Fraud Detection in Online Transactions using Machine Learning Techniques

Samanatha Lee

Table of Contents:

- Abstract
- Introduction
- Data Collection & Preprocessing
- Exploratory Data Analysis (EDA)
- Data Processing
- Feature Engineering
- Model Selection & Evaluation
- Results & Discussion
- Conclusion & Future Direction

Abstract:

This project outlines an approach to improving fraud detection in online payment transactions through machine learning techniques. Utilizing a dataset comprising transactional information, we explore diverse machine learning models and strategies to effectively identify fraudulent transactions. Our methodology encompasses exploratory data analysis (EDA), feature engineering, model selection, and evaluation to develop accurate and reliable fraud detection models. Through empirical analysis and comparison of machine learning algorithms, we demonstrate the efficacy of our approach in detecting fraudulent activities while minimizing false positives and false negatives. Our findings underscore the importance of leveraging advanced machine learning techniques for fraud detection and offer insights for financial institutions to fortify their fraud detection systems.

Introduction:

Fraudulent activities in financial transactions pose significant risks to both financial institutions and consumers. Detecting and preventing fraudulent transactions is important to maintain the integrity of financial systems and protect stakeholders from financial losses. Traditional rule-based approaches to fraud detection often fall short in identifying sophisticated and evolving fraud schemes. Machine learning techniques offer a promising avenue for enhancing fraud detection capabilities by leveraging data-driven insights and predictive modeling algorithms. In this paper, we present a comprehensive framework for enhancing fraud detection in financial transactions using machine learning techniques.

Data Collection & Preprocessing:

The foundation of effective fraud detection lies in the quality and integrity of the data. Utilizing a dataset containing transactional information, including transaction amounts, timestamps, merchant information, and customer demographics. Prior to model training, conducted data preprocessing steps, including handling missing values, encoding categorical variables, and scaling numerical features. These preprocessing steps ensured the consistency and suitability of the dataset for machine learning analysis.

The data pre-processing phase involves loading the 'fraud_dataset.csv' file into a DataFrame named df and examination of its structure and attributes using methods like df.head(), df.info(), and df.describe(). With 6,362,620 entries and 11 columns, including features such as 'step', 'type', 'amount', 'nameOrig', 'oldbalanceOrg', 'newbalanceOrig', 'nameDest', 'oldbalanceDest', 'newbalanceDest', 'isFraud', and 'isFlaggedFraud', the dataset presents a mix of categorical and numerical data.

To ensure consistency and clarity, the 'oldbalanceOrg' column is renamed to 'oldbalanceOrig' using the df.rename() method. Subsequently, the dataset is checked for duplicate entries using df.duplicated().sum(), with zero duplicates found, affirming the absence of redundant rows. This preprocessing establishes a solid foundation for subsequent data analysis and modeling tasks, ensuring data integrity and consistency throughout the project.

Exploratory Data Analysis (EDA):

Exploratory data analysis (EDA) uncovering insights into financial transaction fraud, with a dataset comprising 6,362,620 entries across 11 distinct columns. The distribution of transaction types highlighted the prevalence of CASH_OUT and PAYMENT types, with TRANSFER transactions representing a significant portion shown (Fig.1a). Although fraudulent transactions constituted only 0.13% of the dataset (Fig.1c), they featured substantially higher average transaction amounts compared to non-fraudulent transactions, predominantly involving CASH_OUT and TRANSFER types (Fig.1b). Temporal analysis revealed variations in non-fraudulent transaction frequencies across months, indicating potential temporal trends influenced by external factors (Fig.2b).

Transaction by Type

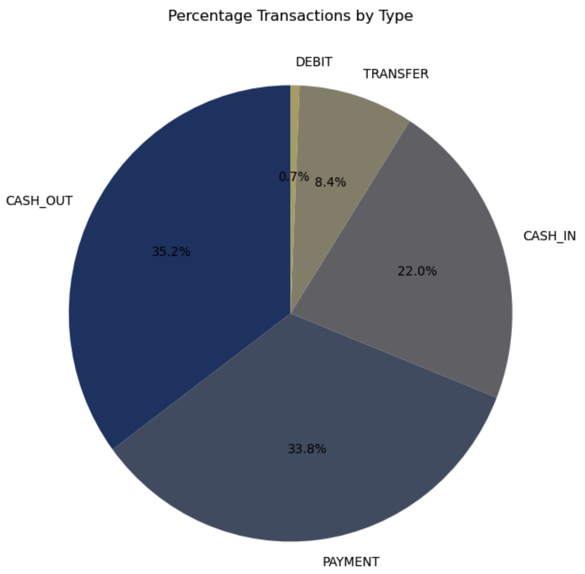


Fig. 1a

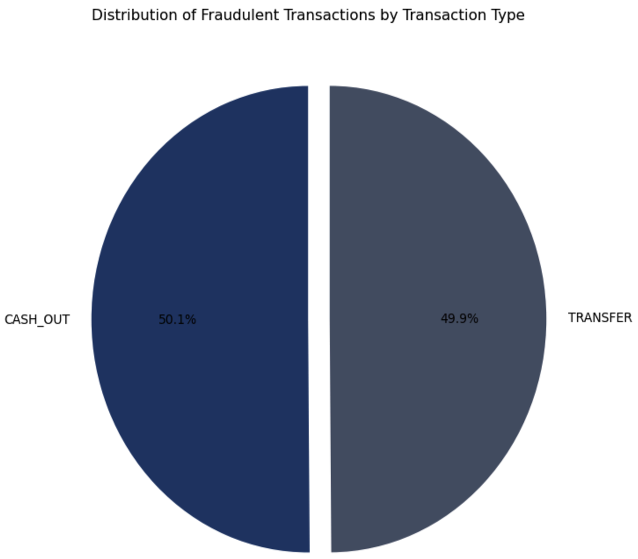


Fig.1b

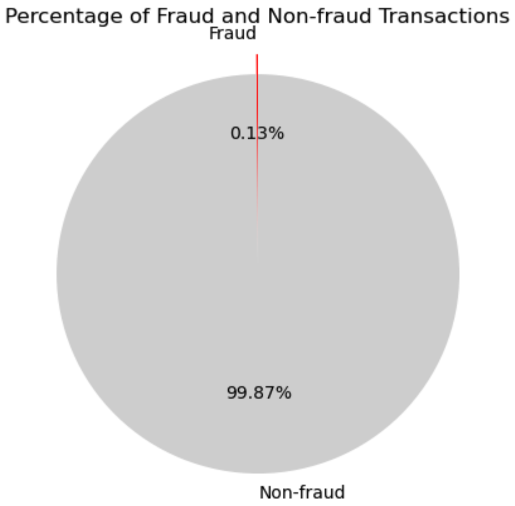


Fig. 1c

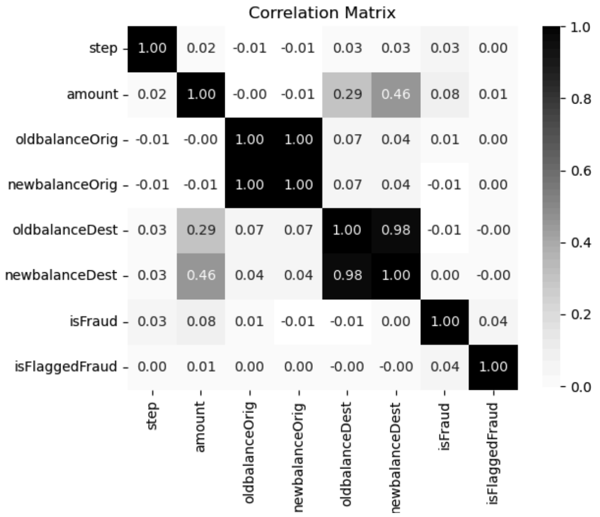


Fig. 2a

Temporal Analysis

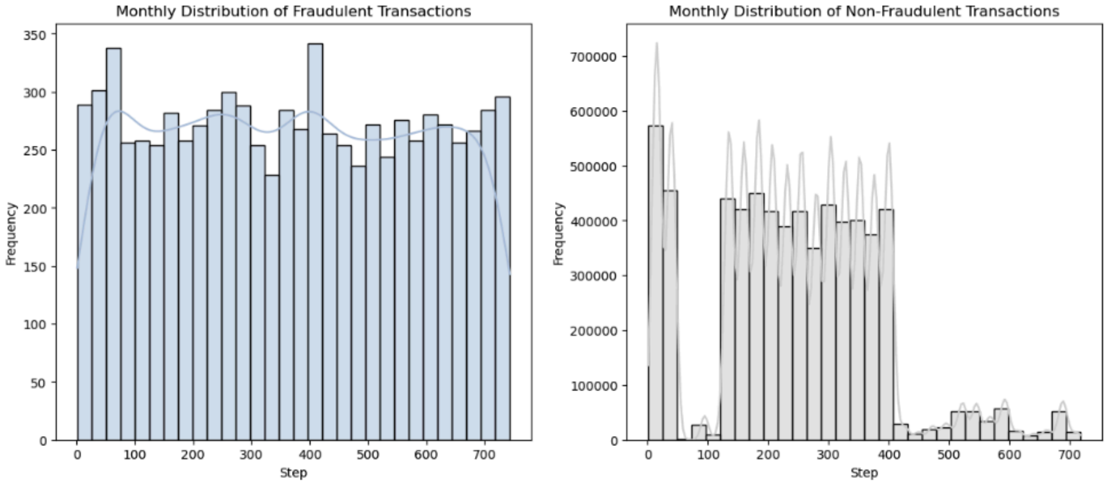


Fig. 2b

Additionally, fraudulent transactions, originating exclusively from customer accounts, often exhibited substantial initial balances that were depleted post-transaction, hinting at suspicious fund movements. Further analysis unveiled correlations between transaction amounts, balances in destination accounts, and fraudulent occurrences. Transaction amount and new balances in destination accounts are moderately positively correlated at 0.46, indicating that higher transaction amounts correspond to increased balances in destination accounts. Additionally, a strong positive correlation of 0.98 exists between new and old balances in destination accounts, emphasizing the link between previous and current balances in these accounts(Fig. 2a).

The findings underscored the critical role of EDA in understanding transaction characteristics, patterns, and correlations, facilitating the development of effective fraud detection methodologies in financial systems.

Fraud and Non-Fraud Transactions

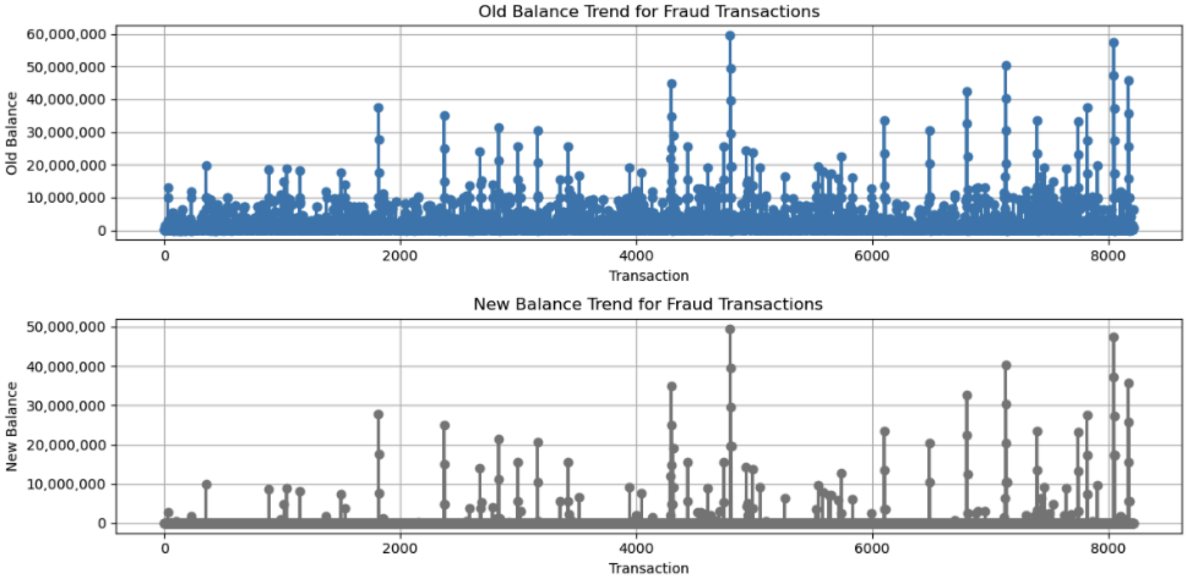


Fig. 3b

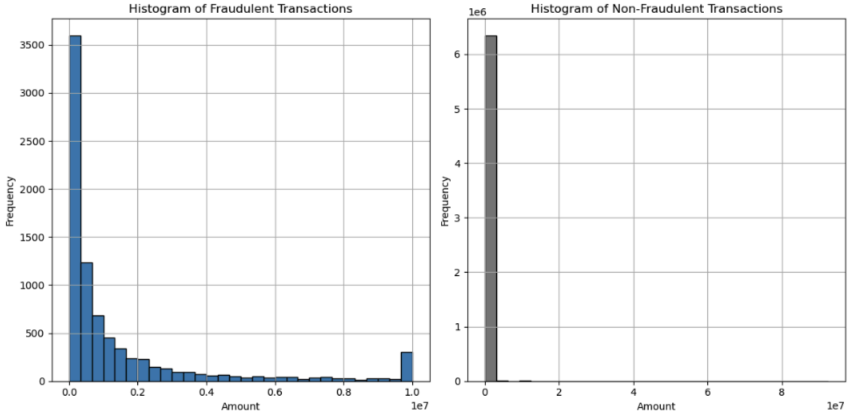


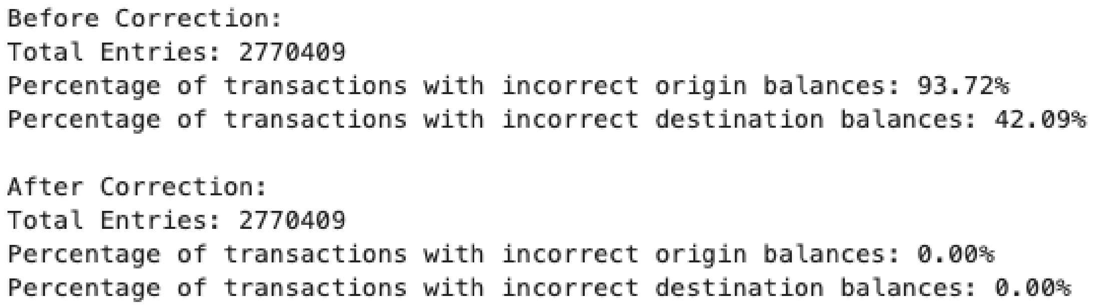
Fig. 3c

The histograms illustrate the distribution of transaction amounts, with fraudulent transactions displaying a broader range from 181.00 to 6,311,409.28, encompassing values like 2,806.00 and 20,128.00. In contrast, non-fraudulent transactions are predominantly concentrated within lower to moderate amounts, ranging from 1,864.24 to 6,311,409.28, with a single bin representation (Fig.3c). This disparity underscores the diverse spectrum of fraudulent transaction amounts compared to the more concentrated nature of non-fraudulent transactions.

Data Processing:

In the data processing phase, the code performs essential tasks on a financial transaction dataset to ensure its integrity and accuracy. Initially, irrelevant columns such as 'isFlaggedFraud', 'nameDest', and 'nameOrig' are removed, streamlining the dataset for further analysis. Moreover, transactions categorized as 'CASH_IN', 'DEBIT', and 'PAYMENT' are filtered out to focus solely on relevant transaction types, enhancing the dataset's relevance to the analysis.

The code then calculates the percentages of transactions with incorrect origin and destination balances before any correction. It identifies transactions where the old balance of the origin account or the destination account does not align with the transaction amount and proceeds to rectify these balances accordingly. After the correction process, the percentages of transactions with incorrect balances reduce to zero (Fig.4). This correction is pivotal for ensuring the dataset's accuracy, especially in fraud detection applications, where precise transactional information is important for effective analysis. By enhancing the reliability of the dataset, the correction process sets a solid foundation for subsequent analysis and modeling tasks.



```
Before Correction:
Total Entries: 2770409
Percentage of transactions with incorrect origin balances: 93.72%
Percentage of transactions with incorrect destination balances: 42.09%

After Correction:
Total Entries: 2770409
Percentage of transactions with incorrect origin balances: 0.00%
Percentage of transactions with incorrect destination balances: 0.00%
```

Fig. 4

Feature Engineering:

In feature engineering, we used LabelEncoder to convert the categorical 'type' feature into numerical values, enabling efficient processing by machine learning algorithms. The dataset was split into feature variables (X) and the target variable (Y), comprising transaction attributes like 'step', 'amount', 'oldbalanceOrig', 'newbalanceOrig', 'oldbalanceDest', and 'newbalanceDest'. Using train_test_split, 80% of the data was used for model training, while 20% used for evaluation. This process establishes the framework for constructing and assessing fraud detection models, essential for detecting fraudulent transactions effectively.

Model Selection & Evaluation:

Using machine learning models, including logistic regression, random forest, XGBoost, and LightGBM, were evaluated for their effectiveness in detecting fraudulent transactions. Techniques such as cross-validation and hyperparameter tuning to optimize model performance and mitigate overfitting. Evaluation metrics such as precision, recall, F1 score, and receiver operating characteristic (ROC) curve analysis were used to assess the performance of the models.

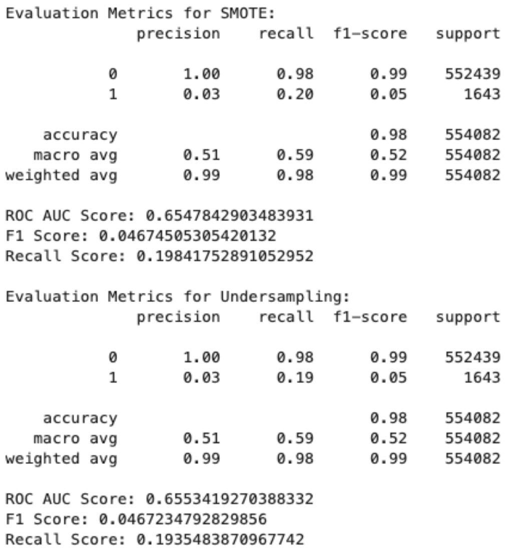
In the process of model selection and evaluation, a diverse array of machine learning algorithms was explored to identify the most effective approach for detecting fraudulent transactions. Among the models considered were logistic regression, random forest, XGBoost, and LightGBM. These models were chosen for their distinct characteristics and applicability in addressing the intricate nature of fraudulent activity detection in financial transactions. To ensure robustness and reliability, techniques such as cross-validation and hyperparameter tuning were utilized to optimize each model's performance and mitigate the risk of overfitting.

Initially, logistic regression was implemented with additional techniques like SMOTE (Synthetic Minority Over-sampling Technique) and undersampling to address the imbalance between fraudulent and non-fraudulent transactions. SMOTE generates synthetic samples of the minority class to balance the dataset, while undersampling involves randomly selecting a subset of the majority class samples. However, due to logistic regression's limitations in handling class imbalance and capturing complex patterns within the data, an alternative approach was pursued. The modeling process continued using undersampled data for the rest of the algorithms. Undersampling was favored for its simplicity and effectiveness in balancing class distributions. By addressing the class imbalance issue, undersampling enhanced the models' ability to accurately identify fraudulent transactions while minimizing bias towards the majority class. Models built on undersampled data are better generalized to unseen instances and improve overall performance in detecting fraudulent activities.

Results & Discussion:

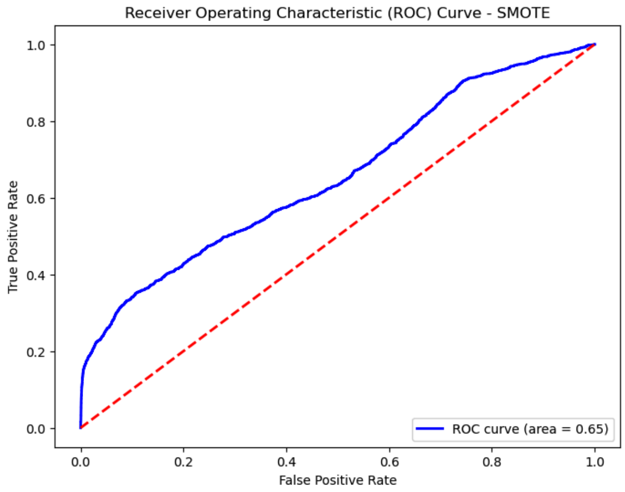
Our empirical analysis demonstrated the effectiveness of machine learning techniques in detecting fraudulent transactions. LightGBM emerged as the optimal model, exhibiting superior performance in terms of precision, recall, and F1 score. The model achieved a high degree of accuracy in identifying fraudulent activities while minimizing false positives and false negatives. Comparative analysis highlighted the strengths and limitations of different machine learning algorithms in fraud detection tasks.

Logistic Regression



```
Evaluation Metrics for SMOTE:
              precision    recall  f1-score   support

           0       1.00      0.98      0.99    552439
           1       0.03      0.20      0.05      1643

    accuracy                           0.98    554082
   macro avg       0.51      0.59      0.52    554082
weighted avg       0.99      0.98      0.99    554082

ROC AUC Score: 0.6547842903483931
F1 Score: 0.04674505305420132
Recall Score: 0.19841752891052952

Evaluation Metrics for Undersampling:
              precision    recall  f1-score   support

           0       1.00      0.98      0.99    552439
           1       0.03      0.19      0.05      1643

    accuracy                           0.98    554082
   macro avg       0.51      0.59      0.52    554082
weighted avg       0.99      0.98      0.99    554082

ROC AUC Score: 0.6553419270388332
F1 Score: 0.0467234792829856
Recall Score: 0.1935483870967742
```

Fig. 5a



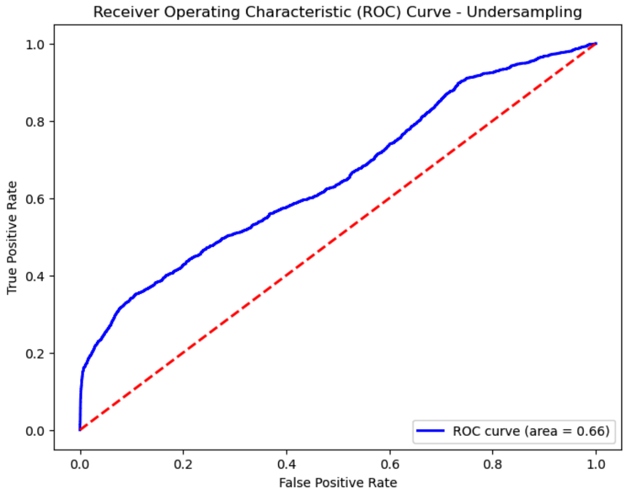


Fig. 5b

Fig. 5c

- **SMOTE Mode (Fig. 5a):** Demonstrated a low precision of 3% with a recall of 20%.
- **Undersampling Model (Fig. 5a, 5c):** Showed slightly better precision and recall rates compared to SMOTE but still struggled to effectively identify fraudulent transactions.
- **Performance Metrics:** Both logistic regression models using SMOTE and undersampling techniques faced challenges in accurately detecting fraudulent transactions. Both models exhibited comparable accuracy, F1 score, and macro-average metrics, highlighting the complexity of addressing class imbalance in fraud detection. Need for more advanced techniques in fraud detection.

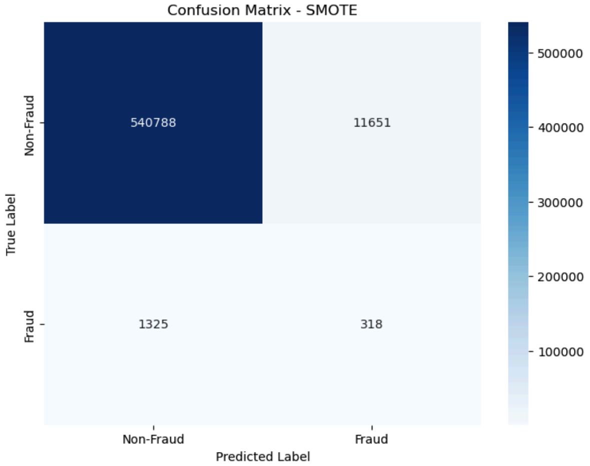


Fig.5d

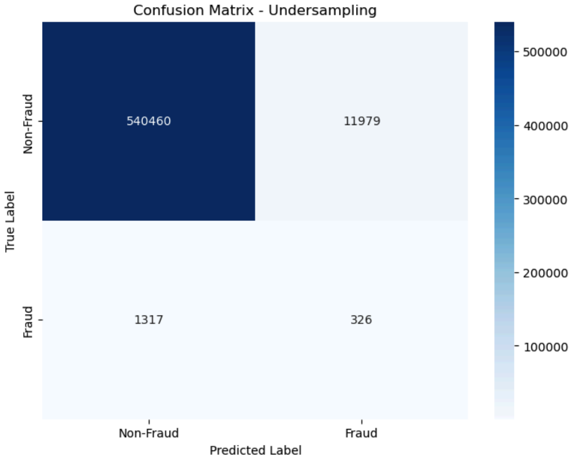
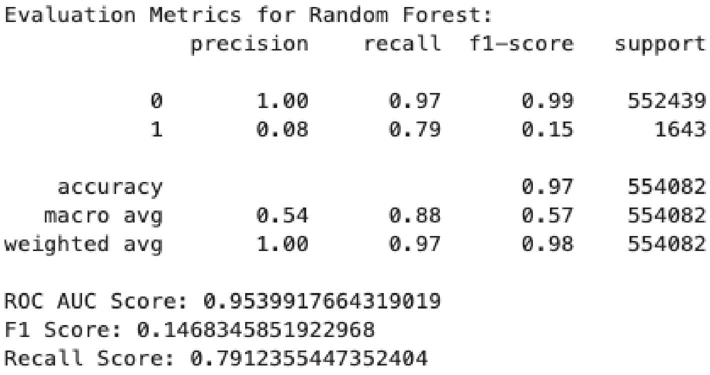


Fig. 5e

- **Confusion Matrix Smote: (Fig. 5d)**
- True Negatives (TN): 540,788 - Correctly identified non-fraudulent transactions.
- False Positives (FP): 11,651 - Incorrectly classified non-fraudulent transactions as fraudulent.
- True Positives (TP): 1,325 - Successfully detected fraudulent transactions.
- False Negatives (FN): 318 - Failed to identify fraudulent transactions.
- **Undersampling Model Confusion Matrix: (Fig. 5e)**
- True Negatives (TN): 540,460
- False Positives (FP): 11,979
- True Positives (TP): 1,317
- False Negatives (FN): 326

Random Forest:



```
Evaluation Metrics for Random Forest:
              precision    recall  f1-score   support

           0       1.00      0.97      0.99    552439
           1       0.08      0.79      0.15      1643

    accuracy                           0.97    554082
   macro avg       0.54      0.88      0.57    554082
weighted avg       1.00      0.97      0.98    554082

ROC AUC Score: 0.9539917664319019
F1 Score: 0.1468345851922968
Recall Score: 0.7912355447352404
```

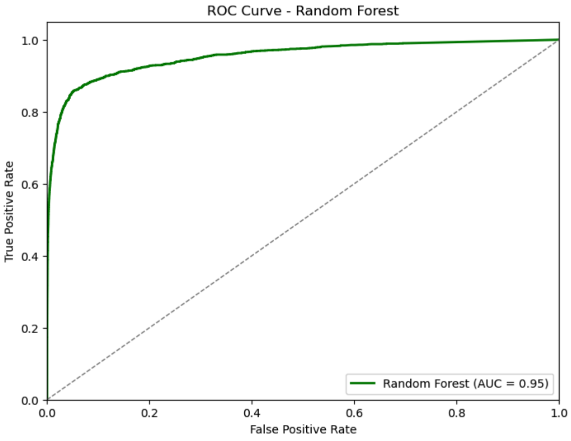
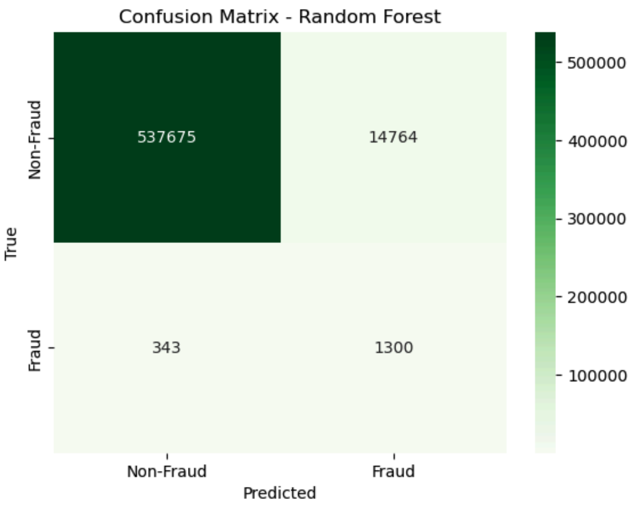


Fig. 6a

Fig. 6b



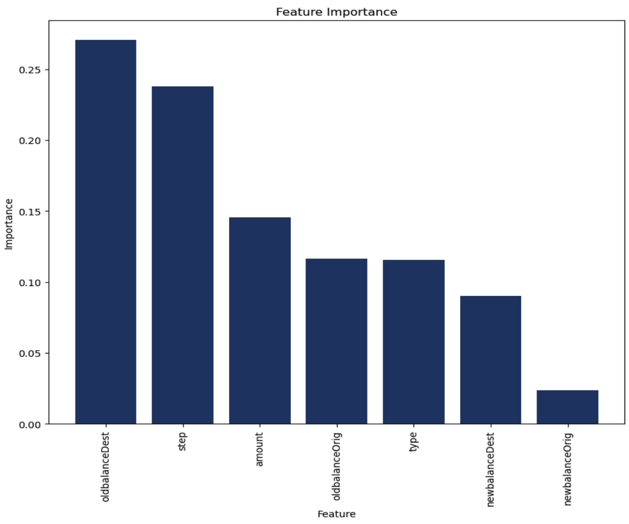
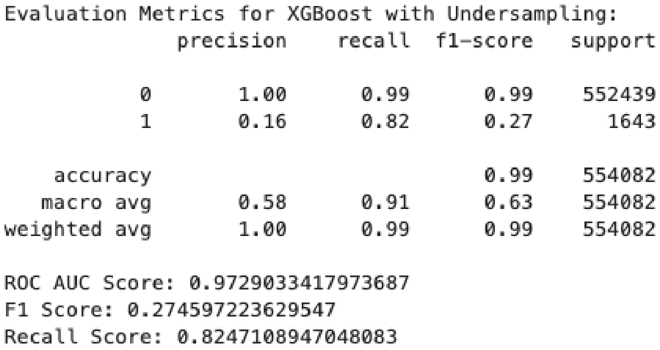


Fig. 6c

Fig. 6d

- **Recall and Precision (Fig. 6a):** The Random Forest model achieved a 79% recall rate and an F1 score of 0.15, outperforming logistic regression in fraud detection.
- **ROC AUC Score(Fig. 6b)**: With a 97% accuracy and a ROC AUC score of 0.95, the model effectively distinguishes between fraudulent and non-fraudulent transactions.
- **Confusion matrix (Fig.6c):** Shows that while the model successfully identifies actual fraud cases, it also has a relatively higher rate of false positives, emphasizing the trade-off between precision and recall in fraud detection.
- **Feature Importance (Fig. 6d):** OldbalanceDest, step, amount, and transaction type are key features driving fraud detection, highlighting the importance of account balances, transaction timing, amount, and type.

XGBoost



```
Evaluation Metrics for XGBoost with Undersampling:
              precision    recall  f1-score   support

           0       1.00      0.99      0.99    552439
           1       0.16      0.82      0.27      1643

    accuracy                           0.99    554082
   macro avg       0.58      0.91      0.63    554082
weighted avg       1.00      0.99      0.99    554082

ROC AUC Score: 0.9729033417973687
F1 Score: 0.274597223629547
Recall Score: 0.8247108947048083
```

Fig. 7a

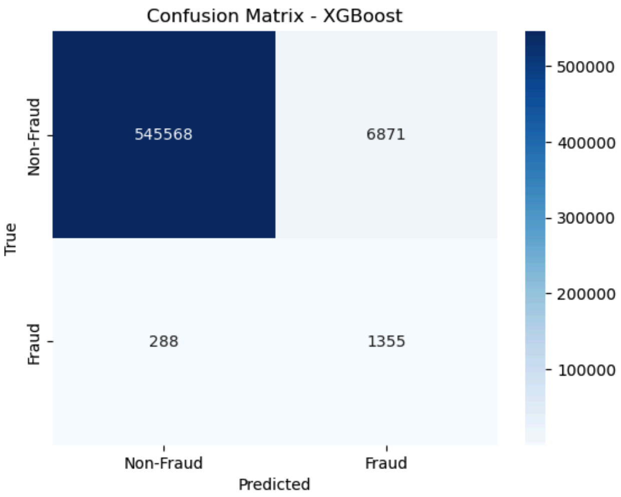


Fig. 7b

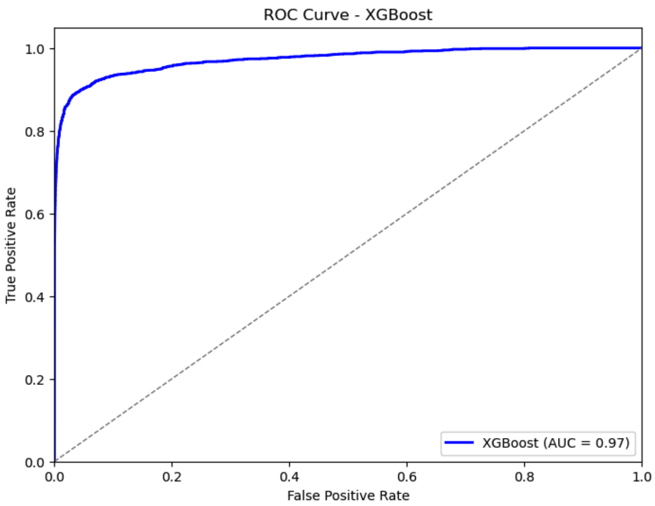
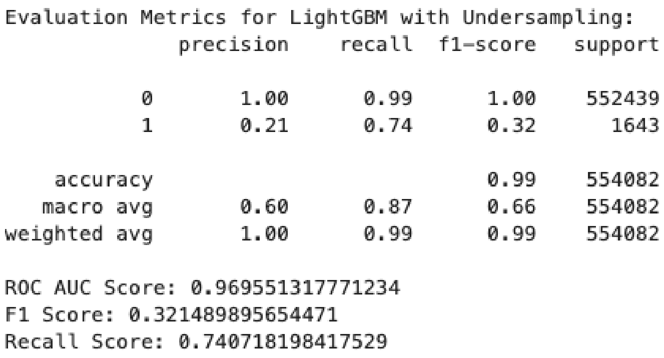


Fig. 7c

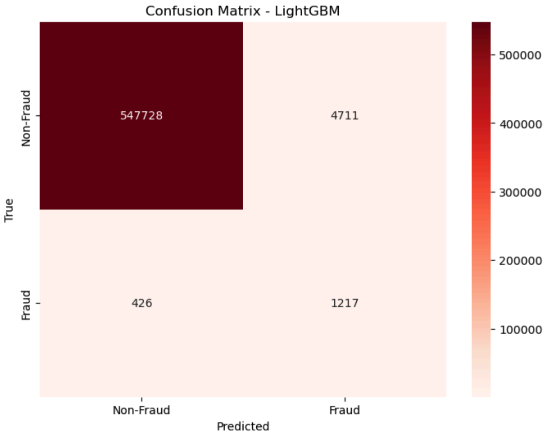
- **XGBoost Performance (Fig. 7a):** XGBoost achieved a high recall of 82% and an F1 score of 0.27, outperforming Logistic Regression and Random Forest.
- **Confusion Matrix(Fig 7.b):** The confusion matrix for XGBoost revealed a relatively low number of false negatives (288), highlighting its effectiveness in identifying fraudulent transactions.
- **ROC AUC Score (Fig. 7c):** XGBoost demonstrated impressive discrimination ability with a ROC AUC of 0.97, indicating great performance in distinguishing between fraudulent and non-fraudulent transactions.
- **Model Comparison:** XGBoost's accuracy and ability to handle complex data structures position it as the preferred model for fraud detection tasks, surpassing both Logistic Regression and Random Forest.

LightGBM



```
Evaluation Metrics for LightGBM with Undersampling:
              precision    recall  f1-score   support

           0       1.00      0.99      1.00    552439
           1       0.21      0.74      0.32      1643

    accuracy                           0.99    554082
   macro avg       0.60      0.87      0.66    554082
weighted avg       1.00      0.99      0.99    554082

ROC AUC Score: 0.969551317771234
F1 Score: 0.321489895654471
Recall Score: 0.740718198417529
```

Fig. 8a



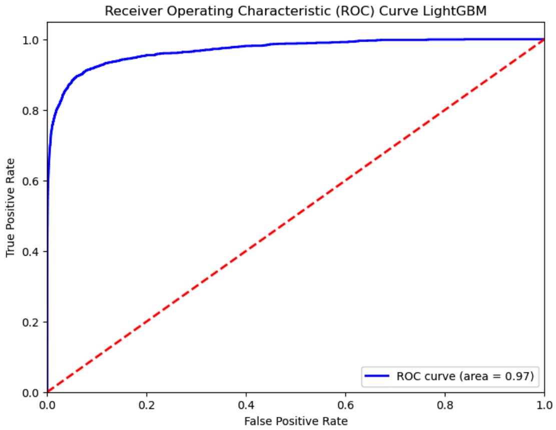


Fig. 8b

Fig. 8c

- **LightGBM Metrics (Fig 8a):** LightGBM achieved a higher F1 score of 0.32 and a recall score of 74%, indicating its effectiveness in accurately identifying fraudulent transactions.
- **Confusion Matrix (Fig 8b)**: LightGBM exhibited fewer false positives compared to XGBoost, indicating its better performance in avoiding misclassification of non-fraudulent transactions.
- **ROC AUC Score(Fig. 8c):** LightGBM demonstrated a commendable ROC AUC score of 0.97, highlighting its ability to discriminate between fraudulent and non-fraudulent transactions effectively.
- **Model Comparison:** While both LightGBM and XGBoost show strengths and weaknesses in different aspects of fraud detection, LightGBM's lower false positive rate makes it a favorable choice.

Conclusion & Future Direction:

In conclusion, this project demonstrates the effectiveness of machine learning techniques in detecting fraudulent transactions within financial datasets. By leveraging supervised learning algorithms and exploratory data analysis, valuable insights were gained into the characteristics and distributions of transaction data, facilitating the identification of fraudulent patterns and behaviors. The project aimed to address fraud detection in financial transactions using machine learning models, including Logistic Regression, Random Forest, XGBoost, and LightGBM. Each model underwent evaluation based on metrics like precision, recall, F1-score, and ROC AUC score.

Logistic Regression provided a baseline, but it struggled with intricate feature relationships and imbalanced datasets.Random Forest, trained on undersampled data, showed a high recall score but also had increased false positives. XGBoost exhibited efficiency in identifying fraudulent transactions, with a low false negative rate, signifying its effectiveness in capturing actual fraud cases.LightGBM surpassed XGBoost with higher F1 and recall scores, indicating its potential for accurate fraud detection.

Future directions involve exploring ensemble methods, advanced feature engineering, hyperparameter tuning, anomaly detection techniques, real-time monitoring, and improving model interpretability. These strategies can enhance fraud detection systems, adapting them to dynamic environments and ensuring continuous accuracy and efficiency in financial transaction security.