

## Model Development Phase

Date	28 July 2025
Project Title	Flight Delays Prediction Using Machine Learning
Maximum Marks	4 Marks

### Initial Model Training Code, Model Validation and Evaluation Report:

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

## Initial Model Training Code:

### 1. Decision Tree

```
sc = StandardScaler()  
x_train = sc.fit_transform(x_train)  
x_test = sc.transform(x_test)
```

23]

```
classifier = DecisionTreeClassifier()  
classifier = DecisionTreeClassifier(random_state = 0)  
classifier.fit(x_train, y_train)
```

24]

..

▼ DecisionTreeClassifier ⓘ ?  
► Parameters

### 2. Random Forest

```
rf = RandomForestClassifier(n_estimators=100, max_depth=None, random_state=0, n_jobs=-1)  
rf.fit(x_train, y_train)
```

25]

.. [d:\Projects\Flight Delays Prediction\env\Lib\site-packages\sklearn\base.py:1363: DataConversionWarning](#)  
return fit\_method(estimator, \*args, \*\*kwargs)

◀

..

▼ RandomForestClassifier ⓘ ?  
► Parameters

### 3. Support Vector Machine

```
svm = SVC(kernel='linear', C=1.0, gamma='scale', random_state=0)
svm.fit(x_train, y_train)
```

[26]

```
... d:\Projects\Flight Delays Prediction\env\Lib\site-packages\sklearn\utils\validation.py:109: UserWarning:
y = column_or_1d(y, warn=True)
```

...

▼ SVC ⓘ ?  
► Parameters

### 4. LightGBM

```
lgbm = LGBMClassifier(n_estimators=100, learning_rate=0.1, random_state=0)
lgbm.fit(x_train, y_train)
```

[42]

## Test the models

```
decisiontree = classifier.predict(x_test)
dt_pred = accuracy_score(y_test, decisiontree)
print("Decision Tree Metrics:")
print("Accuracy:", accuracy_score(y_test, decisiontree))
print("Precision:", precision_score(y_test, decisiontree))
print("Recall:", recall_score(y_test, decisiontree))
print("F1-Score:", f1_score(y_test, decisiontree))
print("\nClassification Report:\n", classification_report(y_test, decisiontree))

randomforest = rf.predict(x_test)
print("Random Forest Metrics:")
print("Accuracy:", accuracy_score(y_test, randomforest))
print("Precision:", precision_score(y_test, randomforest))
print("Recall:", recall_score(y_test, randomforest))
print("F1-Score:", f1_score(y_test, randomforest))
print("\nClassification Report:\n", classification_report(y_test, randomforest))

svmp = svm.predict(x_test)
print("SVM Metrics:")
print("Accuracy:", accuracy_score(y_test, svmp))
print("Precision:", precision_score(y_test, svmp))
print("Recall:", recall_score(y_test, svmp))
print("F1-Score:", f1_score(y_test, svmp))
print("\nClassification Report:\n", classification_report(y_test, svmp))

lgbmp = lgbm.predict(x_test)
print("LightGBM Metrics:")
print("Accuracy:", accuracy_score(y_test, lgbmp))
print("Precision:", precision_score(y_test, lgbmp))
print("Recall:", recall_score(y_test, lgbmp))
print("F1-Score:", f1_score(y_test, lgbmp))
print("\nClassification Report:\n", classification_report(y_test, lgbmp))
```

Model	Classification Report	F1 Score	Confusion Matrix
Decision Tree	<pre> Classification Report:               precision    recall  f1-score   support     False       0.99       0.99       0.99       1802    True        0.97       0.97       0.97        445   accuracy              0.99       2247  macro avg       0.98       0.98       0.98       2247  weighted avg    0.99       0.99       0.99       2247           </pre>	97.3	<pre> Confusion Matrix: [[1790  12]  [  12 433]]           </pre>
Random Forest	<pre> Classification Report:               precision    recall  f1-score   support     False       0.90       0.99       0.94       1802    True        0.93       0.58       0.71        445   accuracy              0.91       2247  macro avg       0.92       0.78       0.83       2247  weighted avg    0.91       0.91       0.90       2247           </pre>	71.2	<pre> Confusion Matrix: [[1782  20]  [ 188 257]]           </pre>
SVM	<pre> Classification Report:               precision    recall  f1-score   support     False       0.80       1.00       0.89       1802    True        0.00       0.00       0.00        445   accuracy              0.80       2247  macro avg       0.40       0.50       0.45       2247  weighted avg    0.64       0.80       0.71       2247           </pre>	0	<pre> Confusion Matrix: [[1802   0]  [ 445   0]]           </pre>

LGBM	<div><div>Classification Report:</div><table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>False</td><td>0.98</td><td>0.99</td><td>0.98</td><td>1802</td></tr><tr><td>True</td><td>0.95</td><td>0.93</td><td>0.94</td><td>445</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.98</td><td>2247</td></tr><tr><td>macro avg</td><td>0.97</td><td>0.96</td><td>0.96</td><td>2247</td></tr><tr><td>weighted avg</td><td>0.98</td><td>0.98</td><td>0.98</td><td>2247</td></tr></tbody></table></div>		precision	recall	f1-score	support	False	0.98	0.99	0.98	1802	True	0.95	0.93	0.94	445	accuracy			0.98	2247	macro avg	0.97	0.96	0.96	2247	weighted avg	0.98	0.98	0.98	2247	93.7	<div><div>Confusion Matrix:</div><pre>[[1780  22]  [  33 412]]</pre></div>
	precision	recall	f1-score	support																													
False	0.98	0.99	0.98	1802																													
True	0.95	0.93	0.94	445																													
accuracy			0.98	2247																													
macro avg	0.97	0.96	0.96	2247																													
weighted avg	0.98	0.98	0.98	2247																													