**Fraud Detection in Banking Report**

**-S B GNANASHREE JAIN**

**1. Introduction**

Fraud detection in banking transactions is crucial for ensuring financial security. This project utilizes a **Random Forest Classifier** to classify transactions as **fraudulent (Class 1) or non-fraudulent (Class 0)** based on a Kaggle dataset.

**Fraud Detection**

* **Impact**: ML models analyze transaction patterns to detect fraudulent activities, reducing financial losses.
* **Example**: Banks use ML to detect unusual transaction behaviors and flag potential fraud.
* **Case Study**: PayPal employs ML to differentiate legitimate transactions from fraudulent ones, significantly reducing fraud cases.

**2. Dataset Overview**

The dataset used is the **Credit Card Fraud Detection Dataset** from Kaggle ([Link](https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud)). It consists of **284,807 transactions**, including 492 fraud cases (**0.172% of the data**), making it highly imbalanced.

**Dataset Columns Explanation**

* **Time**: Seconds elapsed between the first transaction and the current transaction.
* **V1-V28**: Features obtained via **PCA transformation** (principal component analysis) to anonymize sensitive data.
* **Amount**: Transaction amount.
* **Class**: Target variable (0 = Non-Fraud, 1 = Fraud).

**Key Insights from Exploratory Data Analysis (EDA)**

✅ Highly imbalanced dataset (Fraud cases ~0.17%). ✅ No missing values. ✅ Fraudulent transactions tend to have lower amounts. ✅ PCA features (V1-V28) have different distributions for fraud vs. non-fraud cases.

**3. Data Preprocessing**

**Handling Class Imbalance**

The dataset is **heavily imbalanced**, so to balance the data:

* **Undersampling**: Equal samples of fraud and non-fraud transactions were selected.
* **Feature Scaling**: StandardScaler was applied to normalize the **Amount** column.
* **Feature Selection**: Removed the "Time" column as it was not useful.

**4. Model Training**

**Random Forest Classifier**

* **Algorithm Used**: RandomForestClassifier(n\_estimators=100, random\_state=42)
* **Train-Test Split**: 80% training, 20% testing (stratified sampling)
* **Hyperparameters**:
  + n\_estimators=100
  + random\_state=42
  + Default parameters for max depth, min samples split, etc.

**5. Model Evaluation**

**Classification Report**

| **Metric** | **Class 0 (Non-Fraud)** | **Class 1 (Fraud)** |
| --- | --- | --- |
| **Precision** | 94% | 96% |
| **Recall** | 96% | 94% |
| **F1-score** | 95% | 95% |
| **Accuracy** | **95%** Overall |  |

**Insights:** ✅ **High precision (96%) for fraud detection** → Fewer false fraud alerts. ✅ **High recall (94%) for fraud** → Model correctly identifies most frauds. ✅ **Balanced F1-score (95%)** → Effective fraud detection with minimal false alarms.

**Confusion Matrix (Heatmap)**

A visualization of correct and incorrect predictions:

| **Predicted \ Actual** | **Non-Fraud (0)** | **Fraud (1)** |
| --- | --- | --- |
| **Non-Fraud (0)** | True Negatives ✅ | False Negatives ❌ |
| **Fraud (1)** | False Positives ❌ | True Positives ✅ |

✅ Low **False Positives (FP)** means fewer legitimate transactions flagged as fraud. ✅ Low **False Negatives (FN)** ensures frauds are not missed.

**ROC-AUC Score**

* **0.9881** → Indicates a high ability to distinguish between fraud and non-fraud transactions.
* **Closer to 1 = Better Performance.**

**Feature Importance Plot**

* The **most important features** contributing to fraud detection were extracted and visualized.
* PCA-transformed features like **V12, V14, V10** had high importance.
* **Transaction Amount** also played a key role.

**6. Conclusion & Future Work**

✅ **Model performs well with 95% accuracy.** ✅ **Balanced F1-score ensures reliability.** ✅ **ROC-AUC of 0.9881** shows high discrimination power. ✅ **Feature importance analysis helps understand key fraud indicators.**

**Improvements & Future Work**

🔹 **Enhance Recall:** Use SMOTE (Synthetic Minority Oversampling) to further balance data. 🔹 **Hyperparameter Tuning:** Optimize tree depth, feature selection. 🔹 **Real-time Fraud Detection:** Deploy model with online learning.

🚀 **This model is effective for fraud detection and can be further optimized for production use!**