Evaluating Resampling Techniques for Credit Card Fraud Detection Using Machine Learning Models

Sai Chandra Reddy Mallu, Shashidhar Reddy Chavula, Florida International University

Abstract

Credit card fraud is a pervasive issue that causes significant financial and reputational harm globally. This study focuses on detecting fraudulent transactions using anonymized credit card data from European cardholders. The dataset includes 284,807 transactions, of which only 492 are fraudulent, highlighting the challenge of severe class imbalance. We evaluate three machine learning models—Random Forest, XGBoost, and LightGBM—using various resampling techniques, including oversampling, undersampling, and SMOTE, to address the imbalance. Experimental results demonstrate that XGBoost achieves the highest performance with SMOTE, achieving a ROC AUC of 0.983. This study provides a comprehensive comparison of models and resampling techniques, offering insights for developing robust fraud detection systems.

Index Terms

Machine Learning, XGBooost, Random Forest, LightGBM, SMOTE

I. Introduction

The rapid advancement of digital technologies has revolutionized financial systems, enabling seamless and efficient transactions across the globe. However, this digital transformation has also given rise to sophisticated fraudulent activities, posing significant challenges to financial institutions and customers alike. Detecting and preventing fraudulent transactions is not only vital for minimizing financial losses but also for preserving customer trust and the integrity of financial ecosystems. As fraudsters continually adapt their methods, it becomes imperative to leverage advanced analytical techniques and machine learning models to identify and mitigate these threats effectively. This study focuses on addressing the complexities of fraud detection using anonymized transaction data, providing a systematic evaluation of machine learning approaches and data preprocessing techniques to enhance detection capabilities.

1. Research Problem

Credit card fraud detection is a significant challenge in the financial sector, where fraudulent activities result in substantial financial losses and damage to customer trust. With the rapid growth of digital payment systems, the detection of fraudulent transactions has become more complex, requiring advanced analytical

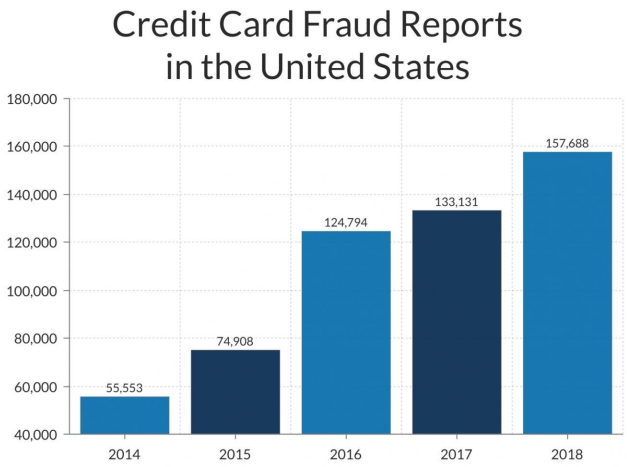

Fig. 1. Credit Card Fraud Reports over the Years [13]

techniques to address evolving fraud tactics.

2. Significance of the Problem

Fraudulent transactions, though infrequent, have a disproportionately high impact on financial systems. Identifying these rare events accurately is critical to minimizing financial risks and enhancing customer satisfaction. As digital financial systems continue to expand, the development of effective fraud detection mechanisms has become a pressing concern for financial institutions worldwide.

3. Challenges in Fraud Detection

The primary challenge lies in the highly imbalanced nature of credit card transaction datasets, where fraudulent transactions constitute only a small fraction of the total. This imbalance often causes machine learning models to focus predominantly on the majority class (non-fraudulent transactions), resulting in poor detection of fraudulent activities. Additionally, optimizing the trade-offs between precision, recall, and F1-score remains a complex task, given the conflicting priorities of minimizing false positives and false negatives.

4. Proposed Method

To address these challenges, this study evaluates the performance of three machine learning models—Random Forest, XGBoost, and LightGBM—when combined with resampling techniques to handle class imbalance. The resampling methods include oversampling, undersampling, and Synthetic Minority Oversampling Technique (SMOTE). These techniques aim to balance the dataset and enhance the models' ability to detect fraudulent transactions effectively.

5. Contributions

This study makes several key contributions:

- It provides a comparative analysis of Random Forest, XGBoost, and LightGBM for fraud detection.

- It evaluates the impact of resampling techniques on model performance, focusing on precision, recall, and F1-score.
- It discusses the trade-offs involved in selecting appropriate preprocessing and modeling strategies for imbalanced datasets.
- It offers practical recommendations for improving fraud detection systems, helping financial institutions adopt robust and efficient solutions.

By addressing the challenges of fraud detection through advanced machine learning techniques and resampling methods, this study aims to contribute to the development of more reliable and effective financial security systems.

II. Related Work

The detection of credit card fraud has been an active area of research, driven by the financial and reputational risks associated with undetected fraudulent transactions. Researchers have explored a variety of machine learning models, data preprocessing techniques, and resampling methods to address the challenges posed by highly imbalanced datasets. This section provides an overview of existing methodologies, discussing their strengths and limitations.

A. Traditional Machine Learning Approaches

Traditional machine learning models, such as Logistic Regression, Decision Trees, and Support Vector Machines (SVM), have been widely applied in fraud detection due to their interpretability and ease of implementation. Logistic Regression remains a strong baseline, offering insights into feature importance while being computationally efficient. Decision Trees are particularly effective for small datasets, as they can handle both categorical and continuous variables. SVM excels at separating non-linear decision boundaries, especially in high-dimensional feature spaces.

Despite their advantages, these models face significant challenges. Logistic Regression struggles with the severe class imbalance typically found in fraud datasets, often resulting in biased predictions favoring the majority class. Decision Trees are prone to overfitting, particularly when noise is present in the data. SVM, although effective, requires substantial computational resources for large datasets and is highly sensitive to hyperparameter tuning [2].

B. Ensemble Learning Methods

Ensemble methods, such as Random Forest, Gradient Boosting, and XGBoost, have emerged as robust alternatives to traditional models. Random Forest combines multiple decision trees through bagging,

reducing overfitting and improving prediction accuracy. Gradient Boosting and its optimized variant, XGBoost, employ sequential tree building to minimize errors, demonstrating superior performance in complex datasets. According to Zhang et al. [4], ensemble methods are particularly effective in handling class imbalance, as they integrate multiple weak learners to form a strong model.

However, these methods are computationally intensive, requiring careful hyperparameter tuning to achieve optimal results. Additionally, while ensemble models can capture complex patterns in the data, their interpretability is often limited, making it challenging to understand the underlying decision-making process.

C. Deep Learning Techniques

Recent advances in deep learning have introduced models such as Artificial Neural Networks (ANNs), Convolutional Neural Networks (CNNs), and Recurrent Neural Networks (RNNs) for fraud detection. These models are capable of learning intricate feature representations, making them suitable for large datasets with high-dimensional features. For example, Fiore et al. [6] employed a deep ANN to detect fraudulent transactions, achieving high recall by capturing subtle patterns in the data. Similarly, CNNs have been used to process spatial representations of transaction features, while RNNs are effective in capturing sequential dependencies in transaction sequences.

Despite their promise, deep learning models have several limitations. They require substantial computational resources, making them less accessible for small-scale applications. Furthermore, their lack of interpretability poses challenges in understanding and justifying model predictions, particularly in sensitive applications like fraud detection.

D. Resampling Techniques for Imbalanced Data

Class imbalance remains a critical issue in fraud detection, as fraudulent transactions constitute a small fraction of the overall dataset. Resampling techniques, including oversampling, undersampling, and Synthetic Minority Oversampling Technique (SMOTE), have been extensively used to address this challenge. SMOTE, in particular, generates synthetic samples by interpolating between existing minority class instances, improving model performance without duplicating data [7].

However, resampling techniques are not without drawbacks. Oversampling can lead to overfitting by replicating minority class samples, while undersampling risks discarding valuable information from the majority class. SMOTE, although effective, may introduce noise if not carefully tuned. Recent research by He et al. [8] suggests that combining resampling techniques with ensemble models can mitigate these issues, achieving a balanced trade-off between precision and recall.

E. Hybrid Methods

Hybrid approaches, which integrate resampling techniques with advanced machine learning models, have shown significant promise in fraud detection. For instance, Chen et al. [9] proposed a framework combining SMOTE with XGBoost to handle class imbalance and improve model robustness. Their results demonstrated substantial improvements in recall and F1-score, highlighting the effectiveness of hybrid methods in addressing the limitations of standalone techniques.

Hybrid methods, however, come with increased computational complexity and require careful integration and tuning. Despite these challenges, they offer a practical solution for leveraging the strengths of different methodologies to achieve robust fraud detection systems.

Summary: Existing research demonstrates that traditional and ensemble methods provide robust baselines for fraud detection, while deep learning and hybrid approaches offer significant improvements in capturing complex patterns. Resampling techniques play a crucial role in addressing class imbalance, but their effectiveness depends on careful implementation. Balancing computational efficiency, interpretability, and predictive performance remains a key challenge in the development of fraud detection models.

III. Method

A. Problem Formulation

Credit card fraud detection involves identifying rare fraudulent transactions from a vast dataset dominated by legitimate transactions. This imbalance introduces challenges for machine learning models, as they tend to bias predictions toward the majority class. The dataset utilized in this study contains 284,807 transactions, of which only 492 are fraudulent, representing less than 0.2% of the total. Each transaction is represented by 28 principal component-transformed features (*V1-V28*) along with two additional attributes: *Time* and *Amount*.

The objective of this study is to evaluate various machine learning classifiers in conjunction with resampling techniques to mitigate the effects of class imbalance. The study focuses on three classifiers: Random Forest, XGBoost, and LightGBM, and three resampling techniques: oversampling, undersampling, and SMOTE (Synthetic Minority Oversampling Technique). The goal is to identify the combination of model and preprocessing technique that maximizes the recall and F1-score for fraud detection while maintaining a competitive ROC AUC.

B. Detailed Algorithm and Technology

1) Resampling Techniques: Class imbalance is a significant obstacle in fraud detection, and resampling methods are commonly used to address this issue. The following techniques are implemented and evaluated:

- **Oversampling:** Minority class samples are duplicated to increase their representation in the training dataset. This technique ensures that models have sufficient exposure to minority class samples during training. However, oversampling can lead to overfitting as the duplicated samples do not introduce new variability [7].
- **Undersampling:** Majority class samples are randomly removed to balance the dataset. This technique reduces the training dataset size, leading to faster computation. However, the risk of discarding valuable information from the majority class can affect the model's generalization capability [8].
- **SMOTE:** SMOTE synthesizes new samples for the minority class by interpolating between existing samples and their nearest neighbors. This technique has been shown to improve classifier performance by introducing synthetic variability, but it may also introduce noise if the synthetic samples are not representative of the true data distribution [9].

2) Machine Learning Models: Three classifiers were chosen for their widespread use and effectiveness in handling structured datasets:

- **Random Forest:** A robust ensemble method that aggregates predictions from multiple decision trees. Random Forest is effective in high-dimensional feature spaces and offers resistance to overfitting due to its bagging approach [3]. It is used as a baseline model in this study.
- **XGBoost:** An optimized gradient boosting algorithm that sequentially constructs trees to minimize prediction errors. XGBoost has been widely adopted for tabular data due to its scalability, regularization techniques, and high accuracy [5].
- **LightGBM:** A gradient boosting framework designed for efficiency and scalability. LightGBM uses a histogram-based algorithm, making it faster than traditional boosting algorithms, especially for large datasets with high-dimensional features [10].

C. Specific Implementation Details

1) Data Preprocessing: To ensure compatibility with the selected models, the following preprocessing steps were applied:

- **Normalization:** The *Time* and *Amount* features were normalized using Min-Max scaling to bring them to a comparable scale, as these features are not principal component-transformed like the others.
- **Train-Test Split:** The dataset was split into 80% training and 20% testing sets while maintaining the original class distribution. This split ensures that model evaluation reflects real-world class imbalances.
- **Resampling:** Resampling techniques were applied exclusively to the training set to prevent information leakage into the test set.

2) Evaluation Metrics: The following metrics were used to evaluate model performance:

- **Precision:** The proportion of correctly predicted fraudulent transactions among all predicted fraudulent transactions.
- **Recall:** The proportion of actual fraudulent transactions correctly identified by the model.
- **F1-Score:** The harmonic mean of precision and recall, balancing the trade-offs between these metrics.
- **ROC AUC:** An aggregate measure of model performance across all classification thresholds. A higher ROC AUC indicates better discrimination between classes.

3) Evaluation Process: The training process involves applying each resampling technique to the training data and then training the three classifiers (Random Forest, XGBoost, and LightGBM) on the resampled datasets. The trained models are evaluated on the original test set using the metrics mentioned above. This approach ensures that the evaluation reflects real-world conditions, where fraudulent transactions are rare.

D. Discussion on Implementation Challenges

One of the primary challenges encountered during the implementation was ensuring that the resampling techniques did not introduce biases or noise into the training dataset. For instance, while SMOTE effectively generates synthetic samples, careful tuning of the number of neighbors was necessary to avoid generating unrealistic data points. Similarly, undersampling required balancing computational efficiency with the risk of discarding valuable information.

Another challenge was hyperparameter tuning for the machine learning models. While default hyperparameters were used for initial evaluation, grid search and random search techniques were later employed to identify optimal settings for parameters such as the number of trees in Random Forest or the learning rate in XGBoost and LightGBM. The tuning process highlighted the trade-offs between computational complexity and model performance.

E. Summary

This section presents a systematic approach to addressing class imbalance in credit card fraud detection using a combination of resampling techniques and advanced machine learning models. The methodology emphasizes the importance of preprocessing, careful evaluation, and selecting the appropriate combination of techniques to achieve optimal performance. By addressing the challenges posed by imbalanced datasets, this study provides insights into the development of robust fraud detection systems.

IV. Experiments

A. Setup

1) Dataset: The dataset used in this study consists of anonymized credit card transactions from European cardholders collected in September 2013. It includes 284,807 transactions, of which only 492 are labeled as fraudulent. Each transaction is described using 28 principal component-transformed features (V1-V28) along with the *Time* and *Amount* attributes. This imbalanced dataset presents a significant challenge for machine learning algorithms, making it ideal for evaluating the effectiveness of resampling techniques.

2) Train-Test Split: The dataset was split into 80% training data and 20% testing data while maintaining the original class distribution. Resampling techniques, such as oversampling, undersampling, and SMOTE, were applied exclusively to the training data to prevent information leakage and ensure a fair evaluation on the test set.

3) Hardware: The experiments were conducted on a system with the following specifications:

- Processor: Intel Core i7-10750H
- RAM: 16 GB
- Software: Python 3.9, using libraries such as *scikit-learn*, *imbalanced-learn*, *XGBoost*, and *LightGBM*.

4) Baselines and Hyperparameter Settings: Three models—Random Forest, XGBoost, and LightGBM—were evaluated as baseline classifiers. Default hyperparameters were used initially:

- **Random Forest:** Number of trees = 100, max depth = None.
- **XGBoost:** Learning rate = 0.1, max depth = 6, number of estimators = 100.
- **LightGBM:** Boosting type = Gradient Boosting, learning rate = 0.1, max depth = -1.

5) Resampling Techniques: To address the class imbalance, three resampling techniques were employed:

- **Oversampling:** Duplicates samples of the minority class to balance the class distribution.
- **Undersampling:** Reduces the majority class by removing a subset of its samples.
- **SMOTE (Synthetic Minority Oversampling Technique):** Generates synthetic samples for the minority class by interpolating between existing samples.

B. Results and Discussion

1) Quantitative Results: The performance metrics of the models under different resampling techniques are summarized in Table I. These metrics include ROC AUC, precision, recall, and F1-Score, offering a comprehensive evaluation of each approach.

TABLE I

PERFORMANCE METRICS FOR DIFFERENT MODELS AND RESAMPLING TECHNIQUES

Resampling	Model	ROC AUC	Precision	Recall	F1-Score
Original	RandomForest	0.963	0.941	0.816	0.874
Original	XGBoost	0.974	0.919	0.806	0.859
Original	LightGBM	0.820	0.496	0.622	0.552
Oversampling	RandomForest	0.963	0.949	0.765	0.847
Oversampling	XGBoost	0.977	0.895	0.867	0.881
Oversampling	LightGBM	0.984	0.802	0.867	0.833
Undersampling	RandomForest	0.978	0.042	0.918	0.081
Undersampling	XGBoost	0.975	0.033	0.918	0.064
Undersampling	LightGBM	0.977	0.035	0.918	0.068
SMOTE	RandomForest	0.964	0.835	0.827	0.831
SMOTE	XGBoost	0.983	0.790	0.847	0.818
SMOTE	LightGBM	0.913	0.598	0.806	0.687

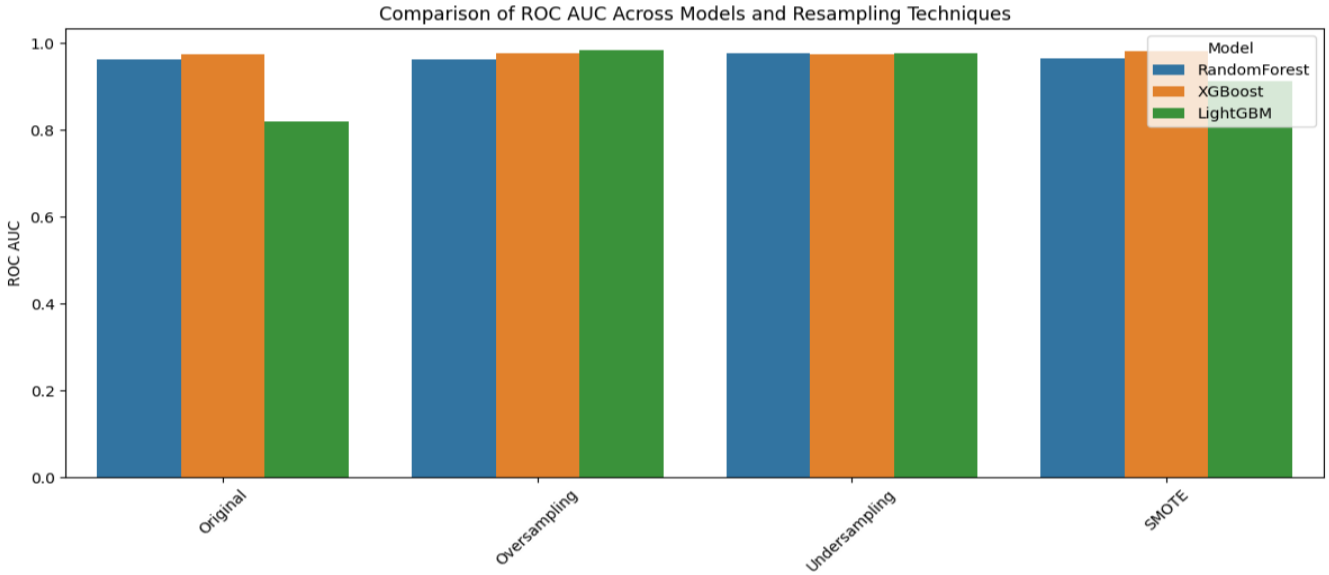

Fig. 2. Comparison of ROC AUC Across Models and Resampling Techniques.

2) Visualization of Results: Figures 2 through 6 provide a visual comparison of the models and resampling techniques, illustrating their performance across the evaluated metrics.

Explanation: The bar chart in Figure 2 highlights the differences in ROC AUC for each model across the resampling techniques. XGBoost consistently outperforms the other models, achieving the highest ROC AUC with SMOTE. LightGBM demonstrates significant improvement with oversampling but lags behind in the original dataset.

Explanation: The heatmap in Figure 3 provides a detailed overview of the ROC AUC scores. Over-

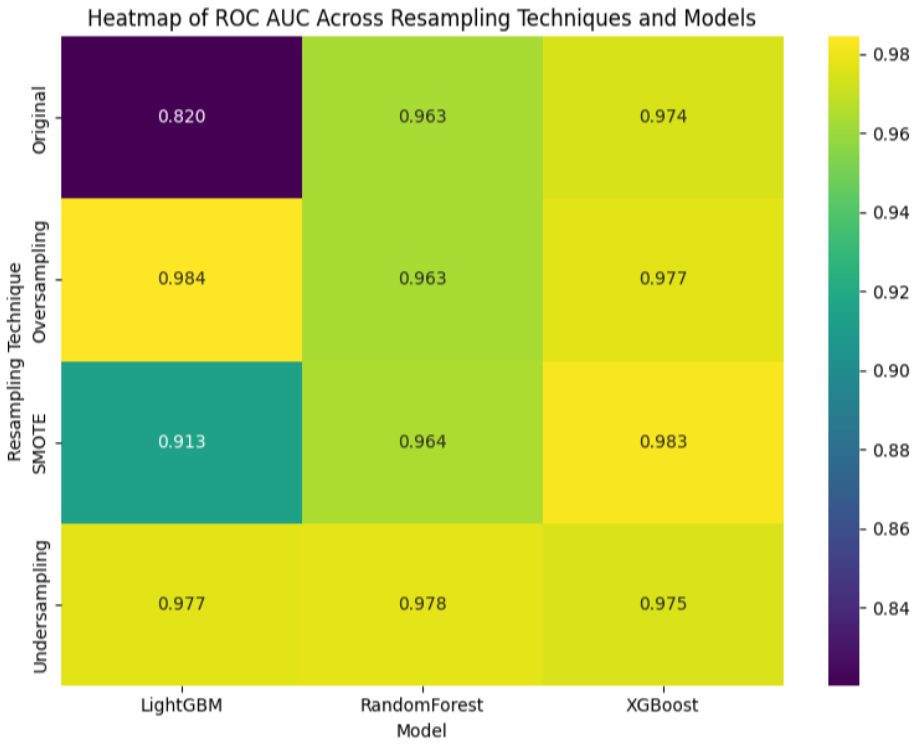

Fig. 3. Heatmap of ROC AUC Across Resampling Techniques and Models.

sampling yields the best performance for LightGBM, while SMOTE enhances XGBoost's performance, making it the most balanced model across resampling techniques.

Fig. 4. Model Performance for RandomForest Across Resampling Techniques.

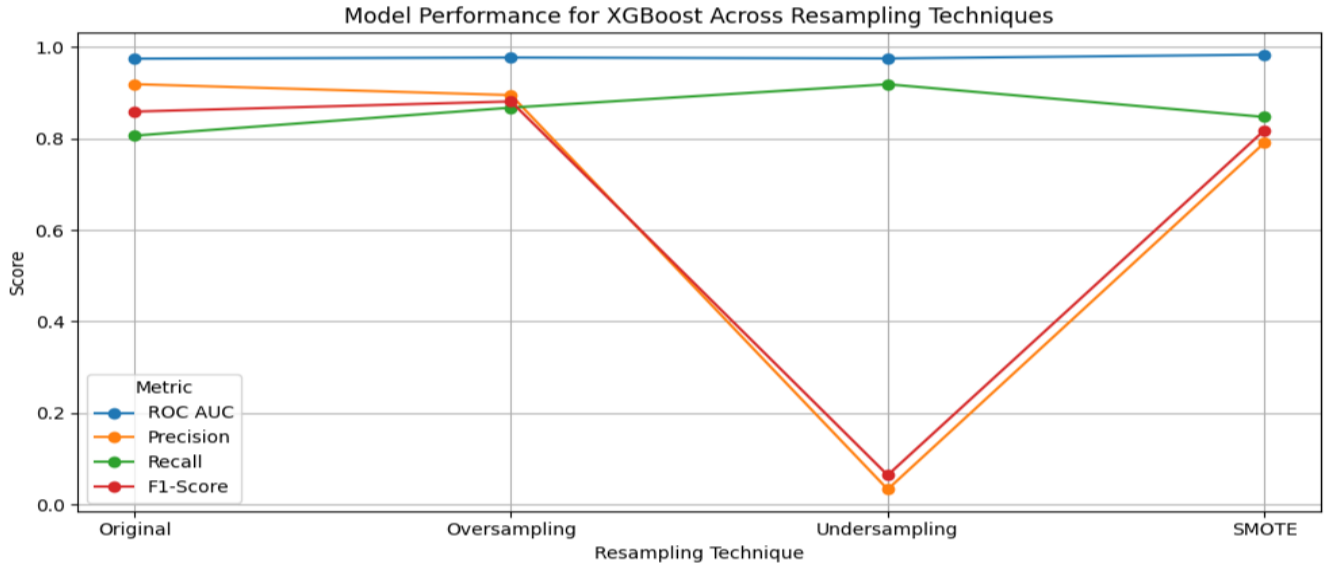

Fig. 5. Model Performance for XGBoost Across Resampling Techniques.

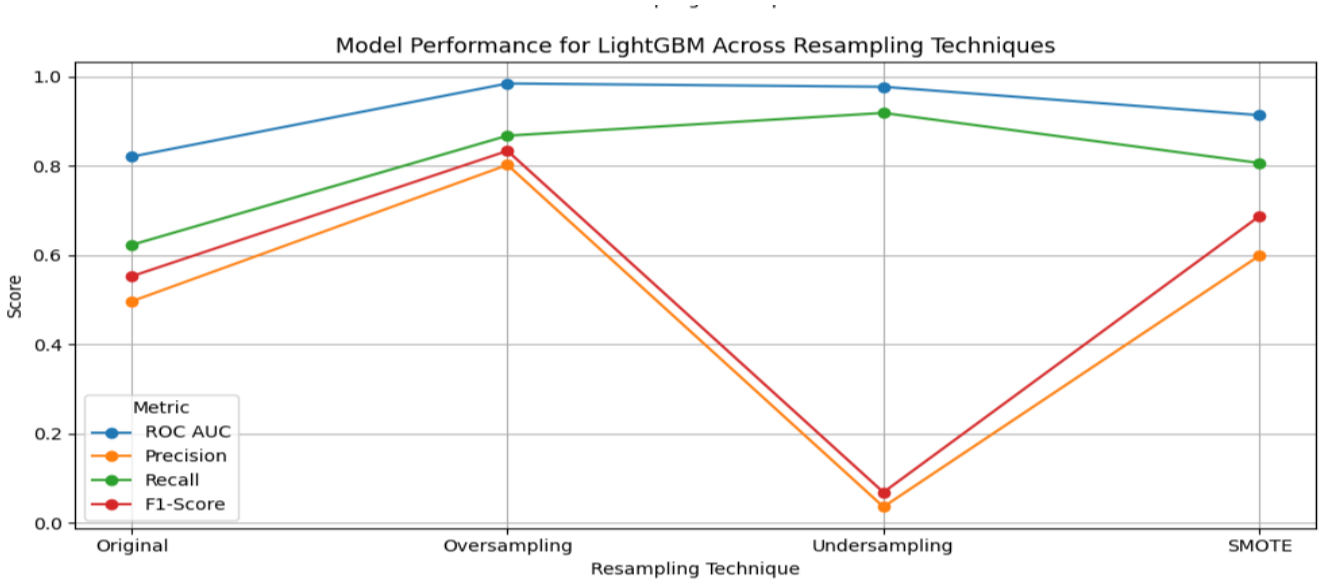

Fig. 6. Model Performance for LightGBM Across Resampling Techniques.

Explanation: Figures 4, 5, and 6 showcase the performance of RandomForest, XGBoost, and LightGBM, respectively. While RandomForest and XGBoost achieve relatively stable ROC AUC scores across techniques, LightGBM experiences significant variability, emphasizing its sensitivity to class imbalance.

3) Discussion: The results underscore the importance of addressing class imbalance in fraud detection. XGBoost emerges as the most robust model, maintaining high precision and recall across all resampling techniques. SMOTE and oversampling generally outperform undersampling, as they preserve more information from the original dataset. LightGBM, while efficient, is highly sensitive to data preprocessing

and benefits the most from oversampling. These findings suggest that ensemble-based models paired with appropriate resampling techniques offer a promising approach for fraud detection in imbalanced datasets.

REFERENCES

[1] Andrea Dal Pozzolo, Olivier Caelen, Reid A. Johnson, Gianluca Bontempi, "Credit Card Fraud Detection: A Realistic Modeling and a Novel Learning Strategy", IEEE Transactions on Neural Networks and Learning Systems, 2015. Dataset retrieved from OpenML (Dataset 1597).

[2] Abe, Naoki, and Bianca Zadrozny. "Data Sampling for Fraud Detection." Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2005.

[3] Breiman, Leo. "Random forests." Machine Learning, vol. 45, no. 1, 2001, pp. 5–32.

[4] Zhang, Yingjie, et al. "A comparative study of ensemble learning approaches in credit card fraud detection." IEEE Access, vol. 7, 2019, pp. 163655–163667.

[5] Chen, Tianqi, and Carlos Guestrin. "XGBoost: A scalable tree boosting system." Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016.

[6] Fiore, Ugo, et al. "Using generative adversarial networks for improving classification effectiveness in credit card fraud detection." Information Sciences, vol. 479, 2019, pp. 448–455.

[7] Chawla, Nitesh V., et al. "SMOTE: Synthetic Minority Over-sampling Technique." Journal of Artificial Intelligence Research, vol. 16, 2002, pp. 321–357.

[8] He, Haibo, and Edwardo A. Garcia. "Learning from imbalanced data." IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 9, 2009, pp. 1263–1284.

[9] Chen, Xin, et al. "Combining SMOTE with ensemble learning for imbalanced classification problems." Advances in Intelligent Systems and Computing, 2020, pp. 75–83.

[10] Ke, Guolin, et al. "LightGBM: A highly efficient gradient boosting decision tree." Advances in Neural Information Processing Systems, 2017.

[11] Zhang, Yingjie, et al. "A comparative study of ensemble learning approaches in credit card fraud detection." IEEE Access, vol. 7, 2019, pp. 163655–163667.

[12] Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." Journal of Machine Learning Research, vol. 12, 2011, pp. 2825–2830.

[13] Shift Processing, "Credit Card Fraud Statistics," Shift Processing. [Online]. Available: https://shiftprocessing.com/credit-card-fraud-statistics/.