Fraud Detection in Credit Card Transactions: Final Project Report

1. Introduction

Fraudulent credit card transactions pose a significant challenge for financial institutions, resulting in financial losses and security risks. This project aims to develop and implement a machine learning model to accurately classify transactions as either fraudulent or non-fraudulent, using a labeled credit card transaction dataset.

2. Problem Statement

The primary objective of this project is to detect fraudulent transactions with high accuracy while balancing precision and recall. Given the highly imbalanced nature of fraud detection datasets, the project explored various machine learning techniques, including logistic regression, decision trees, and balanced random forests, with different data handling strategies to improve classification performance.

Success was measured using key performance metrics such as:

- **Precision** (minimizing false positives)
- **Recall** (ensuring fraud detection coverage)
- **F1-score** (balancing precision and recall)
- **ROC AUC** (overall model performance)

3. Data Cleaning and Exploratory Data Analysis (EDA)

3.1 Data Cleaning

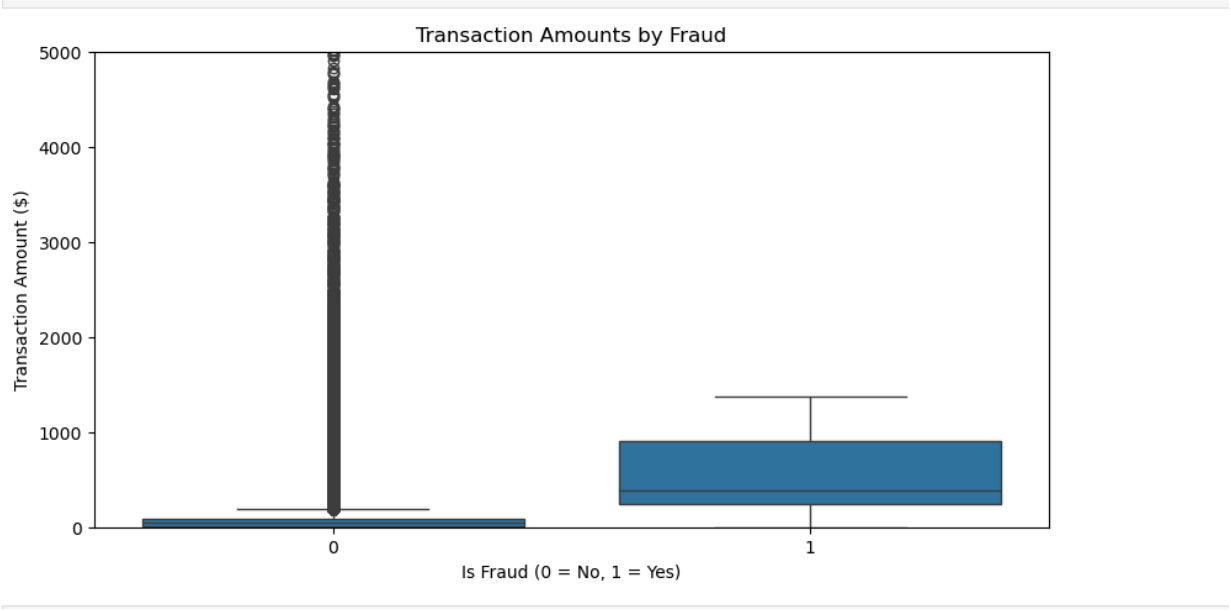
The dataset contained raw transaction records, including features such as transaction amount, time, merchant information, and customer demographics. The following data cleaning steps were performed:

- **Handling Missing Values**: Checked for and imputed missing values where necessary.
- **Removing Duplicates**: Identified and removed duplicate transactions.
- **Fixing Data Types**: Converted categorical features into numerical representations where needed.
- **Standardizing Data**: Normalized transaction amounts and transformed timestamps for better feature extraction.

3.2 Exploratory Data Analysis (EDA)

EDA was conducted to understand the data distribution and identify key trends:

- **Fraud Distribution**: The dataset was highly imbalanced, with fraudulent transactions accounting for less than 1% of the total data.
- **Transaction Patterns**: Analyzed transaction frequency by time of day, transaction amounts, and merchant categories to identify fraudulent patterns.
- **Feature Correlations**: Used correlation heatmaps to understand relationships between numerical variables.
- **Visualization of Fraudulent vs. Non-Fraudulent Transactions**: Plotted histograms, box plots, and scatter plots to highlight distinctions between fraud and legitimate transactions.



4. Feature Engineering and Preprocessing

- **Dropped High-Cardinality Features**: Features like street, city, zip, job, and transaction number were removed to reduce dimensionality and avoid data leakage.
- **Memory Optimization**: Converted large integer and float data types to smaller data types to optimize memory usage.
- **One-Hot Encoding**: Applied to categorical variables while using sparse representation to minimize memory overhead.
- **Feature Scaling**: Standardized numerical features such as transaction amount, city population, age, latitude, and longitude.

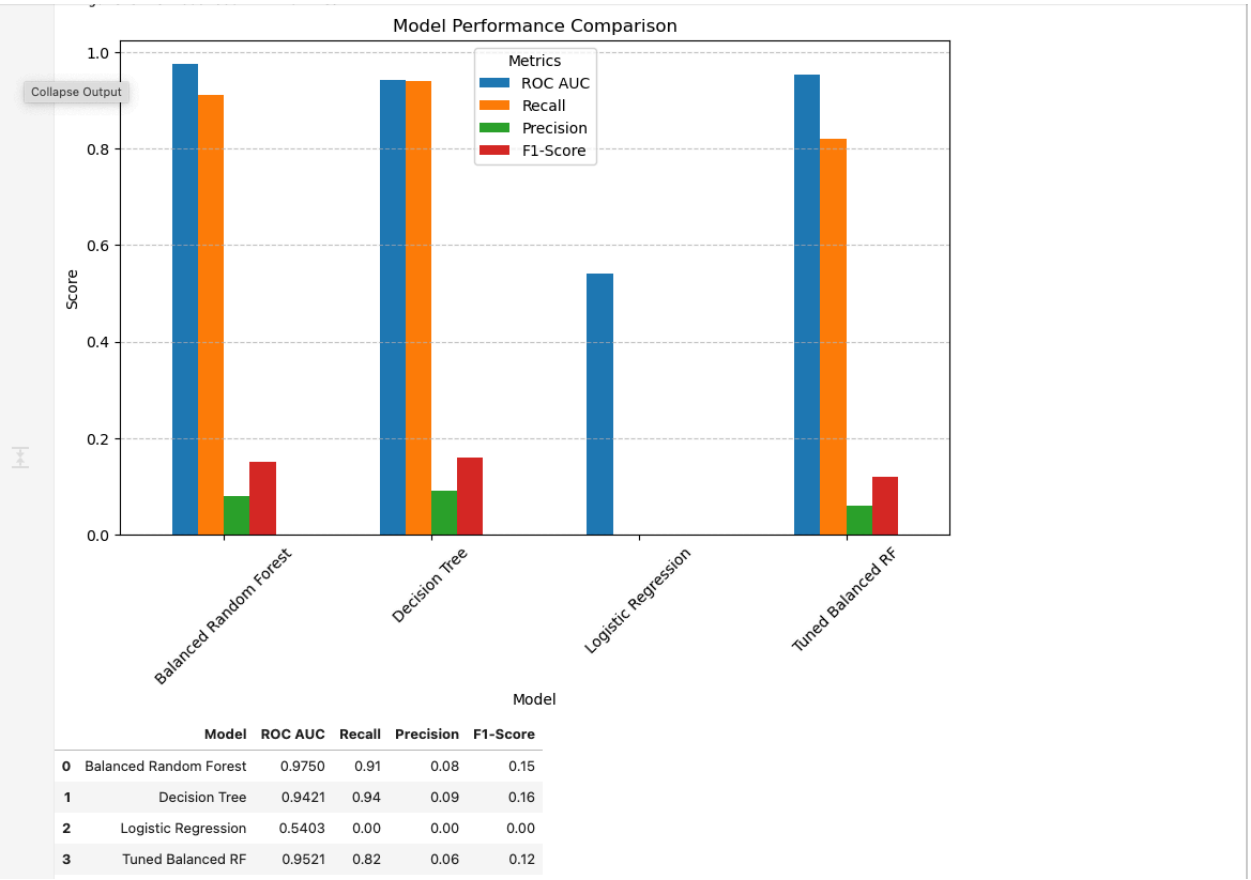
- **Train-Test Split**: Stratified splitting ensured that both training and test datasets maintained class distribution balance.
- **Final Features Used for Modeling**:
    - **Numerical Features**: amt, city_pop, age, trans_hour, lat, long, merch_lat, merch_long.
    - **Categorical Features**: Encoded merchant categories and transaction types.

5. Model Evaluation and Comparison

Model	ROC AUC	Recall	Precision	F1-Score
Balanced Random Forest (BRF)	0.9750	0.91	0.08	0.15
Decision Tree	0.9421	0.94	0.09	0.16
Logistic Regression	0.5403	0.00	0.00	0.00
Tuned Balanced Random Forest	0.9521	0.82	0.06	0.12

6. Final Model Selection

- The Balanced Random Forest (BRF) with default settings performed best, achieving the highest recall (0.91) and a strong ROC AUC score (0.9750).
- Hyperparameter tuning on BRF did not yield significant improvements; in fact, recall dropped from 0.91 to 0.82.
- Logistic Regression was ineffective and was discarded.
- The Decision Tree model was decent but not as strong as BRF.
- Given these results, the Balanced Random Forest (BRF) model was selected as the final model.

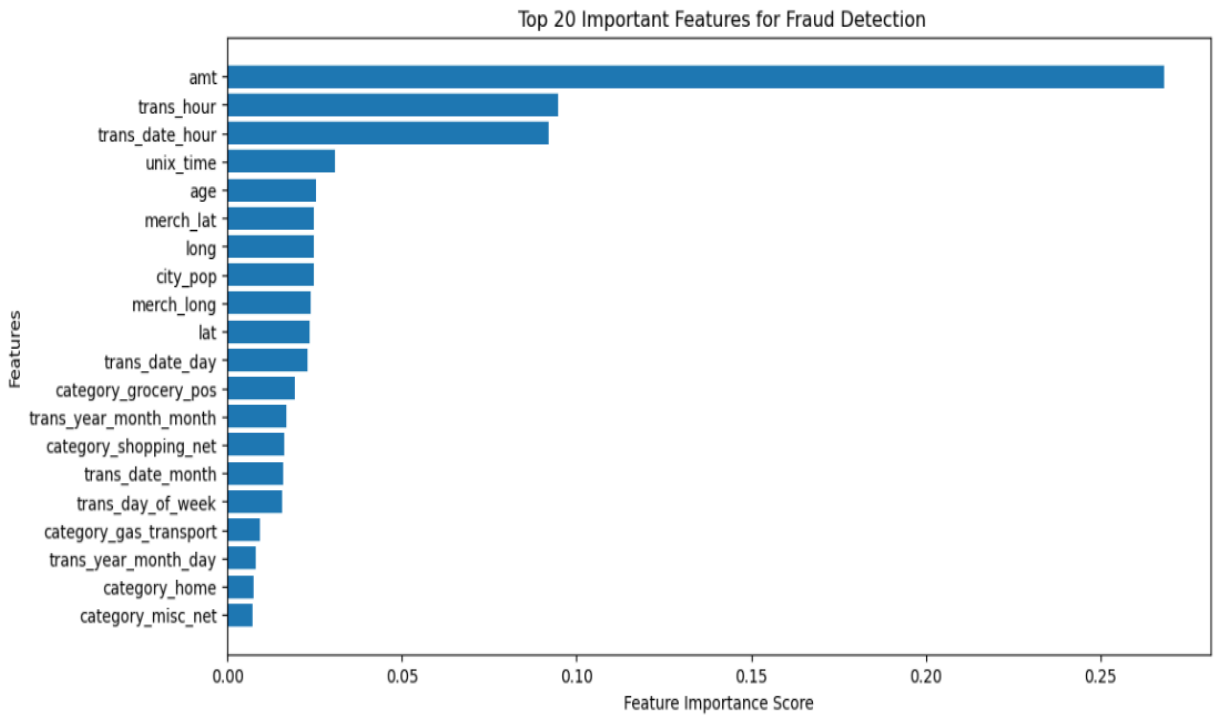


	Model	ROC AUC	Recall	Precision	F1-Score
0	Balanced Random Forest	0.9750	0.91	0.08	0.15
1	Decision Tree	0.9421	0.94	0.09	0.16
2	Logistic Regression	0.5403	0.00	0.00	0.00
3	Tuned Balanced RF	0.9521	0.82	0.06	0.12

7. Feature Importance Analysis

The **top 5 most important features** identified for fraud detection were:

1. **Transaction Amount (amt)** – Highest importance, indicating fraudulent transactions often involve extreme amounts.
2. **Transaction Hour (trans_hour)** – Certain time windows are more prone to fraud.
3. **Transaction Date Hour (trans_date_hour)** – Fraudulent activities peak at specific hours.
4. **Unix Timestamp (unix_time)** – Temporal patterns play a crucial role.
5. **Customer Age (age)** – Older customers tend to have different fraud risks compared to younger users.



8. Recommendations for Fraud Detection Implementation

1. Transaction Monitoring System: The model should be integrated into real-time transaction processing to flag suspicious activities based on top contributing features.
2. Threshold Optimization: To reduce false positives, the classification threshold can be adjusted based on business risk appetite.
3. Periodic Model Recalibration: As fraudsters evolve their tactics, the model should be retrained periodically using fresh data to maintain accuracy.

9. Conclusion

This project successfully developed a fraud detection model using machine learning techniques. The **Balanced Random Forest** model was selected as the final model due to its high recall and strong overall performance. The analysis of feature importance provides actionable insights for financial institutions to refine fraud detection strategies. Future work could explore **deep learning approaches** and **real-time detection systems** to further improve fraud prevention.