Model Development Phase Template

Date	03-10-2024
Team ID	LTVIP2024TMID24897
Project Title	Flight delay prediction using ML
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:

Train Test split for all models:

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.3,random_state = 2)
```

+ Code + Markdown

```
y_train.value_counts()
```

```
0    2294725
1    1358745
Name: Is_Delayed, dtype: int64
```

Decision Tree:

```
from sklearn.tree import DecisionTreeClassifier
classifierDT = DecisionTreeClassifier(criterion = 'entropy', random_state = None)
classifierDT.fit(X_train_sc, y_train)
```

K nearest neighbors:

```
from sklearn.neighbors import KNeighborsClassifier
objClassifier=KNeighborsClassifier(n_neighbors=10,metric='minkowski',p=2)
objClassifier.fit(X_train_sc,y_train)
```

Logistic regression:

```
from sklearn.linear_model import LogisticRegression
classifier = LogisticRegression(random_state = 0)
classifier.fit(X_train_sc, y_train)
```

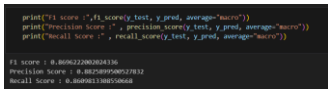
```
# Predicting the Test set results
y_pred = classifier.predict(X_test_sc)

# Making the Confusion Matrix
score = classifier.score(X_test_sc,y_test)
cm = confusion_matrix(y_test, y_pred)
```

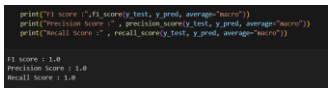
Model Validation and Evaluation Report:

Model	Classification report	Accuracy	Confusion matrix

Decision tree	score
	0.9826194584914554
	F1 score : 0.2725298912293622
	Precision Score : 0.6644134619299129
	Recall Score : 0.5006117468892067
		98%	# Predicting the Test set results
			y_pred = classifierDT.predict(X_test)
			# Making the Confusion Matrix
			cm = confusion_matrix(y_test, y_pred)
			score = classifierDT.score(X_test_sc,y_test)
			array([[1303, 982223],
			[59, 582189]])

K nearest neighbors		86%	from sklearn.metrics import confusion_matrix cm=confusion_matrix(y_test,y_pred) score=objClassifier.score(X_test,y_test) array([[925057, 58469], [127273, 454975]])

Confusion Ma

Logistic Regression		110%