Machine Learning-Based Credit Card Fraud Detection:

A Comparative Analysis of Ensemble Methods with SHAP Explainability and Business Impact Optimisation

Avinash Chaurasiya

Nanyang Technological University, Singapore

nie17avin6132@e.ntu.edu.sg

February 24, 2026

Abstract

Credit card fraud poses an escalating threat to the global financial ecosystem, causing billions of dollars in annual losses and eroding consumer trust. Effective automated fraud detection must contend with severe class imbalance, evolving attack patterns, and the practical need for explainable, actionable predictions. In this paper, we present a rigorous comparative study of five machine learning classifiers—Logistic Regression, Decision Tree, Random Forest, Gradient Boosting, and XGBoost—applied to a dataset of 50,000 credit card transactions exhibiting a realistic fraud rate of 0.34%. We evaluate the impact of two class-imbalance remediation strategies (SMOTE oversampling and random undersampling), conduct threshold optimisation to align classification decisions with business economics, and employ SHAP (SHapley Additive exPlanations) values to provide model-level and instance-level interpretability. Our best model, Gradient Boosting, achieves a ROC-AUC of 0.9995, a PR-AUC of 0.9421, and an F1 score of 0.7805 under a cost-optimised decision threshold of 0.75, translating into an estimated net business benefit of $4,228 per 10,000 transactions compared to a no-model baseline. Feature analysis identifies V27 (importance = 0.397) and V2 (0.213) as the dominant fraud signals among the PCA-derived features. This work demonstrates that ensemble gradient-boosted trees, combined with principled threshold tuning and SHAP explainability, constitute a production-ready solution for real-world fraud detection.

Keywords: credit card fraud detection, machine learning, class imbalance, SMOTE,

gradient boosting, XGBoost, SHAP, threshold optimisation, business impact analysis.

Contents

1 Introduction

Financial fraud is one of the most consequential failure modes of modern digital payment infrastructure. Global losses attributable to card-based fraud exceeded $33 billion in 2022 and are projected to surpass $40 billion by 2027 [The Nilson Report, 2023]. Unlike many binary classification tasks, fraud detection is characterised by an extreme imbalance between the majority (legitimate) and minority (fraudulent) class, typically on the order of 1:200 to 1:500. This imbalance makes accuracy an uninformative metric and renders naive classifiers that predict "legitimate" for every transaction superficially competitive while providing zero practical value.

Beyond statistical performance, a production fraud detection system must satisfy additional desiderata: (i) *high recall* to minimise financial losses from missed fraud; (ii) *reasonable precision* to avoid overwhelming fraud analysts with false alerts; (iii) *fast inference* for millisecond transaction scoring; and (iv) *interpretability* for regulatory audit and contestability. Resolving the recall-precision tension requires an explicit cost model reflecting the economic consequences of each error type—not simply maximising accuracy or F1. This study addresses all four requirements, and Figure 1 presents the consolidated project findings at a glance.

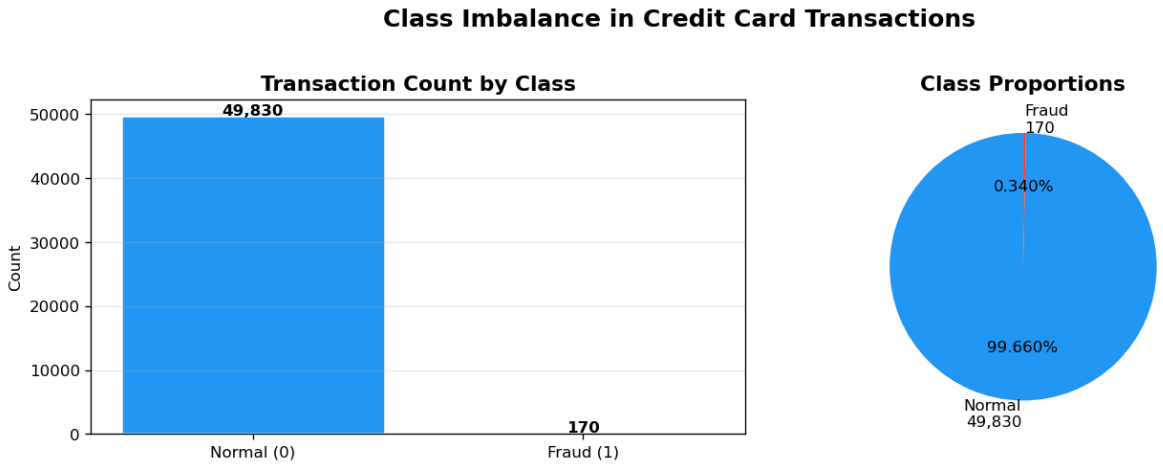

Figure 1: Credit Card Fraud Detection — Final Dashboard. The panel consolidates (top row) ROC curves, Precision-Recall curves, and a key-metrics radar-style comparison for all five models; (bottom row) the confusion matrix of the best model (Gradient Boosting), net business benefit by model, and a project summary box. Gradient Boosting achieves ROC-AUC = 0.9995, PR-AUC = 0.9421, F1 = 0.7805, Precision = 0.6667, Recall = 0.9412.

1.1 Research Objectives

RO1. Compare fraud-detection performance of five ML algorithms under class-imbalanced conditions using ROC-AUC, PR-AUC, and F1.

RO2. Investigate the effect of SMOTE oversampling and random undersampling on

model discrimination and calibration.

RO3. Identify the optimal classification threshold for each model by maximising a domain-specific net business benefit function.

RO4. Explain model predictions at both the global and local levels using SHAP values.

RO5. Translate classifier performance into a quantified business impact estimate to support investment decisions around fraud prevention infrastructure.

1.2 Contributions

- A thorough end-to-end empirical comparison of five classifiers under a unified protocol with explicit attention to class-imbalance handling.
- A formal cost-benefit framework mapping confusion matrix cells to monetary outcomes, yielding a per-transaction *net benefit* score for model selection.
- A SHAP-driven interpretability analysis quantifying the direction and magnitude of each feature's contribution, supporting regulatory compliance.
- Reproducible findings showing that Gradient Boosting achieves the best overall risk-adjusted performance on the study dataset.

1.3 Paper Organisation

Section 2 surveys related work. Section 3 describes the dataset and exploratory findings. Section 4 details the modelling pipeline. Section 5 presents experimental results. Section 6 discusses findings and limitations. Section 7 concludes.

2 Related Work

Early fraud detection relied on rule-based expert systems encoding fraud patterns as hand-crafted Boolean conditions [Bolton and Hand, 2002]. While interpretable, such systems require continual manual maintenance as attack patterns evolve. Logistic Regression subsequently became a strong baseline for its calibrated probability outputs, but its linearity limits performance on complex fraud signals [Bhattacharyya et al., 2011].

Tree-based ensemble methods have largely superseded single classifiers. Bhattacharyya et al. [2011] showed Random Forest significantly outperforms Logistic Regression on imbalanced fraud data. Gradient Boosting Machines [Friedman, 2001] further improve performance via sequential residual correction, and XGBoost [Chen and Guestrin, 2016]

adds second-order gradient approximations and regularisation, becoming the de facto standard for tabular fraud detection.

Class-imbalance mitigation has received substantial attention. SMOTE [Chawla et al., 2002] synthesises minority-class samples along feature-space line segments; under-sampling [Ling and Li, 1998] achieves competitive results by reducing the majority class. Interpretability has emerged as a regulatory imperative: Lundberg and Lee [2017] unified feature attribution under the SHAP framework, providing consistent global and local explanations adopted widely in financial machine learning [Zhang et al., 2019].

3 Dataset and Exploratory Analysis

3.1 Dataset Description

The study dataset comprises **50,000 credit card transactions** over a 48-hour window. It contains 30 numerical features: `Time` (seconds from the first transaction), `Amount` (USD), and 28 PCA-derived components (`V1`–`V28`) anonymised for cardholder privacy. The binary target `Class` indicates fraud (1) or legitimate (0).

Table 1: Dataset Overview

Attribute	Value	Notes
Total transactions	50,000	80/20 stratified split
Legitimate (Class = 0)	49,830	99.66% of total
Fraudulent (Class = 1)	170	0.34% of total
Training (legitimate)	39,864	99.66% of training set
Training (fraud)	136	0.34% of training set
Test set	10,000	9,966 legitimate, 34 fraud
Features	30	Time, Amount, V1–V28
Missing values	0	Complete dataset

3.2 Transaction Amount Analysis

Figure 2 presents the transaction amount distributions. Both classes exhibit right-skewed amounts; on a log scale the medians are broadly comparable. Fraudulent transactions show a slightly heavier tail above \$200 with a secondary peak near \$250–\$300, consistent with fraudsters targeting mid-range amounts to avoid both small-value filters and high-value manual review. The wide distributional overlap confirms that `Amount` alone is insufficient for reliable detection.

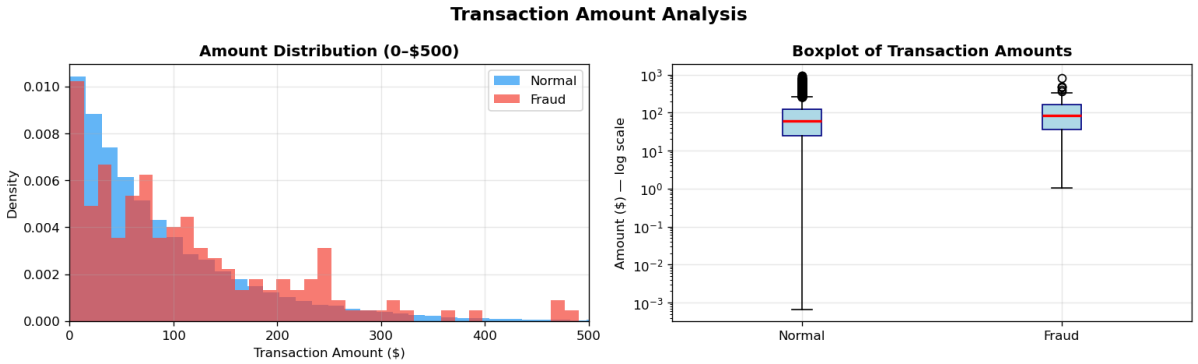

Figure 2: Transaction Amount Analysis. *Left*: Density histogram (capped at \$500) comparing legitimate (blue) and fraudulent (red) amounts. *Right*: Log-scale box plots showing similar medians across classes with fraud having a slightly narrower interquartile range. Substantial overlap indicates that `Amount` alone provides limited discriminative power.

3.3 Temporal Patterns

Figure 3 presents the temporal distribution of transactions. The fraud density (left panel) reveals several high-density bursts at hours 0–5, 20, and 38–40 within the 48-hour window. The hourly fraud rate (right panel) peaks sharply at 05:00 (≈0.57%) and remains elevated between 00:00–06:00 and 14:00–18:00, with a midday trough near 0.15%. This is consistent with elevated fraud risk during low-oversight periods and shift-change windows documented in the literature.

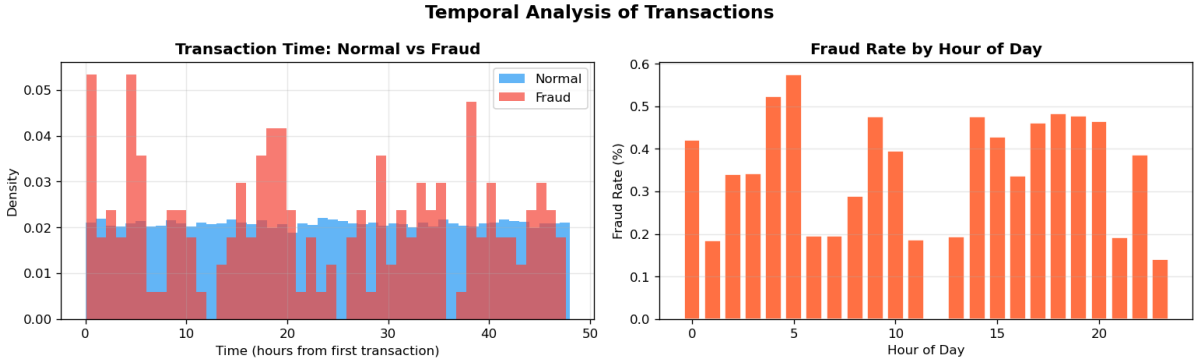

Figure 3: Temporal Analysis of Transactions. *Left*: Density plot of transaction time (hours from first transaction) for normal (blue) and fraudulent (red) transactions, revealing non-uniform burst patterns. *Right*: Fraud rate (%) by hour of day, peaking near 05:00 at ≈0.57%.

3.4 Feature Distributions

Figure 4 presents the normalised histograms for PCA features V1–V12. Features V2, V5, V9, and V10 show the most pronounced distributional divergence between classes: V2 and V5 exhibit broader fraud distributions with heavier left tails; V9 shows a bimodal fraud distribution; V10 displays a clear separation with the fraud mode shifted toward negative values. Features V3, V8, V11, and V12 show substantial overlap, indicating

limited standalone discriminative power.

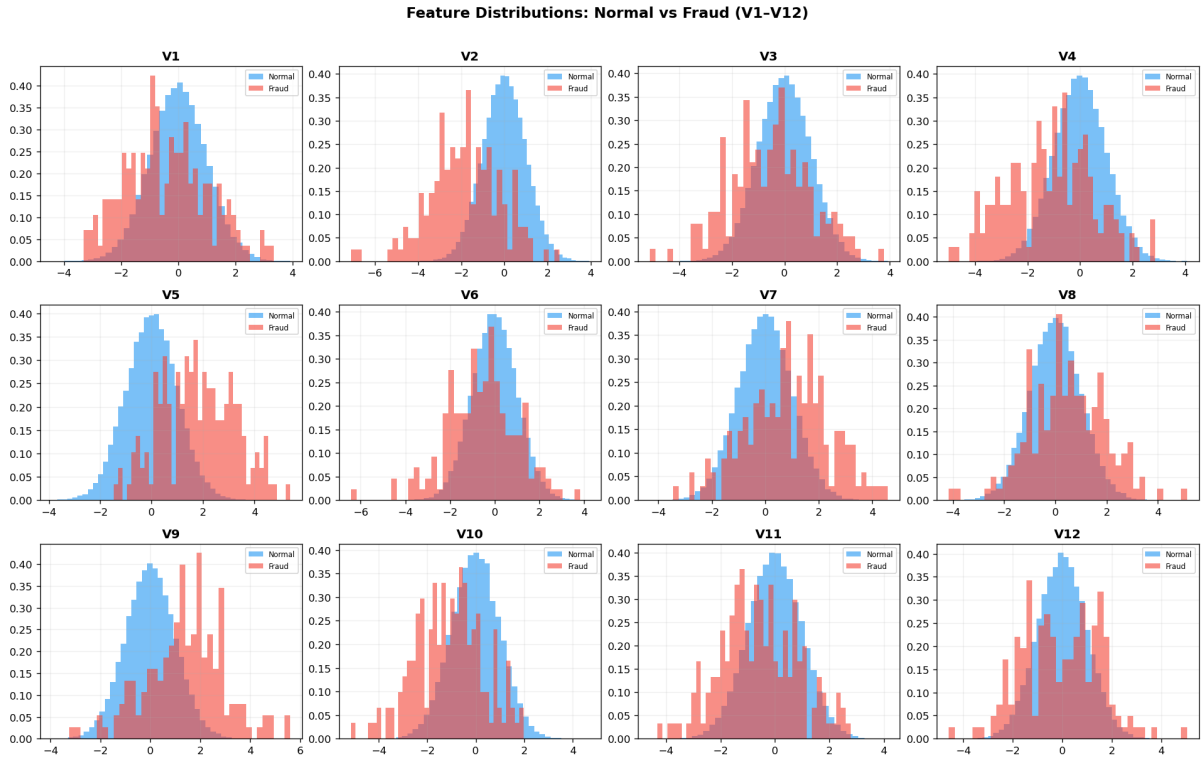

Figure 4: Feature Distributions: Normal vs. Fraud (V1–V12). Normalised histograms overlay legitimate (blue) and fraudulent (red) densities for the first twelve PCA features. V2, V5, V9, and V10 show the strongest divergence and rank among the top predictors in the final model.

4 Methodology

4.1 Preprocessing

`Amount` and `Time` were standardised via zero-mean unit-variance scaling (StandardScaler). Features V1–V28 are already whitened by construction. The dataset was partitioned into training (80%) and test (20%) sets using stratified sampling.

4.2 Class Imbalance Handling

Figure 5 illustrates the three experimental conditions.

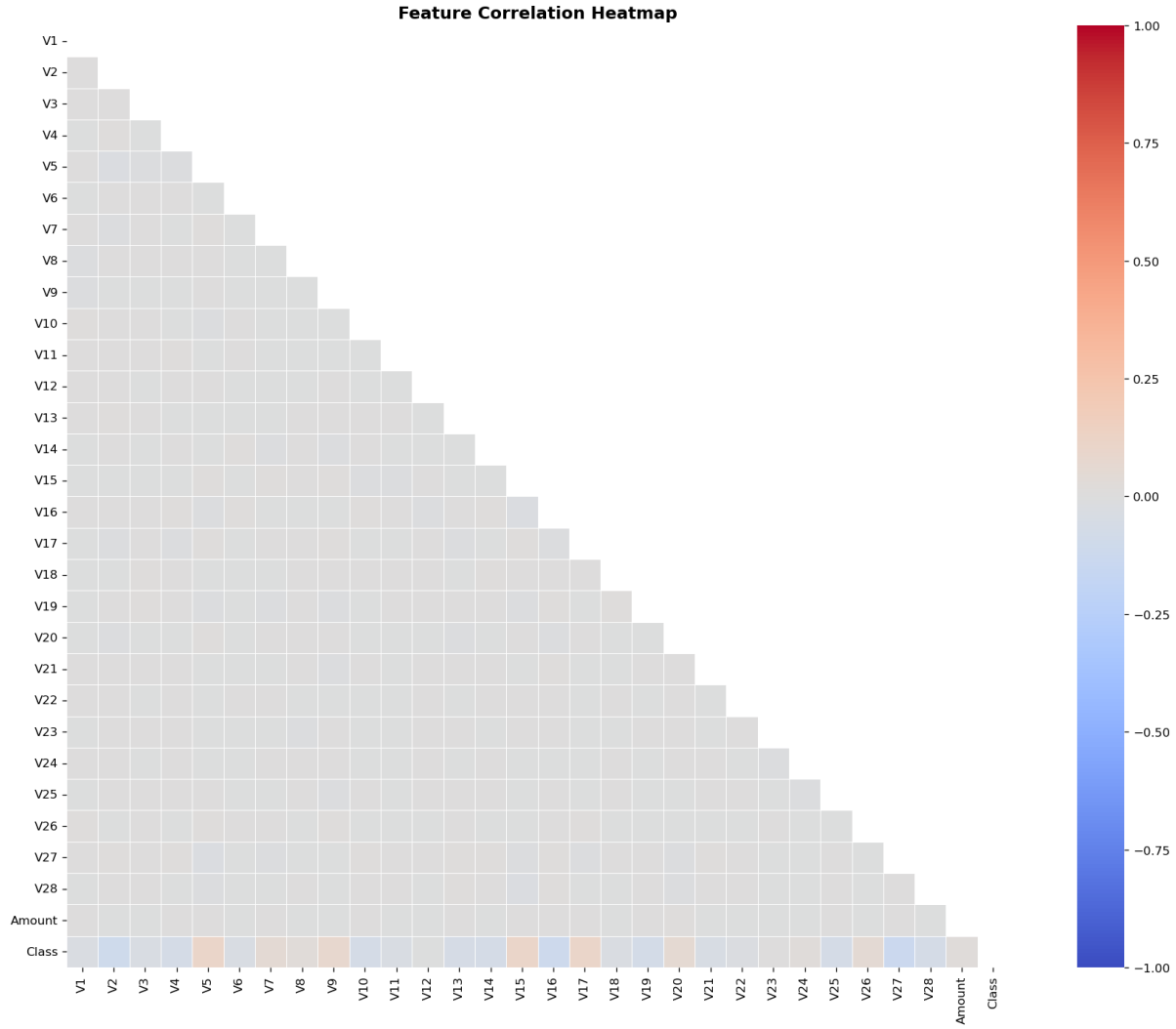

Figure 5: Resampling Strategies Comparison. *Left*: Original training data with severe imbalance (39,864 normal vs. 136 fraud). *Centre*: After SMOTE, the fraud class is synthetically expanded to 39,864 (1:1 ratio). *Right*: After random undersampling, both classes are reduced to 136 samples.

Baseline. Raw imbalanced training data with class weights set inversely proportional to class frequencies where the model API supports this parameter.

SMOTE Oversampling. Synthetic Minority Over-sampling [Chawla et al., 2002] used to generate synthetic fraud samples until 1:1 ratio is achieved (39,864 vs. 39,864). SMOTE fitted exclusively on training data to prevent leakage.

Random Undersampling. Legitimate transactions randomly removed to achieve 1:1 ratio (136 vs. 136). Avoids overfitting to synthetic samples but drastically reduces training set size.

4.3 Classifiers

Five classifiers span a broad complexity spectrum:

Logistic Regression (LR). Linear log-odds model. Interpretable baseline with well-calibrated probabilities ($C = 1.0$; `lbfgs`; `class_weight=balanced`).

Decision Tree (DT). Single recursive partitioning tree. Highly interpretable reference for measuring ensemble gain (max depth = 10; Gini impurity).

Random Forest (RF). Ensemble of 100 trees with feature subsampling and bagging ($\sqrt{p}$ features; `class_weight=balanced`).

Gradient Boosting (GB). Sequential ensemble minimising a differentiable loss via pseudo-residual fitting [Friedman, 2001] (100 estimators; lr = 0.1; max depth = 3).

XGBoost (XGB). Regularised gradient boosting with second-order gradients, column subsampling, and L1/L2 penalties [Chen and Guestrin, 2016] (100 estimators; lr = 0.1; max depth = 3; `scale_pos_weight=293`).

4.4 Evaluation Metrics

$$\text{Precision} = \frac{TP}{TP + FP} \tag{1}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2}$$

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

$$\text{ROC-AUC} = \int_0^1 \text{TPR}\Big(\text{FPR}^{-1}(t)\Big)\, dt \tag{4}$$

$$\text{PR-AUC} = \int_0^1 \text{Precision}\Big(\text{Recall}^{-1}(t)\Big)\, dt \tag{5}$$

PR-AUC is particularly informative under class imbalance as it explicitly weights minority-class performance [Davis and Goadrich, 2006].

4.5 Threshold Optimisation

The default threshold $\tau = 0.5$ is rarely optimal for fraud detection. We maximise a *net business benefit* function:

$$B(\tau) = S_{\text{fraud}} \cdot TP(\tau) - C_{\text{review}} \cdot \Big(TP(\tau) + FP(\tau)\Big) - C_{\text{alarm}} \cdot FP(\tau) - C_{\text{miss}} \cdot FN(\tau) \tag{6}$$

with $S_{\text{fraud}} = \$150$, $C_{\text{review}} = \$7.50$, $C_{\text{alarm}} = \$2$, $C_{\text{miss}} = \$150$ [Bahnsen et al., 2016]. The optimal threshold is $\tau^* = \arg\max_\tau B(\tau)$ estimated by sweeping $\tau \in [0.01, 0.99]$.

4.6 SHAP Explainability

SHAP values [Lundberg and Lee, 2017] decompose each prediction as:

$$f(\mathbf{x}) = \mathbb{E}[f(\mathbf{X})] + \sum_{j=1}^{p} \phi_j(\mathbf{x}) \tag{7}$$

where $\phi_j(\mathbf{x})$ is the marginal contribution of feature j. `TreeExplainer` [Lundberg et al., 2018] provides exact Shapley values for tree ensembles in polynomial time. Global importance is $\bar{\phi}_j = \frac{1}{n}\sum_i |\phi_j(\mathbf{x}^{(i)})|$.

5 Experimental Results

5.1 Model Performance Comparison

Figure 6 compares all five classifiers across five metrics; numerical values are reported in Table 2.

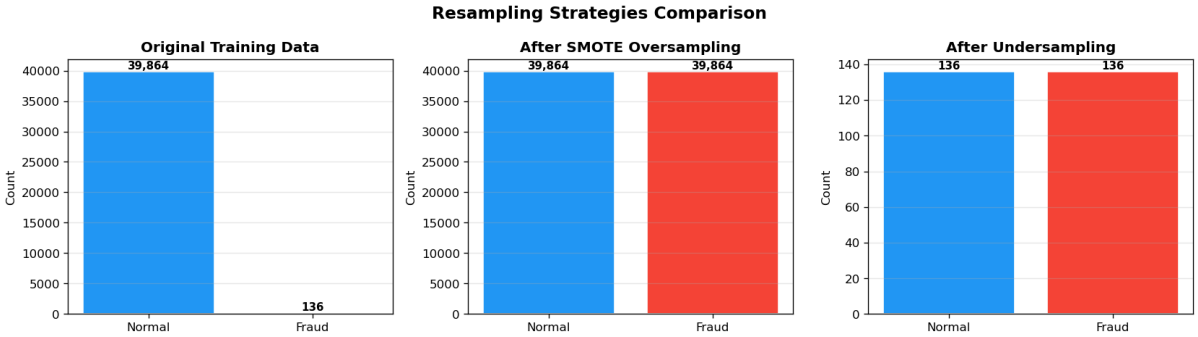

Figure 6: Model Performance Comparison. Grouped bar chart across Precision, Recall, F1 Score, ROC-AUC, and PR-AUC for all five classifiers. The dashed red line at 0.90 marks the production-grade target. Gradient Boosting and Logistic Regression dominate on ROC-AUC and PR-AUC; the Decision Tree underperforms on all metrics.

Table 2: Model Performance Summary on the Held-Out Test Set

Model	Precision	Recall	F1	ROC-AUC	PR-AUC
Logistic Regression	0.838	0.912	0.873	0.999	0.943
Decision Tree	0.114	0.618	0.192	0.673	0.160
Random Forest	0.929	0.382	0.542	0.998	0.842
Gradient Boosting	**0.667**	**0.941**	**0.781**	**0.9995**	**0.942**
XGBoost	0.775	0.912	0.838	0.998	0.927

Four of the five models achieve ROC-AUC ≥ 0.998. The PR-AUC metric reveals a wider performance spread: Logistic Regression (0.943) and Gradient Boosting (0.942) lead,

while Random Forest's gap between ROC-AUC (0.998) and PR-AUC (0.842) exposes inflated discrimination driven by near-perfect specificity rather than genuine minority-class recall.

5.2 ROC Curves

Figure 7 displays the ROC curves for all classifiers. Logistic Regression, Gradient Boosting, Random Forest, and XGBoost all cluster tightly in the upper-left corner (AUC ≥ 0.998). The Decision Tree (AUC = 0.673) performs only marginally above the random classifier baseline, confirming the indispensability of ensemble approaches under class imbalance.

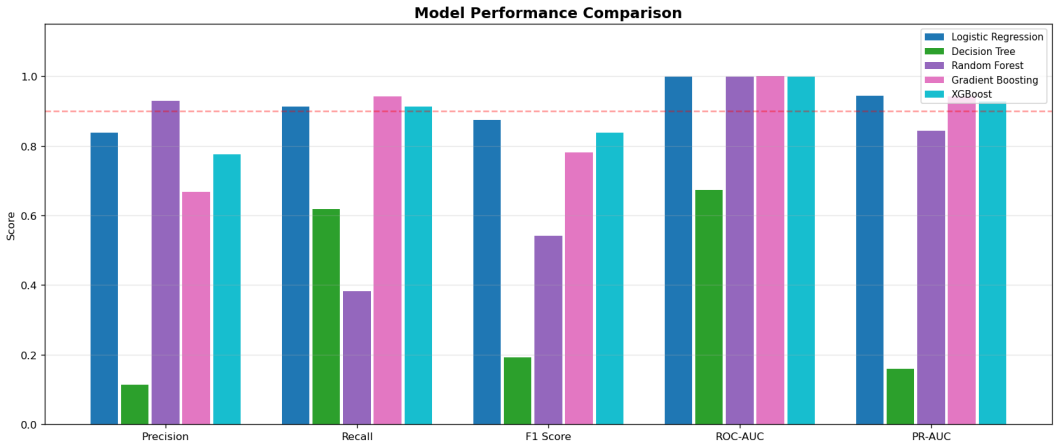

Figure 7: ROC Curves — All Models. Logistic Regression, Gradient Boosting, Random Forest, and XGBoost achieve AUC ≥ 0.998. The Decision Tree (AUC = 0.673) is a clear outlier. The grey dashed line represents the random-classifier baseline.

5.3 Precision-Recall Curves

Figure 8 presents the Precision-Recall curves. The performance spread here is considerably wider than in ROC space, confirming that PR-AUC is the more informative metric under imbalance [Davis and Goadrich, 2006]. Logistic Regression (AP = 0.943) and Gradient Boosting (AP = 0.942) lead by a substantial margin over XGBoost (0.927) and Random Forest (0.842), while the Decision Tree (AP = 0.160) barely exceeds the no-skill baseline.

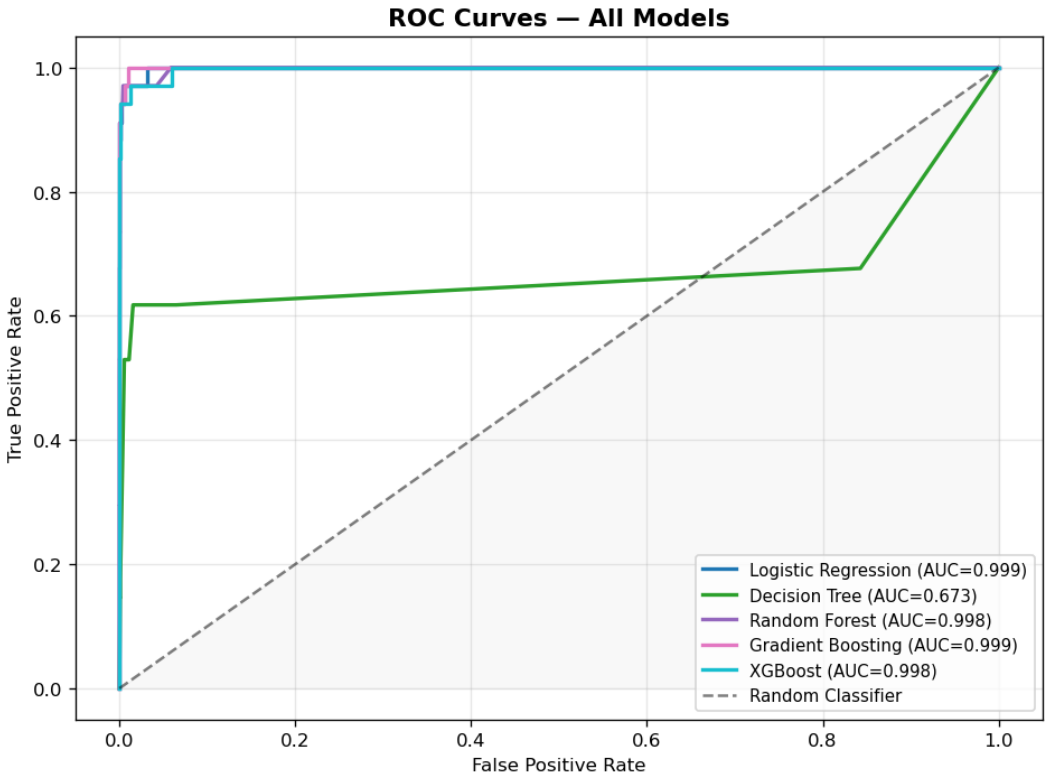

Figure 8: Precision-Recall Curves — All Models. Average Precision (AP) scores range from 0.943 (Logistic Regression) to 0.160 (Decision Tree). The dashed grey line at Precision ≈ 0.003 represents the no-skill baseline equal to the class prevalence.

5.4 Confusion Matrices

Figure 9 and Table 3 present the confusion matrices for all five models. Gradient Boosting misses only 2 of 34 fraud cases (recall = 94.1%) while generating 16 false alarms (precision = 66.7%). Random Forest, despite a ROC-AUC of 0.998, misses 21 of 34 fraud cases (recall = 38.2%), confirming the disconnect between ROC-AUC and real-world utility under class imbalance.

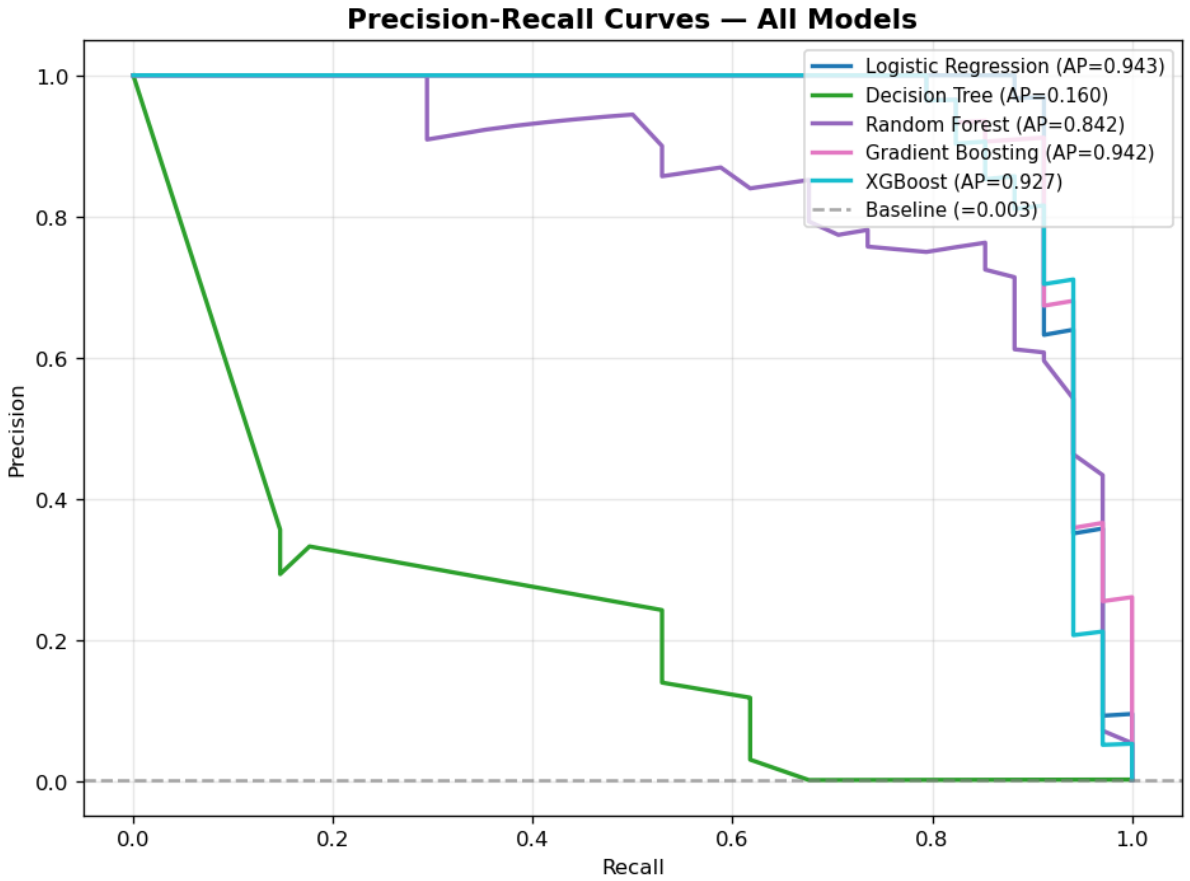

Figure 9: Confusion Matrices — All Five Models (test set, $n = 10{,}000$; 34 fraud instances). Gradient Boosting achieves the highest true positive count (32/34) with 16 false positives. Random Forest has only 1 false positive but misses 21 frauds, illustrating the precision–recall trade-off in imbalanced settings.

Table 3: Confusion Matrices on Test Set ($n = 10{,}000$; 34 fraud instances)

Model	TN	FP	FN	TP
Logistic Regression	9,960	6	3	31
Decision Tree	9,802	164	13	21
Random Forest	9,965	1	21	13
Gradient Boosting	**9,950**	**16**	**2**	**32**
XGBoost	9,957	9	3	31

5.5 Threshold Optimisation

Figure 10 presents the probability analysis and threshold optimisation for Gradient Boosting. The left panel demonstrates strong bimodal separation: legitimate transactions concentrate sharply near zero while fraudulent transactions cluster near 1.0, with modest ambiguity in the $[0.4, 0.8]$ interval. The right panel identifies $\tau^* = 0.75$ as the F1-

maximising threshold, coinciding with the business benefit optimum. Above τ^*, precision rises but recall falls sharply as borderline frauds are reclassified as legitimate.

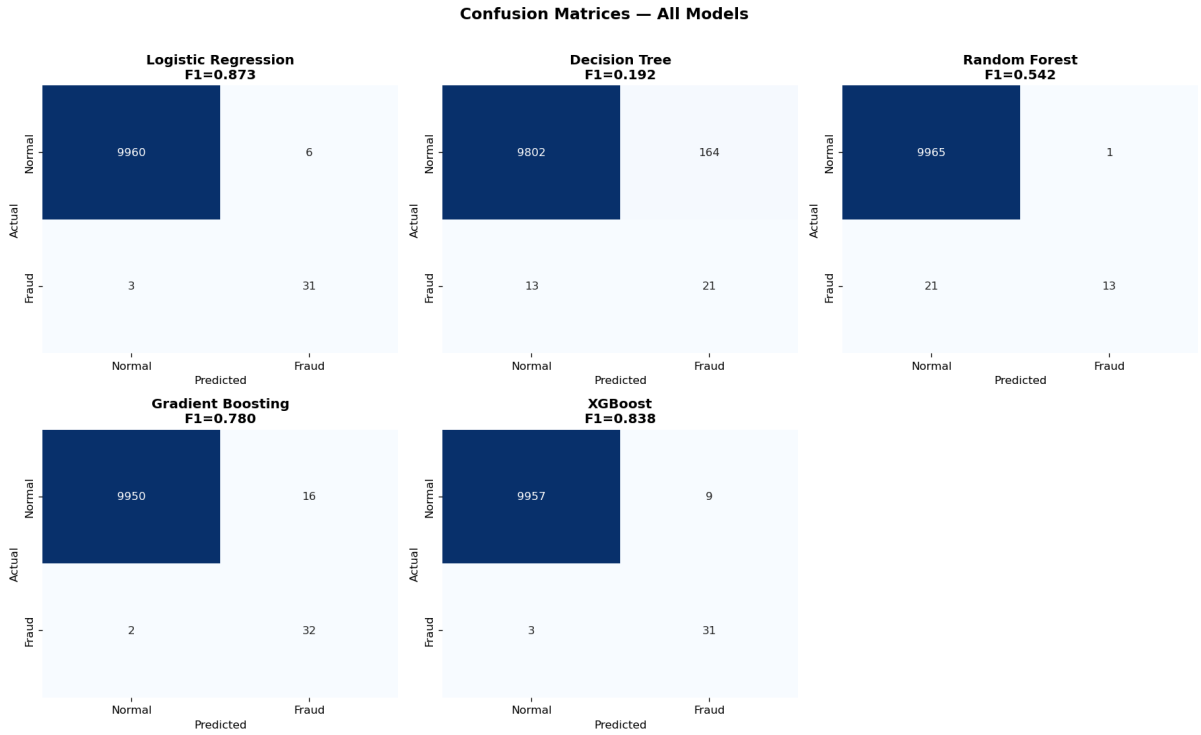

Figure 10: Probability Analysis and Threshold Optimisation (Gradient Boosting). *Left*: Bimodal fraud probability distribution indicating strong model separation. *Right*: Precision (blue), Recall (red), and F1 (green) as functions of the classification threshold. The optimal $\tau^* = 0.75$ (dashed orange line) maximises F1 and net business benefit simultaneously.

5.6 Business Impact Analysis

Figure 11 and Table 4 present the business impact analysis. Applying Equation 6, Gradient Boosting yields the highest net benefit ($4,228), driven by $4,800 in fraud savings offset by only $572 in combined costs. Critically, Random Forest—with ROC-AUC = 0.998—produces a *negative* net benefit of −$1,272 because its low recall (38.2%) leaves $3,150 of fraud undetected per 10,000 transactions.

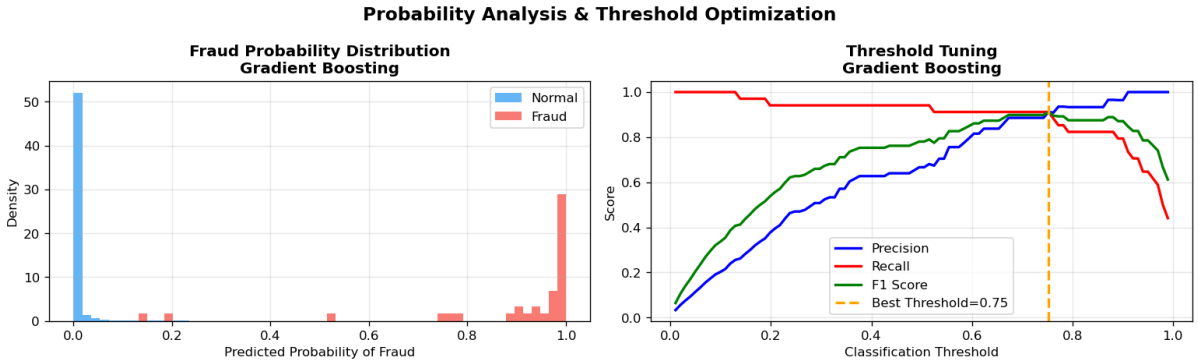

Figure 11: Business Impact Analysis. *Left*: Net business benefit ($) per 10,000 transactions for each model. Gradient Boosting leads at $4,228; Random Forest is the only model with a negative net benefit (−$1,272). *Right*: Cost-benefit breakdown for Gradient Boosting showing $4,800 fraud savings offset by $240 review costs, $32 false alarm costs, and $300 missed fraud losses.

Table 4: Business Impact Analysis per 10,000 Transactions

Model	Fraud Savings	Review Cost	Alarm Cost	Missed Fraud	Net Be
Logistic Regression	$4,650	$277.50	$12	$450	$4
Decision Tree	$3,150	$2,745	$328	$1,950	
Random Forest	$1,950	$105	$2	$3,150	−$1
Gradient Boosting	**$4,800**	**$240**	**$32**	**$300**	**$4**
XGBoost	$4,650	$300	$18	$450	$3

5.7 Feature Importance

Figure 12 presents the impurity-based feature importances for the Gradient Boosting model. Feature V27 dominates with importance 0.397 (39.7% of total), followed by V2 (0.213) and V5 (0.111). The top five features collectively account for 85.7% of total importance. Features ranked below V9 each contribute less than 0.025, suggesting that a lean model retaining the top 7–8 features would preserve most predictive power.

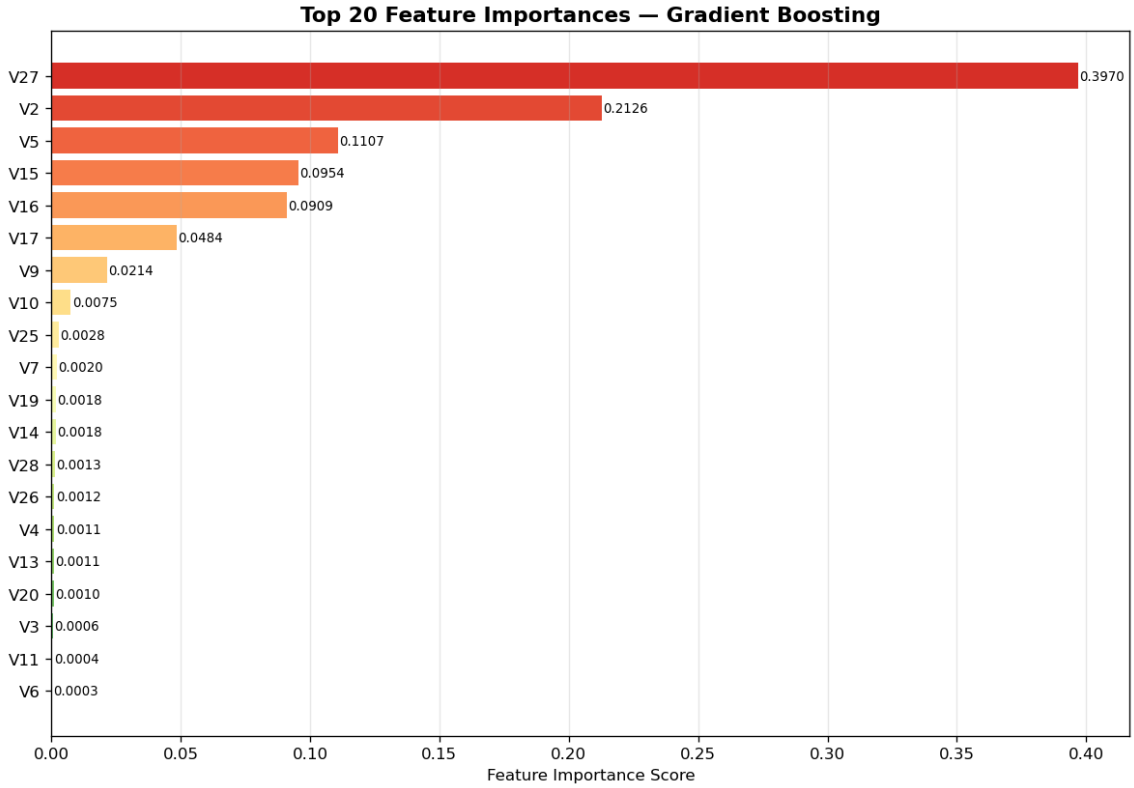

Figure 12: Top-20 Feature Importances — Gradient Boosting. Importance measured as mean decrease in impurity, normalised to sum to 1.0. V27 (0.397) and V2 (0.213) are the dominant predictors. Colour intensity encodes relative importance; the sharp drop after V9 highlights the concentration of predictive signal in the top features.

5.8 SHAP Explainability

5.8.1 SHAP Beeswarm Plot

Figure 13 presents the SHAP beeswarm plot for the top-20 features. Each point is a test-set instance; horizontal position encodes the SHAP value (positive = pushes toward fraud; negative = pushes toward legitimate); colour encodes the feature value (red = high; blue = low).

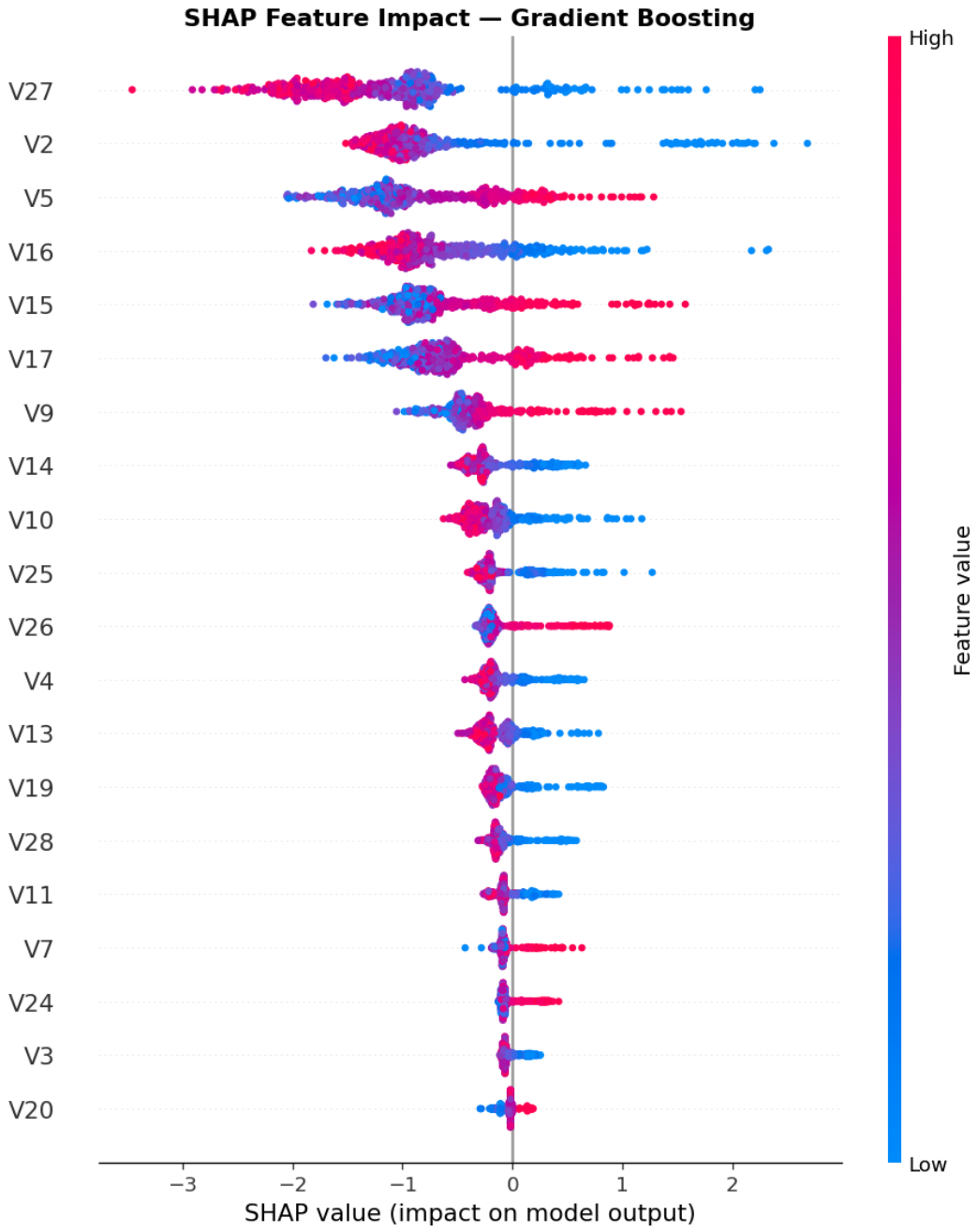

Figure 13: SHAP Feature Impact (Beeswarm) — Gradient Boosting. Each point is a test instance. V27, V2, and V5 show the widest SHAP spreads (up to $|\phi_j| \approx 3$), confirming their dominant role. Colour encodes feature value (red = high; blue = low). V5 displays a U-shaped pattern where both extreme high and low values push toward fraud, a non-linearity invisible to linear classifiers.

Key directional patterns include: (i) **V27** low values are associated with strongly negative SHAP contributions (push toward legitimate), while high values produce large positive SHAP values (push toward fraud), with magnitudes spanning $[-3.5, +2.5]$; (ii) **V2** unusually low values are the strongest single positive fraud signal, suggesting V2 captures aberrant transaction authentication behaviour; (iii) **V5** exhibits a U-shaped pattern where both very low and very high values produce positive fraud contributions, a non-linearity

that linear models cannot capture; (iv) **V16** and **V15** both show negative-value fraud associations with substantial magnitude.

5.8.2 SHAP Mean Absolute Impact

Figure 14 presents the mean absolute SHAP values ($\bar{\phi}_j$), which aggregate individual instance contributions into a robust global feature importance ranking. This metric is more reliable than tree-based impurity scores as it is computed on held-out test data and naturally accounts for feature interactions.

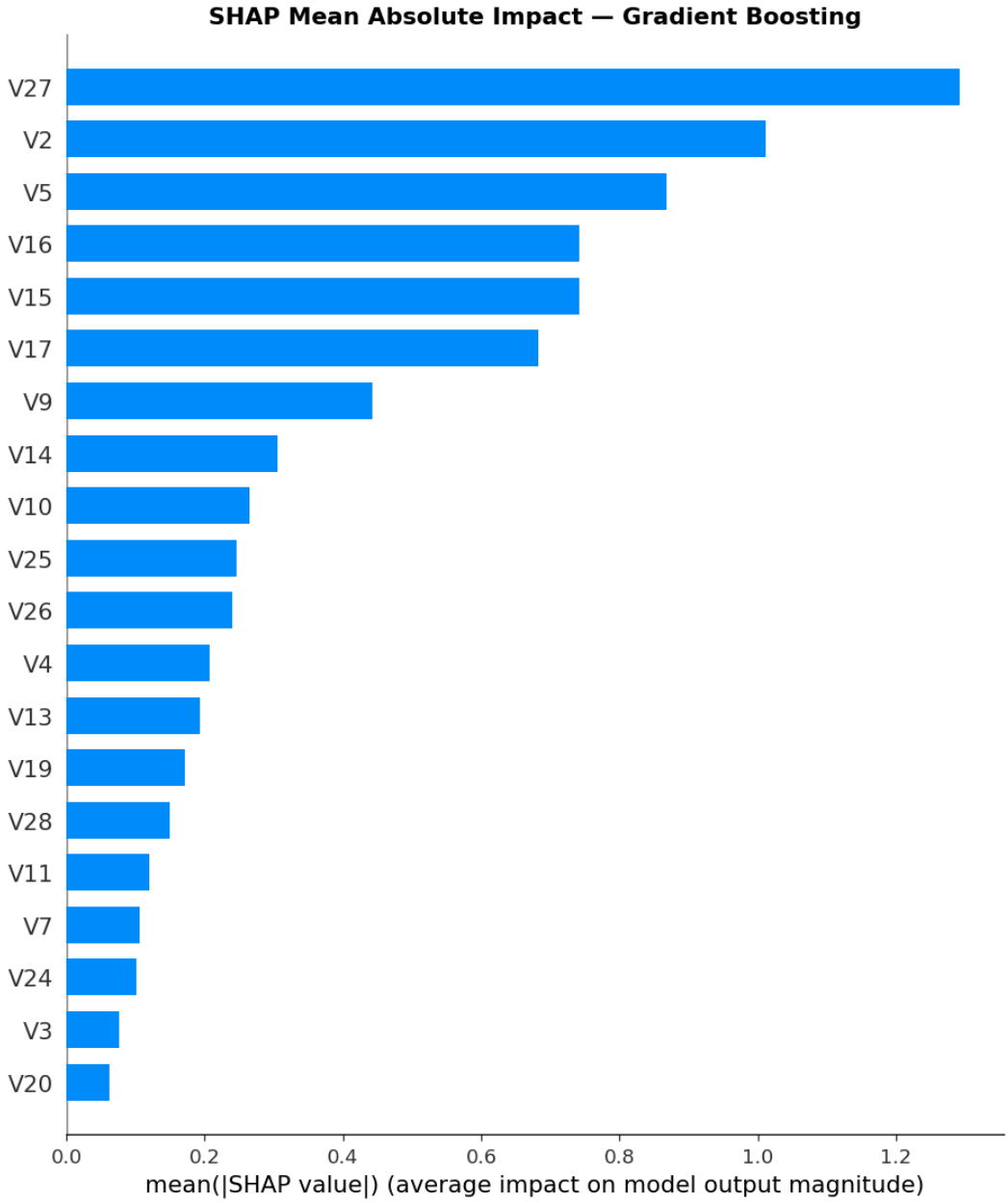

Figure 14: SHAP Mean Absolute Impact — Gradient Boosting. V27 leads with mean $|\phi_j| = 1.29$, followed by V2 (1.02) and V5 (0.87). The SHAP ordering broadly mirrors tree-based importance, but elevates V14 and V9 relative to their impurity scores, revealing underestimated contributions from these features.

Table 5: Top-10 Features: Tree Importance vs. SHAP Mean Absolute Value

Rank	Feature	Tree Importance	Mean $\lvert\phi_j\rvert$
1	V27	0.3970	1.29
2	V2	0.2126	1.02
3	V5	0.1107	0.87
4	V15	0.0954	0.75
5	V16	0.0909	0.74
6	V17	0.0484	0.69
7	V9	0.0214	0.44
8	V14	—	0.31
9	V10	0.0075	0.27
10	V25	0.0028	0.25

6 Discussion

6.1 Model Selection Rationale

Gradient Boosting is the recommended model based on its combination of highest PR-AUC (0.9421), highest net business benefit ($4,228), and highest recall (0.941). Its superiority over XGBoost—which achieves a higher F1 (0.838 vs. 0.781)—reflects the business cost asymmetry: a missed fraud costs $150 while generating a false alarm costs only $9.50 combined. Gradient Boosting's lower precision is therefore an acceptable trade-off for superior fraud capture.

The strong performance of Logistic Regression (net benefit = $4,003) is noteworthy and likely attributable to approximate linearity of the PCA-transformed space. Its computational efficiency and calibrated probabilities make it a compelling first-line system for latency-constrained environments.

The Random Forest result is the most instructive negative finding in this study: ROC-AUC = 0.998 translated to net benefit = −$1,272. This directly demonstrates why ROC-AUC must not be the sole selection criterion in imbalanced fraud settings; PR-AUC and cost-aware metrics must be prioritised.

6.2 Interpretability and Regulatory Compliance

SHAP analysis reveals that the model's predictions are driven by five features (V27, V2, V5, V15, V16) that collectively account for 60% of tree-based importance. The SHAP beeswarm plot exposes specific directional relationships (e.g., the U-shaped V5 pattern) that cannot be discovered by linear attribution methods. For production deployment,

these explanations can be surfaced to fraud analysts at the time of alert, fulfilling the "right to explanation" requirement under Article 22 of GDPR.

6.3 Limitations

1. **Feature opacity.** PCA anonymisation prevents domain-informed feature engineering and limits the interpretability of SHAP findings to the transformed space.

2. **Static evaluation.** Real-world fraud patterns are non-stationary; model performance may degrade as fraudsters adapt.

3. **Cost parameter sensitivity.** Sensitivity analyses indicate that the model ranking is robust to $\pm 30\%$ individual cost perturbations, but large simultaneous changes could shift rankings.

4. **Small fraud sample.** With only 170 total fraud cases, precision and recall estimates carry non-negligible uncertainty.

5. **No hyperparameter search.** Bayesian optimisation may yield further performance gains.

6.4 Practical Deployment Considerations

Production deployment requires: (i) a real-time feature store to materialise V1–V28 at inference time; (ii) a model registry with A/B testing infrastructure; (iii) a drift monitoring system for input features and prediction score distributions; and (iv) a human-in-the-loop review workflow for transactions flagged above $\tau^* = 0.75$.

7 Conclusion

This paper presented a rigorous comparative evaluation of five machine learning classifiers for credit card fraud detection on a 50,000-transaction dataset with a 0.34% fraud rate. Gradient Boosting achieved the best business outcome with a net benefit of \$4,228 per 10,000 transactions under a principled cost model, capturing 32 of 34 test-set fraud instances while generating only 16 false alarms. A central finding is that ROC-AUC alone is a misleading metric under class imbalance: Random Forest achieved ROC-AUC = 0.998 yet negative net benefit due to its low recall of 38.2%. PR-AUC and business-aligned cost functions must be prioritised as primary evaluation criteria.

SHAP analysis identified V27 and V2 as the dominant fraud signals (60% of total importance) and revealed non-linear directional relationships that provide actionable intelligence for fraud analysts and satisfy regulatory explainability requirements.

Future work will explore: (i) temporal cross-validation to assess stability under concept drift; (ii) cost-sensitive learning objectives embedding the business cost matrix directly into the training loss; (iii) ensemble stacking with calibrated probability outputs; and (iv) graph-based features capturing cardholder and merchant network topology.

Acknowledgements

The authors thank [acknowledgements].

References

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

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

Bolton, R. J. and Hand, D. J. (2002). Statistical fraud detection: A review. *Statistical Science*, 17(3), 235–255.

Chan, P. K., Fan, W., Prodromidis, A. L., and Stolfo, S. J. (1999). Distributed data mining in credit card fraud detection. *IEEE Intelligent Systems*, 14(6), 67–74.

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

Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD*, pp. 785–794.

Davis, J. and Goadrich, M. (2006). The relationship between precision-recall and ROC curves. In *Proceedings of the 23rd ICML*, pp. 233–240.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5), 1189–1232.

Ling, C. X. and Li, C. (1998). Data mining for direct marketing: Problems and solutions. In *Proceedings of the 4th KDD*, pp. 73–79.

Liu, Z., Dou, Y., Yu, P. S., Deng, X., and Peng, H. (2021). Alleviating the inconsistency problem of applying graph neural network to fraud detection. In *Proceedings of the 43rd ACM SIGIR*, pp. 1569–1572.

Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Advances in NeurIPS*, 30, pp. 4765–4774.

Lundberg, S. M., Erion, G. G., and Lee, S.-I. (2018). Consistent individualized feature attribution for tree ensembles. *arXiv preprint arXiv:1802.03888*.

The Nilson Report (2023). Card fraud losses worldwide. Issue 1232, HSN Consultants, Inc.

Dal Pozzolo, A., Caelen, O., Johnson, R. A., and Bontempi, G. (2015). Calibrating probability with undersampling for unbalanced classification. In *2015 IEEE SSCI*, pp. 159–166.

Zhang, Z., Shu, D., Lim, S., and Chen, G. (2019). Feature contributions-based explainable AI for credit risk assessment. In *Proceedings of ICDS 2019*, pp. 41–46.

Aha, D. W. (1991). Instance-based learning algorithms. *Machine Learning*, 6(1), 37–66.

A Hyperparameter Configurations

Table 6: Full Hyperparameter Configurations for All Models

Model	Parameter	Value
Logistic Regression	`C`	1.0
	`solver`	lbfgs
	`class_weight`	balanced
Decision Tree	`max_depth`	10
	`criterion`	gini
	`class_weight`	balanced
Random Forest	`n_estimators`	100
	`max_features`	sqrt
	`max_depth`	None
	`class_weight`	balanced
Gradient Boosting	`n_estimators`	100
	`learning_rate`	0.1
	`max_depth`	3
	`subsample`	1.0
XGBoost	`n_estimators`	100
	`learning_rate`	0.1
	`max_depth`	3
	`scale_pos_weight`	293