# Model Optimization and Tuning Phase Report

| | |
|---|---|
| Date | June 2025 |
| Team ID | SWTID1749841176 |
| Project Title | Online Payment Fraud Detection using Machine Learning |
| Maximum Marks | 6 Marks |

**Model Optimization and Tuning Phase:**

The Model Optimization and Tuning Phase involves refining machine learning models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

**Hyperparameter Tuning Documentation (6 Marks):**

| Model | Tuned Hyperparameters |
|---|---|
| Random Forest Classifier | `rfc = RandomForestClassifier(`<br>`  n_estimators=100,`<br>`  max_depth=15,`<br>`  min_samples_split=10,`<br>`  n_jobs=-1,`<br>`  random_state=42,`<br>`  verbose=1`<br>`)`<br>`rfc.fit(X_train, y_train)`<br>`y_pred = rfc.predict(X_test)`<br>`accuracy = accuracy_score(y_test, y_pred)`<br>`print("Accuracy:", accuracy)` |

| | |
|---|---|
| Decision Tree Classifier | `from sklearn.tree import DecisionTreeClassifier`<br>`from sklearn.metrics import accuracy_score`<br>`import numpy as np`<br><br>`dtc = DecisionTreeClassifier(random_state=42)`<br>`dtc.fit(X_train, y_train)`<br>`y_test_pred2 = dtc.predict(X_test)`<br>`accuracy = accuracy_score(y_test, y_test_pred2)`<br>`print("Accuracy:", accuracy)` |
| Extra Trees Classifier | `etc = ExtraTreesClassifier(`<br>`n_estimators=50, # 50 trees (you can push this up to 100 if 3 mins is okay)`<br>`max_depth=None, # Full depth trees (can limit to speed up slightly)`<br>`min_samples_split=5, # Reduces overfitting; more general`<br>`n_jobs=-1, # Use all CPU cores`<br>`random_state=42, # For reproducibility`<br>`verbose=1 # To track progress`<br>`)`<br><br>`etc.fit(X_train, y_train)`<br><br>`y_test_pred3 = etc.predict(X_test)`<br>`accuracy = accuracy_score(y_test, y_test_pred3)`<br>`print("Accuracy:", accuracy)` |
| Support Vector Machine Classifier | `svc = LinearSVC(C=1.0, max_iter=10000, random_state=42, verbose=1)`<br>`svc.fit(X_train, y_train)`<br><br>`y_test_pred4 = svc.predict(X_test)`<br>`accuracy = accuracy_score(y_test, y_test_pred4)`<br>`print("Accuracy:", accuracy)` |
| XgBoost Classifier | `xgb1 = xgb.XGBClassifier()`<br>`xgb1.fit(X_train, y_train_encoded)`<br>`y_test_pred5 = xgb1.predict(X_test)`<br>`accuracy = accuracy_score(y_test_encoded, y_test_pred5)`<br>`print("Accuracy:", accuracy)` |

**Performance Metrics Comparison:**

| Model | Optimized Metric |
|---|---|

| | |
|---|---|
| Random Forest Classifier | |

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 1.00 | 1.00 | 1.00 | 4448056 |
| 1 | 1.00 | 0.76 | 0.87 | 5778 |
| accuracy | | | 1.00 | 4453834 |
| macro avg | 1.00 | 0.88 | 0.93 | 4453834 |
| weighted avg | 1.00 | 1.00 | 1.00 | 4453834 |

| | |
|---|---|
| Decision Tree Classifier | |

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 1.00 | 1.00 | 1.00 | 4448056 |
| 1 | 1.00 | 1.00 | 1.00 | 5778 |
| accuracy | | | 1.00 | 4453834 |
| macro avg | 1.00 | 1.00 | 1.00 | 4453834 |
| weighted avg | 1.00 | 1.00 | 1.00 | 4453834 |

| | |
|---|---|
| Extra Trees Classifier | |

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 1.00 | 1.00 | 1.00 | 4448056 |
| 1 | 1.00 | 0.93 | 0.97 | 5778 |
| accuracy | | | 1.00 | 4453834 |
| macro avg | 1.00 | 0.97 | 0.98 | 4453834 |
| weighted avg | 1.00 | 1.00 | 1.00 | 4453834 |

| | |
|---|---|
| Support Vector Machine Classifier | |

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.91 | 0.94 | 0.92 | 13123 |
| 1 | 0.87 | 0.81 | 0.84 | 6588 |
| accuracy | | | 0.90 | 19711 |
| macro avg | 0.89 | 0.88 | 0.88 | 19711 |
| weighted avg | 0.90 | 0.90 | 0.90 | 19711 |

| | | precision | recall | f1-score | support |
|---|---|---|---|---|---|
| XgBoost Classifier | 0 | 1.00 | 1.00 | 1.00 | 4448056 |
| | 1 | 0.99 | 0.91 | 0.94 | 5778 |
| | accuracy | | | 1.00 | 4453834 |
| | macro avg | 0.99 | 0.95 | 0.97 | 4453834 |
| | weighted avg | 1.00 | 1.00 | 1.00 | 4453834 |

**Final Model Selection:**

| Final Model | Reasoning |
|---|---|
| Decision Tree Classifier | Decision Tree Classifier, is the best choice because it delivers perfect classification performance while maintaining strong generalization. Decision trees are inherently interpretable, allowing you to understand and explain decisions through clear decision paths. They naturally handle both numerical and categorical features without the need for scaling or complex preprocessing. Additionally, they are capable of capturing complex, nonlinear relationships and feature interactions, making them highly effective across a wide range of problems. |