PROJECT DEVELOPMENT PHASE

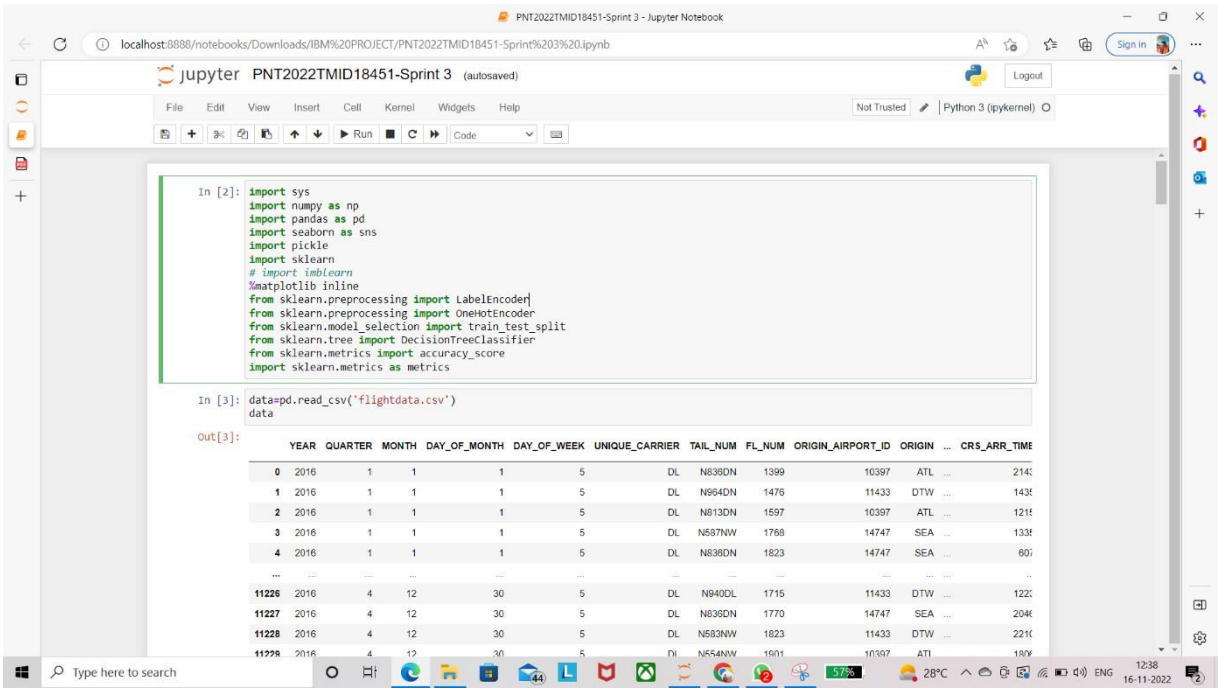
SPRINT 3 – CODE AND TESTCASE

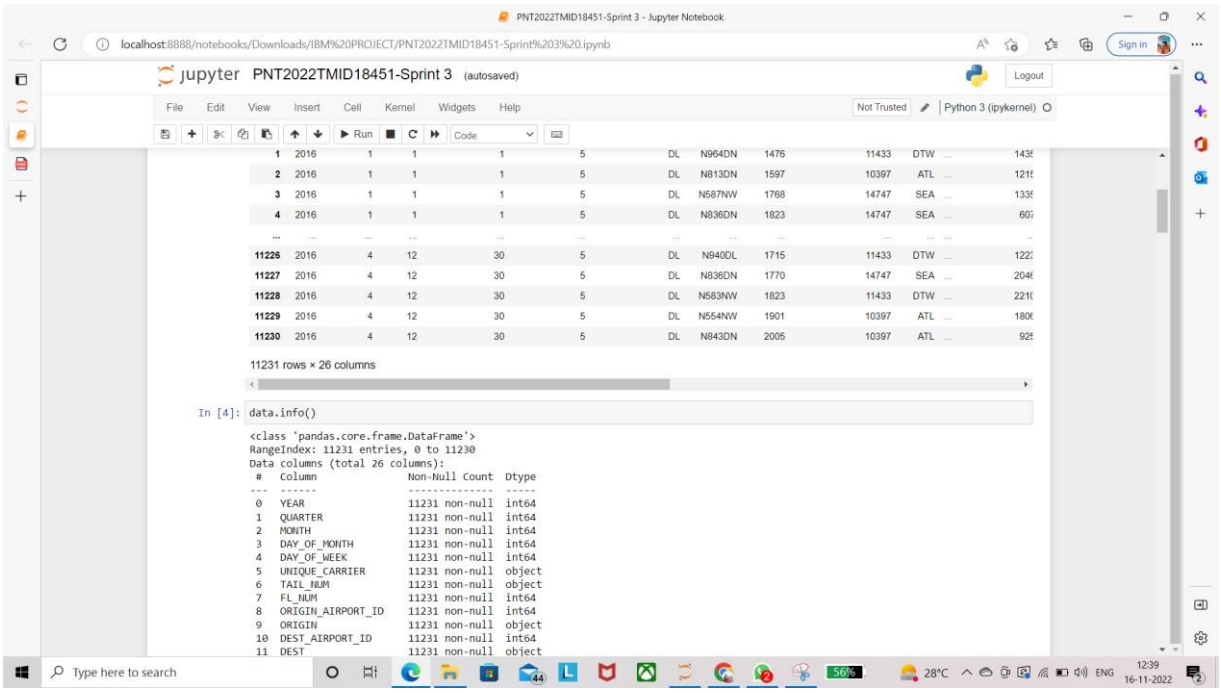
Date	16 November 2022
Team ID	PNT2022TMID18451
Project	Flight delay prediction using Machine learning
Marks	8 Marks

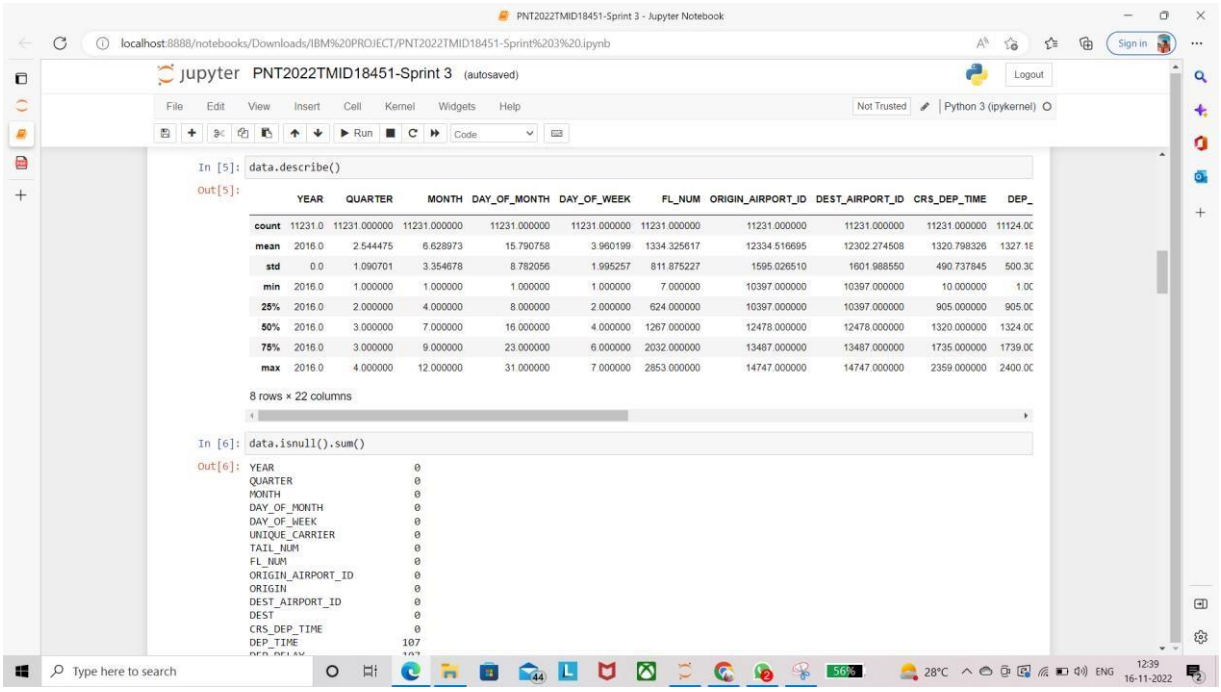
In this Sprint development phase, we have create an model with the help of Pre-processed dataset. We have used Decision Tree Classifier Algorithm for model development. Also we have implement method to check the accuracy of our model and convert the model into pkl file by importing Pickle python library. With the help of pickle model file the prediction is performed by Flask App.

Jupyter notebook :

Screenshots :




PNT2022TMID18451-Sprint 3 - Jupyter Notebook
localhost:8888/notebooks/Downloads/IBM%20PROJECT/PNT2022TMID18451-Sprint%203%20.ipynb
jupyter PNT2022TMID18451-Sprint 3 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
1 2016 1 1 1 5 DL N964DN 1476 11433 DTW ... 143
2 2016 1 1 1 5 DL N813DN 1597 10397 ATL ... 121
3 2016 1 1 1 5 DL N587NW 1768 14747 SEA ... 133
4 2016 1 1 1 5 DL N836DN 1823 14747 SEA ... 607
11226 2016 4 12 30 5 DL N940DL 1715 11433 DTW ... 122
11227 2016 4 12 30 5 DL N836DN 1770 14747 SEA ... 204
11228 2016 4 12 30 5 DL N583NW 1823 11433 DTW ... 221
11229 2016 4 12 30 5 DL N554NW 1901 10397 ATL ... 180
11230 2016 4 12 30 5 DL N843DN 2005 10397 ATL ... 925
11231 rows × 26 columns
In [4]: data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11231 entries, 0 to 11230
Data columns (total 26 columns):
Column Non-Null Count Dtype
0 YEAR 11231 non-null int64
1 QUARTER 11231 non-null int64
2 MONTH 11231 non-null int64
3 DAY_OF_MONTH 11231 non-null int64
4 DAY_OF_WEEK 11231 non-null int64
5 UNIQUE_CARRIER 11231 non-null object
6 TAIL_NUM 11231 non-null object
7 FL_NUM 11231 non-null int64
8 ORIGIN_AIRPORT_ID 11231 non-null int64
9 ORIGIN 11231 non-null object
10 DEST_AIRPORT_ID 11231 non-null int64
11 DEST 11231 non-null object
Type here to search
28°C
ENG
12:39
16-11-2022


PNT2022TMID18451-Sprint 3 - Jupyter Notebook
localhost:8888/notebooks/Downloads/IBM%20PROJECT/PNT2022TMID18451-Sprint%203%20.ipynb
jupyter PNT2022TMID18451-Sprint 3 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
In [5]: data.describe()
Out[5]:
YEAR QUARTER MONTH DAY_OF_MONTH DAY_OF_WEEK FL_NUM ORIGIN_AIRPORT_ID DEST_AIRPORT_ID CRS_DEP_TIME DEP_
count 11231.0 11231.000000 11231.000000 11231.000000 11231.000000 11231.000000 11231.000000 11231.000000 11231.000000 11124.0
mean 2016.0 2.544475 6.628973 15.790758 3.960199 1334.325617 12334.516695 12302.274508 1320.798326 1327.1
std 0.0 1.090701 3.354678 8.782056 1.995257 811.875227 1595.026510 1601.988550 490.737845 500.3
min 2016.0 1.000000 1.000000 1.000000 1.000000 7.000000 10397.000000 10397.000000 10.000000 1.0
25% 2016.0 2.000000 4.000000 8.000000 2.000000 624.000000 10397.000000 10397.000000 905.000000 905.0
50% 2016.0 3.000000 7.000000 16.000000 4.000000 1267.000000 12478.000000 12478.000000 1320.000000 1324.0
75% 2016.0 3.000000 9.000000 23.000000 6.000000 2032.000000 13487.000000 13487.000000 1735.000000 1739.0
max 2016.0 4.000000 12.000000 31.000000 7.000000 2853.000000 14747.000000 14747.000000 2359.000000 2400.0
8 rows × 22 columns
In [6]: data.isnull().sum()
Out[6]:
YEAR 0
QUARTER 0
MONTH 0
DAY_OF_MONTH 0
DAY_OF_WEEK 0
UNIQUE_CARRIER 0
TAIL_NUM 0
FL_NUM 0
ORIGIN_AIRPORT_ID 0
ORIGIN 0
DEST_AIRPORT_ID 0
DEST 0
CRS_DEP_TIME 0
DEP_TIME 107
Type here to search
28°C
ENG
12:39
16-11-2022

PNT2022TMID18451-Sprint 3 - Jupyter Notebook

localhost:8888/notebooks/Downloads/IBM%20PROJECT/PNT2022TMID18451-Sprint%203%20.ipynb

jupyter PNT2022TMID18451-Sprint 3 (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

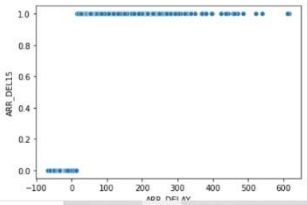
```
DEP_TIME                107
DEP_DELAY               107
DEP_DEL15               107
CRS_ARR_TIME              0
ARR_TIME                115
ARR_DELAY               188
ARR_DEL15               188
CANCELLED                 0
DIVERTED                  0
CRS_ELAPSED_TIME          0
ACTUAL_ELAPSED_TIME     188
DISTANCE                  0
Unnamed: 25           11231
dtype: int64
```

In [7]:
```
data['DEST'].unique()
```

Out[7]:
```
array(['SEA', 'MSP', 'DTW', 'ATL', 'JFK'], dtype=object)
```

In [8]:
```
sns.scatterplot(x='ARR_DELAY',y='ARR_DEL15',data=data)
```

Out[8]:
```
<AxesSubplot:xlabel='ARR_DELAY', ylabel='ARR_DEL15'>
```



Type here to search

28°C ENG 12:39 16-11-2022

PNT2022TMID18451-Sprint 3 - Jupyter Notebook

localhost:8888/notebooks/Downloads/IBM%20PROJECT/PNT2022TMID18451-Sprint%203%20.ipynb

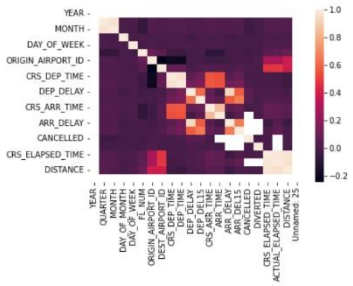
jupyter PNT2022TMID18451-Sprint 3 (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

In [9]:
```
sns.heatmap(data.corr())
```

Out[9]:
```
<AxesSubplot:>
```

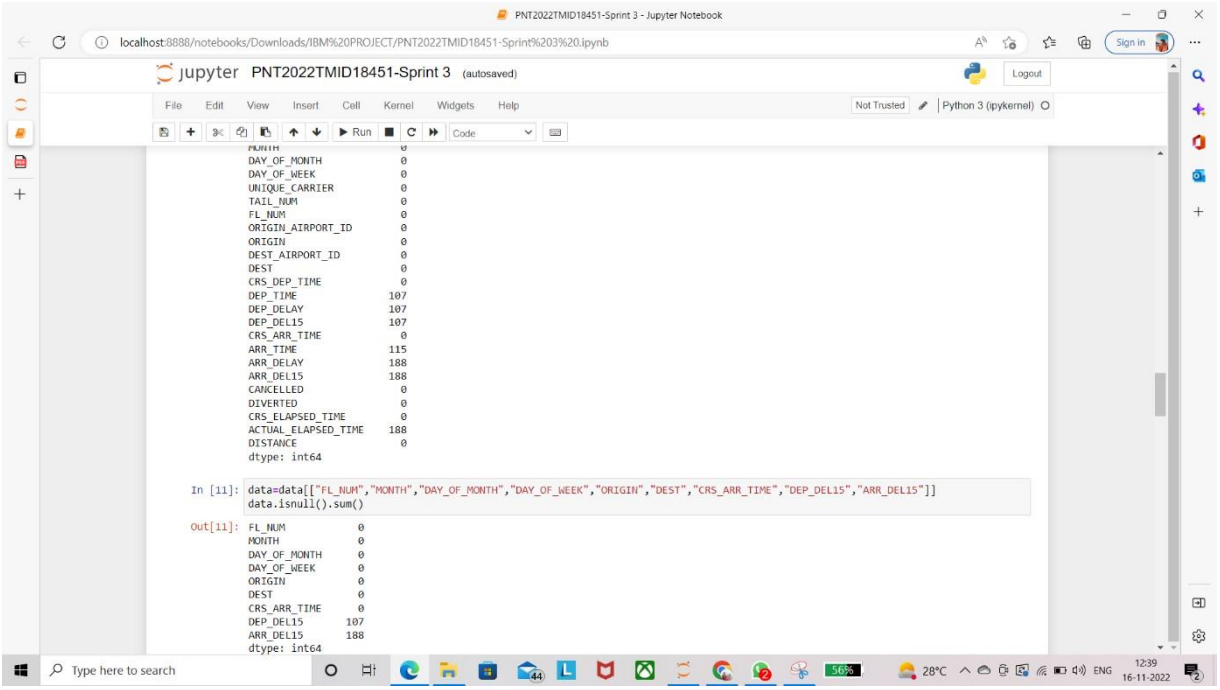


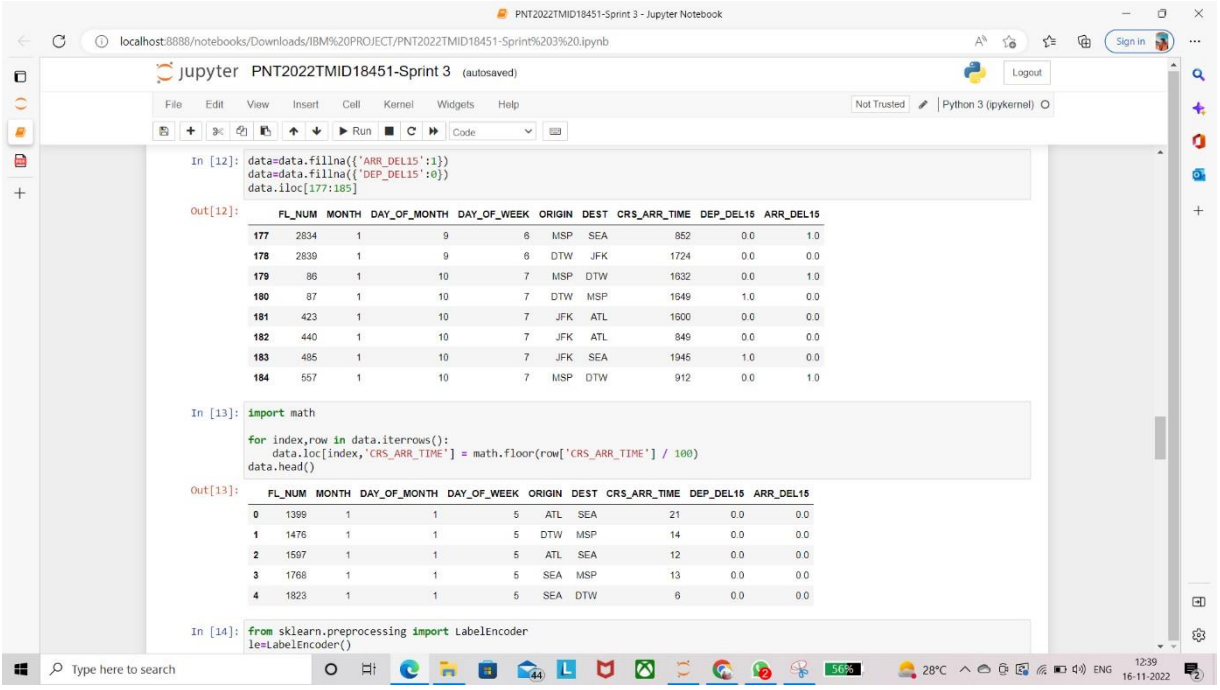
In [10]:
```
data=data.drop('Unnamed: 25',axis=1)
data.isnull().sum()
```

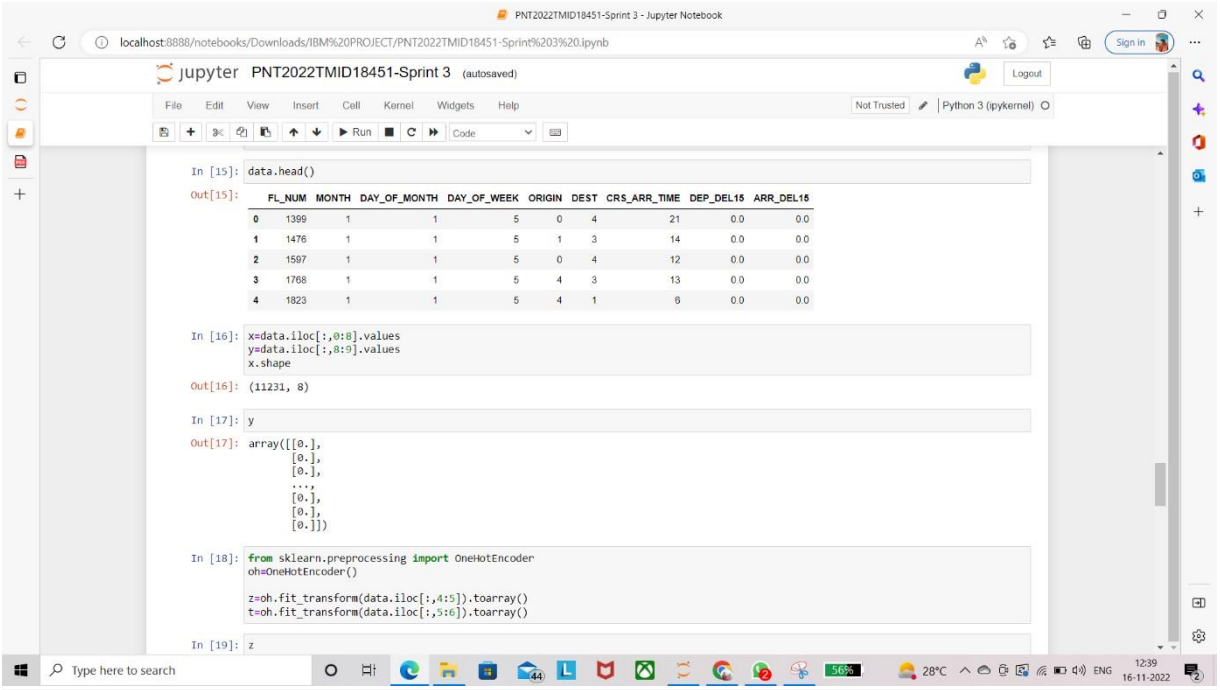
Out[10]:
```
YEAR                  0
QUARTER               0
MONTH                 0
DAY_OF_MONTH          0
DAY_OF_WEEK           0
UNIQUE_CARRIER        0
TAIL_NUM              0
FL_NUM                0
```

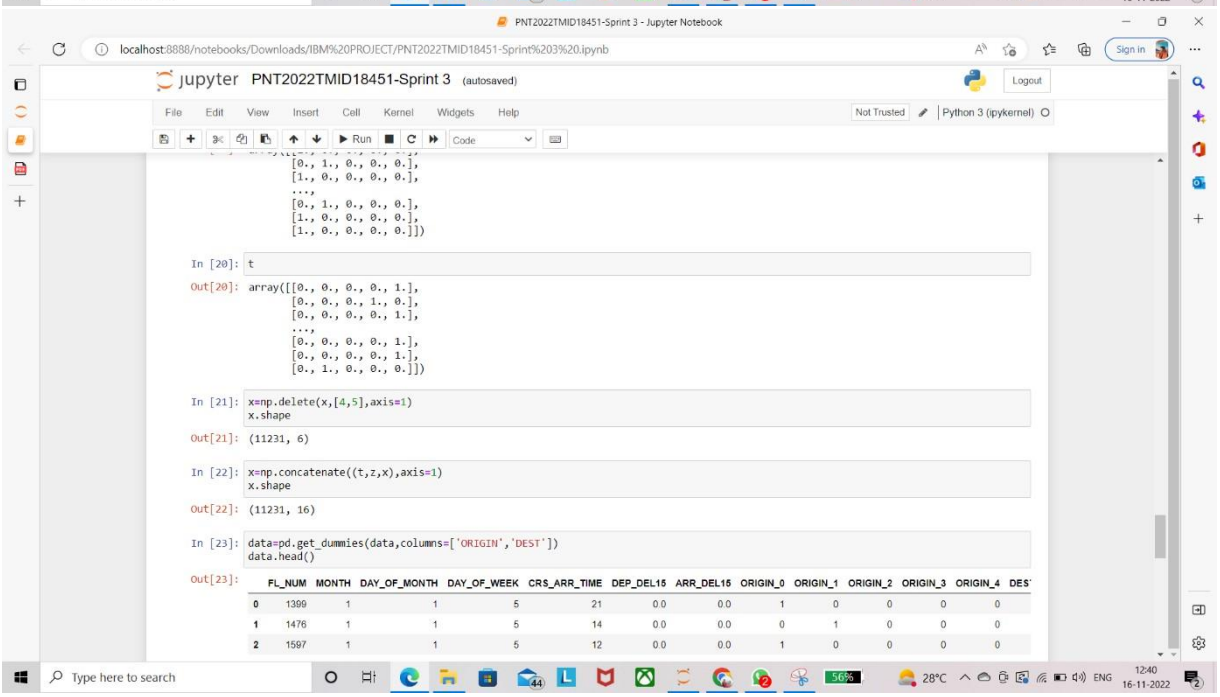
Type here to search

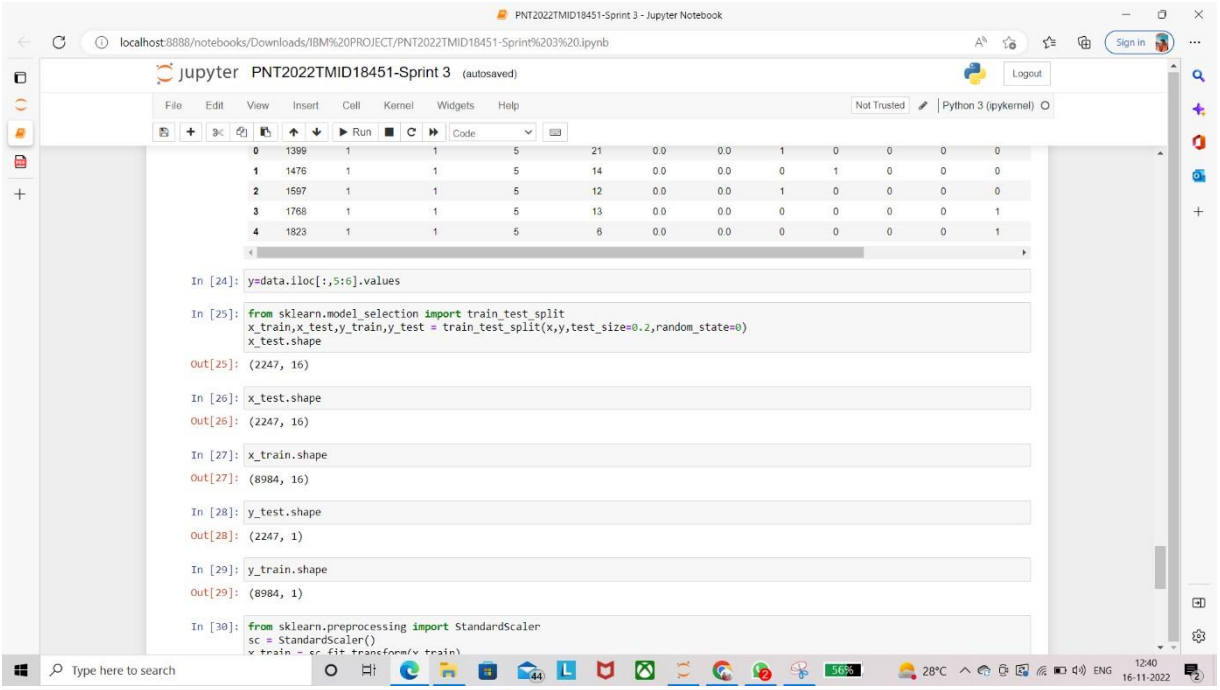
28°C ENG 12:39 16-11-2022

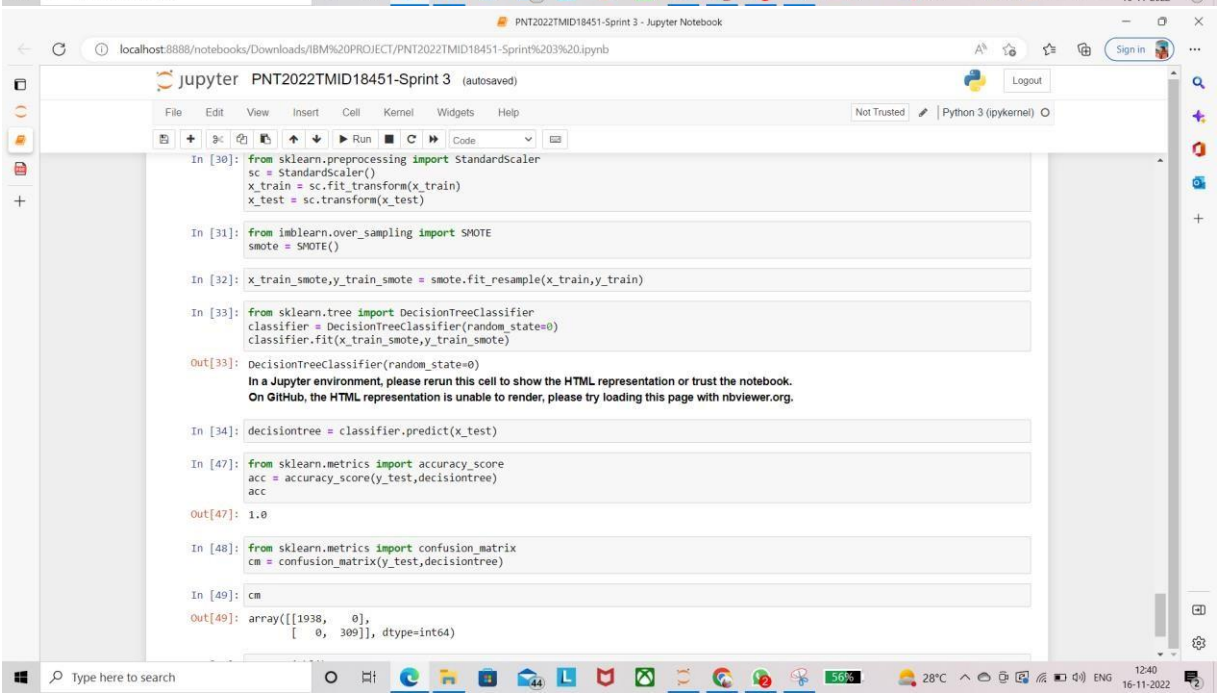

PNT2022TMID18451-Sprint 3 - Jupyter Notebook
localhost:8888/notebooks/Downloads/IBM%20PROJECT/PNT2022TMID18451-Sprint%203%20.ipynb
jupyter PNT2022TMID18451-Sprint 3 (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
MONTH 0
DAY_OF_MONTH 0
DAY_OF_WEEK 0
UNIQUE_CARRIER 0
TAIL_NUM 0
FL_NUM 0
ORIGIN_AIRPORT_ID 0
ORIGIN 0
DEST_AIRPORT_ID 0
DEST 0
CRS_DEP_TIME 0
DEP_TIME 107
DEP_DELAY 107
DEP_DEL15 107
CRS_ARR_TIME 0
ARR_TIME 115
ARR_DELAY 188
ARR_DEL15 188
CANCELLED 0
DIVERTED 0
CRS_ELAPSED_TIME 0
ACTUAL_ELAPSED_TIME 188
DISTANCE 0
dtype: int64
In [11]:
data=data[["FL_NUM","MONTH","DAY_OF_MONTH","DAY_OF_WEEK","ORIGIN","DEST","CRS_ARR_TIME","DEP_DEL15","ARR_DEL15"]]
data.isnull().sum()
Out[11]:
FL_NUM 0
MONTH 0
DAY_OF_MONTH 0
DAY_OF_WEEK 0
ORIGIN 0
DEST 0
CRS_ARR_TIME 0
DEP_DEL15 107
ARR_DEL15 188
dtype: int64
Type here to search
28°C
ENG
12:39
16-11-2022

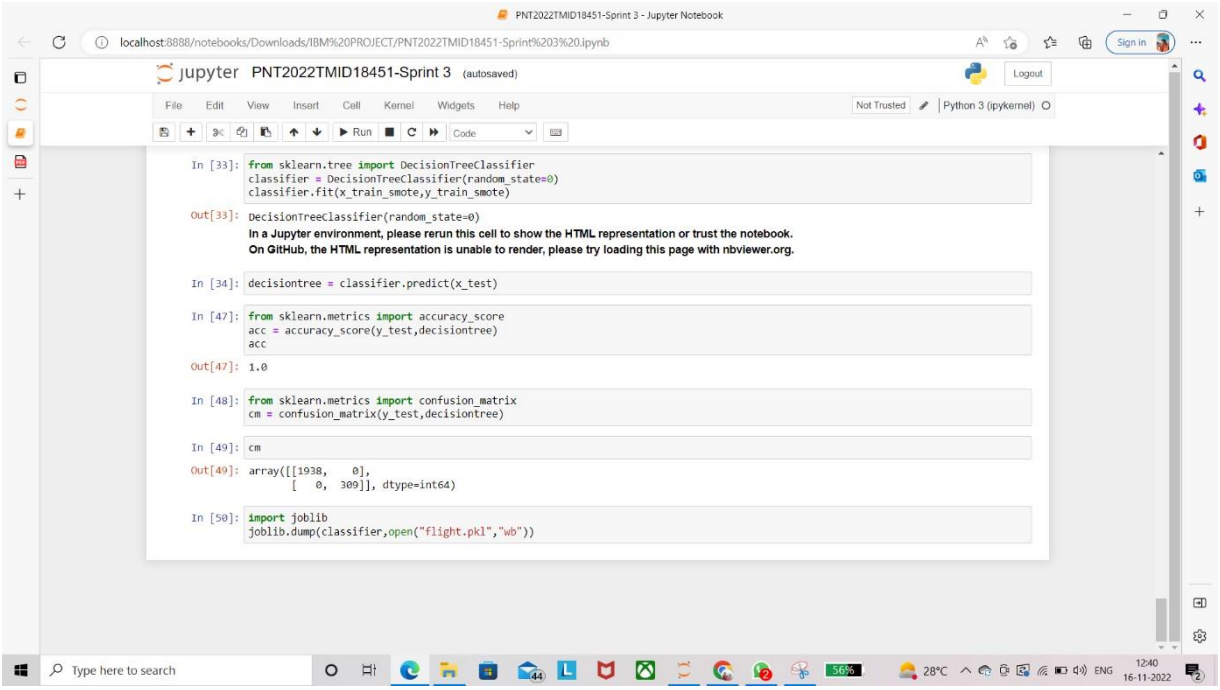

PNT2022TMID18451-Sprint 3 - Jupyter Notebook
localhost:8888/notebooks/Downloads/IBM%20PROJECT/PNT2022TMID18451-Sprint%203%20.ipynb
jupyter PNT2022TMID18451-Sprint 3 (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
In [12]:
data=data.fillna({'ARR_DEL15':1})
data=data.fillna({'DEP_DEL15':0})
data.iloc[177:185]
Out[12]:
FL_NUM MONTH DAY_OF_MONTH DAY_OF_WEEK ORIGIN DEST CRS_ARR_TIME DEP_DEL15 ARR_DEL15
177 2834 1 9 6 MSP SEA 852 0.0 1.0
178 2839 1 9 6 DTW JFK 1724 0.0 0.0
179 86 1 10 7 MSP DTW 1632 0.0 1.0
180 87 1 10 7 DTW MSP 1549 1.0 0.0
181 423 1 10 7 JFK ATL 1600 0.0 0.0
182 440 1 10 7 JFK ATL 849 0.0 0.0
183 485 1 10 7 JFK SEA 1945 1.0 0.0
184 557 1 10 7 MSP DTW 912 0.0 1.0
In [13]:
import math
for index,row in data.iterrows():
data.loc[index,'CRS_ARR_TIME'] = math.floor(row['CRS_ARR_TIME'] / 100)
data.head()
Out[13]:
FL_NUM MONTH DAY_OF_MONTH DAY_OF_WEEK ORIGIN DEST CRS_ARR_TIME DEP_DEL15 ARR_DEL15
0 1399 1 1 5 ATL SEA 21 0.0 0.0
1 1476 1 1 5 DTW MSP 14 0.0 0.0
2 1597 1 1 5 ATL SEA 12 0.0 0.0
3 1768 1 1 5 SEA MSP 13 0.0 0.0
4 1823 1 1 5 SEA DTW 6 0.0 0.0
In [14]:
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
Type here to search
28°C
ENG
12:39
16-11-2022


PNT2022TMID18451-Sprint 3 - Jupyter Notebook
localhost:8888/notebooks/Downloads/IBM%20PROJECT/PNT2022TMID18451-Sprint%203%20.ipynb
jupyter PNT2022TMID18451-Sprint 3 (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
In [15]: data.head()
Out[15]:
FL_NUM MONTH DAY_OF_MONTH DAY_OF_WEEK ORIGIN DEST CRS_ARR_TIME DEP_DEL15 ARR_DEL15
0 1399 1 1 5 0 4 21 0.0 0.0
1 1476 1 1 5 1 3 14 0.0 0.0
2 1597 1 1 5 0 4 12 0.0 0.0
3 1768 1 1 5 4 3 13 0.0 0.0
4 1823 1 1 5 4 1 6 0.0 0.0
In [16]: x=data.iloc[:,0:8].values
y=data.iloc[:,8:9].values
x.shape
Out[16]: (11231, 8)
In [17]: y
Out[17]: array([[0.],
[0.],
[0.],
...,
[0.],
[0.],
[0.]])
In [18]: from sklearn.preprocessing import OneHotEncoder
oh=OneHotEncoder()
z=oh.fit_transform(data.iloc[:,4:5]).toarray()
t=oh.fit_transform(data.iloc[:,5:6]).toarray()
In [19]: z
Type here to search
28°C
ENG
12:39
16-11-2022


PNT2022TMID18451-Sprint 3 - Jupyter Notebook
localhost:8888/notebooks/Downloads/IBM%20PROJECT/PNT2022TMID18451-Sprint%203%20.ipynb
jupyter PNT2022TMID18451-Sprint 3 (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
[0., 1., 0., 0., 0.],
[1., 0., 0., 0., 0.],
...,
[0., 1., 0., 0., 0.],
[1., 0., 0., 0., 0.],
[1., 0., 0., 0., 0.]])
In [20]: t
Out[20]: array([[0., 0., 0., 0., 1.],
[0., 0., 0., 1., 0.],
[0., 0., 0., 0., 1.],
...,
[0., 0., 0., 0., 1.],
[0., 0., 0., 0., 1.],
[0., 1., 0., 0., 0.]])
In [21]: x=np.delete(x,[4,5],axis=1)
x.shape
Out[21]: (11231, 6)
In [22]: x=np.concatenate((t,z,x),axis=1)
x.shape
Out[22]: (11231, 16)
In [23]: data=pd.get_dummies(data,columns=['ORIGIN','DEST'])
data.head()
Out[23]:
FL_NUM MONTH DAY_OF_MONTH DAY_OF_WEEK CRS_ARR_TIME DEP_DEL15 ARR_DEL15 ORIGIN_0 ORIGIN_1 ORIGIN_2 ORIGIN_3 ORIGIN_4 DES
0 1399 1 1 5 21 0.0 0.0 1 0 0 0 0
1 1476 1 1 5 14 0.0 0.0 0 1 0 0 0
2 1597 1 1 5 12 0.0 0.0 1 0 0 0 0
Type here to search
28°C
ENG
12:40
16-11-2022


PNT2022TMID18451-Sprint 3 - Jupyter Notebook
localhost:8888/notebooks/Downloads/IBM%20PROJECT/PNT2022TMID18451-Sprint%203%20.ipynb
jupyter PNT2022TMID18451-Sprint 3 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
0 1399 1 1 5 21 0.0 0.0 1 0 0 0 0
1 1476 1 1 5 14 0.0 0.0 0 1 0 0 0
2 1597 1 1 5 12 0.0 0.0 1 0 0 0 0
3 1768 1 1 5 13 0.0 0.0 0 0 0 0 1
4 1823 1 1 5 6 0.0 0.0 0 0 0 0 1
In [24]: y=data.iloc[:,5:6].values
In [25]: from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state=0)
x_test.shape
Out[25]: (2247, 16)
In [26]: x_test.shape
Out[26]: (2247, 16)
In [27]: x_train.shape
Out[27]: (8984, 16)
In [28]: y_test.shape
Out[28]: (2247, 1)
In [29]: y_train.shape
Out[29]: (8984, 1)
In [30]: from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
Type here to search
28°C
ENG
12:40
16-11-2022


PNT2022TMID18451-Sprint 3 - Jupyter Notebook
localhost:8888/notebooks/Downloads/IBM%20PROJECT/PNT2022TMID18451-Sprint%203%20.ipynb
jupyter PNT2022TMID18451-Sprint 3 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
In [30]: from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
x_train = sc.fit_transform(x_train)
x_test = sc.transform(x_test)
In [31]: from imblearn.over_sampling import SMOTE
smote = SMOTE()
In [32]: x_train_smote,y_train_smote = smote.fit_resample(x_train,y_train)
In [33]: from sklearn.tree import DecisionTreeClassifier
classifier = DecisionTreeClassifier(random_state=0)
classifier.fit(x_train_smote,y_train_smote)
Out[33]: DecisionTreeClassifier(random_state=0)
In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.
In [34]: decisiontree = classifier.predict(x_test)
In [47]: from sklearn.metrics import accuracy_score
acc = accuracy_score(y_test,decisiontree)
acc
Out[47]: 1.0
In [48]: from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test,decisiontree)
In [49]: cm
Out[49]: array([[1938, 0],
[0, 309]], dtype=int64)
Type here to search
28°C
ENG
12:40
16-11-2022



With the help of the flight.pkl file, we have developed the Web pages by using ‘app.py’ flask app to integrate with the our processed model which is pickle file.

App.py(Flask);

With the help of Flask app, the Machine learning model will get the predicted output and integrated with web page and display the Output to the User.

```
 from flask import Flask,request,
render_template  import numpy as np import
pandas as pd import pickle import os
model=pickle.load(open('flight.pkl','rb'))
app=Flask(__name__)

@app.route('/') def
home():
    return render_template('index2.html')

@app.route('/predicts', methods=['POST','GET'])
```

```
def predict():
    name=request.form['name']
month=request.form['month']
dayofmonth=request.form['dayofmonth']
dayofweek=request.form['dayofweek']
origin=request.form['origin']     if(origin=="msp"):
        origin1,origin2,origin3,origin4,origin5=0,0,0,0,1
if(origin=="dtw"):
        origin1,origin2,origin3,origin4,origin5=1,0,0,0,0
if(origin=="jfk"):
        origin1,origin2,origin3,origin4,origin5=0,0,1,0,0
if(origin=="sea"):
        origin1,origin2,origin3,origin4,origin5=0,1,0,0,0
if(origin=="alt"):
        origin1,origin2,origin3,origin4,origin5=0,0,0,1,0


    destination=request.form['destination']
if(destination=="msp"):
        destination1,destination2,destination3,destination4,destination5=0,0,0
,0,1
if(destination=="dtw"):
        destination1,destination2,destination3,destination4,destination5=1,0,0
,0,0
if(destination=="jfk"):
        destination1,destination2,destination3,destination4,destination5=0,0,1
,0,0
if(destination=="sea"):
        destination1,destination2,destination3,destination4,destination5=0,1,0
,0,0
if(destination=="atl"):
        destination1,destination2,destination3,destination4,destination5=0,0,0
,1,0
    dept=request.form['dept']
arrtime=request.form['arrtime']
actdept=request.form['actdept']        dept15 = int(dept) - int(actdept)
total=[[name,month,dayofmonth,dayofweek,origin1,origin2,origin3,origin4,or
igin5,destination1,destination2,destination3,destination4,destination5,dept,ar
rtime]]

y_pred=model.predict(total)
print(y_pred)       if(y_pred
== [0.]):
        ans="The Flight will be on time"
else:
        ans="The Flight will be Delayed"
```

```
    return
render_template("predict.html",showcase=ans)          if
__name__=='__main__':     app.run(debug = True)
```