Credit Card Fraud Detection

Detecting a Fraudulent Credit Card Transaction

Nakul Gupta
Purdue University

Tejas Jadhav
Purdue University

Abstract

Financial fraud poses a persistent challenge in digital transactions, demanding advanced detection methods. This project delves into applying data mining techniques and employing machine learning models like logistic regression, XGBoost, and LightGBM for fraud detection. Utilizing transaction and identity datasets provided by Vesta Corp. over Kaggle, we employ Principal Component Analysis (PCA) and data preprocessing to enhance feature representation. The Synthetic Minority Over-sampling Technique (SMOTE) addresses class imbalance. Our investigation involves training and evaluating multiple models, considering both imbalanced and balanced datasets. Logistic regression models are scrutinized under various balancing strategies, revealing insights into their efficacy. XGBoost and LightGBM models explore various balancing strategies, for instance, weight balancing and SMOTE techniques, revealing insights into their efficacy. Evaluation metrics, including accuracy score, confusion matrix, precision, recall, and f1-score, are utilized to provide a comprehensive assessment and performance of the models used. The research concludes with key findings, emphasizing the significance of preprocessing and balancing in influencing model outcomes and providing valuable insights for advancing the field of fraud detection in financial transactions.

1. Introduction

In the world of financial transactions, the risk of fraud is on the rise, calling for advanced methods to detect and prevent it promptly. In this context, our project emphasizes the crucial role of data mining techniques and machine learning in strengthening the strategies to identify and prevent fraud in credit card transactions. Despite the challenges, we aim to showcase how these sophisticated tools can be vital in securing financial systems.

1.1. Problem Statement

This project is guided by two overarching objectives:

- To critically assess the influence of the use of data mining techniques like Principal Component Analysis (PCA), data preprocessing, and balancing techniques on the overall performance of these machine learning models.
- To implement and rigorously evaluate the following machine learning models tailored for the purpose of fraud detection:
    1. Logistic Regression: Logistic Regression serves as our baseline model due to its simplicity, interpretability, and efficiency, providing a foundational benchmark against which we can evaluate the performance of more complex machine learning algorithms in fraud detection.
    2. XGBoost: XGBoost is well-suited due to its ability to handle imbalanced datasets, learn intricate patterns in data, and deliver high predictive accuracy, making it an effective tool for capturing subtle anomalies indicative of fraudulent activities.
    3. LightGBM: Light Gradient Boosting Machine (LightGBM) is particularly effective, thanks to its efficiency in handling large datasets, optimizing training speed, and providing high accuracy, making it well-equipped to discern complex patterns inherent in fraudulent transactions.

1.2. Overview of the Data

The data set is divided into two parts: Transaction and Identity, which are joined by TransactionID. Not all transactions have corresponding identity information. The following is the number of samples in each of the datasets:

- Transaction: 507k samples
  Transaction features:

  - TransactionDT, TransactionAMT, ProductCD, card1 - card6, addr, dist, P_emaildomain and R_emaildomain, C1-C14, D1-D15, M1-M9, Vxxx.
  - A total of 392 features
- Identity: 144k samples
  Identity features:
  - DeviceType, DeviceInfo, id01 - id38.
  - A total of 40 features

1.3. Contribution

In this project, all of the team members have done an excellent job in contributing towards the success of the project. Following is the contribution by each of the team members who was a part of this project:

1. **Nakul Gupta:** Nakul played a vital role in the success of the project as he was responsible for the data analysis to check the distribution of classes, both for cleaning and checking imbalance in the given data. He also performed the balancing of the data using the SMOTE technique and implemented the XGBoost model for this project.

2. **Tejas Jadhav:** Tejas was a crucial part of the project as he was extensively invested in the implementation of the Principal Component Analysis of the given data. He also performed the correlation analysis of the given features to check which features to choose from the given data to apply PCA on. He was also responsible for implementing the baseline model, Logistic Regression for the project.

3. **Shishir Yadav:** Shishir played an important role as he glued all the analysis done by Nakul and Tejas to make the project happen. He was responsible for performing the data pre-processing (like removing null values, and scaling) based on Nakul's analysis and he assembled all the data (the PCA data with the remaining original data) to get it ready for training. Furthermore, he implemented the LightGBM model for this project.

2. Related Works

Recent studies in predictive modeling and classification strategies have explored innovative approaches to enhance model performance. Cai and He (2022) proposed a hybrid XGBoost and TabNet model, prioritizing AUC scores and accuracy by replacing missing values. However, their evaluation lacked a detailed examination of recall.

Verma and Chandra (2023) introduced the RepuTE Framework for enhancing trust in fog computing, achieving a 99.99% accuracy rate in classifying DoS/DDoS and Sybil attacks. Yet, concerns persist regarding its efficacy in imbalanced scenarios like credit card fraud datasets.

Malik et al. (2022) evaluated seven hybrid models, highlighting the success of AdaBoost with LGBM in ROC scores. Notably, accuracy metrics were omitted, raising questions about the overall performance, especially given the observed discrepancy in the Naive Bayes model's recall and accuracy.

Cochrane et al. (2021) combined LR, DT, and logistic regression models, emphasizing a formula for recall and precision. However, the absence of accuracy metrics leaves uncertainties about the comprehensive performance of their methodology.

3. Methodology

We first performed data preprocessing, feature selection, and handled the data imbalance to train various models. In the data exploration and preprocessing phase, essential techniques were applied to optimize the dataset, including scaling numerical features, handling missing values, and encoding categorical features. Feature selection involved the use of Principal Component Analysis (PCA) to achieve dimensionality reduction and address redundancy within a subset of transaction data. The handling of class imbalance was pivotal, with the Synthetic Minority Over-sampling Technique (SMOTE) and weight-balancing techniques, employed strategically.

Moving to the model training and evaluation phase, logistic regression, XGBoost, and LightGBM models were trained and assessed. Logistic regression models underwent training with and without SMOTE balancing, while XGBoost and LightGBM models were trained with various strategies, including class weights and SMOTE.

This comprehensive approach, integrating preprocessing, feature selection, and handling class imbalance, establishes a robust framework for credit card fraud detection. The chosen techniques align with best practices in the field, contributing to the creation of effective and fair machine-learning models for fraud prevention.

4. Data Exploration and Preprocessing

4.1. Data Preprocessing

In this crucial phase, we employed a series of preprocessing techniques to optimize the dataset for effective model training:

- **Scaling Numerical Features:** Achieving consistency in numerical feature scales is crucial. Through min-max scaling, we normalized the numerical values to prevent disproportionate influence and to promote fair and unbiased model training.

- **Handling Missing Values:** Identifying and addressing missing data ensures completeness and accuracy in our dataset. Various strategies are employed, such as the removal of the features that had more than 80% of their values as 'NaN'. Furthermore, mean imputation of missing values was done to enhance the overall integrity of the dataset.

- **Encoding Categorical Features:** Categorical features, which are non-numeric, are transformed into a format suitable for model interpretation. This conversion facilitates the inclusion of categorical information in our machine-learning models.

By meticulously implementing these preprocessing techniques, we ensure that our data is refined, comprehensive, and well-suited for the subsequent feature selection and training of machine learning models. This process is fundamental in optimizing the dataset's quality and ensuring the robustness of our analytical endeavors.

4.2. Feature Selection

In a targeted approach, Principal Component Analysis (PCA) is employed on a subset of the transaction data. This strategic utilization aims to achieve two primary objectives:

1. **Dimensionality Reduction:** By transforming a set of correlated variables (Vxxx features) into a new set of uncorrelated variables (principal components), PCA mitigates the curse of dimensionality. This not only simplifies the data but also improves computational efficiency.

2. **Addressing Redundancy:** The Vxxx features exhibit correlations (as shown in the correlation matrices below), and since they are engineered from the raw data by the Vesta Corporation, there might be inherent redundancy. PCA serves to capture the essential information from these features while eliminating redundancy, ensuring a more efficient and streamlined representation of the data.

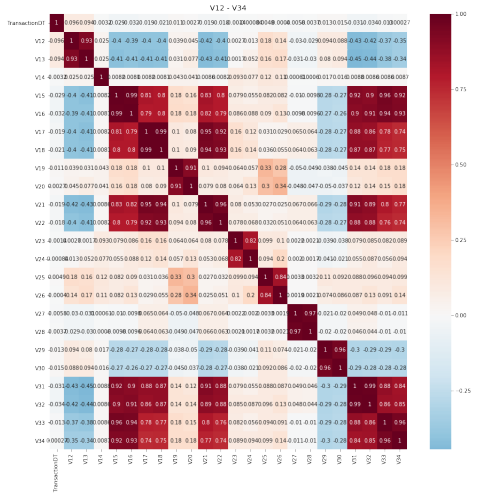

Figure 1: Correlation Matrix for Features V12-V34

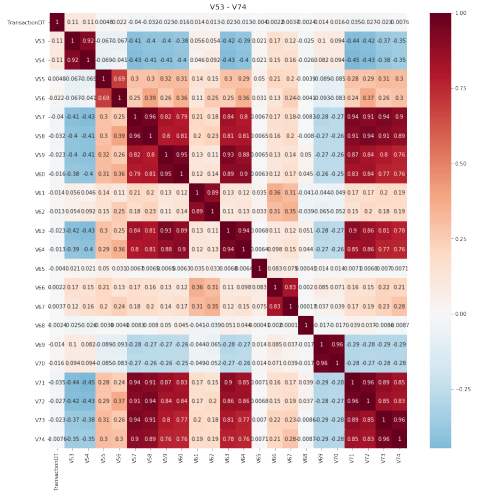

Figure 2: Correlation Matrix for Features V53-V74

This application of PCA aligns with the need to enhance computational efficiency, address feature correlations, and optimize the information gleaned from the engineered features, thereby contributing to a more effective and focused dataset for subsequent analysis.

4.3. Handling Imbalance

In this pivotal phase, we systematically address class imbalance concerns within our target variable ('isFraud') through the following steps:

- **Imbalance Checking:** To guarantee the integrity of our model, we meticulously examine the distribution of the target variable ('isFraud'). By visualizing this distribution using a countplot, we gain insights into potential class imbalance issues. Identifying such imbalances is crucial for fair and unbiased model training.

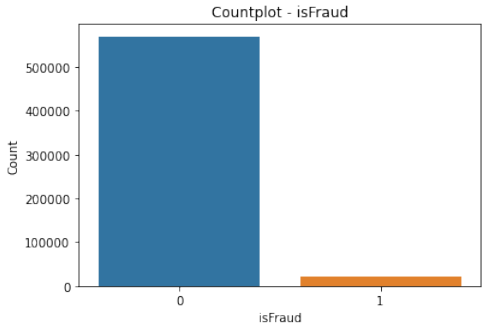

Figure 3: Distribution of the Target Variable 'isFraud'

- **SMOTE Balancing:** To address the class imbalance, we implemented the Synthetic Minority Oversampling Technique (SMOTE), which is an oversampling technique. This technique strategically augments the representation of the minority class, ensuring a more balanced dataset. By synthetically generating instances of the minority class, SMOTE enhances the robustness of our models, allowing them to learn from all classes more effectively.

- **Weight Balancing:** To address the class imbalance, we also used the inbuilt methods available with XGBoost and LightGBM that automatically handle class imbalance through the use of class weights. Class weights assign a higher weight to the minority class during the training process, effectively placing more emphasis on correctly predicting instances of the minority class (fraudulent transactions).

These measures collectively contribute to a more equitable and representative dataset, fostering fairness and accuracy in our fraud detection model.

5. Model Training and Evaluation

In this phase, we trained and evaluated logistic regression, XGBoost, and LightGBM models, employing various balancing techniques to ensure robust performance. The key components of this analysis include:

5.1. Logistic Regression Models

Two logistic regression models undergo training and evaluation:

- **Imbalanced Model:** Trained without any balancing techniques.

- **Balanced Model:** Trained with Synthetic Minority Over-sampling Technique (SMOTE) balancing to address class imbalance.

5.2. XGBoost Models

Three XGBoost models are trained and evaluated:

- **Imbalanced Model:** Trained without addressing class imbalance.

- **Weight-Balanced Model:** Trained by adjusting class weights to handle imbalance using the inbuilt parameter 'scale_pos_weight' while building the model.

- **Balanced Model:** Trained with SMOTE to enhance minority class representation.

5.3. LightGBM Models

Similar to XGBoost models, two LightGBM models are trained and evaluated:

- **Imbalanced Model:** Trained without addressing class imbalance.

- **Weight-Balanced Model:** Trained by adjusting class weights to handle imbalance using the inbuilt parameter 'is_unbalance' while building the model.

- **Balanced Model:** Trained with SMOTE to enhance minority class representation.

6. Results

The performance of each model is meticulously evaluated using a range of metrics, including accuracy score, confusion matrix, precision, recall, and f1-score. This comprehensive analysis provides valuable insights into how balancing techniques influence and optimize model outcomes, guiding our understanding of the most effective strategies for fraud detection.

6.1. Logistic Regression Models

Following are the results for the Logistic Regression models built:

Type of Dataset	Classification Report
Imbalanced	Classification Report of Logistic Regression on Testing Data (Imbalanced) precision recall f1-score support 0 0.96 1.00 0.98 170821 1 0.00 0.00 0.00 6341 accuracy 0.96 177162 macro avg 0.48 0.50 0.49 177162 weighted avg 0.93 0.96 0.95 177162
Balanced	Classification Report of Logistic Regression on Testing Data (Balanced) precision recall f1-score support 0 0.62 0.78 0.69 170944 1 0.70 0.53 0.60 170983 accuracy 0.65 341927 macro avg 0.66 0.65 0.65 341927 weighted avg 0.66 0.65 0.65 341927

Following are the Confusion Matrices for the models:

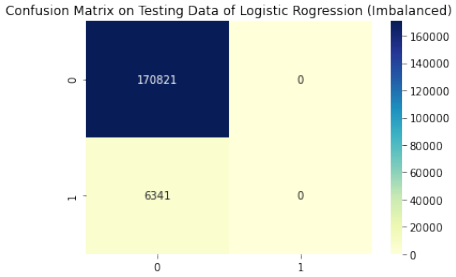

Figure 4: Confusion Matrix for Logistic Regression Model (Imbalanced)

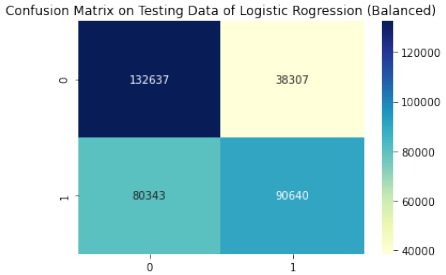

Figure 5: Confusion Matrix for Logistic Regression Model (Balanced)

6.2. XGBoost Models

Following are the results for the XGBoost models built:

Type of Dataset	Classification Report
Imbalanced	Classification Report of XGBoost on Testing Data (Imbalanced) precision recall f1-score support 0 0.98 1.00 0.99 170821 1 0.91 0.49 0.64 6341 accuracy 0.98 177162 macro avg 0.94 0.74 0.81 177162 weighted avg 0.98 0.98 0.98 177162
Weight Balanced	Classification Report of XGBoost on Testing Data (Weight Balanced) precision recall f1-score support 0 0.99 0.93 0.96 170821 1 0.31 0.80 0.44 6341 accuracy 0.93 177162 macro avg 0.65 0.87 0.70 177162 weighted avg 0.97 0.93 0.94 177162
Balanced	Classification Report of XGBoost on Testing Data (Balanced) precision recall f1-score support 0 0.98 0.99 0.99 170944 1 0.99 0.98 0.99 170983 accuracy 0.99 341927 macro avg 0.99 0.99 0.99 341927 weighted avg 0.99 0.99 0.99 341927

Following are the Confusion Matrices for the models:

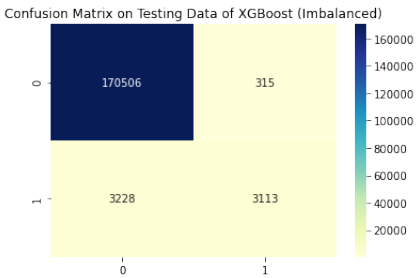

Figure 6: Confusion Matrix for XGBoost Model (Imbalanced)

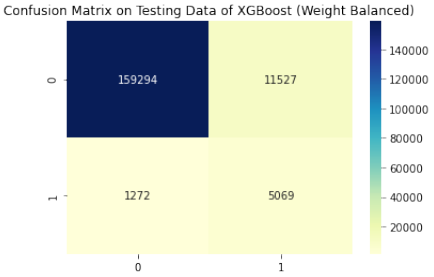

Figure 7: Confusion Matrix for XGBoost Model (Weight Balanced)

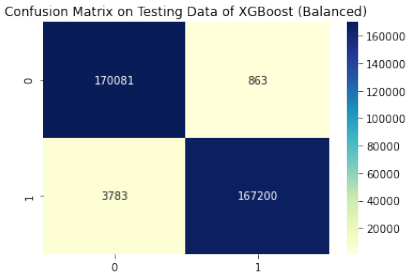

Figure 8: Confusion Matrix for XGBoost Model (Balanced)

6.3. LightGBM Models

Following are the results for the LightGBM models built:

Type of Dataset	Classification Report
Imbalanced	Classification Report of LightGBM on Testing Data (Imbalanced) precision recall f1-score support 0 0.98 1.00 0.99 170821 1 0.89 0.42 0.57 6341 accuracy 0.98 177162 macro avg 0.93 0.71 0.78 177162 weighted avg 0.98 0.98 0.97 177162
Weight Balanced	Classification Report of LightGBM on Testing Data (Weight Balanced) precision recall f1-score support 0 0.99 0.89 0.94 170821 1 0.21 0.81 0.34 6341 accuracy 0.89 177162 macro avg 0.60 0.85 0.64 177162 weighted avg 0.96 0.89 0.92 177162
Balanced	Classification Report of LightGBM on Testing Data (Balanced) precision recall f1-score support 0 0.96 0.99 0.98 170944 1 0.99 0.96 0.98 170983 accuracy 0.98 341927 macro avg 0.98 0.98 0.98 341927 weighted avg 0.98 0.98 0.98 341927

Following are the Confusion Matrices for the models:

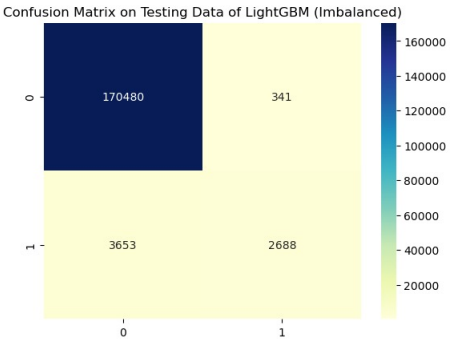

Figure 6: Confusion Matrix for LightGBM Model (Imbalanced)

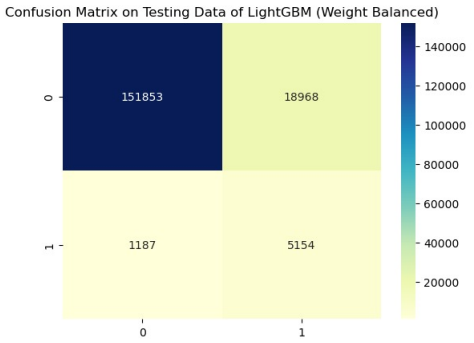

Figure 7: Confusion Matrix for LightGBM Model (Weight Balanced)

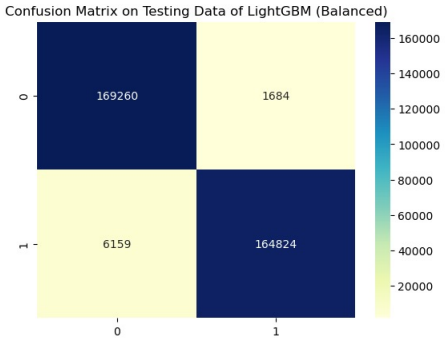

Figure 8: Confusion Matrix for LightGBM Model (Balanced)

7. Conclusion

As we conclude our project, we meticulously summarize the performances of our models under diverse balancing techniques. Our analysis unfolds key findings, giving crucial insights into fraud detection methodologies.

7.1. Key Highlights

LGBM, Our Top Performer: Notably, LightGBM (LGBM) emerges as the frontrunner, exhibiting exceptional accuracy and an elevated F1-score. Particularly noteworthy is its outstanding performance under weight balancing, showcasing a fair value of precision and recall. Even though precision is a little low but high recall is of higher priority as we need to flag as many fraudulent transactions as possible. This explains the balanced precision and recall values based on the bias-variance tradeoff argument.

7.2. Insights from Data Preprocessing and Balancing

- **PCA's Strategic Role:** Our exploration into dimensionality reduction through Principal Component Analysis (PCA) reveals its strategic role in simplifying and optimizing the dataset. By capturing essential features and addressing potential redundancies, PCA contributes to enhancing the efficiency of our fraud detection models.

- **Weight Balancing Triumphs:** In our project, we observed that weight balancing outperforms SMOTE balancing, especially evident in LGBM's superior results. This underscores the significance of selecting an adept balancing technique tailored to the intricacies of credit card fraud detection.

This project, through its insightful findings, not only refines our understanding of effective fraud detection but also underscores the pivotal role of data pre-processing and balancing methodologies, highlighting their indispensable contribution to addressing the inherent challenges in credit card fraud detection.

8. References

1. Bakhtiari, S., Nasiri, Z., & Vahidi, J. (2023). Credit card fraud detection using ensemble data mining methods. Multimedia Tools and Applications, 82(19), 29057–29075. https://doi.org/10.1007/s11042-023-14698-2

2. Cai Q., He J. Credit Payment Fraud detection model based on TabNet and Xgboost; Proceedings of the 2022 2nd International Conference on Consumer Electronics and Computer Engineering (ICCECE); Guangzhou, China. 14–16 January 2022; Piscataway, NJ, USA: IEEE; 2022. pp. 823–826. https://ieeexplore.ieee.org/abstract/document/9712842

3. Cochrane N., Gomez T., Warmerdam J., Flores M., Mccullough P., Weinberger V., Pirouz M. Pattern Analysis for Transaction Fraud Detection; Proceedings of the IEEE Annual Computing and Communication Workshop and Conference (CCWC); Las Vegas, NV, USA. 27–30 January 2021 https://ieeexplore.ieee.org/abstract/document/9376045

4. Dornadula, V. N., & S, G. (2019). Credit card fraud detection using machine learning algorithms. Procedia Computer Science, 165, 631–641. https://doi.org/10.1016/j.procs.2020.01.057

5. Malik E.F., Khaw K.W., Belaton B., Wong W.P., Chew X. Credit card fraud detection using a new hybrid machine learning architecture. Mathematics. 2022;10:1480. doi: 10.3390/math10091480. https://www.mdpi.com/2227-7390/10/9/1480

6. Mohammed, N. H., & Reddy Maram, S. C. (2022). Fraud detection of credit cards using logistic regression. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.4135514

7. Purwar, A., & M. (2023). Credit card fraud detection using XGBoost for Imbalanced Data Set. Proceedings of the 2023 Fifteenth International Conference on Contemporary Computing. https://doi.org/10.1145/3607947.3607986

8. Verma R., Chandra S. RepuTE: A soft voting ensemble learning framework for reputation-based attack detection in fog-IoT milieu. Eng. Appl. Artif. Intell. 2023;118:105670. doi: 10.1016/j.engappai.2022.105670. https://doi.org/10.1016/j.engappai.2022.105670