

Fraud Detection Model - Documentation

Introduction

This report documents the design, methodology, and findings of the fraud detection system implemented in the provided Python file. The system is designed to identify fraudulent credit card transactions using machine learning techniques. It provides flexibility in handling datasets, performing exploratory analysis, training models, and evaluating their effectiveness.

Approach

The fraud detection system is encapsulated within a class named **FraudDetectionModel**. The approach involves:

- Loading datasets from CSV/Excel or generating synthetic data if no dataset is available.
- Validating datasets to ensure the presence of the 'Class' column with binary labels.
- Performing exploratory data analysis (EDA) including summary statistics and visualizations.
- Preprocessing with scaling and balancing techniques (SMOTE or undersampling).
- Training multiple models including Logistic Regression, Random Forest, SVM, and Isolation Forest.
- Evaluating models using accuracy, F1-score, ROC-AUC, and confusion matrices.
- Performing hyperparameter tuning with GridSearchCV.
- Conducting feature importance analysis.
- Detecting fraud in new transactions.

Methodology

The methodology follows the machine learning project pipeline:

1. **Data Handling:** Load or generate synthetic datasets.
2. **Validation:** Ensure appropriate structure and binary target labels.
3. **Exploration:** Summarize data, visualize distributions, and assess correlations.
4. **Preprocessing:** Scale features and address class imbalance.
5. **Model Training:** Train multiple supervised and unsupervised algorithms.
6. **Evaluation:** Compare model performance with confusion matrices, reports, and ROC curves.
7. **Hyperparameter Tuning:** Optimize Random Forest and Logistic Regression parameters.
8. **Feature Importance:** Identify influential predictors.
9. **Fraud Detection:** Deploy models to assess new transactions.

Findings

The framework produced the following findings:

- Synthetic dataset creation successfully simulates fraud imbalance (~0.2%).
- Exploratory analysis reveals class imbalance and correlations.

- Random Forest and SVM achieved the strongest F1-scores among supervised methods.
- Isolation Forest provides unsupervised anomaly detection without labels.
- Hyperparameter tuning improved Random Forest performance significantly.
- Feature importance analysis identified key variables (e.g., transaction amount) driving fraud detection.

Conclusion

The **FraudDetectionModel** provides a robust fraud detection pipeline that is modular, extendable, and applicable across financial datasets. It combines supervised and unsupervised learning techniques, supports resampling methods, and integrates hyperparameter tuning. Random Forest emerged as the most effective model, balancing precision and recall, making the solution practical for real-world fraud detection tasks.