

AN INNOVATIVE ANALYSIS OF PARTIAL PENALTY ON IMBALANCED DATA IN CREDIT CARD FRAUD PREDICTION

Jiawei Zhang¹, Xin Zhang² and Xinyin Miao³

¹Senior Investment Analyst, PRA Group (Nasdaq: PRAA), Norfolk, Virginia, USA

²Data Scientist, PRA Group (Nasdaq: PRAA), Norfolk, Virginia, USA

³Senior Data Analyst, American Airlines Group Inc (Nasdaq: AAL), Dallas, Texas, USA

ABSTRACT

This paper provides an innovative methodology of partial penalty on machine learning models to handle the data imbalance scenario occurring in credit card fraud detection implementation. Unlike the normal over-sampling or under-sampling methodologies, partial penalty directs the machine learning model to focus on learning the minor class of target variable even when the class distribution is extremely imbalanced. Besides comparing the partial penalty approach with over-sampling and under-sampling approaches to handle data imbalance scenario, we've implemented this new approach under five machine learning classification models, including Logistic Regression, Random Forest, kNN, Decision Tree, and Light Gradient Boosting Model. The new partial penalty approach realizes a performance of 88.35% F1 score and 98.79% AUC score with Light GBM, higher than either over-sampling or under-sampling approaches in similar articles.

KEYWORDS

Partial Penalty, Gradient Boosting, Data Imbalance, Credit Card Fraud Detection, SMOTE

1. INTRODUCTION

In the past decade, credit card payment has been becoming a more dominant payment methods amount all consumer transactions, occupying from 18.18% of all transactions in 2016 to 32.61% of all transactions in 2023 in the US [1]. In 2024, while cash accounted for only 14% of all US consumer payments, credit cards accounted for 35%, which represented over 65% of all payments after combining with debit cards payment [2]. Coming with the increasing volume of credit card transactions is the higher than ever risk of credit card fraud, which is the unauthorized use of credit card or debit card information to make withdrawals or purchases, typically including physical card theft, online theft, application fraud by using others' personal information, and account takeover transactions [3]. In 2024, the number of complaints of identity theft related to existing credit card misuse or new card applications has increased from 416,582 to 449,032 since 2024 according to Federal Trade Commission's consumer sentinel network data book in 2024, counting for \$12.5 billion in total consumer loss [4]. However, there're several challenges for the credit card fraud detection. For example, advanced fraud technologies such as biometrics to bypass traditional security measures, the increasing volume of daily fraud transactions, or balancing the transaction security measures and customer experience are some of the challenges making the daily capturing

of credit card fraud difficult to fully handle by traditional monitoring method [5]. While machine learning serves as a good way to handle such amount of transaction detection, the data is so imbalanced that the accuracy of machine learning might be falsely too high while the model cannot effectively capture those small amounts of fraud.

2. LITERATURE REVIEW

Due to the minority of true positive credit card fraud in large amount of non-fraud transaction, machine learning has increasingly been implemented to detect the minor occurrence of fraud to avoid the large amount of human work and inefficient strictly monitoring criteria. Some machine learning models have been explored using the same European credit card fraud dataset as implemented in this article. For example, ensembled machine learning models, including SVM, kNN, and boosting based ensemble learning models, have been able to provide an AUC score of 96% tested on European credit card fraud dataset [6]. Some early work in 2022 also tested the SaaS platform to conduct model training and testing for credit card fraud detection, realizing 84% recall rate 97.3% AUC score in testing dataset [7]. Feature selection experiments have also been explored by combining linear correlation, Information Gain score, and random forest feature importance to select the expected features [8]. Other feature selection paths, including ANOVA, were also tested to conduct dimension reduction to improve credit card fraud detection accuracy by Xiaomei Feng and Song-Kyoo Kim [9].

Besides the implementation of machine learning models, data imbalance handling should also be carefully considered when training and evaluating credit card fraud detection model performance. Based on the research conducted by Nazim Uddin Niaz, imbalanced data could cause the machine learning model to be more influenced by majority class while neglecting the minority class [10]. Over-sampling methodology, such as SMOTE, has been proved to increase the fraud detection model performance by creating similar data points to balance the minor target class with the major class in training data [11]. Under-sampling methodology has been compared with oversampling showing different levels of supportive effects on different machine learning algorithms, of which the F-1 score could be increased to 87% [12].

3. METHODOLOGY

The purpose of this article is to explore a new methodology, namely weighing classification penalty between minority and majority target classes, within different model algorithms to emphasize on learning the minority detection during model training.

Such an algorithm will further be combined with hyperparameter tuning, over-sampling methodology and k-fold cross validation in this article to provide a comprehensive picture of how much model improvement can be realized by each step of the analysis workflow along with the training time. The analysis workflow in this article can be found in Figure 1.

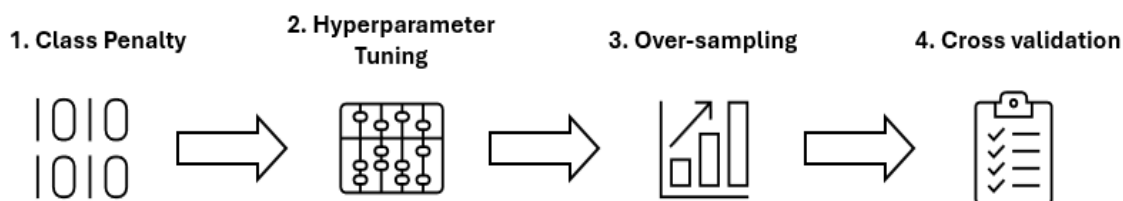


Figure 1. Analysis flow from (1) Model class penalty (2) hyperparameter tuning (3) over-sampling to handle data imbalance to (4) cross validation to get model performance.

In this analysis, we've used the masked real-world September 2013 European Credit Card Fraud dataset collected by Worldline and the Machine Learning Group of ULB (Université Libre de Bruxelles) on big data mining and fraud detection. [13]. The original dataset has recorded 284,807 real transactions including 492 (0.17%) fraud transactions as shown in Figure 2.

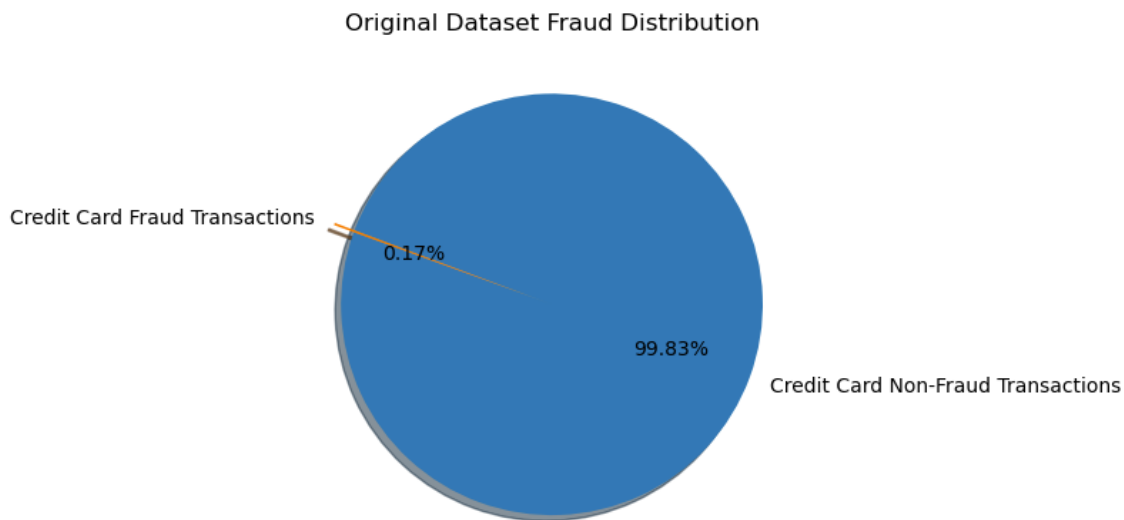


Figure 2. Original dataset fraud distribution (1) Credit card fraud transactions: 492 (0.17%) of all transactions (2) Credit card non-fraud transactions: 284,315 (99.83%) of all transactions.

3.1. Partial Penalty

The major challenge of Credit Card Fraud detection is that because of the scarcity of the positive fraud records compared with the large number of non-fraud ones, common machine learning models may simply predict most or even all results as positive to realize a high but useless accuracy because it sacrifices the prediction power of capturing the minor fraud transactions.

To handle such target class imbalance, over-sampling, such as SMOTE, and under-sampling can be implemented to either artificially create unreal data points to expand minor class or remove real data points to reduce major class [14]. However, both paths have their own risks. Over-sampling can bring in unrealistic noises into training data that may not happen in testing environment, while under-sampling can face the risk of losing massive amounts of real information by aggressively removing major target class data points.

In this study, to keep the training data as close to the real world environment as possible, an innovative solution is to train the machine learning models by partially customizing the model's penalty of wrong prediction on minor target classes so that the influence of false negative results will be amplified during the model training process to help the model focus more on studying minor target classes while keeping the training data the same as origin. Because the ratio between major class and minor class in this data is $284,315 / 492 = 577.87$, we will implement 577.87 times penalty on fraud prediction and 1 time penalty on non-fraud prediction.

The conceptual differences among over-sampling, under-sampling, and partial penalty implementation can be visualized as below:

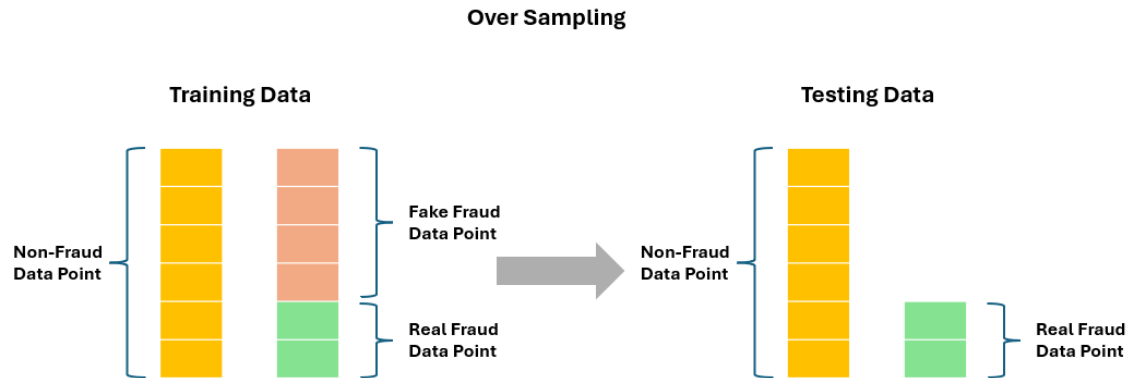


Figure 3. Over-sampling. Over-sampling will create fake data points in training data (red bars), which will introduce risk that such training data will not appear in real testing scenarios and lead to biasness of model training.

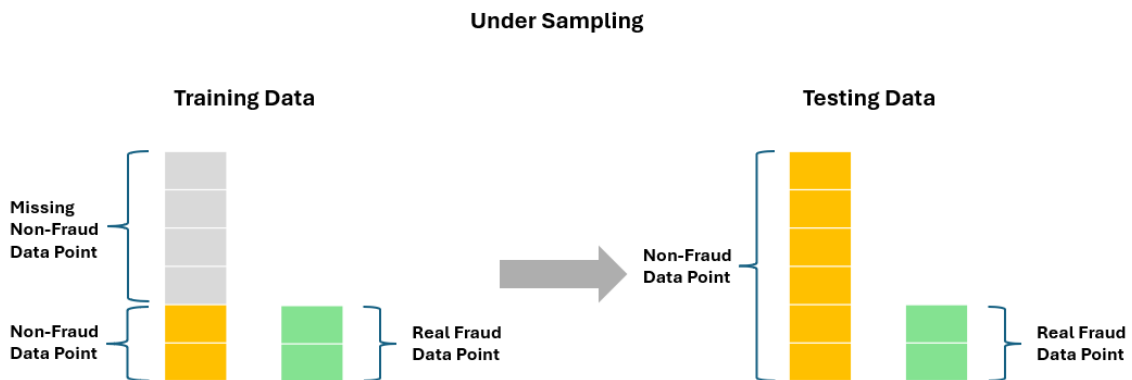


Figure 4. Under-sampling. Under-Sampling will remove major amount of majority class data point, bringing in the risk of missing significant amount of information in training data.

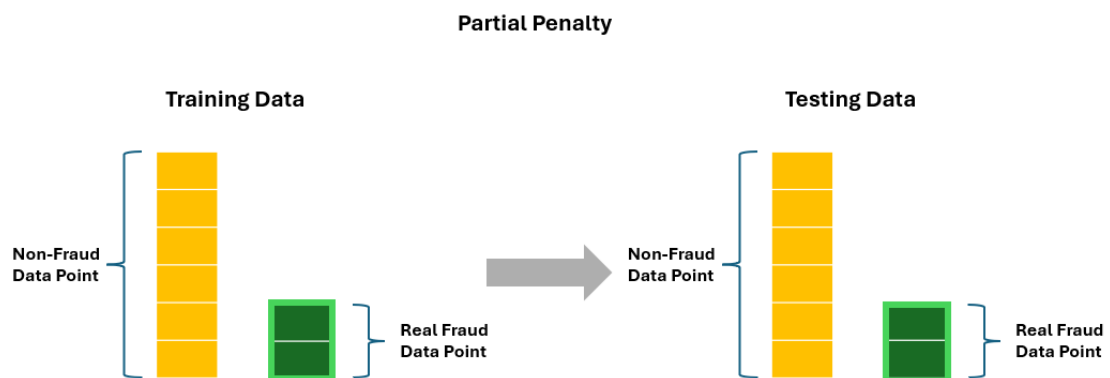


Figure 5. Partial penalty. Partial penalty model will capture the minor class while keeping the training data same as testing data.

3.2. Hyperparameter Tuning

After keeping the training data distribution the same as that of testing data, we continue to explore the best hyperparameters within class weighted models, including Logistic Regression (LR), Random Forest (RF), k Nearest Neighbor (kNN), Decision Tree (DT) and Light Gradient Boosting Model (LGB), to study how hyper parameter would impact class weighted machine learning models in credit card detection.

According to Annika Stuke's research, Bayesian Optimization hyper parameter tuning algorithms performs better than other tuning algorithms, especially in the computing speed and high dimension prediction [15]. Because our credit card fraud data has 25 dimensions with a relatively wide initial range of continuous hyper parameter values, we've decided to use Bayesian Optimization to tune the hyper parameters as shown in Table 1.

Table 1. Hyperparameter tuning selection range

Model	Hyperparameter Range
LR	tol:(0, 1), C:(0, 1), l1_ratio:(0, 1), max_iter:(100, 300)
RF	max_depth:(5, 15), min_impurity_decrease:(0, 1e-6), min_samples_leaf:(1, 10), n_estimators:(300, 400)
kNN	n_neighbors:(3, 100), algorithm:(('auto', 'ball_tree', 'kd_tree'))
DT	max_depth:(1, 15), min_impurity_decrease:(0, 1e-6), min_samples_leaf:(1, 10)
LGB	max_depth:(5, 15), num_leaves:(200, 300), learning_rate:(0.01, 0.1), n_estimators:(300, 400)

3.3. Comparison with Sampling Methodologies

In order to compare the performance among over-sampling, under-sampling and class weighted modelling, we also implemented the Synthetic Minority Over-sampling Technique (SMOTE) as well as under-sampling in the original training data based on the best hyper parameters retrieved from 3.2. section above.

In addition to the traditional way of implementing only one method to deal with the credit card fraud data imbalance, we've also introduced the test group to hybrid the class weighted modelling process with the sampling methodologies so that we finally get the final 25 versions of models for performance comparison listed as in Table 2:

Table 2. Credit card fraud modelling methodologies

Methodology	Model	Class Weight	Sampling	Hyperparameter Tuning
LR_NNN	LR			
RF_NNN	RF			
kNN_NNN	kNN	N/A	N/A	Not Tuned
DT_NNN	DT			
LGB_NNN	LGB			

LR_WNN	LR			
RF_WNN	RF			
kNN_WNN	kNN	Weighted	N/A	Not Tuned
DT_WNN	DT			
LGB_WNN	LGB			
LR_WNT	LR			
RF_WNT	RF			
kNN_WNT	kNN	Weighted	N/A	Tuned
DT_WNT	DT			
LGB_WNT	LGB			
LR_NOT	LR			
RF_NOT	RF			
kNN_NOT	kNN	N/A	Over-Sampling	Tuned
DT_NOT	DT			
LGB_NOT	LGB			
LR_NUT	LR			
RF_NUT	RF			
kNN_NUT	kNN	N/A	Under-Sampling	Tuned
DT_NUT	DT			
LGB_NUT	LGB			

3.4. Cross Validation

To evaluate the methodology performance without data leakage, we've selected a hybrid evaluation check by two steps:

- (1) Splitting the original dataset into 80% as training data and holding 20% as testing data
- (2) Cross validating each of the methodologies based on 5-fold cross validation on training data

Based on Janio Martinez Bachmann's evaluation results, in order to check the over-sampling and under-sampling methodologies without data leakage in the prediction results and over-fitting in the validation results, the sampling process should be implemented during the cross-validation not before cross-validation [16].

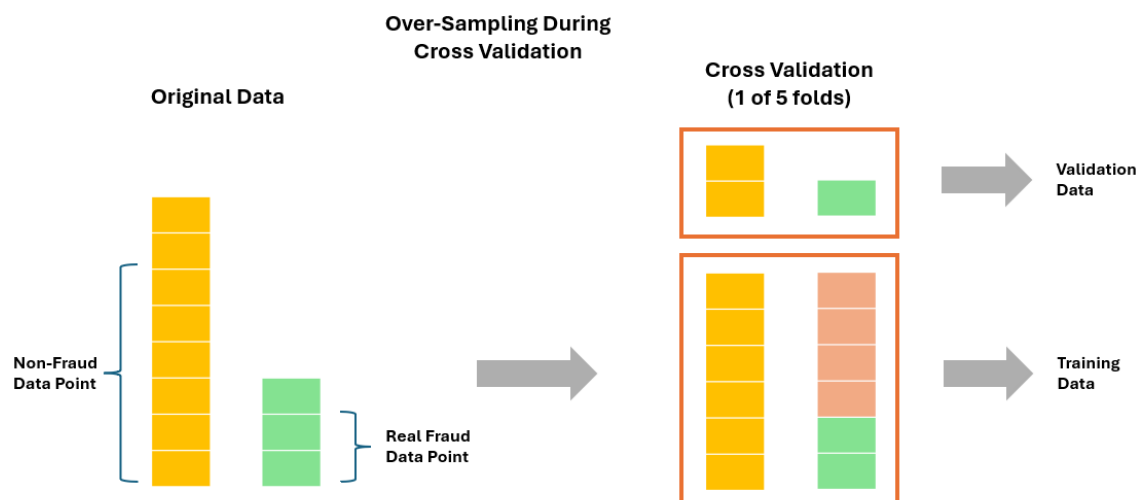


Figure 6. Over-Sampling during cross validation. For the whole training data, we split the data into 5 folds and within each iteration, we over-sampled the training data to train the model and predict based on real validation data.

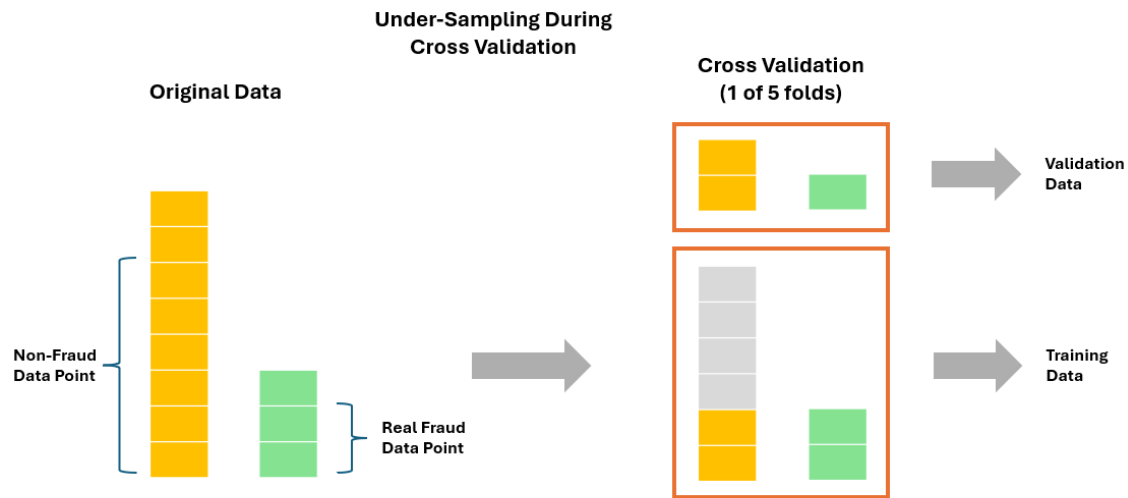


Figure 7. Under-Sampling during cross validation. For the whole training data, we split the data into 5 folds and within each iteration, we under-sampled the training data to train the model and predict based on real validation data.

As shown in Figure 6 and 7, we've combined the cross-validation process with the over-sampling and under-sampling to implement the sampling to training data within each of the 5 iterations and predict on real validation data so that the cross-validation results contain only real data points to be compared with the actual real fraud data points.

3.5. Evaluation Metrics

To understand the model prediction performance of credit card fraud in different aspects, we've used 5 evaluation metrics: ROC_AUC, Accuracy, Precision, Recall, and F-1 score.

- (1) ROC_AUC (Receiver Operating Characteristic – Area Under the Curve) is the area under the ROC curve to represent the true positive rate against the false positive rate at various classification thresholds.
- (2) Accuracy is the metric to measure how much percentage of testing target is correctly predicted by the classification model.
- (3) Precision is the metric to measure how much percentage of positive predictions (Credit Card Fraud = 1) are actual positive results.
- (4) Recall is the metric to measure how much percentage of positive results (Credit Card Fraud = 1) are correctly labelled by the classification model as positive.
- (5) F-1 score is the harmonic mean of precision and recall measuring the model's performance on positive prediction (Credit Card Fraud = 1), especially if the target variable is imbalanced.

We've applied the five-evaluation metrics to the following two validation and testing datasets:

- (1) (5-Fold CV Validation Results) 5-Fold cross validation results using the 80% training data of original dataset
- (2) (Testing Results) Testing results using the 20% testing data of original dataset

4. RESULTS

4.1. Partial Penalty and Hyperparameter Tuning

Using original dataset, class weighted partial penalty and hyper parameter tuning, we've observed the performance changes as in Figure 8 below:

Sampling	Hyperparameter	Modeling	accuracy_score	precision_score	recall_score	f1_score
Original-Data	Basic-Hyperparameter	Logistic Regression	0.999209	0.843636	0.648045	0.733017
		Random Forest	0.999593	0.944262	0.804469	0.868778
		kNN	0.999307	0.910156	0.650838	0.758958
		Decision Tree	0.999199	0.760446	0.762570	0.761506
		Light GBM	0.995950	0.221734	0.564246	<u>0.318361</u>
Class-Weighted	Basic-Hyperparameter	Logistic Regression	0.999363	0.818966	0.796089	0.807365
		Random Forest	0.999569	0.965035	0.770950	0.857143
		kNN	0.999345	0.922481	0.664804	0.772727
		Decision Tree	0.999265	0.786325	0.770950	0.778561
		Light GBM	0.999565	0.895522	0.837989	<u>0.865801</u>
	Hyperparameter-Tuned	Logistic Regression	0.999391	0.841317	0.784916	0.812139
		Random Forest	0.999490	0.865103	0.824022	0.844063
		kNN	0.999410	0.886667	0.743017	0.808511
		Decision Tree	0.999448	0.900000	0.754190	0.820669
		Light GBM	0.999640	0.963696	0.815642	<u>0.883510</u>

Figure 8. Performance improvement. 15 models are compared using sampling and hyper parameter tuning methodology.

The auc curve-based performance for class-weighted partial penalty and hyperparameter tuning can be found in the following Figure 9–11.

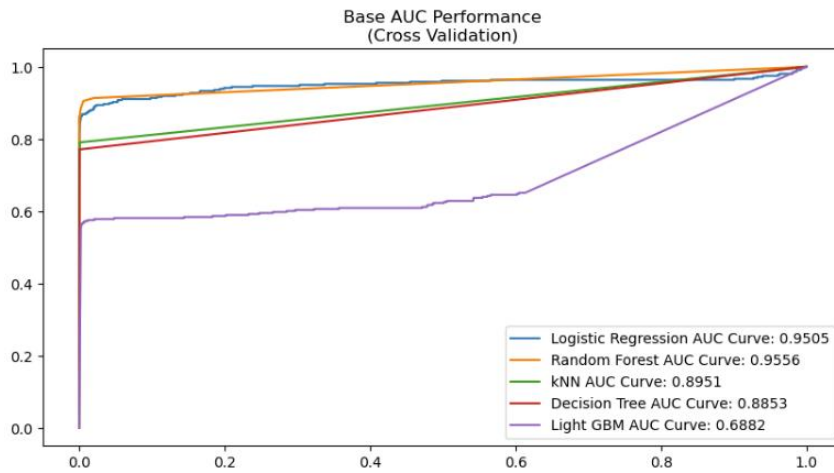


Figure 9. Base model AUC performance.

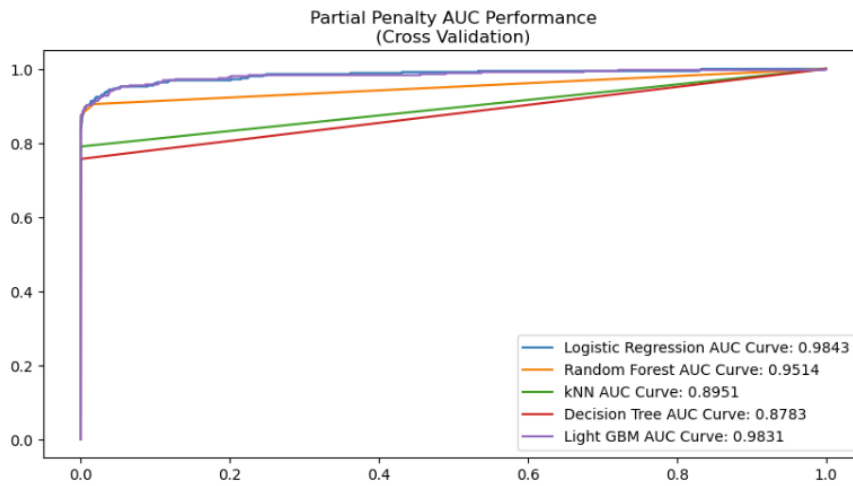


Figure 10. Partial penalty AUC performance.

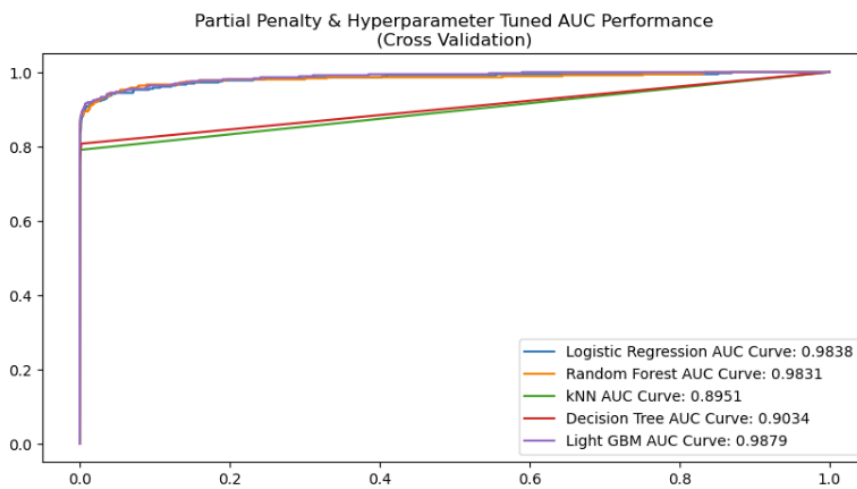


Figure 11. Partial penalty and hyperparameter tuned AUC performance.

As shown in Figure 8, because of the imbalance in original data, machine learning models have relatively low precision, recall and f1 validation performance, of which Light GBM provides the lowest 0.2217 precision, 0.5642 recall and 0.3183 f1 score among all five machine learning models. Such low prediction performance was also reflected in Figure 9 where Light GBM's AUC is 0.6882.

After implementing the class-weighted methodology of partial penalty, all those five models' performances have been improved. When it comes to Light GBM, the partial penalty approach increased the precision to 0.8955, recall to 0.8380, f1 to 0.8658, and AUC score to 0.9831.

In addition to the partial penalty, we've conducted the hyperparameter tuning for all five models, which further improved the accuracy, precision, recall, f1 score, and AUC through cross validation. As we can see in Figure 8 and 11, the best model after partial penalty and hyperparameter tuning is Light GBM, realizing 0.9637 precision, 0.8156 recall, 0.8835 f1 score, and 0.9879 AUC score.

4.2. Comparison with Sampling Methodologies

To compare with the other sampling methodologies, we've also conducted cross validation for the over-sampling (SMOTE) and under-sampling methodologies in addition to partial penalty while keeping the optimal hyperparameters the same.

The performance impact due to sampling strategy change can be found in Figure 12.

Sampling	Hyperparameter	Modeling	auc_score	accuracy_score	precision_score	recall_score	f1_score
Original-Data	Basic-Hyperparameter	Logistic Regression	0.951850	0.999190	0.829181	0.650838	0.729264
		Random Forest	0.951312	0.999583	0.943894	0.798883	0.865356
		kNN	0.895105	0.999307	0.910156	0.650838	0.758958
		Decision Tree	0.881083	0.999204	0.762570	0.762570	0.762570
		Light GBM	0.688185	0.995950	0.221734	0.564246	0.318361
Partial-Penalty	Hyperparameter-Tuned	Logistic Regression	0.983791	0.999340	0.835913	0.754190	0.792952
		Random Forest	0.983122	0.999073	0.675439	0.860335	0.756757
		kNN	0.895145	0.999410	0.886667	0.743017	0.808511
		Decision Tree	0.903360	0.999448	0.900000	0.754190	0.820669
		Light GBM	0.987899	0.999640	0.963696	0.815642	0.883510
Over-Sampling	Hyperparameter-Tuned	Logistic Regression	0.976744	0.999361	0.852654	0.780619	0.810948
		Random Forest	0.984225	0.998118	0.502423	0.845503	0.616069
		kNN	0.927478	0.987872	0.116249	0.844666	0.201683
		Decision Tree	0.867360	0.995597	0.249443	0.743537	0.368871
		Light GBM	0.983973	0.999319	0.817728	0.799014	0.805068
Under-Sampling	Hyperparameter-Tuned	Logistic Regression	0.966003	0.998589	0.633512	0.607543	0.615650
		Random Forest	0.976215	0.999361	0.857499	0.755516	0.798890
		kNN	0.942040	0.986377	0.136050	0.678623	0.211640
		Decision Tree	0.930044	0.940233	0.027787	0.890414	0.053563
		Light GBM	0.982840	0.999259	0.838275	0.703830	0.760178

Figure 12. Performance comparison. 20 models with baseline models, partial-penalty, over-sampling and under-sampling approaches.

As we can see in Figure 12, hyperparameter tuned combined with data imbalance adjustment methodologies, including partial penalty, over-sampling and under-sampling, have in general much better AUC and F1 score performances than baseline models without hyperparameter tuning. Out of all listed models, Light GBM with partial penalty methodology provides the best performance with 0.9879 AUC, 0.9996 accuracy, 0.9637 precision, 0.8156 recall and 0.8835 f1 score based on the cross validation.

The detailed ROC_AUC curves are listed in Figure 13 for four data imbalance adjustment methodologies along with the highest three AUC scores with each methodology.

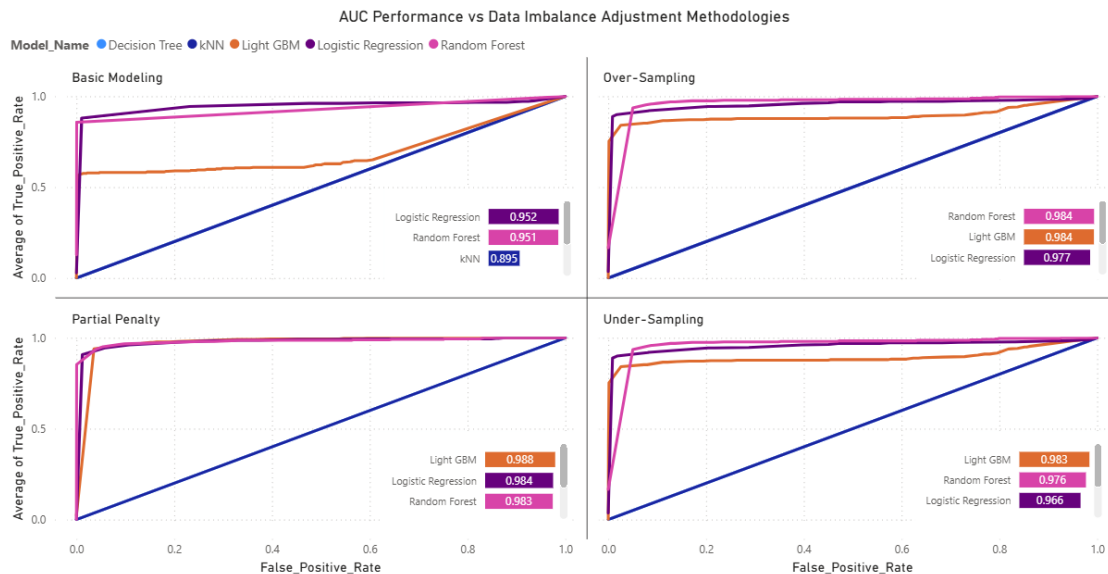


Figure 13. AUC curve comparison with highest three AUC scores shown in the lower right corner of each segment.

As shown in Figure 13, while the best model in baseline situation is logistic regression, Light GBM has been improved using the right data imbalance adjustment methodologies and outperforms Logistic Regression model in partial penalty (0.988), over-sampling (0.984), and under-sampling (0.983) methodologies.

5. CONCLUSIONS

This study compares three different methodologies to handle extreme data imbalance situations in credit card fraud prediction by traditional over-sampling, under-sampling, and an innovative way of partial penalty on the minor class of target variable. To validate the data imbalance methodology on various machine learning implementation scenario, it also used five classification models, including Logistic Regression, Random Forest, kNN, Decision Tree, and Light GBM under each imbalance handling approach. Among those three approaches, it turns out that partial penalty has better accuracy, ROC_AUC score, prediction, recall and f1 score than either the over-sampling or under-sampling. Using Light GBM, the partial penalty approach provides the highest performance of 88.35% F1 score and 98.79% AUC score.

However, such study has its limitation on the size of data. The dataset used in this study contains 284,807 transaction records with 492 fraud transactions, which can be further expanded to larger amount of transaction from real world to better validate the stability of such partial penalty framework along with the light GBM.

In conclusion, this study investigates the benefits of using partial penalty framework to improve imbalanced data for machine learning prediction. In the results, such innovative work improved the F1 score performance by 8% compared to SMOTE approach and by 11% compared to under-sampling approach. Such new methodology has also shown an over improvement across five evaluation metrics and five machine learning classification models. Such result shows a promising result that can be potentially expanded and used in larger amount of data to validate and provide better contribution to machine learning data imbalance handling process.

REFERENCES

- [1] Robert Luong. (2024). U.S. Credit card statistics and trends 2025. Retrieved from: <https://www.helcim.com/guides/credit-card-statistics-and-trends/#:~:text=card%20usage%20statistics-,Key%20findings:,during%20the%20pandemic%20in%202019>
- [2] Federal Reserve Financial Services. (2025). 2025 Diary of Consumer Payment Choice reveals U.S. consumer trends in cash usage and digitalization. Retrieved from: <https://www.frbservices.org/news/fed360/issues/060325/cash-2025-findings-diary-consumer-payment-choice>
- [3] Sanctions Editorial Team. (2025) What Is Credit Card Fraud? Common Schemes and How to Protect Your Business. Retrieved from: <https://www.sanctions.io/blog/what-is-credit-card-fraud>
- [4] Federal Trade Commission. (2025) Consumer Sentinel Network Data Book 2024. Retrieved from: https://www.ftc.gov/system/files/ftc_gov/pdf/csn-annual-data-book-2024.pdf
- [5] Fraud Net. (2024) Fraud Detection in Banking: Key Challenges and Solutions. Retrieved from: <https://www.fraud.net/resources/fraud-detection-in-banking-key-challenges-and-solutions#the-digital-era-has-become-a-curse-and-a-blessing-at-the-same-time->
- [6] Yih Bing Chu, Zhi Min Lim, Bryan Keane, Ping Hao Kong, Ahmed Rafat Elkilany, Osama Hisham Abusetta. (2023). Credit Card Fraud Detection on Original European Credit Card Holder Dataset Using Ensemble Machine Learning Technique. Retrieved from: <https://www.techscience.com/JCS/v5n1/54443/html#s3>
- [7] Vasilios Plakandaras, Periklis Gogas, Theophilos Papadimitriou & Ioannis Tsamardinos. (2022). Credit Card Fraud Detection with Automated Machine Learning Systems. Retrieved from: <https://www.tandfonline.com/doi/pdf/10.1080/08839514.2022.2086354?needAccess=true>
- [8] Al Mahmud Siam, Pankaj Bhowmik, Md Palash Uddin. (2025). Hybrid feature selection framework for enhanced credit card fraud detection using machine learning models. Retrieved from: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0326975>
- [9] Xiaomei Feng and Song-Kyoo Kim. (2024). Novel Machine Learning Based Credit Card Fraud Detection Systems. Retrieved from: https://mdpi-res.com/mathematics/mathematics-12-01869/article_deploy/mathematics-12-01869-v4.pdf?version=1719301038
- [10] Nazim Uddin Niaz, K.M. Nadim Shahariar, Muhammed J. A. Patwary. (2022). Class Imbalance Problems in Machine Learning: A Review of Methods And Future Challenges. Retrieved from: <https://dl.acm.org/doi/pdf/10.1145/3542954.3543024>
- [11] Abdul Rehman Khalid, Nsikak Owoh, Omair Uthmani, Moses Ashawa, Jude Osamor and John Adejoh. (2024). Enhancing Credit Card Fraud Detection: An Ensemble Machine Learning Approach. Retrieved from: https://mdpi-res.com/BDCC/BDCC-08-00006/article_deploy/BDCC-08-00006.pdf?version=1704285744
- [12] Noor Saleh Alfaiz and Suliman Mohamed Fati. (2022). Enhanced Credit Card Fraud Detection Model Using Machine Learning. Retrieved from: https://mdpi-res.com/electronics/electronics-11-00662/article_deploy/electronics-11-00662.pdf?version=1645426223

- [13] European Credit Card Fraud Dataset. (2021). Kaggle Credit Card Fraud Detection. Retrieved from: <https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/data>
- [14] Zahra Salekshahrezaee, Joffrey L. Leevy and Taghi M. Khoshgoftaar. (2023). The effect of feature extraction and data sampling on credit card fraud detection. Retrieved from: <https://journalofbigdata.springeropen.com/counter/pdf/10.1186/s40537-023-00684-w.pdf>
- [15] Annika Stuke, Patrick Rinke and Milica Todorovi. (2021). Efficient hyperparameter tuning for kernel ridge regression with Bayesian optimization. Retrieved from <https://iopscience.iop.org/article/10.1088/2632-2153/abee59/pdf>
- [16] Janio Martinez Bachmann. (2019). Credit Fraud || Dealing with Imbalanced Datasets. Retrieved from: <https://www.kaggle.com/code/janiobachmann/credit-fraud-dealing-with-imbalanced-datasets>