursed (Thousands of Peruvian Soles).



Source: Compiled by authors.

Table 2
Final Set of Predictors

Variable	Type	Role in the model	Treatment applied (reader-friendly)
Disbursed Capital	Numeric	Predictor	Standardized (centered and scaled) so on the same scale as other numerics
Disbursement Month	Numeric	Predictor	Converted to dummy indicators (one indicator per month) to capture seasonality
Number of Installments	Numeric	Predictor	Standardized (centered and scaled)
Payment Frequency	Categorical	Predictor	Converted to dummy indicators (daily, weekly, biweekly, monthly)
Credit Type	Categorical	Predictor	Converted to dummy indicators
Product	Categorical	Predictor	Converted to dummy indicators
Credit Score	Categorical	Predictor	Converted to dummy indicators (includes the "No Score" category)
Financial Information	Categorical	Predictor	Converted to dummy indicators (Yes/No)
Guarantor	Categorical (Yes/No)	Predictor	Converted to dummy indicators (Yes/No)
Delay (Target Variable)	Categorical (binary)	Dependent variable	Defined as 1 = delinquent and 0 = non-delinquent. Used only as the outcome label, not transformed

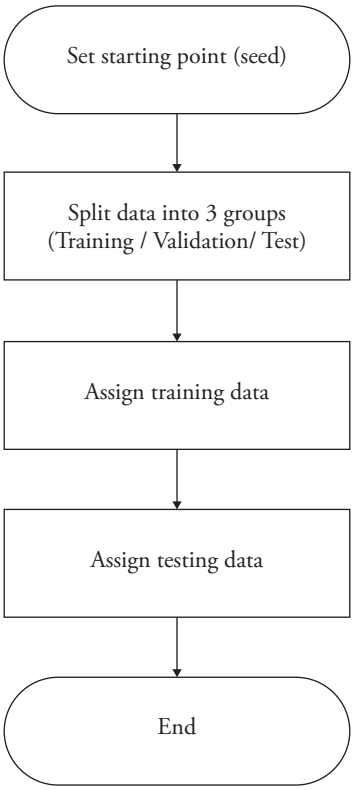
Source: Compiled by authors.

Data Splitting

We partitioned the final dataset into three subsets: 70% for training, 15% for validation, and 15% for testing. We employed the training set to fit the predictive models, while the validation set served to optimize the classification threshold and tune model performance. Finally, we employed the testing set—kept completely separate during the training and validation phases—to assess each model's generalization capacity on unseen data.

This three-way split strategy ensured that the evaluation metrics were unbiased and reduced the risk of overfitting, since the optimal cutoff point was selected based on validation results rather than test outcomes. We applied stratification by target variable (Delay) to preserve the same class distribution across all partitions.

Figure 6
Process Flow for Dataset Splitting into Training and Testing Sets



Source: Compiled by authors.

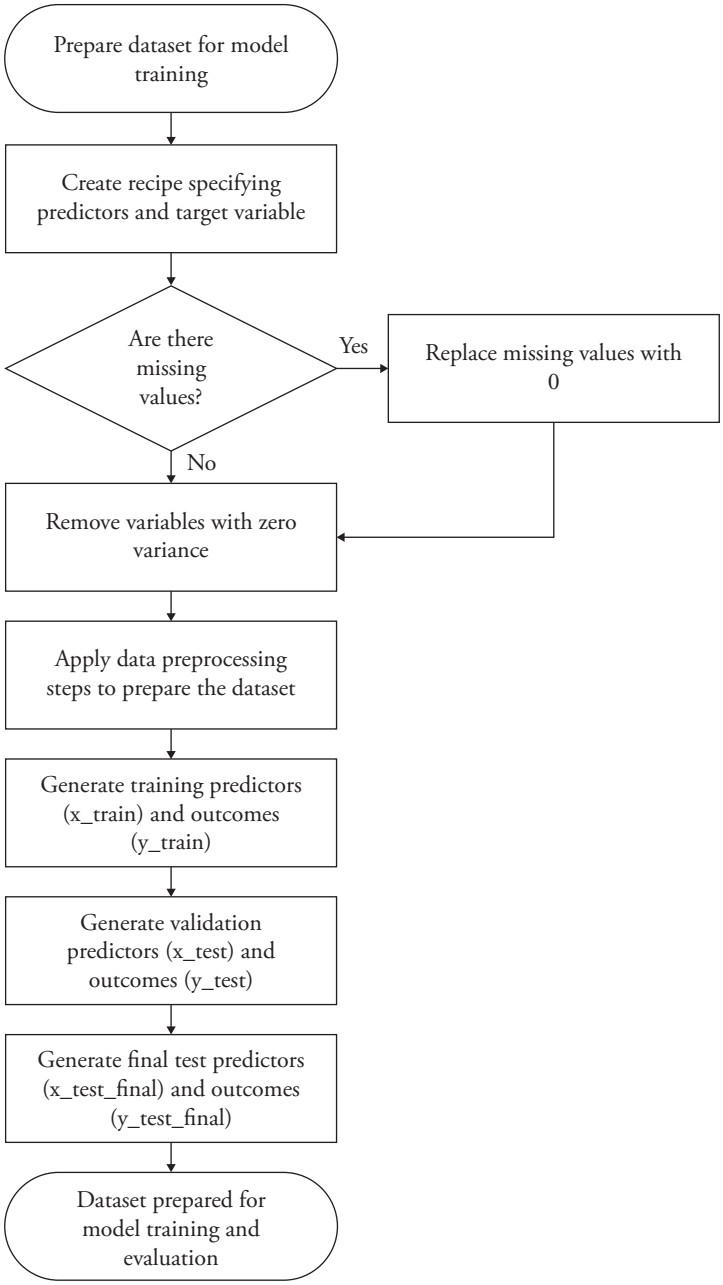
Data Preprocessing ("Recipe")

The recipes package in R was employed to construct a standardized and reproducible data preparation workflow (Kuhn & Wickham, 2020). The preprocessing stage involved the following steps:

- Missing values: All missing entries were replaced with zero, in alignment with the financial institution's data recording practice.
- Zero variance predictors: Variables with no variability across observations were removed to avoid redundant information.
- Categorical encoding: Categorical predictors (e.g., product type, credit score, guarantor) were transformed into dummy variables, resulting in a final set of 41 features after preprocessing.
- Consistent transformations: Identical preprocessing steps were applied to training, validation, and testing sets, ensuring no data leakage.
- Data stratification: The class distribution of the target variable (Delay) was preserved across partitions.

The output of this workflow generated six objects: predictors and outcomes for the training set (x_train,y_train), validation set (x_test,y_test), and final test set (x_test_final,y_test_final). This pipeline guaranteed reproducibility, comparability between models, and methodological transparency.

Figure 7
Data Preprocessing Workflow for Model Training in R



Source: Compiled by authors.

Model Training

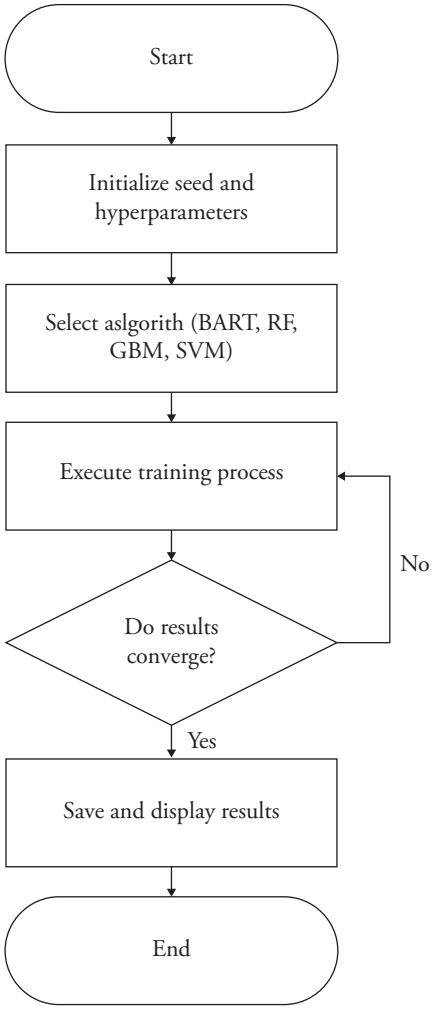
We implemented and validated four machine learning algorithms were implemented following best practices in credit scoring research (Lessmann et al., 2015; Huang et al., 2007; Rinaldo et al., 2018). As noted, the models we used in this study were BART, RF, GBM, and SVM.

All models were trained on the training set (70%) using the standardized preprocessing workflow described earlier. We initialized hyperparameters according to prior studies and package defaults, with additional tuning performed when applicable. We used a fixed random seed was used to ensure reproducibility.

First, we assessed performance on the validation set (15%), which served to optimize the classification threshold and verify convergence. Then we applied the models to the independent test set (15%), using the selected cut-off to compute final performance metrics.

This three-step process—training, validation, and independent testing—ensured both comparability across models and transparency of evaluation.

Figure 8
Flowchart: Algorithm Training Process



Source: Compiled by authors,

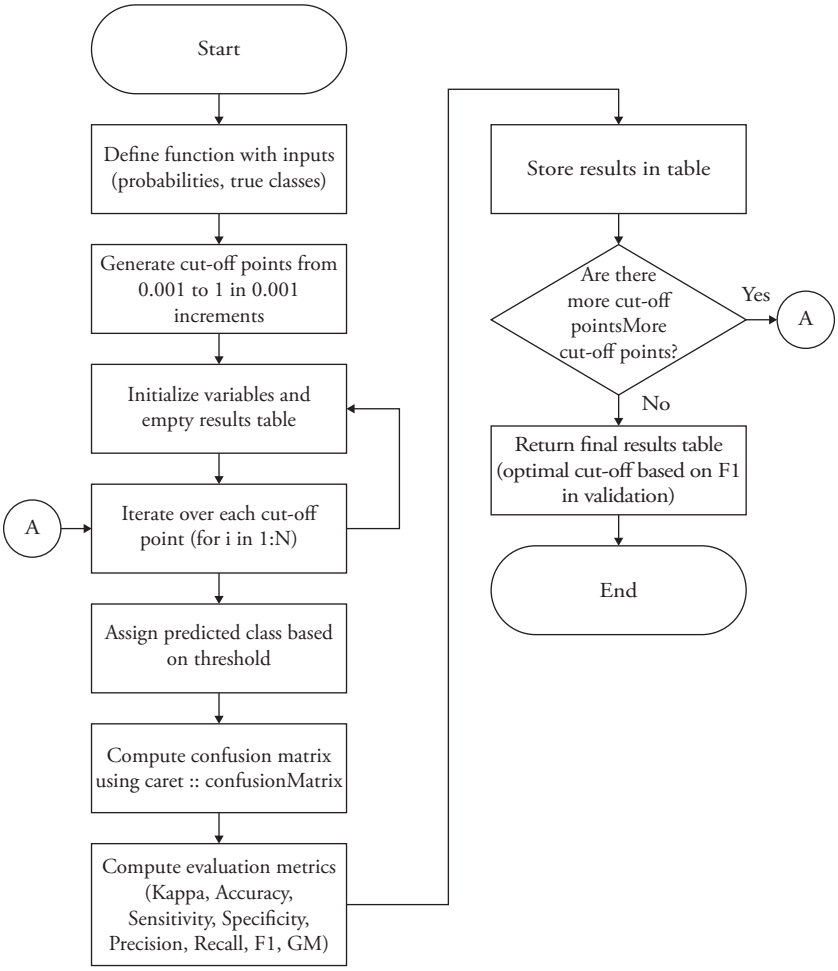
Cut-off Point Optimization

Because the models return probabilities rather than binary classifications, we evaluated multiple thresholds ranging from 0.001 to 1 (in increments of 0.001) on the validation set. The objective was to identify the cut-off point that maximized predictive performance—particularly in terms of the F1-score, which balances precision and recall and is especially relevant in imbalanced classification problems.

For each threshold candidate, we generated predicted classes, computed a confusion matrix, and recorded evaluation metrics. In addition to F1, we took into account the following indicators: Cohen's kappa, GM, accuracy, sensitivity, specificity, precision, and recall. Once the optimal cut-off was identified in validation, it was applied to the test set to obtain the final performance results.

This procedure ensured that threshold selection was not biased by the test data, thereby preserving the integrity of the generalization assessment.

Figure 9
Flowchart: Optimization of Cut-Off Point in Predictions



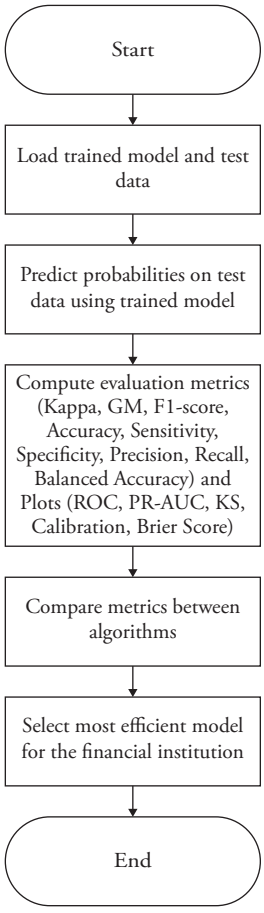
Source: Compiled by authors,

Final Validation

We conducted the final evaluation of the models on the independent test set, using the cut-off threshold previously optimized during the validation stage. This procedure ensured that the test data remained untouched during training and threshold selection, thereby preserving its role as an unbiased measure of generalization.

We recalculated the performance metrics on the test set, including Cohen's kappa, GM, F1-score, accuracy, sensitivity, specificity, precision, and recall. These results provided the basis for the final comparison of algorithms and for identifying the most effective model configuration for the financial institution.

Figure 10
Flowchart: Validation with Test



Source: Compiled by authors.

Results

This section presents the performance results of the four selected predictive models: BART, GBM, SVM, and RF. We conducted the evaluation on the test dataset using various metrics, including accuracy, kappa, sensitivity, specificity, positive predictive value, negative predictive value, and balanced accuracy. Table 3 outlines the key metrics for each model.

Table 3
Performance Metrics of Predictive Models on the Test Set

Metrics	GBM	BART	RF	SVM
Threshold	0.294	0.296	0.284	0.231
Accuracy	0.7021	0.6953	0.6777	0.6931
95% CI	(0.6915, 0.7125)	(0.6846, 0.7058)	(0.6668, 0.6884)	(0.6824, 0.7036)
No Information Rate	0.7127	0.7127	0.7127	0.7127
P-Value	0.9783	0.9995	1	0.9999
Kappa	0.3234	0.3171	0.3028	0.2474
McNemar's Test P-Value	< 2.2e-16	< 2.2e-16	< 2.2e-16	0.555
Sensitivity	0.6075	0.6198	0.655	0.4589
Specificity	0.7402	0.7257	0.6868	0.7874
Pos Pred Value	0.4852	0.4766	0.4574	0.4654
Neg Pred Value	0.8239	0.8256	0.8316	0.7831
Precision	0.4852	0.4766	0.4574	0.4654
Recall	0.6075	0.6198	0.655	0.4589
F1	0.5395	0.5389	0.5386	0.4621
Prevalence	0.2873	0.2873	0.2873	0.2873
Detection Rate	0.1745	0.1781	0.1882	0.1319
Detection Prevalence	0.3597	0.3736	0.4114	0.2833
Balanced Accuracy	0.6738	0.6727	0.6709	0.6232
GM	0.6706	0.6707	0.6707	0.6012

Source: Compiled by authors.

In general terms, the results reveal differentiated strengths and weaknesses across the four algorithms. SVM achieved the highest accuracy (69.31%) and specificity (78.74%), demonstrating strong performance in identifying

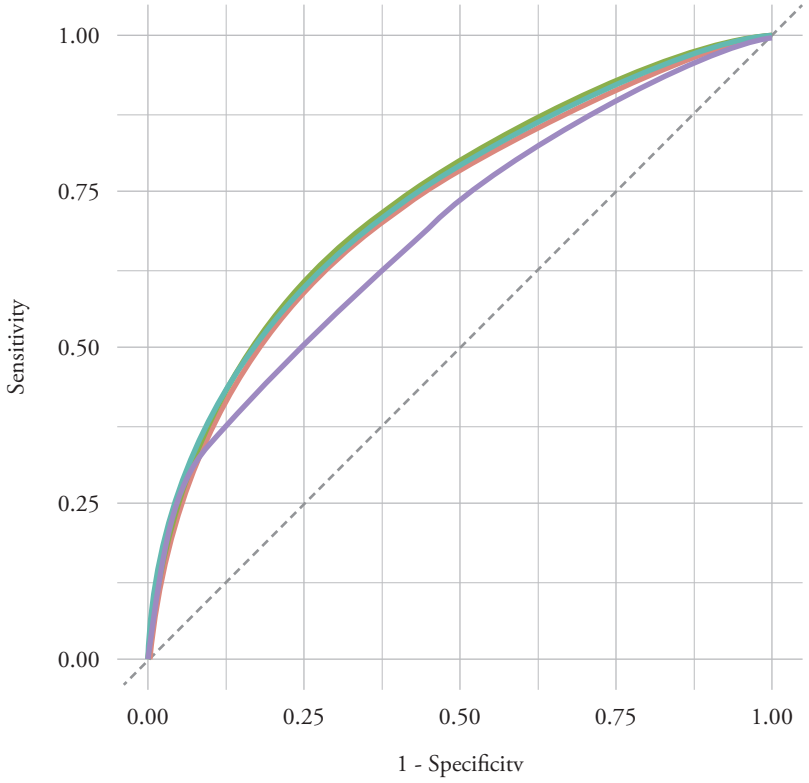
non-delinquent clients—although the sensitivity was the lowest of all models (45.89%), which limits its ability to capture delinquent cases. Gradient boosting presented a more balanced performance, with an accuracy of 70.21% and the highest Kappa value (0.3234), indicating a better level of agreement beyond chance. BART showed comparable accuracy (69.53%) and a Kappa of 0.3171, with sensitivity (61.98%) higher than that of GBM and SVM, although at the cost of lower specificity. RF obtained the highest sensitivity (65.50%) but relatively lower specificity (68.68%), resulting in the best-balanced accuracy (0.6709) among the models.

It is worth noting that some metrics, such as prevalence (28.73%) and the no information rate (71.27%), were identical across all four models. This is because these measures depend entirely on the class distribution within the dataset (the proportion of delinquent and non-delinquent clients) and not on the specific performance of each algorithm. In other words, they reflect characteristics inherent to the test dataset and provide a baseline against which to evaluate the predictive capacity of the models.

In addition to the tabular results, we conducted graphical analyses to further assess model performance:

- **ROC Curves:**
  The ROC analysis shows that all models achieved moderate discriminative capacity, with AUC values between 0.684 and 0.732. GBM (AUC = 0.732) and RF (AUC = 0.730) outperformed the other methods, closely followed by BART (AUC = 0.726). SVM obtained the lowest AUC (0.684), indicating a weaker ability to distinguish between delinquent and non-delinquent clients.

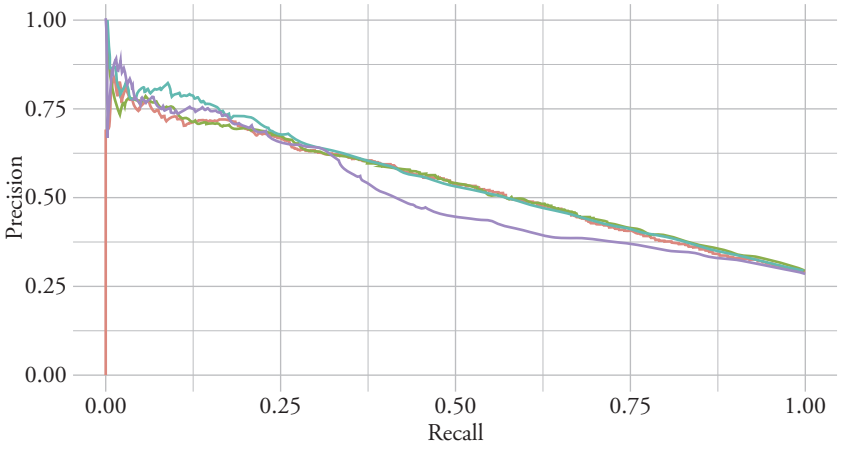
Figure 11
ROC Curves on Test



Source: Compiled by authors.

- **Precision–Recall Curves**
  The Precision–Recall analysis highlights the stronger performance of ensemble-based models. RF (PRAUC = 0.549) and GBM (PRAUC = 0.540) achieved the best balance between precision and recall, followed by BART (PRAUC = 0.535). SVM showed the lowest PRAUC (0.507), confirming its weaker ability to handle class imbalance in the dataset.

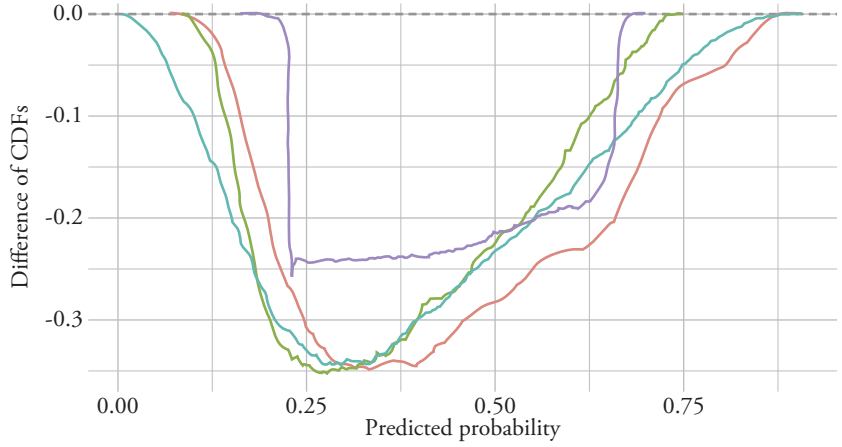
Figure 12
Precision–Recall Curves



Source: Compiled by authors.

- **KS Curves:**
  The Kolmogorov–Smirnov (KS) analysis shows there is clear separation between the cumulative distributions of positive and negative classes. GBM and RF presented the highest KS values, indicating stronger discriminatory power. BART achieved similar though slightly lower performance, while SVM displayed the weakest separation, confirming its limited ability to distinguish risk groups.

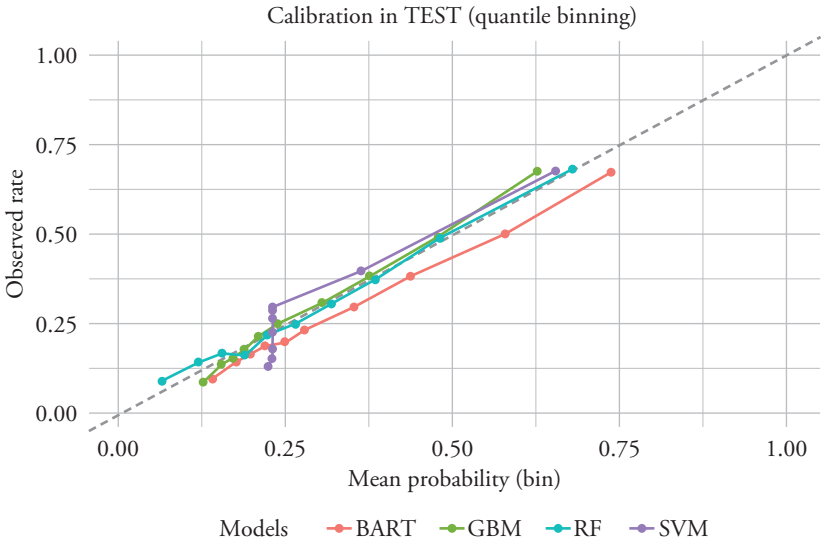
Figure 13
KS Curve On Test



Source: Compiled by authors.

- **Calibration Plots:**
  The calibration plots indicate that RF and GBM provided the closest alignment between predicted probabilities and observed default rates, suggesting better reliability in probability estimation. BART exhibited moderate deviations, while SVM exhibited greater instability, particularly at lower probability bins, reflecting miscalibration in predicting default likelihoods.

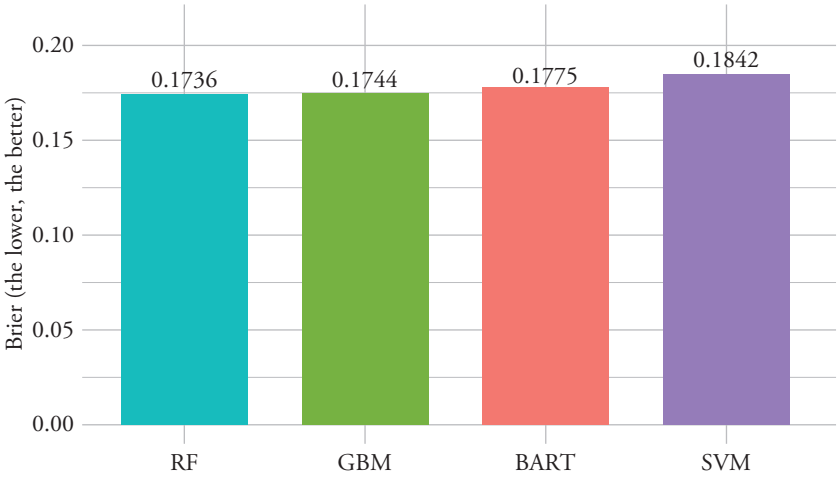
Figure 14
Calibration On Test



Source: Compiled by authors.

- **Brier Score Analysis**
  Figure 14 shows the Brier score of the four predictive models, where lower values indicate better calibration. RF (0.1736) and GBM (0.1744) achieved the lowest scores, suggesting that their predicted probabilities are more closely aligned with the observed outcomes. By contrast, BART (0.1775) and SVM (0.1842) exhibited slightly higher values, indicating less precise probability estimates. Overall, ensemble methods demonstrated superior calibration compared to BART and SVM.

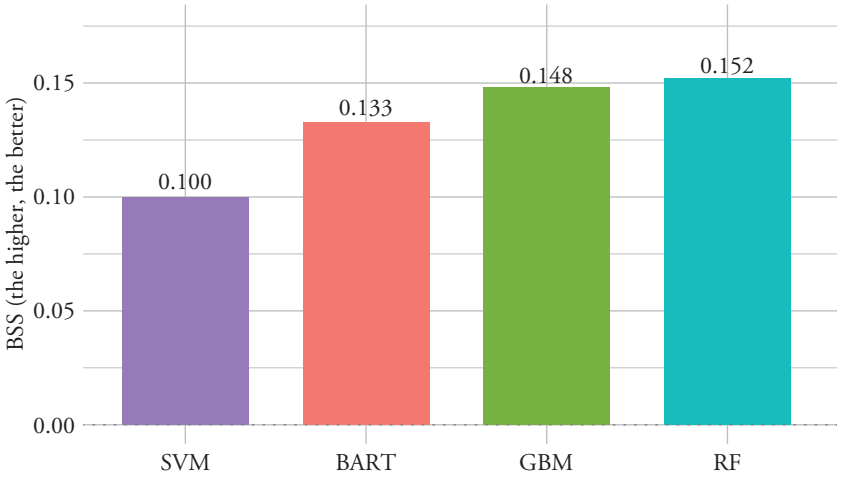
Figure 15
Brier Skill Score On Test



Source: Compiled by authors.

- **Brier Skill Score Analysis**
  Figure 15 presents the Brier skill score (BSS), which evaluates the performance of the models relative to a baseline defined by prevalence. Higher values indicate better performance. The results confirm that RF (0.152) and GBM (0.148) outperform BART (0.133) and SVM (0.100), reinforcing the conclusion that ensemble-based algorithms not only reduce prediction error but also provide more skillful probability estimates compared to simpler approaches.

Figure 16
Brier skill score on test (Baseline = prevalence)



Source: Compiled by authors

Discussion

The comparative analysis of the four predictive models (GBM, BART, RF, and SVM) provides several insights regarding their performance and calibration on the test dataset.

First, although the SVM achieved relatively competitive accuracy (0.6931) and the highest specificity (0.7874), its sensitivity (0.4589) and F1-score (0.4621) were considerably lower. This indicates that while the SVM model is effective at correctly identifying negative cases, it fails to capture a significant portion of positive cases, making it less suitable when the primary objective is the early detection of high-risk clients. This finding is consistent with previous literature, where SVM tends to favor class separation at the expense of recall in imbalanced datasets (Huang et al., 2007).

Conversely, GBM and BART demonstrated a more balanced trade-off between sensitivity and specificity. GBM achieved the highest accuracy (0.7021), balanced accuracy (0.6738), and F1-score (0.5395), closely followed by BART (accuracy = 0.6953; F1 = 0.5389). These results align with Friedman (2001) and Rinaldo et al. (2018), who highlighted the ability of boosting and Bayesian ensemble methods to adapt to complex patterns while reducing classification errors in heterogeneous data.

The RF model, while slightly lower in accuracy (0.6777), showed stable results across sensitivity (0.6550) and specificity (0.6868), highlighting its robustness. Moreover, RF achieved the best calibration performance, reflected in it having the lowest Brier score (0.1736) and the highest Brier Skill score (0.152). GBM also ranked competitively with similar Brier-based metrics, confirming that ensemble methods outperform both BART and SVM in terms of probability calibration.

As to discrimination ability, ROC and PR curves provide additional insights. ROC-AUC values were highest for GBM (0.732) and RF (0.730), followed by BART (0.726), while SVM lagged behind (0.684). Similarly, precision–recall AUC (PRAUC) favored RF (0.549) and GBM (0.540) over BART (0.535) and SVM (0.509), reinforcing the advantage of ensemble approaches under imbalanced conditions. The KS statistic further highlighted GBM and RF as the most efficient models in separating the distributions of positive and negative classes.

Calibration plots corroborated these findings, showing that GBM and RF probabilities aligned more closely with observed event rates across quantiles. BART displayed slightly weaker calibration, while SVM deviated the most from the ideal diagonal, evidencing systematic bias in probability estimates.

Overall, the results suggest that GBM and RF provide the most reliable balance between accuracy, discrimination, and calibration metrics, with GBM slightly superior in classification performance (accuracy, F1, ROC/PR curves) and RF excelling in probability calibration (Brier Score, BSS). On the other hand, SVM—despite strong specificity—proved the least suitable model due to its low sensitivity and weaker calibration. BART achieved intermediate performance but did not surpass GBM or RF across key indicators.

Despite these findings, some limitations must be acknowledged. First, the analysis was conducted using data from a single microfinance institution (MFI), which may limit the generalizability of the results to other financial contexts. Second, the models were evaluated under modest performance levels, with metrics such as accuracy and F1-score remaining relatively low, which reflects the inherent difficulty of predicting credit risk in highly imbalanced datasets. Third, potential biases could arise from the specific structure of the dataset, including variable selection and class distribution. These limitations suggest that the results should be interpreted with caution and highlight the need for future studies to incorporate multiple institutions, larger datasets, and alternative modeling strategies to strengthen the robustness and external validity of the findings.

Conclusions

Although the results of this study show that the overall performance of the analyzed models remains modest and is not yet definitive, they still serve as useful evidence to guide the design of predictive schemes in microfinance. Among the four algorithms evaluated, GBM emerges as the most suitable option when the primary objective is the early detection of clients with a higher probability of arrears. This model achieved the best balance between sensitivity (0.6075), F1-score (0.5395), accuracy (0.7021), and discrimination metrics (AUC-ROC = 0.732; PRAUC = 0.540), suggesting a stronger capacity to identify high-risk cases without sacrificing stability in other relevant measures.

In turn, RF performed competitively and excelled in probability calibration (Brier score = 0.1736; BSS = 0.152), though its F1 and sensitivity values were lower than those of GBM, reducing its usefulness when the priority is to minimize false negatives. BART offered intermediate results, while SVM, despite its high specificity, exhibited considerably lower sensitivity (0.4589), which limits its applicability in contexts where the cost of missing arrears is critical.

These findings reinforce the importance of combining both classification and calibration metrics in the evaluation of credit risk models. They also confirm that the optimization of decision thresholds, within the range of 0.001 to 1, is as relevant as the choice of the algorithm itself in enhancing the identification of clients at risk of default.

Finally, in conceptual and forward-looking terms, this study is framed within the proposal of artificial intelligence agent-based systems for risk analysis in microfinance. Although no such framework was implemented as part of this research, the results suggest that models like GBM—which prioritize early detection of arrears and maintain stability across different metrics—can serve as a robust input for multi-agent architectures aimed at continuous risk monitoring, proactive client interactions, and financial loss mitigation.

References

Armendáriz, B., & Morduch, J. (2010). *The economics of microfinance* (2nd ed.). MIT Press.

Brown, I., & Mues, C. (2012). An experimental comparison of classification algorithms for imbalanced credit scoring data.

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* (pp. 785–794). ACM. https://doi.org/10.1145/2939672.2939785

Economist Intelligence Unit. (2012). *Global microscope on the microfinance business environment 2012*. https://www.eiu.com/n/campaigns/microscope2012/

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics, 29*(5), 1189–1232. https://doi.org/10.1214/aos/1013203451

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning* (2nd ed.).

He, H., & Garcia, E. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering 21*(9), 1263–1284 10.1109/TKDE.2008.239

Huang, C. L., Chen, M. C., & Wang, C. J. (2007). Credit scoring with a data mining approach based on support vector machines. *Expert Systems with Applications, 33*(4), 847–856. https://doi.org/10.1016/j.eswa.2006.07.007

J-PAL. (2022). *Microcredit: Impacts and promising innovations*. Abdul Latif Jameel Poverty Action Lab. https://www.povertyactionlab.org/policy-insight/microcredit

Khandani, A., Kim, A., & Lo, A. (2010). Consumer credit-risk models via ML. *Journal of Banking & Finance 34*(11), 2767–2787 https://doi.org/10.1016/j.jbankfin.2010.06.001

Lessmann, S., Baesens, B., Seow, H., & Thomas, L. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society, 66*(4), 740–758. https://doi.org/10.1057/jors.2014.22

Malekipirbazari, M., & Aksakalli, V. (2015). Risk assessment in social lending via random forests. *Expert Systems with Applications, 42*(10), 4621–4631. https://doi.org/10.1016/j.eswa.2015.01.002

Nhung, D. H., & Simioni, M. (2021). A comparison of Random Forest and logistic regression model in credit scoring. *HAL Open Archive.* https://hal.science/hal-03178971

Rinaldo, A., Passos, L., Lopes, H. F., & Giudici, P. (2018). Application of Bayesian additive regression trees in the development of credit scoring models in Brazil. *Brazilian Journal of Probability and Statistics, 32*(2), 264–280. https://doi.org/10.1214/17-BJPS354

Sharma, D. (2013). *Improving credit scoring with random forests* [Masters thesis, San José State University]. SJSU ScholarWorks. https://scholarworks.sjsu.edu/etd_projects/353

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*. *45*(4), 427–437. https://doi.org/10.1016/j.ipm.2009.03.002