Project Title: Fraud Detection Model for Financial Transactions

Objective:

The purpose of this project is to develop a machine learning model capable of identifying fraudulent transactions from a dataset containing genuine and fraudulent financial transactions. The model should accurately predict whether a given transaction is fraudulent, helping to protect against financial losses and ensure transactional security.

Tasks:

1. Exploratory Data Analysis (EDA):

- **Objective:** Gain an understanding of the dataset, identify patterns, and detect any potential issues or anomalies.
- **Tasks:**
- Load and preview the data to understand its structure and key characteristics.
- Summarize the data, including mean, median, minimum, maximum, and missing values, if any.
- Perform univariate and multivariate analysis to understand distributions and relationships among variables.
- Visualize the distributions of transaction types, amounts, and other relevant features.
- Identify and handle any class imbalances, if applicable.

2. Data Preparation:

- **Objective:** Prepare the data for training, ensuring it is clean and structured.
- **Tasks:**
- Handle missing or erroneous data.
- Encode categorical variables appropriately.
- Normalize or standardize features as required.
- Split the data into training and testing sets to evaluate model performance.

3. Feature Selection & Engineering:

- **Objective:** Identify and create features that can enhance model performance.
- **Tasks:**
- Analyse feature importance, correlation, and redundancy to identify useful features.
- Engineer new features if required, such as aggregating transaction amounts or creating time-based features.

- Address any multicollinearity issues that might affect the model's predictive power.

4. Base Model:

- **Objective:** Establish a baseline model to compare subsequent models.
- **Tasks:**
- Train a simple model (e.g., Logistic Regression) to set a benchmark for model performance.
- Evaluate its performance on key metrics (e.g., accuracy, precision, recall, F1-score).

5. Model Selection:

- **Objective:** Select and test different algorithms to determine the best performing model.
- **Tasks:**
- Test multiple algorithms, such as Decision Trees, Random Forest, Gradient Boosting, or XGBoost.
- Use cross-validation to assess each model's robustness and avoid overfitting.
- Choose a model that provides the best balance of performance metrics.

6. Model Training & Evaluation:

- **Objective:** Train the chosen model and evaluate its performance on test data.
- **Tasks:**
- Train the selected model on the training set.
- Evaluate model performance using metrics suitable for imbalanced data (e.g., precision-recall curve, ROC-AUC score).
- Document the performance metrics, including confusion matrix, precision, recall, and F1-score.

7. Final Model and Metrics:

- **Objective:** Present a finalized model that can be deployed in a real-world setting.
- **Tasks:**
- Finalize the model and provide detailed explanations of the features, parameters, and decisions made.
- Generate and report final metrics on the test dataset, focusing on metrics that highlight the model's effectiveness in identifying fraud.

Expected Deliverables:

1. **Project Notebook**:

- A comprehensive Jupyter Notebook that includes:
- Data loading, EDA, data preparation, and feature engineering.
- Base model development, comparison of different models, and detailed evaluation.
- Clear documentation for each section explaining the choices and reasoning behind them.
- A final model and evaluation metrics.

2. **Report**:
   - A summary of the analysis, including:
   - Key findings from the EDA.
   - Decisions taken during data preparation and feature engineering.
   - Comparison of model performances and the final model's evaluation metrics.
   - An explanation of why the final model was chosen over others.
3. **Code Documentation**:
   - Well-commented code to ensure clarity and understanding of each step taken in the notebook.

Evaluation Metrics:

- Accuracy, Precision, Recall, F1-score, and ROC-AUC should be provided.
- Emphasis should be placed on metrics suitable for imbalanced data (such as Precision, Recall, F1-score, and ROC-AUC).

Submission Guidelines:

Candidates should submit:

- The completed Jupyter Notebook file (.ipynb).
- Any associated files, such as CSVs or Python scripts.
- A summary report (in PDF or Word format).

Good luck! We look forward to seeing your approaches and solutions!