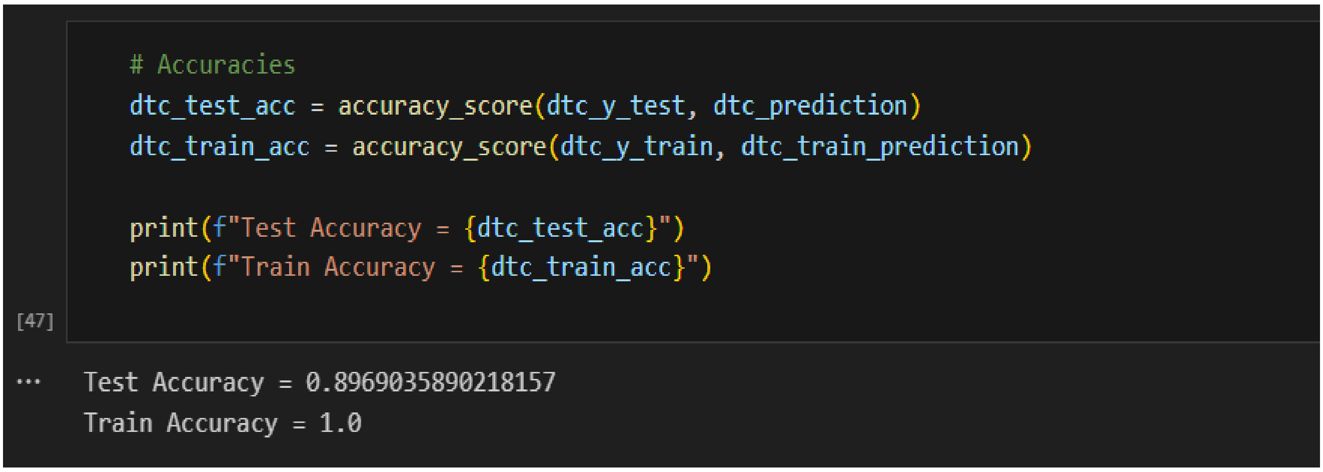
Project Development Phase
Model Performance Test

Date	November 2023
Team ID	Team-592444
Project Name	Anticipating Business Bankruptcy
Maximum Marks	10 Marks

Model Performance Testing:

Accuracy Scores:

- DECISION TREE CLASSIFIER:

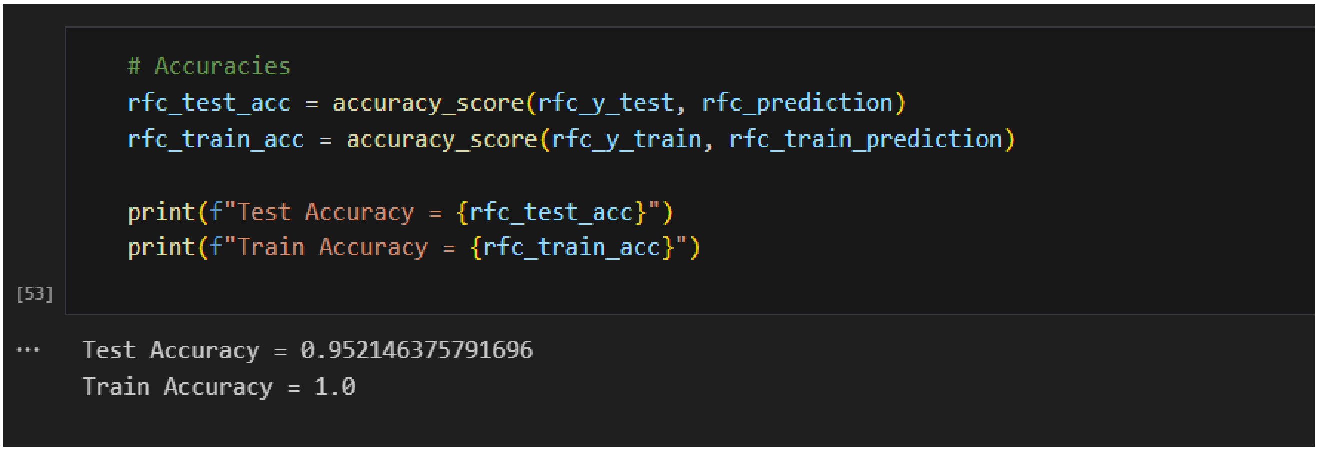


```
# Accuracies
dtc_test_acc = accuracy_score(dtc_y_test, dtc_prediction)
dtc_train_acc = accuracy_score(dtc_y_train, dtc_train_prediction)

print(f"Test Accuracy = {dtc_test_acc}")
print(f"Train Accuracy = {dtc_train_acc}")
```

```
Test Accuracy = 0.8969035890218157
Train Accuracy = 1.0
```

- RANDOM FOREST CLASSIFIER

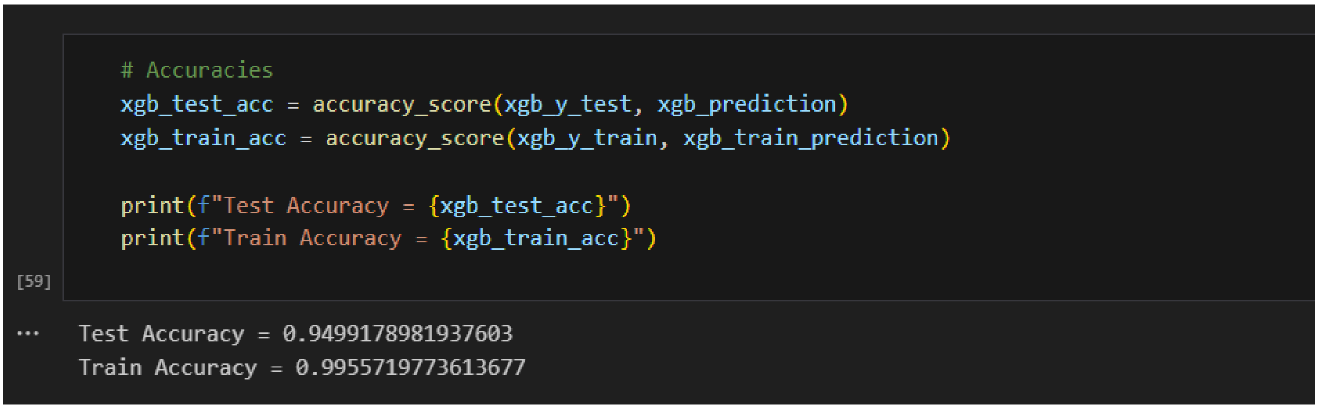


```
# Accuracies
rfc_test_acc = accuracy_score(rfc_y_test, rfc_prediction)
rfc_train_acc = accuracy_score(rfc_y_train, rfc_train_prediction)

print(f"Test Accuracy = {rfc_test_acc}")
print(f"Train Accuracy = {rfc_train_acc}")
```

```
Test Accuracy = 0.952146375791696
Train Accuracy = 1.0
```

- XGB CLASSIFIER



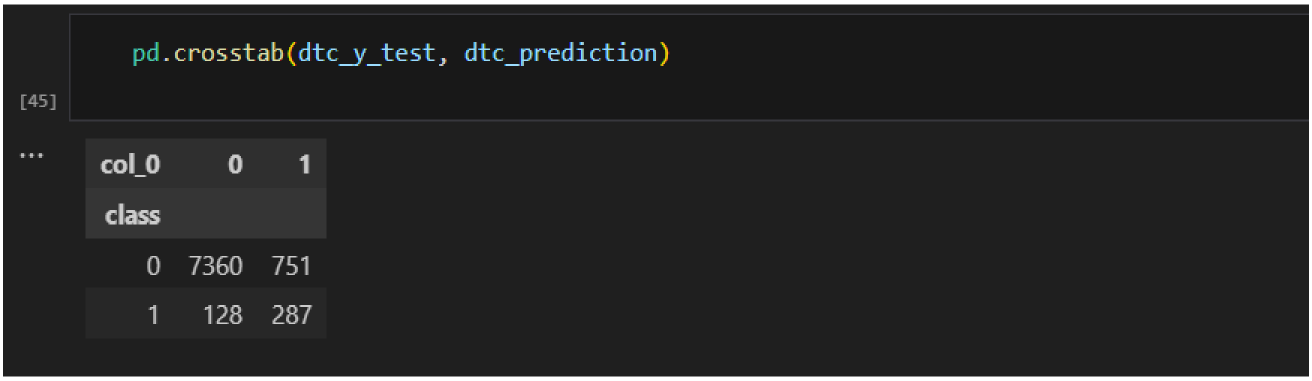
```
# Accuracies
xgb_test_acc = accuracy_score(xgb_y_test, xgb_prediction)
xgb_train_acc = accuracy_score(xgb_y_train, xgb_train_prediction)

print(f"Test Accuracy = {xgb_test_acc}")
print(f"Train Accuracy = {xgb_train_acc}")
```

```
Test Accuracy = 0.9499178981937603
Train Accuracy = 0.9955719773613677
```

Confusion Matrix:

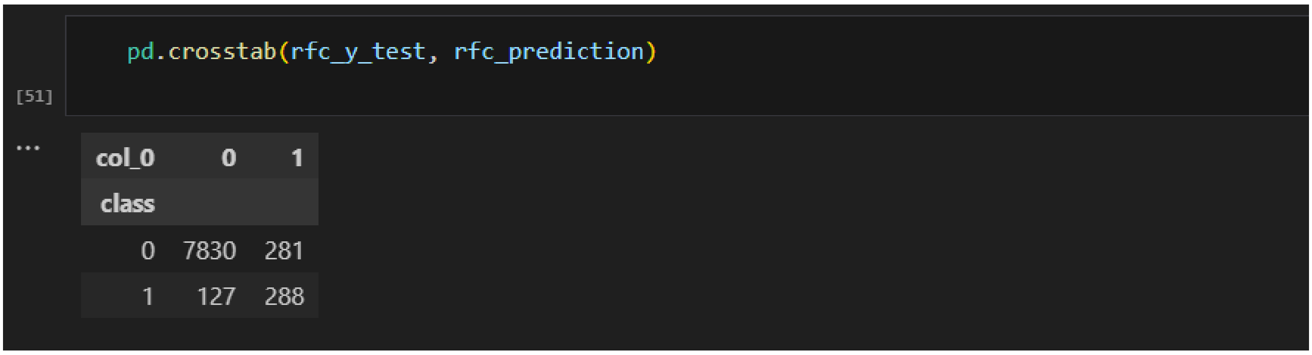
- DECISION TREE CLASSIFIER



```
[45]
pd.crosstab(dtc_y_test, dtc_prediction)
```

```
col_0     0    1
class
    0  7360  751
    1   128  287
```

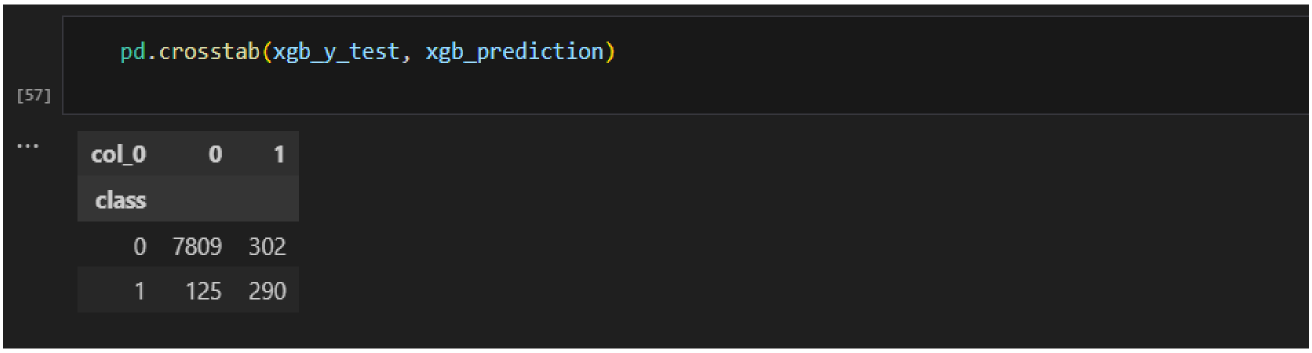
- RANDOM FOREST CLASSIFIER



```
[51]
pd.crosstab(rfc_y_test, rfc_prediction)
```

```
col_0     0    1
class
    0  7830  281
    1   127  288
```

- XGB CLASSIFIER

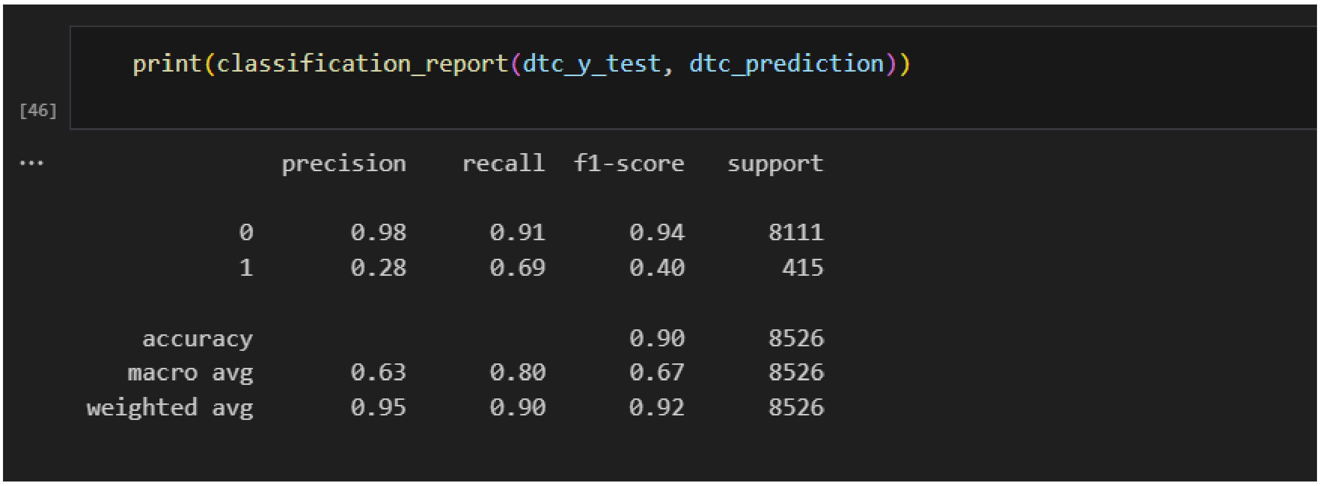


```
[57]
pd.crosstab(xgb_y_test, xgb_prediction)
```

```
col_0     0    1
class
    0  7809  302
    1   125  290
```

Classification Report:

- DECISION TREE CLASSIFIER



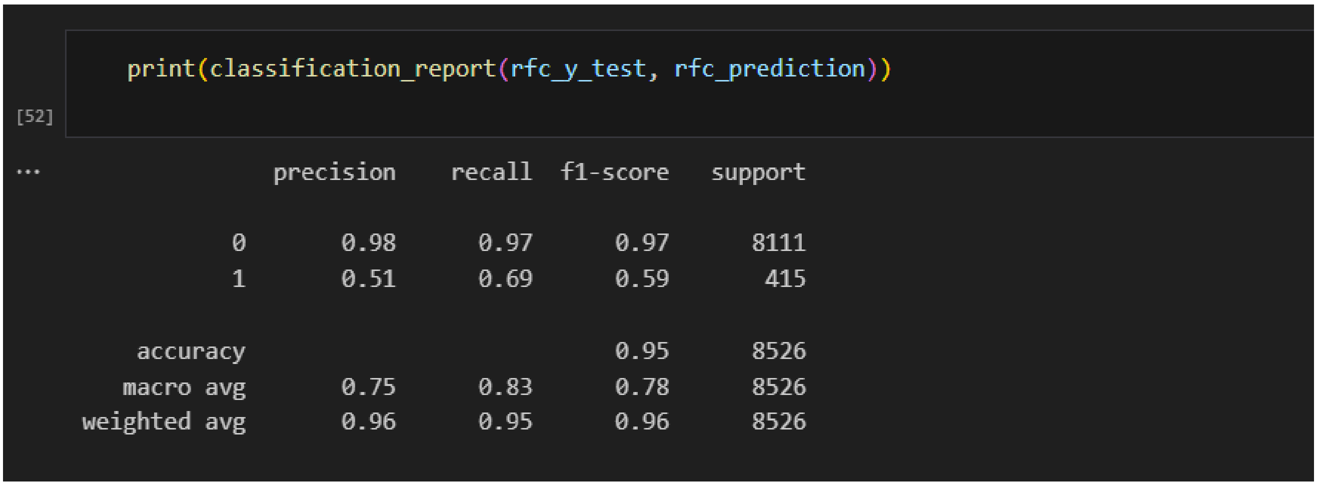
```
print(classification_report(dtc_y_test, dtc_prediction))
```

```
              precision    recall  f1-score   support

           0       0.98      0.91      0.94      8111
           1       0.28      0.69      0.40       415

    accuracy                           0.90      8526
   macro avg       0.63      0.80      0.67      8526
weighted avg       0.95      0.90      0.92      8526
```

- RANDOM FOREST CLASSIFIER



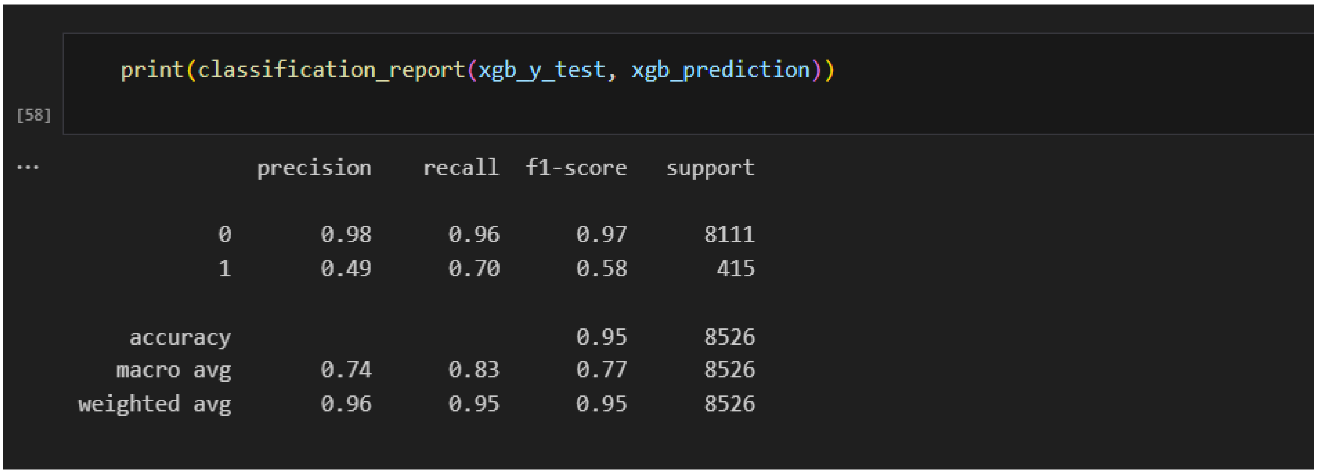
```
print(classification_report(rfc_y_test, rfc_prediction))
```

```
              precision    recall  f1-score   support

           0       0.98      0.97      0.97      8111
           1       0.51      0.69      0.59       415

    accuracy                           0.95      8526
   macro avg       0.75      0.83      0.78      8526
weighted avg       0.96      0.95      0.96      8526
```

- XGB CLASSIFIER



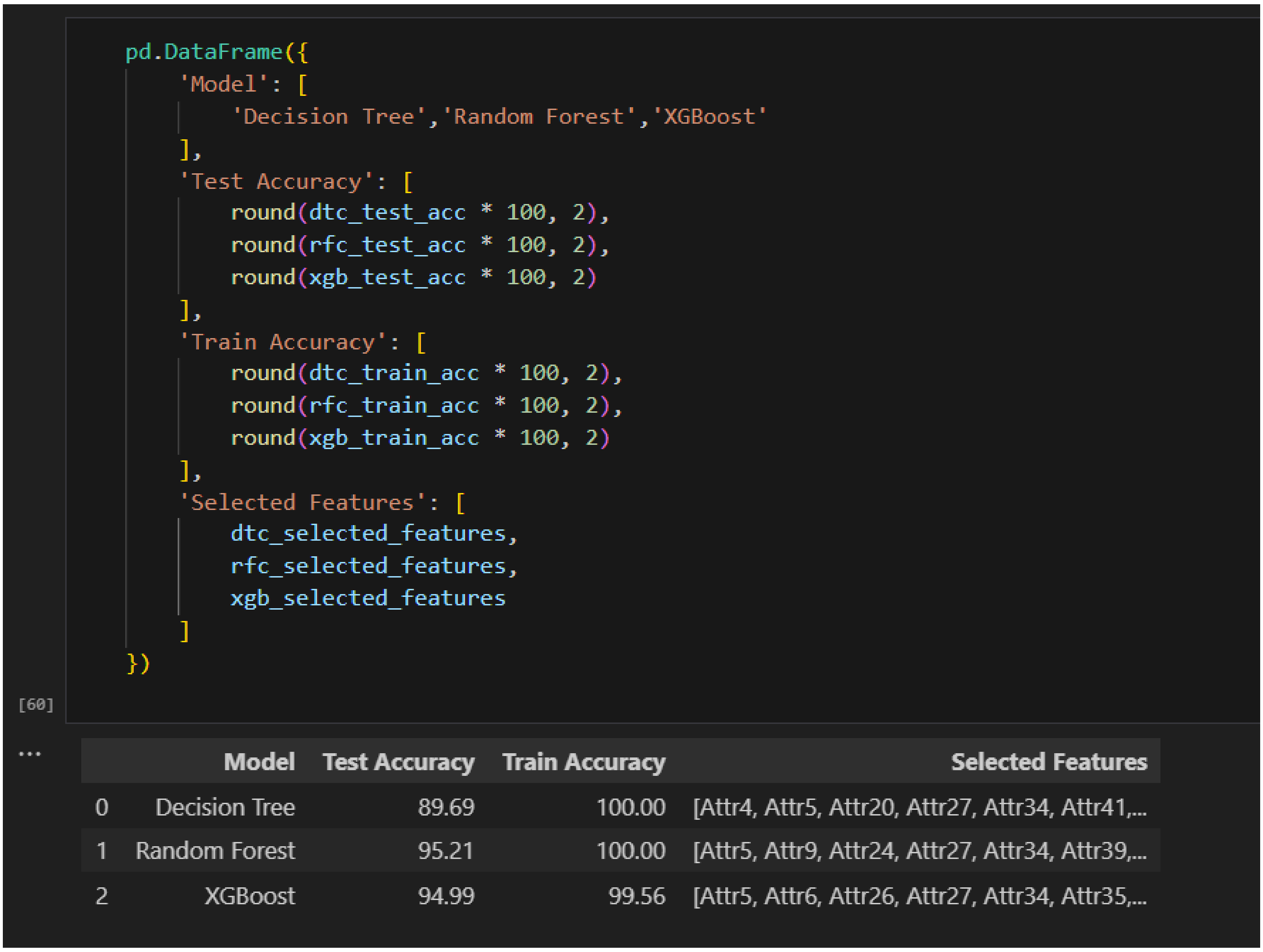
```
print(classification_report(xgb_y_test, xgb_prediction))
```

```
              precision    recall  f1-score   support

           0       0.98      0.96      0.97      8111
           1       0.49      0.70      0.58       415

    accuracy                           0.95      8526
   macro avg       0.74      0.83      0.77      8526
weighted avg       0.96      0.95      0.95      8526
```

Comparison:



```
pd.DataFrame({
    'Model': [
        'Decision Tree','Random Forest','XGBoost'
    ],
    'Test Accuracy': [
        round(dtc_test_acc * 100, 2),
        round(rfc_test_acc * 100, 2),
        round(xgb_test_acc * 100, 2)
    ],
    'Train Accuracy': [
        round(dtc_train_acc * 100, 2),
        round(rfc_train_acc * 100, 2),
        round(xgb_train_acc * 100, 2)
    ],
    'Selected Features': [
        dtc_selected_features,
        rfc_selected_features,
        xgb_selected_features
    ]
})
```

	Model	Test Accuracy	Train Accuracy	Selected Features
0	Decision Tree	89.69	100.00	[Attr4, Attr5, Attr20, Attr27, Attr34, Attr41,...
1	Random Forest	95.21	100.00	[Attr5, Attr9, Attr24, Attr27, Attr34, Attr39,...
2	XGBoost	94.99	99.56	[Attr5, Attr6, Attr26, Attr27, Attr34, Attr35,...

We can conclude that the Extreme Gradient Boosting Classifier emerges as the most promising choice. Its ability to maintain high test accuracy while avoiding overfitting, as evident in the training data, highlights its robustness and suitability for this binary classification task. Consequently, we conclude that the Extreme Gradient Boosting Classifier is the most effective and reliable model among the evaluated classifiers, providing superior results for this specific problem.