Fraud Detection Report

ARYAN KHAPARDE
IIT BOMBAY
Email: 21B090010@iitb.ac.in

***Abstract*—This report presents a comprehensive analysis of building a fraud detection system using machine learning techniques to identify fraudulent financial transactions. Various preprocessing steps, feature engineering, and machine learning models are employed to detect fraudulent behavior effectively. The project demonstrates the performance of several models, with XGBoost emerging as the top performer for fraud detection.**

I. Introduction

Fraud detection in financial transactions is a critical application of machine learning techniques. This project focuses on developing a fraud detection system using a dataset containing various transaction details. The primary objective is to identify patterns and features that distinguish fraudulent transactions from legitimate ones.

II. Data Preparation

A. Data Loading and Cleaning

The dataset `fraud_0.1origbase.csv` contains 636,262 rows and 11 columns representing financial transaction details. These columns include `step`, `type`, `amount`, `name_orig`, `oldbalance_org`, `newbalance_orig`, `name_dest`, `oldbalance_dest`, `newbalance_dest`, `is_fraud`, and `is_flagged_fraud`. For consistency, column names are converted into snake_case, and categorical columns `is_fraud` and `is_flagged_fraud` are mapped to 'yes' and 'no' labels.

B. Data Splitting and Transformation

The data is divided into training, validation, and test sets using an 80-20 split. Stratified sampling ensures that the class distribution of the target variable (`is_fraud`) is preserved. One-hot encoding is applied to the `type` column, and numerical features are scaled using Min-Max scaling.

C. Feature Selection

The final selected features include:

- `step`
- `oldbalance_org`
- `newbalance_org`
- `newbalance_dest`
- `diff_new_old_balance`
- `diff_new_old_destiny`
- `type_TRANSFER`

These features are used to build and evaluate machine learning models.

III. Exploratory Data Analysis (EDA)

A. Numerical Attributes

Descriptive statistics for numerical attributes are calculated, including mean, standard deviation, minimum, and maximum values. Additional statistical measures such as range, variation coefficient, skewness, and kurtosis are computed to gain insights into the data distribution.

B. Categorical Attributes

The distribution of the target variable `is_fraud` is visualized to examine class imbalance, as fraudulent transactions are far fewer compared to non-fraudulent ones.

IV. Machine Learning Modeling

A. Baseline Model

A **DummyClassifier** is used as the baseline model to predict the majority class (non-fraudulent transactions). Performance metrics such as precision, recall, and F1-score for the minority class (fraudulent transactions) are 0, indicating the challenges posed by imbalanced data.

B. Logistic Regression

The **Logistic Regression** model performs moderately with a balanced accuracy of 0.584 and an F1-score of 0.288. However, recall for the fraud class is low at 0.168, suggesting difficulty in detecting fraudulent transactions.

C. K-Nearest Neighbors (KNN)

The **KNN** model has similar performance to Logistic Regression with a balanced accuracy of 0.565 and an F1-score of 0.23. The recall score remains low, indicating challenges in fraud detection.

D. Support Vector Machine (SVM)

The **SVM** model performs poorly with a balanced accuracy of 0.5 and low precision and recall for fraudulent transactions.

E. Random Forest

The **Random Forest** model performs significantly better with a balanced accuracy of 0.844 and an F1-score of 0.807. It achieves a good balance between precision and recall in detecting fraudulent transactions.

F. XGBoost

The **XGBoost** model delivers excellent performance with a balanced accuracy of 0.863 and an F1-score of 0.83, demonstrating high precision and recall for detecting fraudulent transactions.

G. LightGBM

The **LightGBM** model performs relatively poorly, achieving a balanced accuracy of 0.69 and recall of 0.382. This indicates that it struggles to detect fraud, despite performing reasonably on the majority class. The model's inability to detect fraud effectively highlights its limitations in dealing with imbalanced data.

H. Model Performance Comparison

XGBoost and **Random Forest** are identified as the top models, with **XGBoost** slightly outperforming **Random Forest** in both recall and precision for fraudulent transactions.

V. Hyperparameter Tuning

A. Fine-Tuning XGBoost

Grid search is used to fine-tune the hyperparameters of the **XGBoost** model. The best parameters identified are:

- Booster: 'gbtree'
- Eta: 0.3
- Scale_pos_weight: 1

Despite fine-tuning, the performance of the tuned **XGBoost** model remains strong, maintaining an excellent F1-score and balanced accuracy.

VI. Conclusion

Based on the evaluation of multiple machine learning models, **XGBoost** and **Random Forest** emerge as the top-performing models for fraud detection. **XGBoost** outperforms other models, achieving the highest balanced accuracy and F1-score. **Random Forest** also performs well, making it a viable alternative. The fine-tuning of **XGBoost** confirms its effectiveness, and further optimizations can improve performance. This work lays the foundation for building an effective fraud detection system using machine learning.