

# Fraud Detection in Bank Account Transactions

## Objective:

Develop a predictive model to identify fraudulent transactions in bank accounts using the provided dataset. Your model should accurately classify transactions as either legitimate or fraudulent.

## Dataset Overview:

You are provided with a comprehensive database that includes the following information:

- **Customer Data:** Information of customers.
- **Account Activities:** Records of transactions and account operations.
- **Fraud Indicators:** Flags and features that may indicate fraudulent behaviour.
- **Merchant Information:** Information about merchants involved in the transactions.

## Task Breakdown:

Your project will consist of several key steps to achieve effective fraud detection. Follow the outlined steps below:

### 1. Data Pre-processing

Prepare the dataset for analysis by performing the following tasks:

- **Handling Missing Values:** Identify and address any missing or incomplete data entries.
- **Statistical Aggregations:** Compute summary statistics such as mean, standard deviation, and other relevant metrics to understand data distributions.
- **Multicollinearity Evaluation:** Assess the relationships between variables to identify and address multicollinearity issues that may affect model performance.
- **Normality Testing:** Conduct tests (e.g., Shapiro-Wilk) to determine if the data follows a normal distribution.
- **Additional Pre-processing:** Apply any other necessary data cleaning or transformation techniques to enhance data quality.

### 2. Supervised Classification Methods

Apply various classification algorithms to the pre-processed data to build predictive models:

- **Model Selection:** Choose several supervised classification methods (e.g., Logistic Regression, Decision Trees, Random Forest, Support Vector Machines).
- **Training:** Train each selected model using the training subset of your data.
- **Evaluation Strategy:** Implement an evaluation approach (e.g., cross-validation) to assess the performance of each model systematically.

### 3. Model Evaluation

Assess the performance of each classification model using appropriate metrics:

- **Confusion Matrix:** For each model, create a confusion matrix to visualize true positives, true negatives, false positives, and false negatives.
- **Performance Metrics:** Calculate metrics such as accuracy, precision, recall, F1-score, and ROC AUC to compare model effectiveness.

#### 4. Model Tuning

Optimize your models to improve their predictive performance:

- **Identify Issues:** Analyze potential problems such as overfitting, underfitting, or imbalance in the dataset.
- **Hyperparameter Tuning:** Adjust model parameters (e.g., number of trees in Random Forest, regularization strength in Logistic Regression) to enhance performance.
- **Techniques:** Utilize methods like Grid Search or Random Search for systematic hyperparameter optimization.

#### 5. Model Selection

Choose the best-performing model based on your evaluations:

- **Comparison:** Compare all tuned models using the evaluation metrics.
- **Justification:** Provide a rationale for selecting the final model, considering factors like accuracy, interpretability, and computational efficiency.

#### Deliverables:

Your submission should include the following components packaged in a single ZIP file:

1. **Python Implementation:**
  - A well-documented Python script or Jupyter Notebook (`fraud_detection.py` or `fraud_detection.ipynb`) that includes all steps from data pre-processing to model selection.
  - Ensure your code is organized, with clear comments explaining each section and function.
2. **Project Report (PDF):**
  - A comprehensive PDF document detailing the steps you took to complete the project.
  - **Report Structure:**
    - **Introduction:** Brief overview of the project and its objectives.
    - **Data Pre-processing:** Describe the methods used to clean and prepare the data, including handling missing values and feature engineering.
    - **Modeling:** Explain the classification algorithms chosen and the rationale behind selecting them.
    - **Evaluation:** Present the performance metrics and confusion matrices for each model.
    - **Model Tuning:** Discuss the hyperparameter tuning process and how it improved model performance.

- **Model Selection:** Justify the final model choice based on your evaluations.
- **Conclusion:** Summarize your findings and suggest potential improvements or future work.

### Submission Instructions:

1. **Prepare Your Work:**
  - Ensure your Python script or Jupyter Notebook runs without errors and includes all necessary components.
  - Compile your project report into a PDF document, ensuring it is well-formatted and free of typos.
2. **Create a ZIP File:**
  - Include both your Python implementation and PDF report in a single ZIP archive.
  - Name the ZIP file following any specific guidelines provided (e.g., `YourName_FraudDetection_Project.zip`).
3. **Upload:**
  - Submit the ZIP file through the designated submission portal or as instructed by your course guidelines.

### Additional Guidelines:

- **Reproducibility:** Ensure that your code can be executed on a different machine without issues. Include any necessary instructions or requirements.
- **Documentation:** Provide clear comments in your code and ensure your report is thorough and well-organized.
- **Academic Integrity:** Do your own work and cite any external resources or references you use.