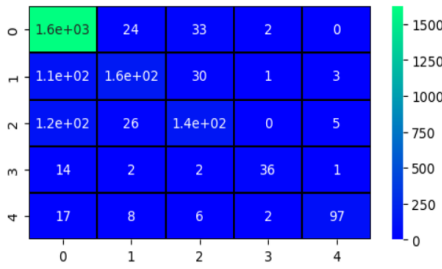


## Project Development Phase Model Performance Test

Date	10 November 2022
Team ID	PNT2022TMID32527
Project Name	Developing a Flight Delay Prediction Model using Machine Learning
Maximum Marks	10 Marks

### Model Performance Testing:

Project team shall fill the following information in model performance testing template.

S.No.	Parameter	Values	Screenshot																																													
1.	Metrics	<b>Classification Model:</b> Confusion Matrix – , Accuray Score- & Classification Report -	<pre>In [83]: from sklearn.metrics import confusion_matrix, accuracy_score, classification_report pred=rf.predict(x_test) cm=confusion_matrix(y_test, pred) plt.figure(figsize=(6,3)) sns.heatmap(cm, annot=True, cmap='winter', linewidths=0.3, linecolor='black', annot_kws={"size": 10}) TP=cm[0][0] TN=cm[1][1] FN=cm[1][0] FP=cm[0][1] #print(round(accuracy_score(prediction3,y_test)*100,2)) #print('Testing Accuracy for knn',(TP+TN)/(TP+TN+FN+FP)) print('Testing Sensitivity for Random Forest',(TP/(TP+FN))) print('Testing Specificity for Random Forest',(TN/(TN+FP))) print('Testing Precision for Random Forest',(TP/(TP+FP))) print('Testing accuracy for Random Forest',accuracy_score(y_test, pred))</pre> <p>Testing Sensitivity for Random Forest 0.9360230547550432 Testing Specificity for Random Forest 0.8716577540106952 Testing Precision for Random Forest 0.9854368932038835 Testing accuracy for Random Forest 0.8368506493506493</p>  <pre>In [69]: print(classification_report(y_test,pred))#RandomForest</pre> <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>0.86</td><td>0.96</td><td>0.91</td><td>1683</td></tr><tr><td>1.0</td><td>0.73</td><td>0.53</td><td>0.61</td><td>308</td></tr><tr><td>2.0</td><td>0.67</td><td>0.49</td><td>0.57</td><td>288</td></tr><tr><td>3.0</td><td>0.88</td><td>0.65</td><td>0.75</td><td>55</td></tr><tr><td>4.0</td><td>0.92</td><td>0.75</td><td>0.82</td><td>130</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.84</td><td>2464</td></tr><tr><td>macro avg</td><td>0.81</td><td>0.68</td><td>0.73</td><td>2464</td></tr><tr><td>weighted avg</td><td>0.83</td><td>0.84</td><td>0.83</td><td>2464</td></tr></tbody></table>		precision	recall	f1-score	support	0.0	0.86	0.96	0.91	1683	1.0	0.73	0.53	0.61	308	2.0	0.67	0.49	0.57	288	3.0	0.88	0.65	0.75	55	4.0	0.92	0.75	0.82	130	accuracy			0.84	2464	macro avg	0.81	0.68	0.73	2464	weighted avg	0.83	0.84	0.83	2464
	precision	recall	f1-score	support																																												
0.0	0.86	0.96	0.91	1683																																												
1.0	0.73	0.53	0.61	308																																												
2.0	0.67	0.49	0.57	288																																												
3.0	0.88	0.65	0.75	55																																												
4.0	0.92	0.75	0.82	130																																												
accuracy			0.84	2464																																												
macro avg	0.81	0.68	0.73	2464																																												
weighted avg	0.83	0.84	0.83	2464																																												

2.	Tune the Model	Hyperparameter Tuning - Validation Method -	<div><div>In [74]: <pre>from sklearn.model_selection import KFold, GridSearchCV, cross_val_score kfe = KFold(n_splits=5, shuffle=True, random_state=42) param={     'max_depth':[4,5,7,9,11],     'max_leaf_nodes':[5,6,8,9],     'random_state':[42,56,72],     'criterion':['gini','entropy','log_loss'] }</pre></div><div>In [77]: <pre>grcv=GridSearchCV(rf,param,cv=kf) grcv.fit(x_train,y_train) grcv.best_params_</pre></div><div>Out [77]: <pre>{'criterion': 'entropy',   'max_depth': 7,   'max_leaf_nodes': 9,   'random_state': 56}</pre></div><div>click to scroll output; double click to hide</div><div>In [76]: <pre>cv_result=cross_val_score(rf,x_train,y_train,cv=kf) cv_result</pre></div><div>Out [76]: <pre>array([0.81135903, 0.80974125, 0.8021309 , 0.81836631, 0.81887367])</pre></div></div>
----	----------------	---	---