Mathematical Modeling and Algorithm Application
ISSN: 3006-0842 | Vol. 10, No. 1, 2026

Application of Cost-Sensitive Threshold Calibration in Fraud Detection

Jiayi Xing*

School of Systems Science and Statistics, Beijing Wuzi University, Beijing, 101100, China
* **Corresponding author**: Jiayi Xing (Email: 3144865361@qq.com)

Abstract: Detecting credit card fraud in practice must confront two core challenges: extreme rarity of the positive class and losses that differ sharply between the two types of mistake. The conventional practice of employing a fixed 0.5 cutoff while optimising for accuracy is ill-suited to this setting, as it cannot reconcile the economic damage of a missed fraud with the operational expense of a false alert. The present work develops a threshold calibration framework driven by expected cost minimisation. A misclassification cost matrix is first constructed. Three base classifiers—Logistic Regression, Random Forest, and XGBoost—provide per-transaction fraud likelihoods, which are then passed through Platt Scaling to improve reliability. A Bayes minimum-risk decision rule yields the optimal static cutoff, and an additional dynamic mechanism that ties the false-negative penalty to the transaction amount is introduced. Evaluated on the ULB Credit Card Fraud Detection set , probability calibration cuts the dynamic-threshold cost of Logistic Regression by 92.76%. The dynamic strategy attains the lowest overall expense of €2,655.28 on XGBoost, a 27.04% drop relative to the default 0.5 threshold. The proposed framework demonstrably lowers total misclassification cost and supplies an interpretable, cost-aware foundation for threshold decisions in high-stakes applications.

Keywords: Cost-Sensitive Learning; Threshold Calibration; Credit Card Fraud Detection; Class Imbalance; Bayesian Decision

1. Introduction

Electronic payments have expanded at an extraordinary pace over the past decade, and the deepening of Internet finance has cemented credit cards as a primary consumer payment instrument across the globe [1]. This expansion, however, has also nurtured a parallel growth in fraudulent activity, imposing serious financial burdens on cardholders, merchants, and issuing banks alike [2].

Viewed through the lens of machine learning, credit card fraud detection can be cast as a supervised binary classification task: each incoming transaction must be labelled as either legitimate or fraudulent [3]. Yet the problem carries two properties that set it apart from ordinary classification exercises.

First, Class Imbalance is severe. In genuine transactional streams, fraud accounts for only a tiny sliver of the total volume. For the ULB Credit Card Fraud Detection dataset adopted here, 492 of the 284,807 records are fraudulent—roughly 0.172% [4]. Under these conditions, a naive classifier that trivially flags every transaction as legitimate would reach an accuracy above 99.8% while completely missing actual fraud. Accuracy, as a metric, becomes all but meaningless [5].

Second, the two types of misclassification errors carry very different costs. In real fraud detection, false negatives and false positives have very different economic consequences [6]. A fraudulent transaction that goes undetected means the issuing institution bears the full transaction amount; mistakenly flagging a legitimate transaction only triggers a manual review, whose cost is relatively small [7]. Because of this gap, the decision boundary of a classifier cannot be set apart from the cost structure; it must match the actual loss profile of the business.

Conventional binary classifiers typically use 0.5 as the decision threshold and optimize overall accuracy [8]. When class distributions are roughly balanced and the two error types carry similar costs, this approach works fine; in the fraud detection setting, however, it falls short [9]. To deal with class imbalance, the existing literature offers two main types of solutions—resampling and cost-sensitive learning—but pays little attention to a key question at the prediction stage: once a probabilistic model has been trained, how should the threshold be chosen [10]. Recent studies have begun to separate the optimization of ranking ability from the choice of cutoff point, so that cost minimization and detection quality can be improved at the same time [11].

Taken together, these points motivate us to examine cost-sensitive threshold calibration more closely, with the goal of building a systematic method that minimizes total misclassification costs. Our work contributes to the literature in several respects. We construct a misclassification cost matrix that reflects the specific economics of credit card fraud. We also design a threshold calibration procedure—rooted in Bayesian minimum-risk theory—that yields both a static optimum and a dynamic adjustment linked to transaction size. Finally, instead of relying on accuracy-based metrics, we evaluate performance purely from an economic-cost standpoint, which we believe offers a more realistic picture.

2. Literature Review

2.1. Evolution of Credit Card Fraud Detection Techniques

The development of Credit Card Fraud Detection techniques has undergone an evolutionary progression from rule-based methods to machine learning-based approaches [12]. A recurring challenge in fraud detection is how to keep pace with rapidly evolving attack patterns. Early rule-based systems, while transparent and easy to deploy, were brittle in the face of such changes [13]. Transactions that exceeded a

preset limit or originated from unusual locations would be flagged, but these rigid criteria were quickly outmanoeuvred as fraudsters adapted. This weakness spurred the adoption of more flexible algorithms—decision trees, SVMs, neural networks, and ensembles—that could uncover non-linear signals in high-dimensional transaction data [14]. Among these, ensemble methods have consistently emerged as particularly competitive in comparative studies. Hand and Henley [15] reviewed statistical classification methods in consumer credit assessment, providing methodological foundations that later extended to fraud detection. Another study [16] proposed a systematic framework for engineering features from raw transaction records, showing that temporal and behavioural aggregate features meaningfully boost detection performance.

2.2. Class Imbalance and Cost-Sensitive Learning

Research on the class imbalance problem in fraud detection has followed two main paths: one at the data level and the other at the algorithm level [17]. At the data level, the SMOTE method proposed by Chawla et al. is a representative approach. It generates synthetic samples through linear interpolation between minority-class instances, and compared with simple replication-based oversampling, it effectively reduces overfitting [18]. In high-dimensional feature spaces, however, this interpolation approach becomes less effective. Subsequent studies have proposed improved sampling schemes [19] and introduced a cost-sensitive C-score metric to guide threshold selection, improving the identification of fraudulent samples.

At the algorithm level, cost-sensitive learning incorporates class weights or cost matrices directly into the objective function. Elkan [20] provided a systematic theoretical derivation, giving the Bayesian optimal decision rule under a given cost matrix and establishing the rescaling theorem, which links the cost ratio to class prior probabilities. Bahnsen et al. applied this framework to credit card fraud and proposed a cost-oriented evaluation scheme that matches the economic realities of the business [21]. A recent line of work, for instance, combines deep reinforcement learning with integrated gradients to enable adaptive decision-making [22], a step toward handling concept drift in fraud patterns. A survey by [23] takes stock of current ML, deep learning, and metaheuristic applications in this domain, and arrives at a sobering conclusion: existing models are still ill-equipped to cope with dynamic fraud behaviours.

In terms of threshold optimization, Bayesian decision theory provides a principled foundation, yet the existing literature has at least three gaps. For one thing, tree-based and boosting models often output raw probabilities that are poorly calibrated [24]. This issue undermines any subsequent threshold selection that assumes these probabilities are reliable. Our study attempts to address these gaps by bringing together Bayesian decision theory, probability calibration techniques, and dynamic cost modelling within a single threshold calibration framework.

3. Method

3.1. Dataset Description and Preprocessing

The experiments use the credit card fraud detection dataset first released in 2013, a collaborative release from Worldline and the Machine Learning Group at Université Libre de Bruxelles. The data contain 284,807 transactions from two consecutive days in September 2013, of which 284,315 are genuine and 492 are fraudulent—a fraud rate of about 0.172% (Fig 1). Each record has 31 fields: time, amount, 28 anonymized PCA components labelled V1–V28, and a binary class label.



Figure 1 Transaction class distribution

Preprocessing proceeded as follows. The V1–V28 features, having already been PCA-transformed, required no further scaling. The time variable was normalized to [0, 1] via Min-Max scaling. For the amount variable, we used Robust Scaler with the median (€22.00) and interquartile range as parameters—this choice mitigates the influence of extreme outliers. To avoid time leakage, the data were sorted chronologically and split in temporal order: the first 80% of samples (227,845 transactions) formed the training set, and the remaining 20% (56,962 transactions) formed the test set. No missing values were present.

A word on the dataset's age: it is now over a decade old, and the transaction patterns and fraud tactics it captures belong to an earlier era. The payment landscape has since changed considerably—mobile wallets, contactless terminals, and cross-border e-commerce have opened up new fraud vectors. Still, the dataset retains value for several reasons. The PCA-transformed features are statistically detached from specific payment channels, which preserves some generalizability. Its fraud rate of roughly 0.17% is close to what is commonly reported in the industry. And as a widely used benchmark, it allows direct comparison with previous studies. At the same time, its limitations are clear—it comes from a single European institution, spans only two days, and does not cover newer payment methods. Further validation on more recent and geographically diverse data is an important next step.

3.2. Model Evaluation Metric System

Given the twin challenges of severe class imbalance and asymmetric costs, we adopted a two-tier evaluation framework. The first tier consists of conventional metrics: precision, recall, F1-score, AUC-ROC, and AUC-PR. AUC-ROC measures how well the model separates classes across all possible thresholds, but it can paint an overly rosy picture under extreme imbalance. AUC-PR is more sensitive to performance on the minority class. Reporting both together gives a fuller picture of model quality.

The second tier is cost-sensitive, with Total Misclassification Cost (TMC) as the primary metric. For static threshold strategies, TMC is defined as follows:

$$\mathrm{TMC}_{\mathrm{static}} = \mathrm{FP} \times C_{\mathrm{FP}} + \mathrm{FN} \times C_{\mathrm{FN}} \quad (1)$$

where FP is the number of false positives (genuine

transactions misclassified as fraudulent), FN is the number of false negatives (fraudulent transactions misclassified as genuine), and C_{FP} and C_{FN} correspond to the unit costs of the two types of errors, respectively. For the dynamic threshold strategy, the cost of false negatives is tied to the transaction amount, and TMC takes the following form:

$$TMC_{dynamic} = FP \times C_{FP} + \sum_{i \in \mathcal{FN}} Amount_i \quad (2)$$

i indexes a fraudulent transaction that is misclassified as legitimate.

3.3. Construction of Base Classification Models

We selected three classifiers as base models—logistic regression, random forest, and XGBoost—to generate fraud probability estimates for each transaction, which then feed into the threshold calibration procedure. These three were chosen because they represent three distinct modelling paradigms: linear models, bagging ensembles, and boosting ensembles (Fig 2 and Fig 3).

For logistic regression, we used the "balanced" mode to adjust class weights automatically, with L2 regularization to keep overfitting in check. Random forest tackles minority-class identification through a combination of class weighting and stratified sampling. XGBoost sets the positive class weight dynamically according to the ratio of negative to positive samples, with a learning rate of 0.1, a maximum tree depth of 6, and early stopping to prevent overfitting during training. All hyperparameters were determined via grid search under five-fold cross-validation. Table 1 summarizes the performance of the three base models on the test set.

Table 1 Performance summary of the three models (test set)

Models	AUC-ROC	AUC-PR	Precision	Recall	F1
Logistic Regression	0.9853	0.5966	0.2605	0.8267	0.3962
Random Forest	0.9759	0.7848	0.8169	0.7778	0.7969
XGBoost	0.9865	0.8001	0.7703	0.7600	0.7651

As shown in Table 1, XGBoost achieves the best AUC-ROC (0.9865) and AUC-PR (0.8001); the random forest model exhibits the highest precision (0.8169), while the logistic regression model attains the highest recall (0.8267) but with a precision of only 0.2605. This gain in recall comes at the cost of generating a substantial number of false positives. A trade-off between precision and recall exists across different models, making it difficult for a single threshold to satisfy both metrics simultaneously. Furthermore, the F1-optimal thresholds for logistic regression, random forest, and XGBoost are 0.89, 0.55, and 0.78, respectively, all deviating from the default value of 0.5.

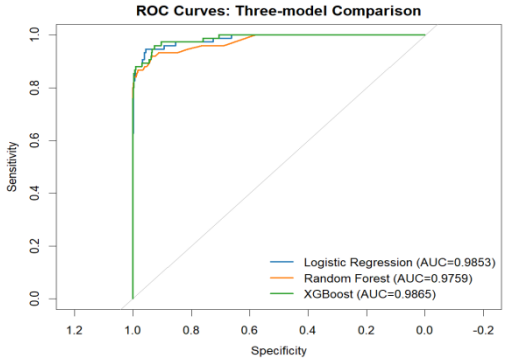


Figure 2 ROC Curves

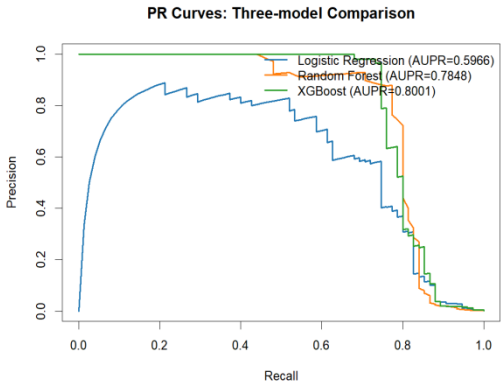


Figure 3 PR Curves

3.4. Cost-Sensitive Threshold Calibration Framework

3.4.1. Misclassification Cost Matrix

In binary classification, let C(i, j) denote the cost of misclassifying true class jj as class ii, with zero cost for correct classifications. Two misclassification costs are defined: C_{FN} for false negatives (fraud misclassified as legitimate) and CFP for false positives (legitimate misclassified as fraud). For the static threshold, a relative cost ratio is adopted. C_{FN}=5.56 is derived from the ratio of the average fraud amount (€122.21) to the median transaction amount (€22.00), meaning a missed fraud costs 5.56 times a false positive; C_{FP} is normalized to 1. For the dynamic threshold, $C_FN(Amount) = Amount(ineuros)$, while C_{FN} =€1(manual review cost), ensuring dimensional consistency.

3.4.2. Probability Calibration

The Bayesian threshold calibration method assumes that the classifier's probability estimate p(x) accurately reflects the true posterior probability of fraud. However, Random Forest and XGBoost often exhibit calibration bias. Uncalibrated probabilities cause the Bayesian threshold to deviate from the optimum, as the decision relies on absolute probability values rather than merely their ranking. Thus, probability calibration is a prerequisite for the Bayesian optimal threshold.

In this study, the Platt Scaling method is adopted to calibrate the probability outputs. The calibrated probability estimate is given by:

$$p_{calibrated}(x) = \frac{1}{1+\exp(A \cdot f(x)+B)} \quad (3)$$

where f(x) denotes the raw output probability of the base classifier, and A and B are the calibration parameters to be estimated. To avoid overfitting, a 5-fold cross-validation scheme is adopted to generate Out-of-Fold (OOF) probabilities: the training set is partitioned into 5 folds; in each iteration, the base model is trained on 4 folds and used to generate predicted probabilities for the remaining 1 fold (Fig 4). After cycling through all 5 folds, the OOF

probabilities for the entire training set are obtained. The Platt Scaling parameters are then fitted using these OOF probabilities along with the true labels, and subsequently applied to the test set. The calibrated parameters for the three models are presented in Table 2.

Table 2 Calibration parameters of Platt Scaling

Models	A	B	Calibrated Probability Range
Logistic Regression	11.9165	-10.5422	[0, 0.7981]
Random Forest	11.5972	-10.1711	[0, 0.4508]
XGBoost	10.6401	-8.5268	[0.0002, 0.8922]

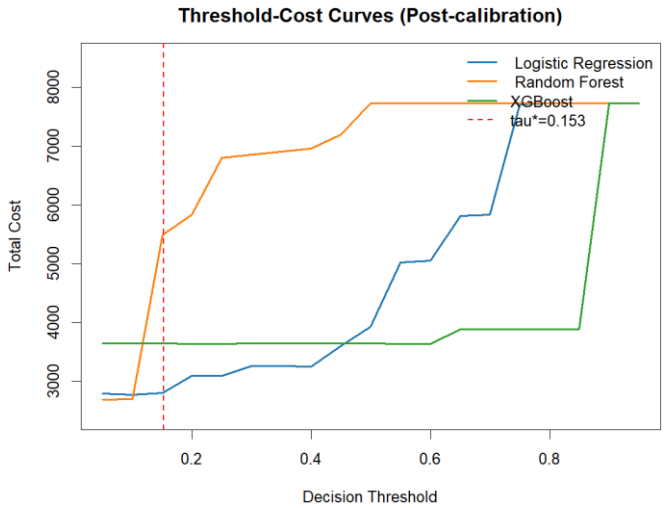


Figure 4. Reliability curves of the three models before and after calibration.

3.4.3. Bayesian Optimal Threshold Derivation

Upon completion of probability calibration, Bayesian minimum-risk decision theory is introduced to provide a theoretical framework for optimal threshold selection. For a transaction whose calibrated predicted probability is p, if the decision is made with a threshold τ (i.e., intercepting when $p \geq \tau$, and releasing when $p < \tau$), the expected misclassification cost can be expressed as:

$$E[Cost] = (1 - p) \cdot \mathbb{I}(p \geq \tau) \cdot C_{FP} + p \cdot \mathbb{I}(p < \tau) \cdot C_{FN} \quad (4)$$

To minimize the expected cost, the transaction should be intercepted when $(1 - p) \times C_{FP} \leq p \times C_{FN}$,i.e, when $p \geq \frac{C_{FP}}{C_{FP}+C_{FN}}$Therefore, the Bayesian optimal threshold is given by:

$$\tau^* = \frac{C_{FP}}{C_{FP}+C_{FN}} \quad (5)$$

This formulation carries a clear economic interpretation: the larger the false negative cost C_{FN} relative to the false positive cost C_{FP}, the lower the optimal threshold $\tau*$, and consequently the more inclined the model is to intercept transactions. Under the static setting of this study, with $C_FP = 1$, $C_{FN} = 5.56$, the optimal threshold is $\tau* = 1/(1 + 5.56) = 0.1526$ (Fig 5). This threshold is substantially lower than the conventional default value of 0.5, reflecting the realistic characteristic that the cost of false negatives far outweighs that of false positives in fraud detection.

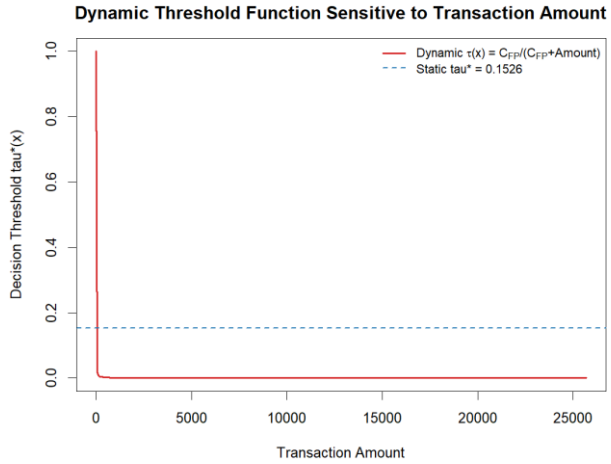


Figure 5 Threshold-cost curves after calibration

3.4.4. Dynamic Threshold Mechanism

The static threshold approach assumes an identical misclassification cost structure across all transactions; however, in real-world scenarios, the false negative losses associated with transactions of varying amounts differ substantially. To fully leverage the transaction amount information, this study proposes a transaction-amount-

sensitive dynamic threshold mechanism built upon the Bayesian optimal threshold framework. Under this dynamic threshold mechanism, the false negative cost $C_{FN}(x)$ is no longer a fixed constant but is tied to the transaction amount, $C_{FN}(x) = \text{Amount}(x)$ (in euros), while the false positive cost C_{FP} remains fixed at €1. Consequently, the optimal threshold for each transaction becomes a function of the transaction amount:

$$\tau^*(x) = \frac{C_{FP}}{C_{FP}+\text{Amount}(x)} = \frac{1}{1+\text{Amount}(x)} \quad (6)$$

This functional form embodies the core principle of "the larger the transaction amount, the stricter the risk control." The total cost comparison of the three threshold strategies—the default 0.5, the static τ, and the dynamic τ(x)—on the test set is presented in Table 3.

Table 3 Comparison of total misclassification costs across the three threshold strategies (calibrated probabilities, in euros).

Model	Default threshold (0.5)	Static threshold (τ*)	Dynamic threshold (τ*(x))	Cost change before vs. after calibration
Logistic Regression	3,928.69	2,844.60	2,751.54	Reduction of 92.76%
Random Forest	6,944.26	3,947.67	2,478.78	Reduction of 83.66%
XGBoost	3,638.54	3,638.54	2,655.28	Increase of 22.96%

4. Conclusion

This study shows that probability calibration is a necessary step before cost-sensitive threshold calibration. Using Platt Scaling with 5-fold cross-validation, the probability quality of Random Forest and Logistic Regression improves notably. After calibration, the total cost of Logistic Regression under the dynamic threshold drops from €38,018.48 to €2,751.54, a reduction of 92.76%. Random Forest shows a similar pattern, falling from €15,173.96 to €2,478.78, a reduction of 83.66%. These results confirm that calibrated probability estimates matter for cost-sensitive decision-making. For XGBoost, however, the total cost rises from €2,159.49 to €2,655.28 after calibration. This increase is unexpected but can be explained: when a model's uncalibrated probabilities already sit close to the cost-based threshold, calibration may shift predictions for borderline cases in an unfavorable direction.

The Bayesian optimal threshold performs much better than the default threshold of 0.5. For Random Forest, the dynamic threshold yields a total cost of €2,478.78, the lowest across all three models and three threshold strategies. At the same time, its precision drops from 0.9184 to 0.3647 and its F1-score from 0.7258 to 0.3875 (Table 3), meaning that the cost saving comes with a clear loss in predictive accuracy. XGBoost with the dynamic threshold, by comparison, keeps a precision of 0.7612 and an F1-score of 0.7183. Its total cost is slightly higher than that of Random Forest, but its predictive reliability is better. The lowest total cost does not always mean the best overall performance. In practice, the choice depends on how much the business can tolerate false positives versus missed frauds.

The amount-sensitive dynamic threshold improves on the static approach by lowering the decision barrier for high-amount transactions, catching more of the most damaging frauds while keeping false positives low on routine small-value purchases. On XGBoost, this strategy reaches a total cost of €2,655.28, a 27.04% improvement over the default cutoff of 0.5.

This study has several limitations. First, C_{FP} is fixed at €1, but in practice review costs may vary by channel, time of day, or customer segment. Second, the dynamic threshold relies only on transaction amount; adding other factors such as cardholder behavioural history, merchant risk profiles, and temporal context could improve cost estimates further. Third, some studies have questioned whether assigning the full transaction amount to the false-negative cost always produces the best classifier performance. The evidence on this issue is mixed, and the right cost specification likely depends on the dataset and the fraud distribution. Future work could also introduce an online learning framework to update thresholds adaptively as fraud patterns change. Finally, the ULB dataset used in this study was released in 2013 and reflects the transaction patterns and fraud tactics of that time. As payment technologies and fraud strategies evolve, the findings may not fully generalize to the current payment environment, and future research should validate the proposed method on more recent datasets to confirm its applicability and robustness.

References

[1] Nilson Report. (2024). Card fraud losses reach $32.34 billion worldwide. https://nilsonreport.com/

[2] Abdallah, A., Maarof, M. A., & Zainal, A. (2016). Fraud detection system: A survey. Journal of Network and Computer Applications, 68, 90–113. https://doi.org/10.1016/j.jnca.2016.04.007

[3] Bolton, R. J., & Hand, D. J. (2002). Statistical fraud detection: A review. Statistical Science, 17(3), 235–255. https://doi.org/10.1214/ss/1042727940

[4] ULB Machine Learning Group. (2016). Credit card fraud detection dataset [Data set]. Kaggle. https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

[5] Korynetskyi, D., & Stetsenko, I. (2025). Detection method of fraudulent payment transaction based on C-score metric. Information, Computing and Intelligent Systems, 6, 75–86. https://doi.org/10.20535/2786-8729.6.2025.333898

[6] Bahnsen, A. C., Stojanovic, A., Aouada, D., & Ottersten, B. (2013). Cost sensitive credit card fraud detection using Bayes minimum risk. In Proceedings of the 12th International Conference on Machine Learning and Applications (ICMLA) (pp. 333–338). Miami, FL. https://doi.org/10.1109/ICMLA.2013.134

[7] Elbadraoui, A., Mouhssine, Y., El Alaoui, A., & Ouatik El Alaoui, S. (2026). Cost-aware deep neural network for credit card fraud detection under chronological evaluation. Statistical, Optimisation and Information Computing, 15(4), 3174–3193. https://doi.org/10.19139/soic-2310-5070-3362

[8] Duda, R. O., Hart, P. E., & Stork, D. G. (2001). Pattern classification (2nd ed.). Wiley. https://www.wiley.com/en-us/Pattern+Classification%2C+2nd+Edition-p-9780471056690

[9] Sheokand, S., & Beniwal, S. (2025). Credit card fraud detection using machine learning and deep learning techniques: A review. Pertanika Journal of Science & Technology, 33(6). http://pertanika2.upm.edu.my/pjst/browse/regular-issue?article=JST-5675-2024

[10] Sang, V. N. T. (2026). Enhancing credit card fraud detection under severe class imbalance using cost-sensitive learning and threshold optimization. EAI Endorsed Transactions on Security and Machine Learning. https://doi.org/10.4108/eetismla.12078

[11] Danang, D., & Putra, T. W. A. (2025). Cost-sensitive fraud detection with reliability calibration: A practical pipeline with XGBoost and focal-proxy reweighting. International Journal of Information Technology and Business, 8(1), 13–24. https://doi.org/10.24246/ijiteb.812025.13-24

[12] IEEE. (2026). A systematic review of ML/DL for credit card fraud detection. In 2026 IEEE Conference. https://ieeexplore.ieee.org/document/11430904

[13] IEEE. (2026). Real-time credit card fraud detection using graph attention networks with cost-sensitive learning. In 2026 IEEE Conference. https://ieeexplore.ieee.org/document/11546393

[14] Bhattacharyya, S., Jha, S., Tharakunnel, K., & Westland, J. C. (2011). Data mining for credit card fraud: A comparative study. Decision Support Systems, 50(3), 602–613.

[15] Hand, D. J., & Henley, W. E. (1997). Statistical classification methods in consumer credit scoring: A review. Journal of the Royal Statistical Society: Series A, 160(4), 523–541.

[16] Bahnsen, A. C., Aouada, D., Stojanovic, A., & Ottersten, B. (2016). Feature engineering strategies for credit card fraud detection. Expert Systems with Applications, 51, 134–142.

[17] Sheokand, S., & Beniwal, S. (2025). Credit card fraud detection using machine learning and deep learning techniques: A review. Pertanika Journal of Science & Technology, 33(6).

[18] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321–357.

[19] Korynetskyi, D., & Stetsenko, I. (2025). Detection method of fraudulent payment transaction based on C-score metric. Information, Computing and Intelligent Systems, 6.

[20] Elkan, C. (2001). The foundations of cost-sensitive learning. In Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI) (pp. 973–978). Seattle, USA.

[21] Bahnsen, A. C., Aouada, D., & Ottersten, B. (2015). A cost sensitive framework for credit card fraud detection. In Proceedings of the 4th International Conference on Pattern Recognition Applications and Methods (ICPRAM) (pp. 51–58). Lisbon, Portugal.

[22] IEEE. (2026). Credit card fraud detection using DQNN and integrated gradients for adaptive decisions. In 2026 IEEE Conference. https://ieeexplore.ieee.org/document/11566609

[23] Hafez, I. Y., Hafez, A. Y., Saleh, A., & Abd Elrahman, S. (2025). A systematic review of AI-enhanced techniques in credit card fraud detection. Journal of Big Data, 12(1), 1–35.

[24] Platt, J. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Advances in Large Margin Classifiers, 10(3), 61–74.