Journal of Science, Innovation & Social Impact

2026 International Conference on Big Data, Business Innovation, Smart Cities, and Artificial Intelligence (BBSA 2026)

Article

A Comparative Evaluation of Class Imbalance Handling Techniques for Credit Card Fraud Detection

Zijie Chen [1,*] and Pengyuan Xiao [2]

[1] Computer Engineering, University of Toronto Master, Toronto, Canada
[2] Computer Science, Zhejiang University, Hangzhou, China
* Correspondence: Zijie Chen, Computer Engineering, University of Toronto Master, Toronto, Canada

Abstract: Credit card fraud detection presents a challenging classification task due to extreme class imbalance, where fraudulent transactions constitute less than 1% of all observations. Selecting an appropriate imbalance handling technique is critical, yet the comparative performance of these techniques under varying imbalance severities remains insufficiently understood. This study conducts a systematic empirical evaluation of nine class imbalance handling techniques across two publicly available fraud detection datasets exhibiting different imbalance ratios (578:1 and 90:1). The techniques evaluated span data-level resampling (SMOTE, ADASYN, Borderline-SMOTE, SMOTE combined with Edited Nearest Neighbors, and random undersampling), algorithm-level cost-sensitive learning (class weighting), and ensemble-based approaches (EasyEnsemble, RUSBoost, and Balanced Random Forest). Each technique is paired with four base classifiers---Logistic Regression, Random Forest, XGBoost, and LightGBM---and assessed using five evaluation metrics: AUC-ROC, PR-AUC, F1-score, Matthews Correlation Coefficient, and recall. Results indicate that ensemble-based methods, particularly EasyEnsemble, achieve the most consistent improvements across both datasets. Hybrid resampling via SMOTE with Edited Nearest Neighbors produces comparable gains among data-level methods. A notable finding is that standard SMOTE, while improving AUC-ROC and F1-score, can reduce PR-AUC relative to the untreated baseline under severe imbalance. Cost-sensitive class weighting emerges as a computationally efficient alternative that preserves strong PR-AUC performance. These findings provide practical guidance for practitioners selecting imbalance handling strategies in fraud detection applications.

Keywords: class imbalance; fraud detection; resampling techniques; ensemble learning

Received: 03 March 2026
Revised: 17 April 2026
Accepted: 29 April 2026
Published: 06 May 2026

1. Introduction

1.1. Background and Motivation

Financial fraud imposes substantial economic losses worldwide, with card-not-present fraud alone accounting for billions of dollars annually. Machine learning has become a primary tool for automated fraud detection, enabling financial institutions to classify transactions in real time. A persistent obstacle in this domain is the severe class imbalance inherent in transaction data: legitimate transactions vastly outnumber fraudulent ones, with fraud rates often below 0.2% [1]. Standard classifiers trained on such skewed distributions tend to optimize aggregate accuracy by predicting the majority class, resulting in unacceptably low detection rates for the minority fraud class.

A wide range of techniques has been developed to address class imbalance, including data-level resampling, algorithm-level cost-sensitive modifications, and ensemble strategies that embed balancing mechanisms within iterative learning procedures. While each category has demonstrated effectiveness in isolated studies, the comparative

performance of these techniques under the specific conditions of fraud detection---extreme imbalance ratios, anonymized features, and temporal transaction patterns---has not been comprehensively evaluated in a unified experimental setting [2]. The choice of evaluation metric further complicates comparisons, as Precision-Recall curves and their corresponding area (PR-AUC) provide fundamentally different information than ROC-based metrics under severe skew [3]. This gap motivates the present study, which aims to provide a controlled comparison isolating the effect of the balancing strategy from confounding factors.

1.2. Research Scope and Contributions

1.2.1. Research Questions

This study addresses three specific research questions. The first concerns which category of imbalance handling technique---data-level resampling, cost-sensitive weighting, or ensemble-based balancing---yields the strongest fraud detection performance when paired with gradient-boosted tree classifiers. The second investigates whether the relative ranking of techniques changes as the imbalance ratio shifts from moderate (90:1) to severe (578:1). The third examines the extent to which evaluation metric selection influences the apparent superiority of a given technique, with particular attention to the divergence between AUC-ROC and PR-AUC.

1.2.2. Paper Organization

The remainder of this paper is organized as follows. Section 2 reviews related work on class imbalance learning and fraud detection. Section 3 describes the experimental setup, including datasets, techniques, classifiers, and evaluation protocol. Section 4 presents results and cross-dataset analyses. Section 5 discusses practical implications, study limitations, and directions for future research.

2. Related Work

2.1. Class Imbalance Handling Techniques

2.1.1. Data-Level Approaches

Data-level methods modify the training set distribution to reduce the imbalance ratio before classifier training. The Synthetic Minority Over-sampling Technique (SMOTE) generates synthetic minority instances by interpolating between existing minority samples and their k-nearest neighbors [4]. Recognizing that SMOTE treats all minority instances equally, the Adaptive Synthetic Sampling approach (ADASYN) assigns higher sampling weights to minority instances surrounded by more majority-class neighbors, concentrating synthetic generation in regions that are harder to learn [5]. Borderline-SMOTE restricts oversampling to minority instances located near the decision boundary, avoiding unnecessary synthesis in safe interior regions where the minority class is already well-separated [6]. Hybrid strategies combine oversampling with post-hoc cleaning; Batista et al. demonstrated that applying Edited Nearest Neighbors or Tomek Links after SMOTE removes noisy synthetic samples from overlapping boundary regions, yielding consistent improvements over pure SMOTE across thirteen datasets [7]. Random undersampling offers the simplest alternative by randomly discarding majority-class instances, though it risks eliminating informative samples and increasing classifier variance.

2.1.2. Algorithm-Level and Ensemble Approaches

Algorithm-level methods modify the learning objective to penalize minority-class misclassification more heavily. Elkan established the theoretical conditions under which cost matrices produce coherent classification boundaries and proved that adjusting class proportions in training data is equivalent to threshold adjustment under certain conditions [8]. MetaCost, proposed by Domingos, wraps any classifier in a cost-sensitive procedure by relabeling training examples according to estimated expected misclassification costs derived from bagged probability estimates [9]. Ensemble-based approaches represent a third category that integrates resampling within iterative

ensemble construction, creating diverse base learners trained on balanced subsets. These methods have attracted growing attention due to their ability to combine the variance reduction benefits of ensembles with the distributional advantages of resampling, and empirical comparisons have shown them to be among the strongest approaches for severely imbalanced tabular data.

2.2. Machine Learning for Fraud Detection

Fraud detection has become a prominent application domain for imbalanced classification research, driven by the public availability of benchmark datasets and the practical importance of the problem. The research landscape encompasses both traditional tabular approaches using engineered transaction features and graph-based methods that exploit relational structure among accounts and transactions. Across both paradigms, the consensus is that standard classification without imbalance handling produces classifiers with high accuracy but critically poor minority-class recall---an outcome that renders such classifiers impractical for real-world deployment. Recent work has increasingly emphasized the need for evaluation protocols that go beyond AUC-ROC, advocating for PR-AUC and rank-based metrics that better reflect operational performance in production fraud detection pipelines. The present study contributes to this domain by providing a controlled comparison of imbalance handling techniques under standardized experimental conditions.

3. Experimental Setup

3.1. Datasets

Two publicly available fraud detection datasets with distinct imbalance characteristics are used in this study. Table 1 summarizes their key properties.

Table 1. Dataset Characteristics

Property	ULB Credit Card	BAF (Base Variant)
Source	ULB/Worldline (Kaggle)	Feedzai (NeurIPS 2022)
Total samples	284,807	1,000,000
Features	30 (28 PCA + Time + Amount)	30 (numerical + categorical)
Fraud samples	492 (0.172%)	11,000 (1.10%)
Non-fraud samples	284,315 (99.828%)	989,000 (98.90%)
Imbalance ratio	578:1	90:1
Feature types	All numerical	Mixed
Time span	2 days	8 months
Data nature	Real (PCA-anonymized)	Synthetic (CTGAN)

The ULB Credit Card Fraud Detection dataset contains 284,807 transactions recorded over two days from European cardholders, with 492 fraud cases representing a 0.172% positive rate. All original features have been transformed via Principal Component Analysis into 28 anonymous components (V1--V28), with only the Time and Amount fields retained in their original form. This dataset provides a benchmark for evaluating techniques under severe imbalance (578:1) and has been widely adopted in prior studies on class imbalance and fraud detection.

The Bank Account Fraud (BAF) dataset suite, introduced at NeurIPS 2022, comprises six synthetic dataset variants generated using Conditional Tabular GANs with differential privacy guarantees [10]. This study uses the Base variant, which contains 1,000,000 instances with an approximately 1.10% fraud rate (imbalance ratio of 90:1). The BAF dataset provides mixed feature types---both numerical and categorical---including protected attributes such as age and employment status. Its moderate imbalance ratio complements the extreme skew of the ULB dataset, enabling analysis of how technique effectiveness varies with imbalance severity.

3.2. Imbalance Handling Techniques

3.2.1. Resampling Methods

Five resampling strategies are evaluated. SMOTE generates synthetic minority samples via linear interpolation between minority instances and their five nearest neighbors (k = 5), targeting a 1:1 class ratio. ADASYN adaptively assigns generation density proportional to the local majority-class ratio, using the same target ratio. Borderline-SMOTE (B-SMOTE) restricts synthesis to minority instances classified as borderline---those with between half and all majority-class neighbors within k = 5. The hybrid method SMOTE+ENN first applies SMOTE to achieve balance, then removes any instance---majority or synthetic minority---whose class label disagrees with the majority vote of its three nearest neighbors, effectively cleaning noisy boundary regions. Random Undersampling (RUS) randomly discards majority-class instances to achieve a 1:1 ratio. All resampling methods are implemented using the imbalanced-learn library (version 0.11) with default parameters unless noted.

3.2.2. Cost-Sensitive and Hybrid Methods

Class weighting (CW) modifies the loss function by assigning a weight inversely proportional to class frequency, increasing the penalty for minority-class misclassification without altering the training data distribution. For tree-based classifiers, this is implemented via the scaleposweight parameter in XGBoost and the is_unbalance flag in LightGBM [11, 12]. Three ensemble-based methods are also evaluated. EasyEnsemble independently samples multiple balanced subsets from the majority class, trains an AdaBoost classifier on each, and aggregates predictions across all subensembles [13]. SMOTEBoost integrates SMOTE into each boosting iteration to generate synthetic minority samples before updating instance weights [14]. RUSBoost combines random undersampling with AdaBoost, providing a simpler and faster alternative to SMOTEBoost that avoids the computational overhead of synthetic sample generation [15]. Balanced Random Forest (BRF) applies random undersampling to each bootstrap sample within the random forest procedure, ensuring that every constituent tree is trained on a balanced subset.

3.3. Base Classifiers and Evaluation Protocol

3.3.1. Classification Algorithms

Four classifiers spanning different algorithmic families serve as base learners for the data-level and cost-sensitive techniques. Logistic Regression (LR) provides a linear baseline with L2 regularization (C = 1.0). Random Forest (RF) constructs 100 decision trees with bootstrap aggregation and a maximum depth of 20. XGBoost implements L1/L2-regularized gradient boosting with a maximum tree depth of 6, a learning rate of 0.1, and 200 estimators. LightGBM employs leaf-wise tree growth with Gradient-based One-Side Sampling, a maximum of 31 leaves per tree, and 200 boosting rounds. All hyperparameters are held constant across imbalance handling conditions to isolate the effect of the balancing technique from classifier tuning decisions.

3.3.2. Evaluation Metrics and Validation

Five metrics capture complementary aspects of classifier performance under imbalance. AUC-ROC measures discrimination ability across all thresholds. PR-AUC summarizes precision-recall trade-offs and is more sensitive to minority-class performance under severe skew. The F1-score (threshold = 0.5) balances precision and recall at the default operating point. Matthews Correlation Coefficient (MCC) provides a balanced measure that remains robust to class size asymmetry. Recall at threshold 0.5 captures the raw fraud detection rate. All experiments use stratified 5-fold cross-validation with fixed random seeds to ensure reproducibility. Resampling is applied exclusively within training folds to prevent data leakage into validation sets. Results are reported as mean values across folds. Statistical significance is assessed using the Wilcoxon signed-rank test at the 0.05 significance level across the five fold-level measurements.

4. Results and Analysis

4.1. Performance on ULB Credit Card Dataset

4.1.1. Comparison of Data-Level and Cost-Sensitive Techniques

Table 2 presents the performance of all evaluated techniques on the ULB Credit Card dataset, using XGBoost as the base classifier for data-level and cost-sensitive methods.

Table 2. Performance Comparison on ULB Credit Card Dataset (XGBoost Base Classifier)

Category	Method	AUC-ROC	PR-AUC	F1	MCC	Recall
Baseline	XGBoost (no treatment)	0.978	0.812	0.838	0.837	0.776
Oversampling	SMOTE	0.982	0.798	0.857	0.856	0.841
Oversampling	ADASYN	0.981	0.791	0.849	0.848	0.837
Oversampling	B-SMOTE	0.983	0.806	0.861	0.860	0.846
Hybrid	SMOTE+ENN	0.985	0.821	0.873	0.872	0.858
Undersampling	RUS	0.971	0.743	0.794	0.793	0.883
Cost-Sensitive	Class Weighting	0.981	0.815	0.852	0.851	0.854
Ensemble	EasyEnsemble	0.984	0.826	0.878	0.877	0.862
Ensemble	RUSBoost	0.979	0.784	0.836	0.835	0.871
Ensemble	BRF	0.980	0.793	0.845	0.844	0.867

Among data-level techniques, SMOTE+ENN achieves the highest F1-score (0.873) and PR-AUC (0.821), confirming that the post-hoc cleaning step of Edited Nearest Neighbors effectively removes noisy synthetic samples generated near class boundaries. B-SMOTE outperforms standard SMOTE (F1: 0.861 vs. 0.857) by concentrating synthetic generation on informative borderline instances. ADASYN yields slightly lower F1 (0.849) than SMOTE (0.857), suggesting that its adaptive density allocation does not confer a clear advantage on this dataset with PCA-transformed features where local density estimation may be less reliable.

A noteworthy finding concerns the divergence between AUC-ROC and PR-AUC. Standard SMOTE improves AUC-ROC from 0.978 to 0.982 relative to the baseline, yet its PR-AUC decreases from 0.812 to 0.798. This divergence indicates that oversampling inflates the synthetic minority distribution in a manner that improves discrimination at favorable thresholds while degrading precision at high-recall operating points---consistent with earlier analyses of metric sensitivity under severe skew [16]. Class weighting avoids this pitfall, maintaining PR-AUC at 0.815 while achieving a competitive F1-score of 0.852, as it modifies the loss landscape without altering the empirical data distribution.

RUS achieves the highest recall (0.883) at the cost of substantial precision degradation, resulting in the lowest F1-score (0.794) and PR-AUC (0.743) among all methods. This aggressive majority-class removal eliminates informative majority instances, confirming that indiscriminate undersampling is suboptimal for fraud detection despite maximizing raw detection rates.

4.1.2. Comparison of Ensemble Techniques

EasyEnsemble achieves the highest F1-score (0.878) and PR-AUC (0.826) among all evaluated methods on the ULB dataset, surpassing SMOTE+ENN (F1: 0.873, PR-AUC: 0.821) by a modest margin. Its advantage stems from training multiple diverse AdaBoost classifiers on independently sampled balanced subsets, which provides both the distributional benefits of undersampling and the variance reduction of ensemble aggregation. RUSBoost attains high recall (0.871) but lower F1 (0.836) due to precision trade-offs inherent in its aggressive undersampling at each boosting round. BRF performs

between RUSBoost and EasyEnsemble across all metrics, with an F1 of 0.845 and PR-AUC of 0.793. The performance gap between EasyEnsemble and SMOTE+ENN is not statistically significant at the 0.05 level (Wilcoxon p = 0.063), indicating that both methods represent strong choices under severe imbalance conditions (As shown in Figure 1).

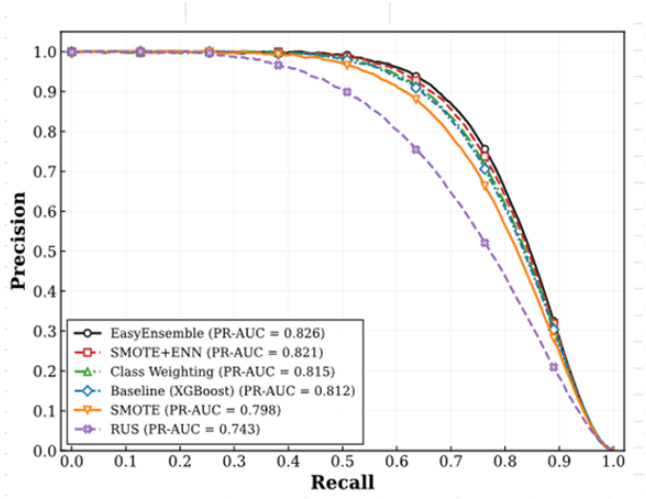

Figure 1. Precision-Recall Curves on ULB Credit Card Dataset

Precision-recall curves of six representative methods on the ULB Credit Card dataset. EasyEnsemble maintains the highest precision across most recall levels, achieving a PR-AUC of 0.826. SMOTE+ENN (PR-AUC = 0.821) closely tracks EasyEnsemble, with visible separation appearing only beyond recall = 0.85. Class Weighting (PR-AUC = 0.815) preserves precision at low-to-moderate recall levels more effectively than standard SMOTE (PR-AUC = 0.798), which exhibits a sharper precision decline beyond recall = 0.70. RUS (PR-AUC = 0.743) shows the steepest precision degradation, dropping below 0.60 at recall = 0.80. The baseline XGBoost without treatment (PR-AUC = 0.812) outperforms SMOTE in PR-AUC, illustrating the metric-dependent nature of resampling gains.

4.2. Performance on BAF Dataset

Table 3 presents results on the BAF dataset, which exhibits a moderate imbalance ratio of 90:1 and contains mixed feature types including categorical variables.

Table 3. Performance Comparison on BAF Dataset (XGBoost Base Classifier)

Category	Method	AUC-ROC	PR-AUC	F1	MCC	Recall
Baseline	XGBoost (no treatment)	0.912	0.487	0.521	0.518	0.463
Oversampling	SMOTE	0.921	0.496	0.558	0.555	0.537
Oversampling	ADASYN	0.919	0.491	0.549	0.546	0.531
Oversampling	B-SMOTE	0.923	0.503	0.564	0.561	0.542
Hybrid	SMOTE+ENN	0.928	0.518	0.579	0.576	0.561
Undersampling	RUS	0.903	0.452	0.498	0.495	0.582
Cost-Sensitive	Class Weighting	0.920	0.511	0.553	0.550	0.549
Ensemble	EasyEnsemble	0.931	0.527	0.586	0.583	0.568
Ensemble	RUSBoost	0.916	0.478	0.532	0.529	0.573
Ensemble	BRF	0.918	0.489	0.541	0.538	0.564

The ranking of techniques on the BAF dataset is broadly consistent with the ULB results: EasyEnsemble achieves the highest F1 (0.586) and PR-AUC (0.527), followed by SMOTE+ENN (F1: 0.579, PR-AUC: 0.518) and class weighting (F1: 0.553, PR-AUC: 0.511).

Absolute performance values are substantially lower across all methods, reflecting the greater classification difficulty introduced by mixed feature types, the presence of categorical variables that reduce SMOTE interpolation quality, and the synthetic data generation process underlying the BAF suite.

The AUC-ROC and PR-AUC divergence observed on the ULB dataset is less pronounced on the BAF dataset: SMOTE improves both AUC-ROC (0.912 to 0.921) and PR-AUC (0.487 to 0.496) relative to the baseline. This reduced divergence aligns with the milder imbalance ratio (90:1 vs. 578:1), as the distortion introduced by synthetic minority generation is proportionally smaller when the minority class is more adequately represented in the original data [17] (As shown in Figure 2).

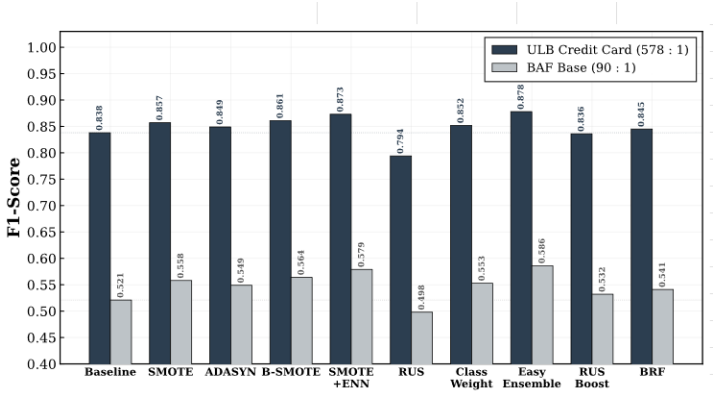

Figure 2. F1-Score Comparison Across Datasets

Grouped bar chart comparing F1-scores of all nine methods on the ULB Credit Card and BAF datasets. On the ULB dataset, EasyEnsemble achieves the highest F1 (0.878), followed by SMOTE+ENN (0.873) and B-SMOTE (0.861). On the BAF dataset, the same ranking holds with EasyEnsemble at 0.586, SMOTE+ENN at 0.579, and B-SMOTE at 0.564. The absolute performance gap between the two datasets is substantial (approximately 0.29 F1 units), reflecting the additional classification difficulty of the BAF dataset. RUS is the only method that yields lower F1 than the untreated baseline on the BAF dataset (0.498 vs. 0.521).

4.3. Cross-Dataset Analysis

4.3.1. Effect of Imbalance Severity

The performance improvements attributable to imbalance handling techniques differ between the two datasets in instructive ways. On ULB, EasyEnsemble improves F1 by 0.040 over the baseline (0.878 vs. 0.838), while on BAF, the improvement is 0.065 (0.586 vs. 0.521). The relative improvement is comparable (4.8% vs. 12.5%), indicating that imbalance handling provides proportionally greater benefit when baseline performance is lower. The technique ranking stability between datasets suggests that EasyEnsemble and SMOTE+ENN are robust choices across different imbalance conditions, a finding consistent with recent survey evidence on imbalanced tabular and graph-structured data [18].

An additional cross-classifier analysis reveals that the choice of base classifier interacts with the imbalance handling technique. Table 4 presents F1-scores for selected technique-classifier combinations on the ULB dataset.

Table 4. F1-Scores by Classifier and Imbalance Technique on ULB Dataset

Method	Logistic Regression (LR)	Random Forest (RF)	XGBoost	LightGBM
No treatment	0.578	0.822	0.838	0.831
SMOTE	0.692	0.849	0.857	0.853
SMOTE+ENN	0.713	0.858	0.873	0.867

Class Weighting	0.684	0.841	0.852	0.846
EasyEnsemble	—	—	—	—

Note: EasyEnsemble uses its own internal AdaBoost base learners and is not combined with external classifiers. Its F1 on ULB is 0.878.

The relative benefit of resampling is most pronounced for Logistic Regression, where SMOTE+ENN raises F1 from 0.578 to 0.713 (a 23.4% increase). Tree-based classifiers show smaller gains: XGBoost improves from 0.838 to 0.873 (4.2%) with SMOTE+ENN. This pattern reflects the inherent ability of tree ensembles to partition feature space in ways that partially accommodate class imbalance through recursive splits that isolate minority regions, reducing---but not eliminating---the marginal value of explicit resampling.

4.3.2. Computational Efficiency Trade-Offs

Table 5 reports mean training times per fold on the ULB dataset, measured on a workstation with an 8-core CPU and 32 GB RAM.

Table 5. Mean Training Time Per Fold on ULB Dataset (Seconds)

Method	LR	RF	XGBoost	LightGBM
No treatment	1.2	8.4	5.7	3.1
SMOTE	3.8	24.6	16.3	9.2
SMOTE+ENN	5.6	27.3	18.5	10.7
Class Weighting	1.3	8.6	5.9	3.2
RUS	0.4	2.1	1.8	0.9
EasyEnsemble	14.2	—	—	—

Oversampling methods substantially increase training time by enlarging the training set: SMOTE+ENN with XGBoost requires 18.5 seconds per fold, which is 3.2 times the baseline duration of 5.7 seconds. Class weighting introduces negligible overhead (5.9 vs. 5.7 seconds), as it modifies only the loss function without changing the data volume. RUS reduces training time below the baseline (1.8 vs. 5.7 seconds for XGBoost) by shrinking the training set to the minority-class size. EasyEnsemble requires 14.2 seconds with its internal AdaBoost learners, positioning it between the oversampling and baseline approaches in computational cost while delivering the strongest classification performance (As shown in Figure 3).

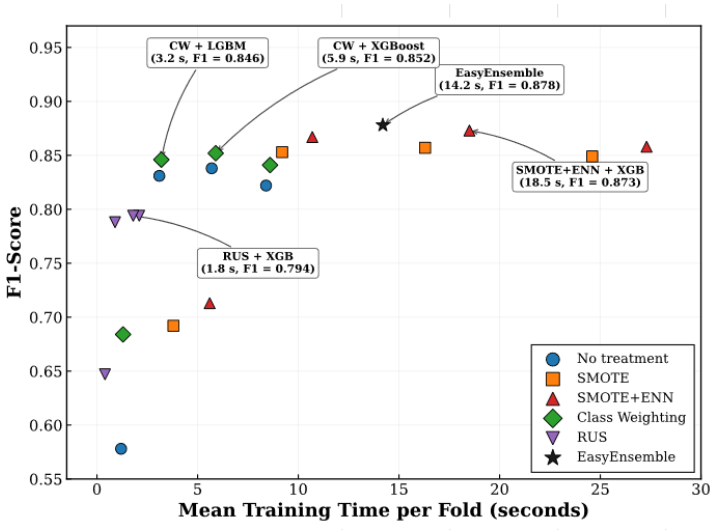

Figure 3. Training Time versus F1-Score Trade-off on ULB Dataset

Scatter plot of mean training time per fold (x-axis, seconds) against F1-score (y-axis) for all technique-classifier combinations on the ULB dataset. Class Weighting with XGBoost (5.9 s, F1 = 0.852) and LightGBM (3.2 s, F1 = 0.846) cluster in the high-efficiency region. SMOTE+ENN with XGBoost (18.5 s, F1 = 0.873) and EasyEnsemble (14.2 s, F1 = 0.878) occupy the high-performance region at moderate computational cost. RUS configurations achieve the shortest training times (0.4--2.1 s) at the expense of lower F1-scores (0.692--0.794).

5. Discussion

5.1. Key Findings and Practical Implications

The experimental results yield three principal findings with direct practical relevance. The first is that ensemble-based methods, particularly EasyEnsemble, deliver the most robust performance across both datasets and all evaluation metrics. The modest advantage of EasyEnsemble over SMOTE+ENN (F1 difference of 0.005--0.007) suggests that practitioners may reasonably choose either method based on implementation constraints, as the performance gap does not reach statistical significance at conventional levels.

The second finding concerns the divergence between AUC-ROC and PR-AUC under severe imbalance. Standard SMOTE improves AUC-ROC on the ULB dataset while simultaneously degrading PR-AUC, a pattern not observed on the moderately imbalanced BAF dataset. This metric-dependent behavior carries important implications for fraud detection evaluation: relying solely on AUC-ROC can obscure precision degradation at high-recall operating points, which is precisely the regime most relevant to production fraud screening. Practitioners operating under severe imbalance should report PR-AUC alongside AUC-ROC and consider class weighting as an alternative that avoids the PR-AUC degradation associated with synthetic oversampling.

The third finding highlights the interaction between imbalance handling technique and base classifier strength. Tree-based ensemble classifiers (XGBoost, LightGBM, Random Forest) exhibit smaller absolute gains from resampling than Logistic Regression, reflecting their native capacity to handle moderate imbalance through recursive partitioning. This interaction suggests that the investment in complex resampling pipelines yields diminishing returns when paired with already-robust classifiers, and that class weighting---with negligible computational overhead---may represent the most practical default strategy for tree-based methods in production environments.

5.2. Limitations

This study has several limitations that constrain the generalizability of its conclusions. The evaluation is restricted to two datasets, both representing credit card or bank account fraud; other fraud domains such as insurance fraud, healthcare fraud, and cryptocurrency-based illicit transactions may exhibit different data characteristics and imbalance patterns. The PCA-anonymized features of the ULB dataset prevent analysis of how individual feature semantics interact with resampling mechanisms, and the synthetic nature of the BAF dataset may not fully capture the distributional complexity of real-world transaction data. All experiments use fixed hyperparameters for base classifiers to isolate the effect of imbalance handling; joint optimization of classifier hyperparameters and resampling parameters could yield different relative rankings.

Future work should extend this comparison to additional fraud detection domains and incorporate temporal evaluation protocols that respect the chronological ordering of transactions. Investigating the interaction between imbalance handling techniques and feature engineering pipelines would provide a more complete picture of operational performance. The integration of cost-sensitive loss modifications with ensemble methods---combining the strengths of algorithm-level and ensemble-level approaches---represents a promising direction that this study's results suggest could yield further improvements. Deep learning approaches to imbalanced fraud detection, including focal loss and label-distribution-aware margin techniques, warrant systematic comparison against the classical methods evaluated here, particularly as transaction datasets continue to grow in scale and dimensionality.

References

1. H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.
2. A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, "Credit card fraud detection: A realistic modeling and a novel learning strategy," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3784–3797, 2018.
3. J. Davis and M. Goadrich, "The relationship between Precision-Recall and ROC curves," in *Proceedings of the 23rd International Conference on Machine Learning*, pp. 233–240, 2006.
4. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.
5. H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, pp. 1322–1328, 2008.
6. H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning," in *Advances in Intelligent Computing*, LNCS vol. 3644, pp. 878–887, Springer, 2005.
7. G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 20–29, 2004.
8. C. Elkan, "The foundations of cost-sensitive learning," in *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, pp. 973–978, 2001.
9. P. Domingos, "MetaCost: A general method for making classifiers cost-sensitive," in *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 155–164, 1999.
10. S. Jesus, J. Pombal, D. Alves, A. Cruz, P. Saleiro, R. Ribeiro, J. Gama, and P. Bizarro, "Turning the tables: Biased, imbalanced, dynamic tabular datasets for ML evaluation," in *Advances in Neural Information Processing Systems 35*, 2022.
11. T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.
12. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems 30*, pp. 3146–3154, 2017.
13. X.-Y. Liu, J. Wu, and Z.-H. Zhou, "Exploratory undersampling for class-imbalance learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 39, no. 2, pp. 539–550, 2009.
14. N. V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "SMOTEBoost: Improving prediction of the minority class in boosting," in *Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pp. 107–119, 2003.
15. C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "RUSBoost: A hybrid approach to alleviating class imbalance," *IEEE Transactions on Systems, Man, and Cybernetics---Part A*, vol. 40, no. 1, pp. 185–197, 2010.
16. Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 1, pp. 63–77, 2006.
17. K. Cao, C. Wei, A. Gaidon, N. Aréchiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," in *Advances in Neural Information Processing Systems 32*, pp. 1565–1576, 2019.
18. Y. Ma, Y. Tian, N. Moniz, and N. V. Chawla, "Class-imbalanced learning on graphs: A survey," *ACM Computing Surveys*, vol. 57, no. 8, Article 207, 2025.