**Documentation for Fraudulent Transaction Detection Project**

**Introduction**

This project aims to detect fraudulent transactions using both supervised and unsupervised machine learning techniques. The dataset used originates from Kaggle and includes various features such as transaction distance, retailer information, and chip usage. The pipeline involves data preprocessing, model training, evaluation, and explainability.

**About the Dataset**

* **Source:** Kaggle.
* **Features:**
  + distance\_from\_home: Numeric feature indicating the distance of the transaction from the user’s home.
  + distance\_from\_last\_transaction: Distance from the previous transaction.
  + ratio\_to\_median\_purchase\_price: Purchase price normalized by the median purchase price.
  + repeat\_retailer: Indicates if the transaction occurred at a repeat retailer.
  + used\_chip: Indicates whether a chip was used in the transaction.
  + used\_pin\_number: Indicates whether a PIN number was used.
  + online\_order: Indicates if the transaction was an online order.
  + fraud: Target variable indicating fraudulent (1) or legitimate (0) transactions.
* **Class Imbalance:** Approximately 91.26% of the transactions are legitimate, while 8.74% are fraudulent, resulting in an imbalance ratio of 10.44.

**Steps in the Code**

**1. Data Exploration and Preprocessing**

* **Data Exploration:**
  + Checked for null values and duplicates.
  + Summary statistics and boxplots were created for numeric features to detect outliers.
  + Pie charts and histograms analyzed categorical features and their distribution.
  + Correlation matrices identified feature relationships and relevance to the target.
* **Outlier Removal:**
  + Top 5 extreme values for each numeric feature were removed to reduce noise.
* **Data Splitting:**
  + Data split into training (70%), validation (15%), and test (15%) sets, stratified by the target.
* **Feature Scaling:**
  + Applied StandardScaler to normalize numeric features.

**2. Model Development (Supervised)**

* **Baseline Logistic Regression:**
  + Applied class weights to address imbalance.
  + Achieved an accuracy of 93%, recall of 95%, and F1-score of 72%.
* **XGBoost Model:**
  + Used scale\_pos\_weight to handle class imbalance.
  + Achieved near-perfect performance with an accuracy of 100%, recall of 100%, and F1-score of 99%.
  + Included regularization techniques to mitigate overfitting.

**3. Unsupervised Learning**

* **Isolation Forest:**
  + Detected anomalies without using labeled data.
  + Anomaly scores were computed, and transactions were classified as normal (0) or anomalous (1).
  + Results compared with ground truth for evaluation.

**4. Model Evaluation**

* **Confusion Matrices:**
  + Evaluated performance for Logistic Regression and XGBoost.
  + Highlighted true positives, false positives, and overall prediction balance.
* **Metrics:**
  + Accuracy, precision, recall, and F1-score were computed.
  + ROC and Precision-Recall (PR) curves were plotted to visualize model performance.

**5. Explainability**

* **Feature Importance:**
  + XGBoost’s feature importance ranked significant contributors to fraud detection.
* **SHAP Analysis:**
  + Summary plots provided global interpretability of feature impact.
  + Force plots explained individual predictions, highlighting key influential features.

**Model Performance Summary**

* **Logistic Regression:**
  + **Accuracy:** 93%
  + **Precision:** 57%
  + **Recall:** 95%
  + **F1-Score:** 72%
* **XGBoost:**
  + **Accuracy:** 100%
  + **Precision:** 98%
  + **Recall:** 100%
  + **F1-Score:** 99%

**Visualizations and Plots**

1. **Confusion Matrices:** Showed classification results for legitimate and fraudulent transactions.
2. **ROC and PR Curves:** Visualized trade-offs between precision, recall, and false positive rate.
3. **SHAP Summary Plot:** Highlighted the most influential features across the dataset.
4. **SHAP Force Plot:** Illustrated individual predictions.

**Unsupervised Approach Overview**

* **Isolation Forest Results:**
  + Identified anomalies effectively, providing insights into potential fraudulent behaviors.
  + Examples of anomalies include unusually high distances or non-standard transaction patterns.

**Conclusion**

The project successfully implemented machine learning to detect fraud, leveraging both supervised and unsupervised techniques. XGBoost emerged as the best-performing model, demonstrating its robustness in handling class imbalance and complex relationships. The use of SHAP provided explainability, ensuring the model's decisions align with real-world expectations.