

Model Development Phase Template

Date	18 June 2024
Team ID	739634
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:

StandardScalar:

```
from sklearn.preprocessing import StandardScaler
le=LabelEncoder()
for i in range(x_train.shape[1]):
    if isinstance(x_train[0,i],str):
        x_train[:,i]=le.fit_transform(x_train[:,i])
        x_test[:,i]=le.fit_transform(x_test[:,i])
sc=StandardScaler()
x_train=sc.fit_transform(x_train)
x_test=sc.transform(x_test)
```

Logistic regression:

```
from sklearn.linear_model import LogisticRegression

lr = LogisticRegression()
lr.fit(x_train, y_train)

y_pred = lr.predict(x_test)

accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)
```

RandomForest:

```
from sklearn.ensemble import RandomForestClassifier
rf = RandomForestClassifier(n_estimators=100, random_state=0)
rf.fit(x_train, y_train)
y_pred_rf = rf.predict(x_test)
accuracy_rf = accuracy_score(y_test, y_pred_rf)
print("Accuracy:", accuracy_rf)
```

DecisionTree:

```
from sklearn.tree import DecisionTreeClassifier
classifier=DecisionTreeClassifier(random_state=0)
classifier.fit(x_train,y_train)
y_pred_classifier = classifier.predict(x_test)
accuracy_classifier = accuracy_score(y_test, y_pred_classifier)
print("Accuracy:", accuracy_classifier)
```

Model Validation and Evaluation Report:

Model	Classification Report	F1 Score	Confusion matrix
Logistic Regression	<pre>print("Classification report:\n", metrics.classification_report(y_test, y_pred))</pre>	92%	
	✓ 0.0s		
	Classification report:		<pre>cm = metrics.confusion_matrix(y_test, y_pred)</pre> <pre>print("Confusion matrix:\n", cm)</pre>
	precision recall f1-score support		✓ 0.0s
	0.0 0.96 0.95 0.95 1932		Confusion matrix:
1.0 0.69 0.71 0.70 293	[[1838 94]		
			[84 209]]
	accuracy 0.92 2225		
	macro avg 0.82 0.83 0.83 2225		
	weighted avg 0.92 0.92 0.92 2225		
Random forest	<pre>print("Classification report:\n", metrics.classification_report(y_test, y_pred_rf))</pre>	91%	
	✓ 0.0s		
	Classification report:		<pre>cm_rf = metrics.confusion_matrix(y_test, y_pred_rf)</pre> <pre>print("Confusion matrix:\n", cm_rf)</pre>
	precision recall f1-score support		✓ 0.0s
	0.0 0.94 0.96 0.95 1932		Confusion matrix:
1.0 0.71 0.59 0.65 293	[[1862 70]		
			[119 174]]
	accuracy 0.92 2225		
	macro avg 0.83 0.78 0.80 2225		
	weighted avg 0.91 0.92 0.91 2225		

Decision Tree	<pre>print("Classification report:\n", metrics.classification_report(y_test, y_pred_classifier))</pre> <p>✓ 0.0s</p> <p>Classification report:</p> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0.0</td><td>0.92</td><td>0.92</td><td>0.92</td><td>1932</td></tr><tr><td>1.0</td><td>0.47</td><td>0.50</td><td>0.49</td><td>293</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.86</td><td>2225</td></tr><tr><td>macro avg</td><td>0.70</td><td>0.71</td><td>0.70</td><td>2225</td></tr><tr><td>weighted avg</td><td>0.86</td><td>0.86</td><td>0.86</td><td>2225</td></tr></tbody></table>		precision	recall	f1-score	support	0.0	0.92	0.92	0.92	1932	1.0	0.47	0.50	0.49	293	accuracy			0.86	2225	macro avg	0.70	0.71	0.70	2225	weighted avg	0.86	0.86	0.86	2225	86%	<pre>cm_classifier = metrics.confusion_matrix(y_test, y_pred_classifier)</pre> <p>✓ 0.0s</p> <p>Confusion matrix:</p> <pre>[[1768 164] [146 147]]</pre>
	precision	recall	f1-score	support																													
0.0	0.92	0.92	0.92	1932																													
1.0	0.47	0.50	0.49	293																													
accuracy			0.86	2225																													
macro avg	0.70	0.71	0.70	2225																													
weighted avg	0.86	0.86	0.86	2225																													