

PERFORMANCE TESTING :

1 Selecting the Features and Target

- **X (Features):** Transaction inputs like amount, type, balances, etc.
 - **y (Target):** Fraud label
 - 0 = Legitimate
 - 1 = Fraud
-

2 Splitting the Dataset

- Divide data into:
 - **Training set** (used to train the model)
 - **Testing set** (used to evaluate performance)
 - Common split: **80% train, 20% test**
-

3 Choosing the ML Algorithm

Select a suitable model for fraud detection, such as:

- Decision Tree
 - Random Forest
 - Logistic Regression
-

4 Creating the Model

- Import the algorithm from sklearn
 - Define the model object with parameters (example: max depth, random state)
-

5 Training the Model

- Train the model using the training dataset:
 - `model.fit(X_train, y_train)`
 - During training, the model learns fraud patterns from transaction data.
-

6 Making Predictions

- Use the trained model to predict fraud on test data:
 - `y_pred = model.predict(X_test)`
-

7 Model Evaluation

Check model performance using:

- Accuracy score
- Confusion matrix
- Precision, Recall, F1-score

(Important in fraud detection because fraud data is imbalanced.)

8 Saving the Trained Model (Optional)

- Save the trained model using pickle for future use in Flask/web app:
 - `model.pkl`