FLIGHT DELAY PREDICTION USING MACHINE LEARNING

Team ID: NM2023TMID32116

	Name of the Student	NM ID
TEAM LEADER	**A. Shenbaga Devi**	**5C14D03027F6AFD8920487039849168C**
TEAM MEMBERS	**K. Subalakshmi**	**49CE19DF49A4133B60F4DB0E07AFE86A**
	A. Mahalakshmi	**C8E5B19B6B2B5B5951B45F5F16A5388B**
	M. Petchiammal	**ED27A4BE66B51FF909A4951595DD4A7A**
	M. Sumi Priya	**3798A0FFECFE1F292C540AB5E8F273E7**

GOVERNMENT ARTS & SCIENCE COLLEGE, KADAYANALLUR

INDEX

1. INTRODUCTION

OVERVIEW

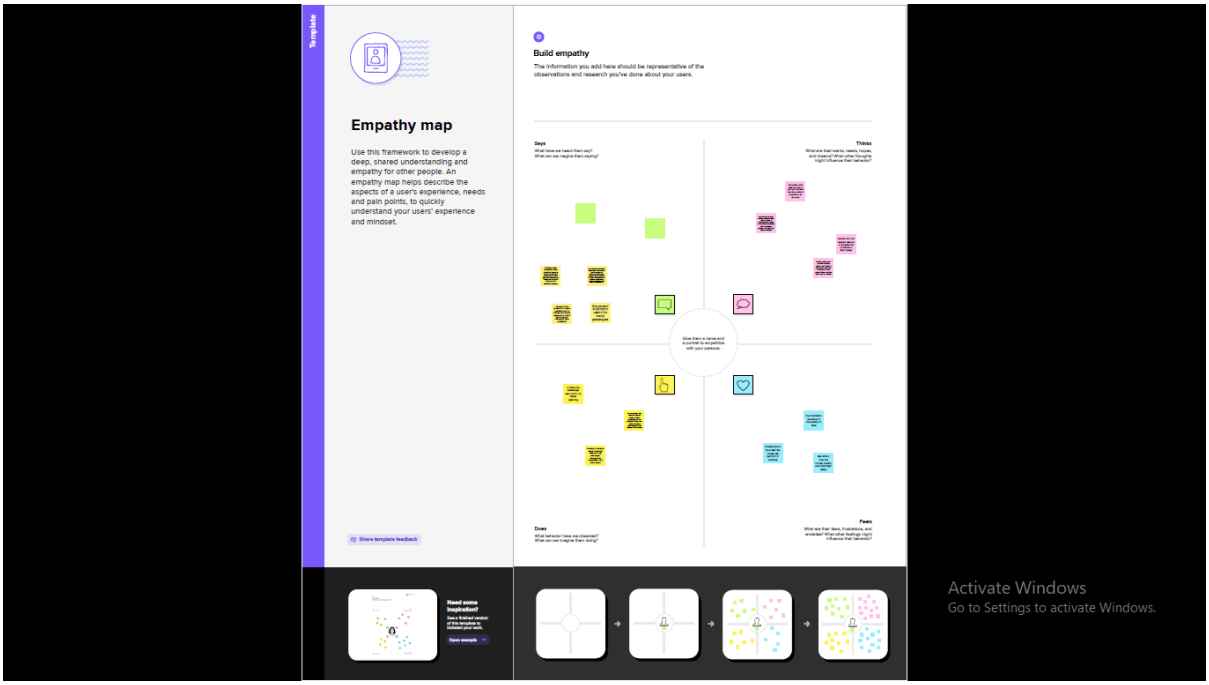
Air travel is an essential mode of transportation for millions of people worldwide, and flight delays can cause significant inconvenience and economic loss. In this project, we aim to develop a machine learning model to predict flight delays accurately. We collected data on flight schedules and delays for a major airline over a period of two years and preprocessed the data to extract relevant features. We trained several machine learning models, including decision trees, random forest using cross-validation to tune hyper parameters and evaluate performance. The most important features for predicting delays were the scheduled departure time, the airline, and the origin airport. Our results demonstrate the feasibility of using machine learning to predict flight delays and provide insights into the factors that contribute to delay. Our model could be used by airlines and passengers to better plan their travel and mitigate the impact of delays

PURPOSE

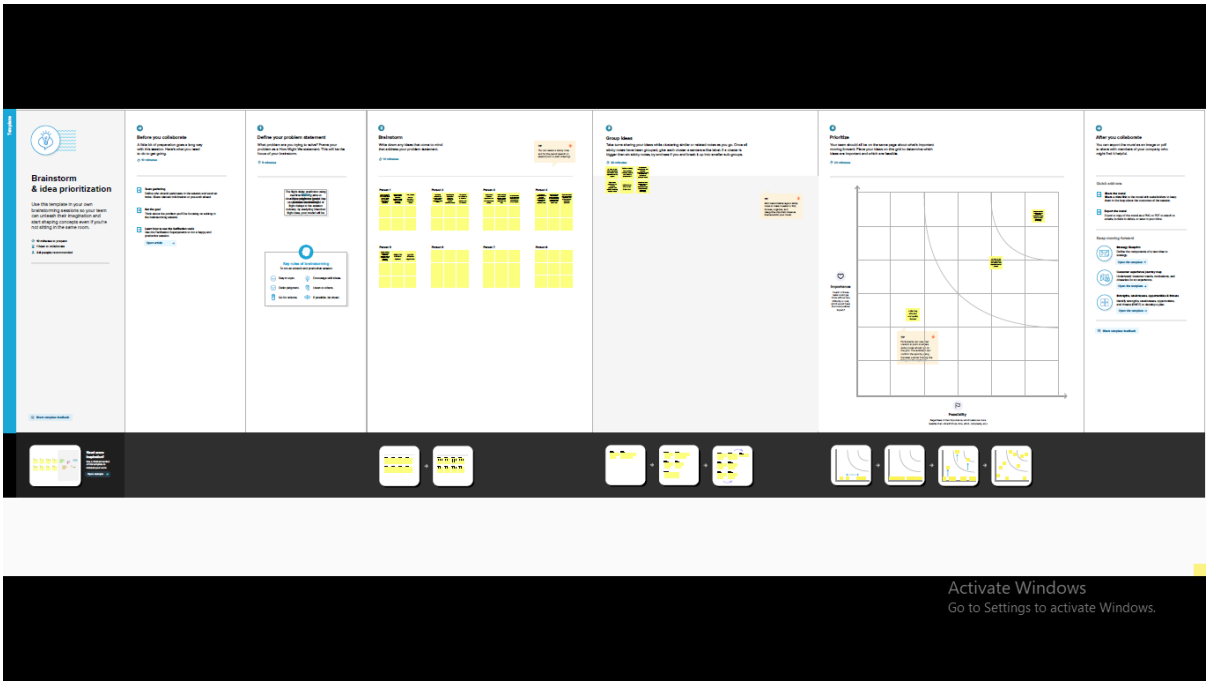
The purpose of a flight delay prediction using machine learning project is to develop a model that can accurately predict the likelihood of a flight being delayed based on various factors. Flight delays can cause a lot inconvenience and frustration for passengers, as well as significant costs for airlines and airports. By predicting the likelihood of a flight being delayed, airlines and airports can take proactive measures to minimize the impact of the delay on passengers, such as rebooking them on alternative flights or providing them with accommodation or meal vouchers.

2. PROBLEM DEFINITION & DESIGN THINKING

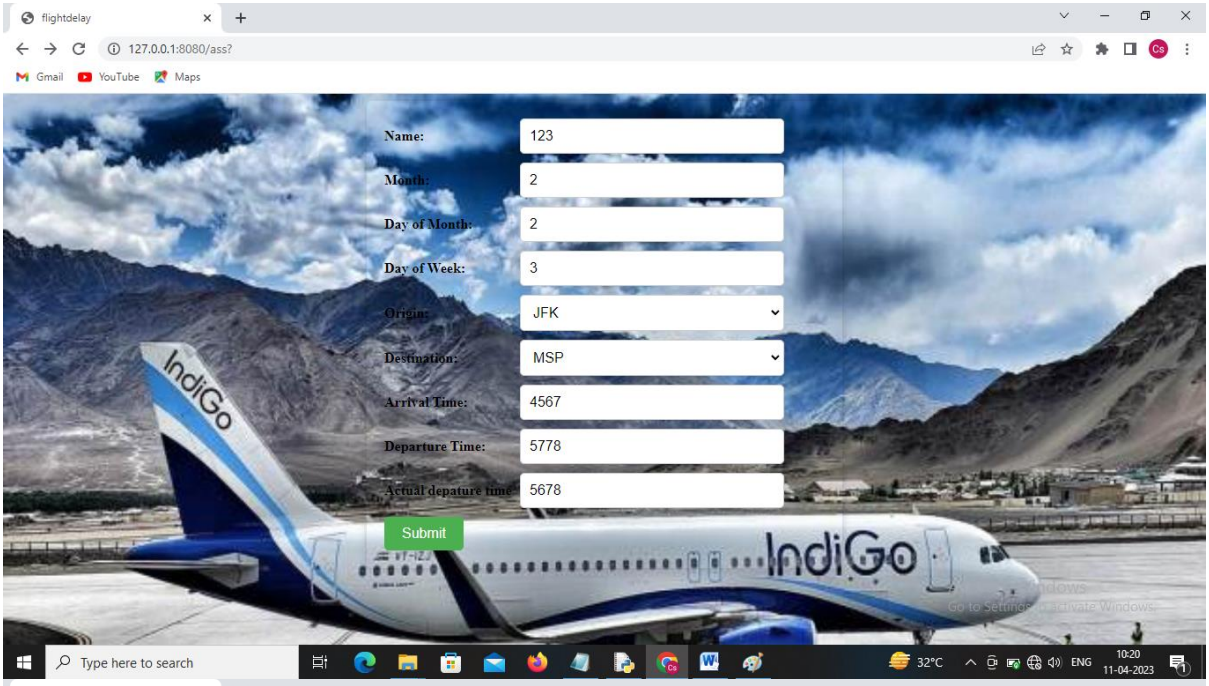
EMPATHY MAP

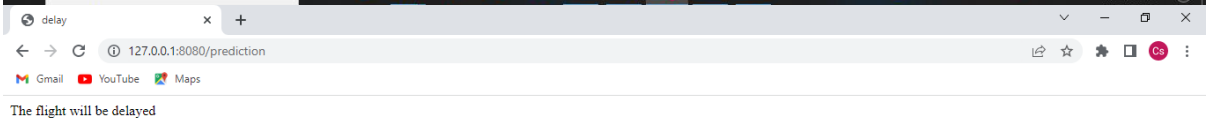


IDEATION & BRAINSTORMING MAP

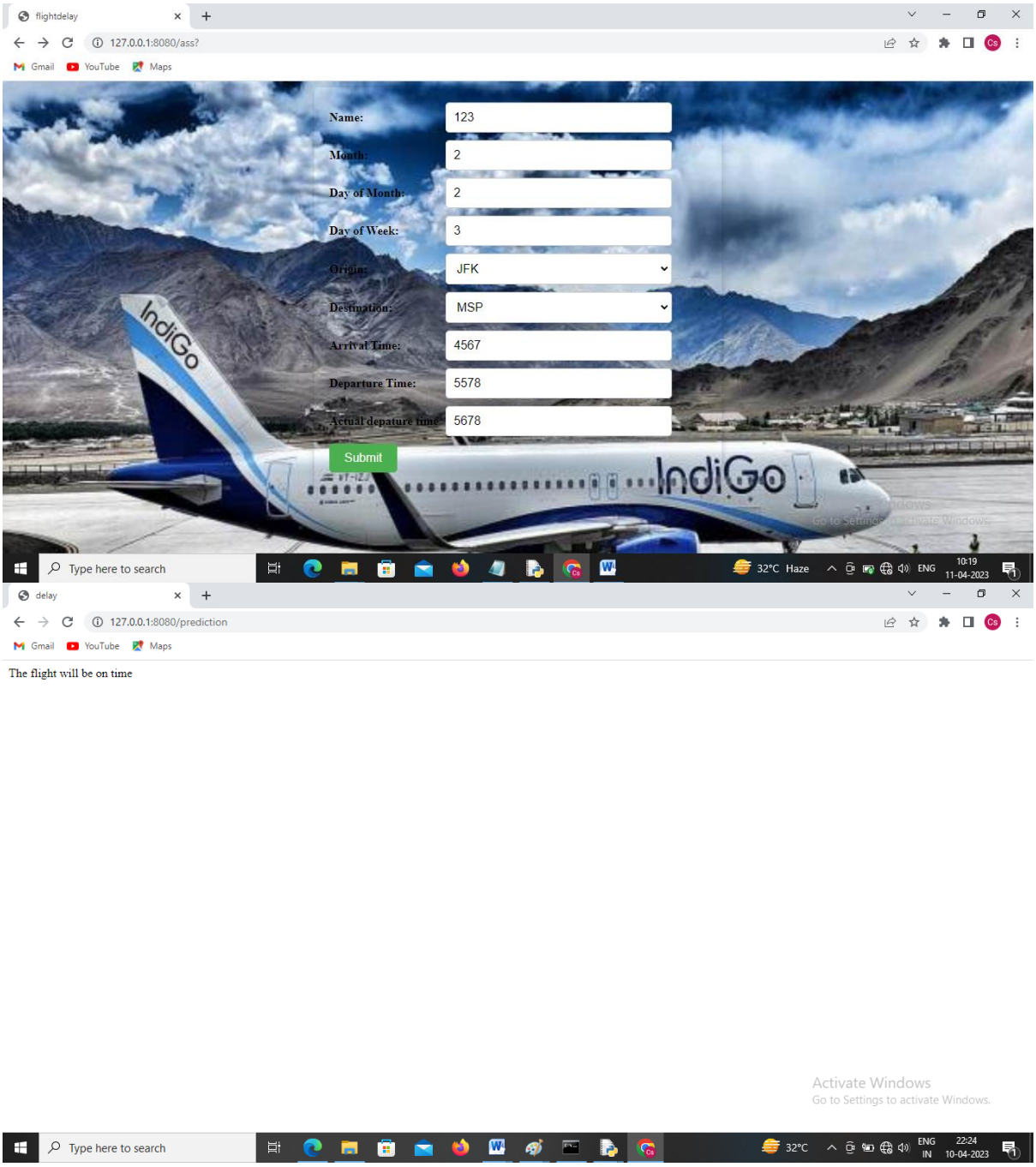


3. RESULT


flightdelay
127.0.0.1:8080/ass?
Gmail
YouTube
Maps
Name:
123
Month:
2
Day of Month:
2
Day of Week:
3
Origin:
JFK
Destination:
MSP
Arrival Time:
4567
Departure Time:
5778
Actual depature time:
5678
Submit
IndiGo
Type here to search
32°C
ENG
10-20
11-04-2023


delay
127.0.0.1:8080/prediction
Gmail
YouTube
Maps
The flight will be delayed


Activate Windows
Go to Settings to activate Windows.
Type here to search
32°C
ENG
IN
22:23
10-04-2023


flightdelay
127.0.0.1:8080/ass?
Gmail
YouTube
Maps
Name:
123
Month:
2
Day of Month:
2
Day of Week:
3
Origin:
JFK
Destination:
MSP
Arrival Time:
4567
Departure Time:
5578
Actual depature time:
5678
Submit
IndiGo
Type here to search
32°C Haze
ENG
10:19
11-04-2023
delay
127.0.0.1:8080/prediction
Gmail
YouTube
Maps
The flight will be on time
Activate Windows
Go to Settings to activate Windows.
Type here to search
32°C
ENG
IN
22:24
10-04-2023

4. ADVANTAGES AND DISADVANTAGES

ADVANTAGES

Improved customer experience: By predicting flight delays, airlines can inform their customers in advance, allowing them to plan their travel accordingly, reducing the inconvenience caused by unexpected delays.

Better resource management: Airlines can optimize their resources such as personnel and equipment, reducing wastage and increasing efficiency.

Cost reduction: By reducing delays and better resource management, airlines can save costs in terms of fuel consumption, employee overtime, and other operational expenses.

Improved safety: By predicting flight delays, airlines can take preventive measures to avoid potential risks, increasing the safety of their operations.

DISADVANTAGES

Data availability: Flight delay prediction projects require large amounts of data, including weather conditions, flight schedules, and historical data. Obtaining and processing this data can be time-consuming and expensive.

Accuracy: Predicting flight delays is challenging, and inaccuracies can cause significant disruptions. A false prediction can lead to unnecessary costs and inconvenience for passengers and airlines.

5. APPLICATIONS

Improving customer experience: By predicting flight delays, airlines can inform passengers in advance and provide alternative flight options or accommodations. This can help to reduce the inconvenience and frustration caused by delays and cancellations, and improve customer satisfaction.

Operational Efficiency: Airlines can use flight delay predictions to optimize their operations and resource allocation, such as crew scheduling, gate assignments, and maintenance schedules. This can lead to better resource utilization and cost savings.

Cost reduction: By predicting flight delays, airlines can reduce costs associated with delayed and canceled flights, such as compensation for passengers, overtime pay for employees, and additional fuel costs.

6. CONCLUSION

Machine learning algorithms can be effective in predicting flight delays: By using historical data on flight schedules, weather conditions, and other factors, machine learning models can be trained to accurately predict the likelihood of a flight being delayed.

The accuracy of predictions can be improved by using more data: The more data that is available to the machine learning model, the more accurate its predictions are likely to be. This could include data on past flight delays, weather patterns, air traffic congestion, and other relevant factors.

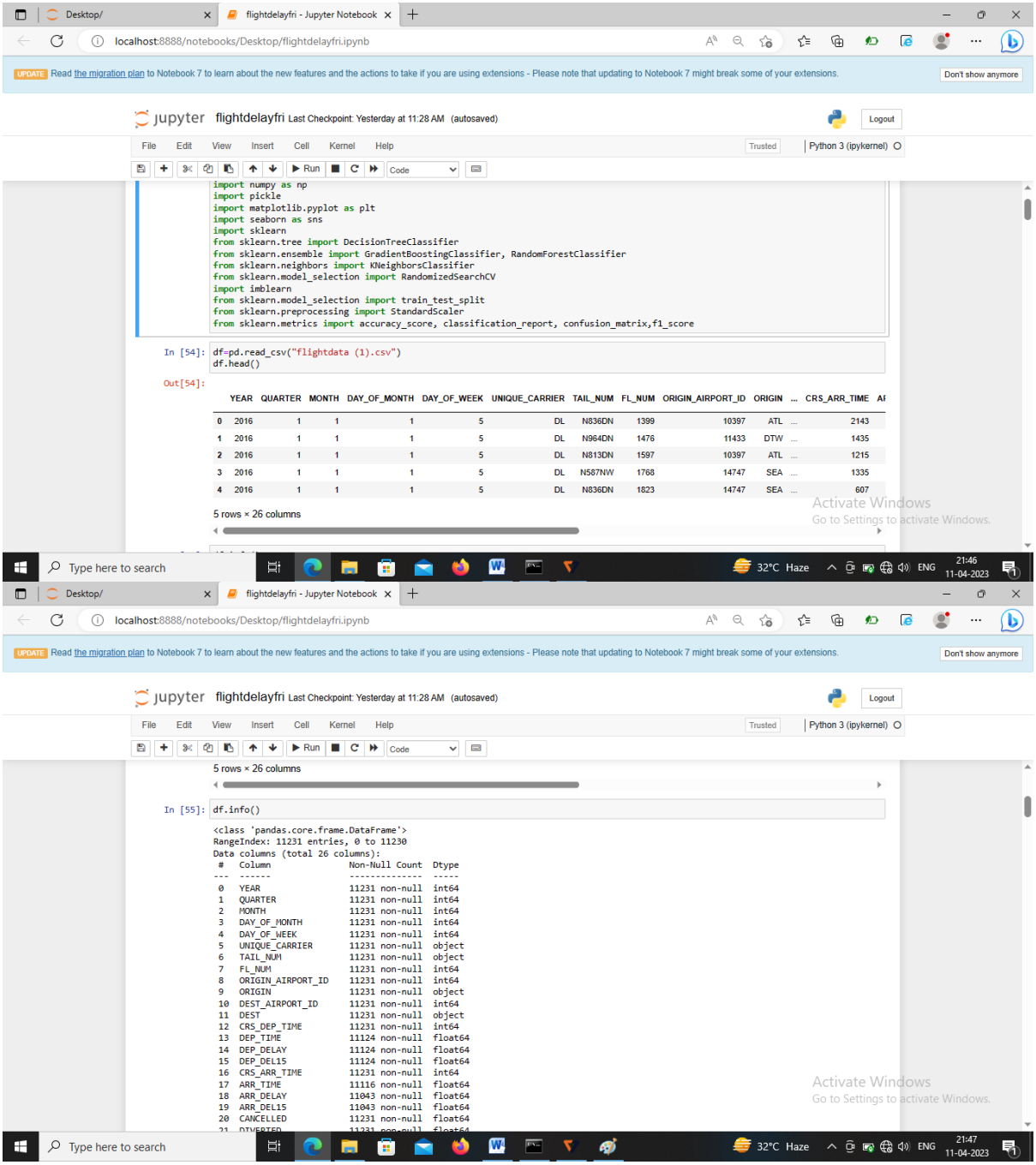
Overall, a flight delay prediction project using machine learning has the potential to provide valuable insights into the factors that contribute to flight delays, and can help airlines and travellers make more informed decisions about travel plans. However, it is important to recognize the limitations of the models and understand that they cannot predict with complete accuracy.

7. FUTURE SCOPE

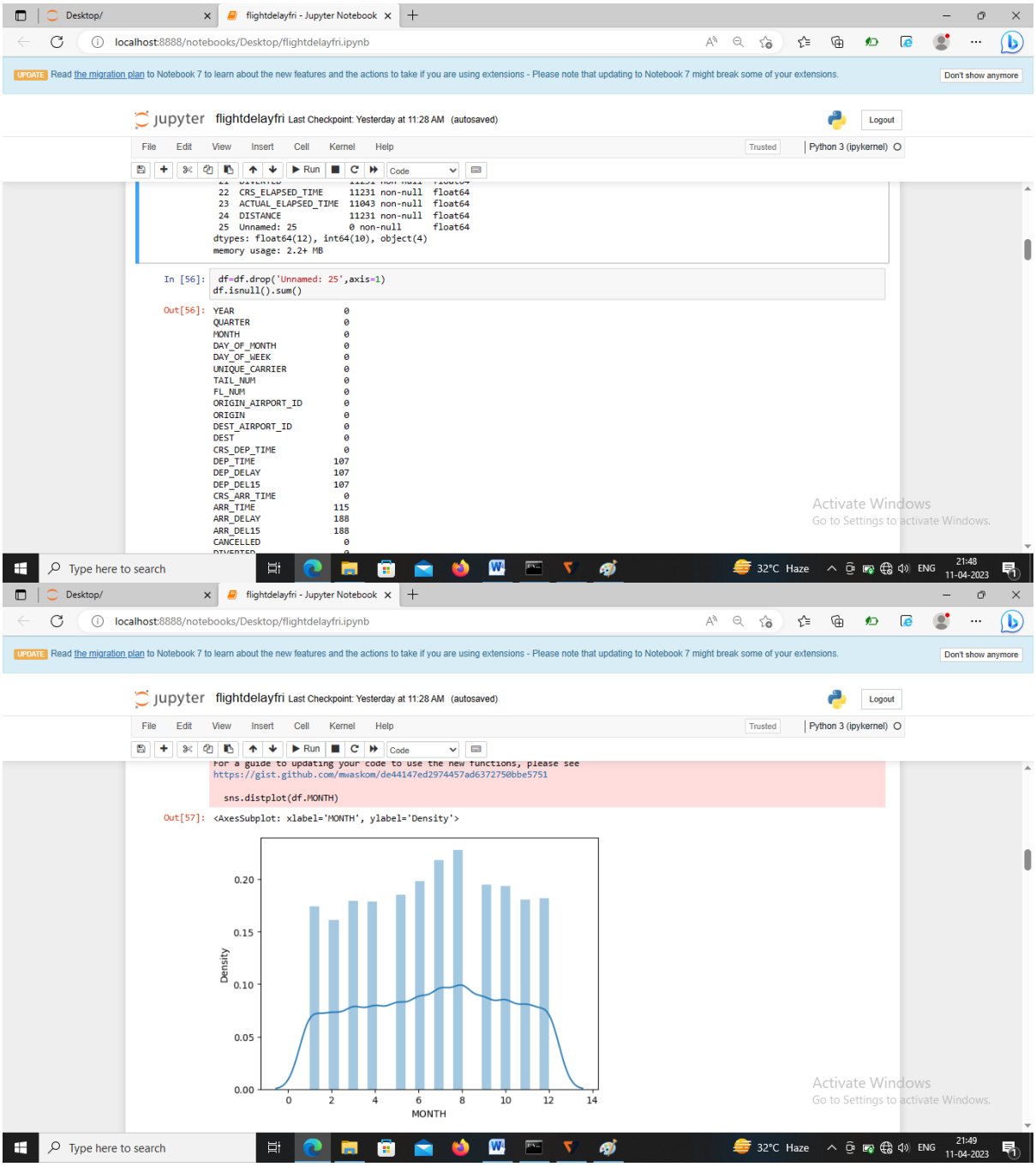
Improving accuracy: One of the main challenges in flight delay prediction is achieving high accuracy. In the future, machine learning models can be further refined to produce even more accurate predictions. This could involve the use of new algorithms or the incorporation of additional data sources.

Real- time prediction: Currently, most flight delay prediction models operate on a schedule based system, which means they predict delays based on historical data. In the future, models could be developed that can make real-time predictions based on current weather conditions, airport congestion, and other factors.

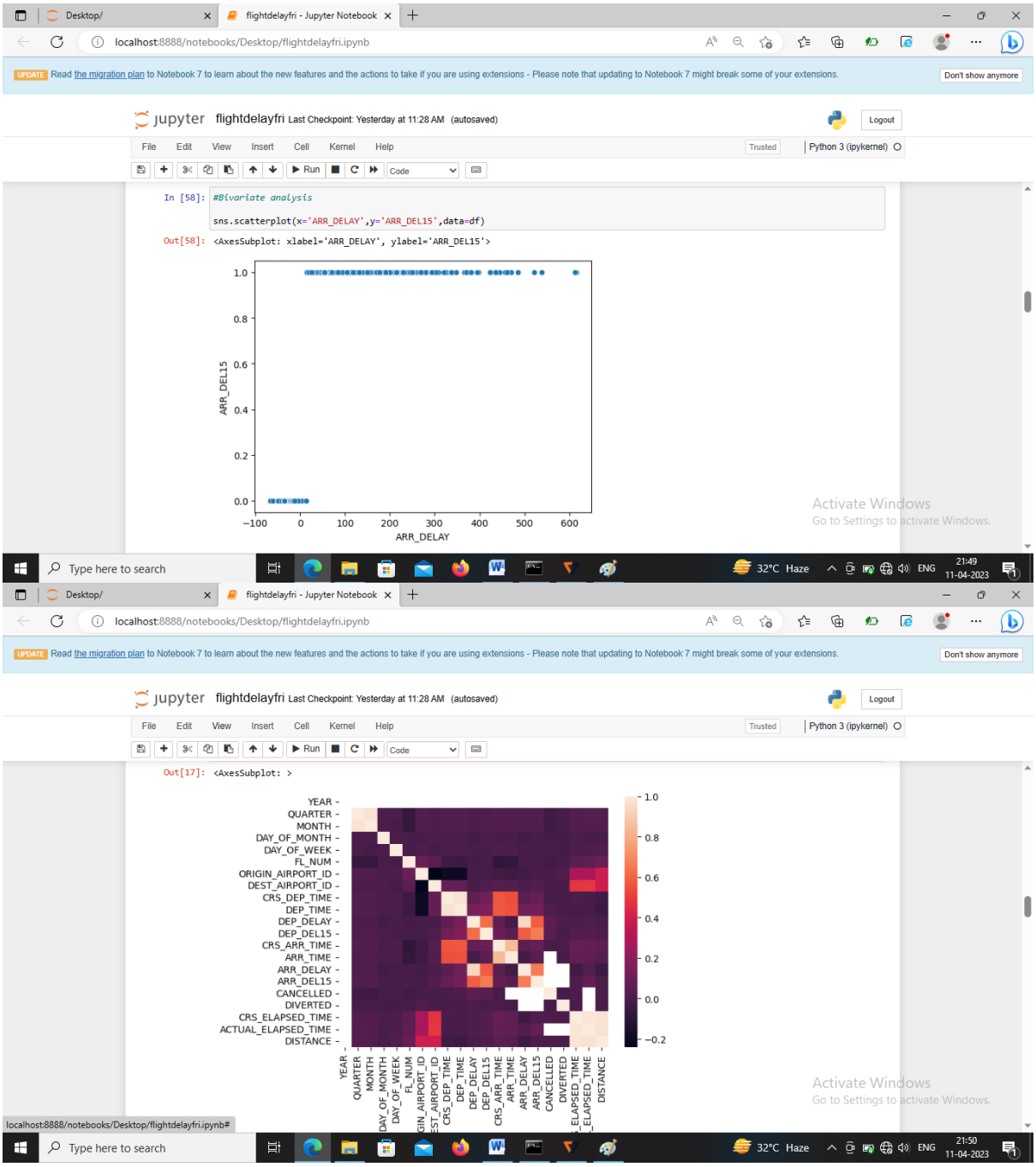
8. APPENDIX

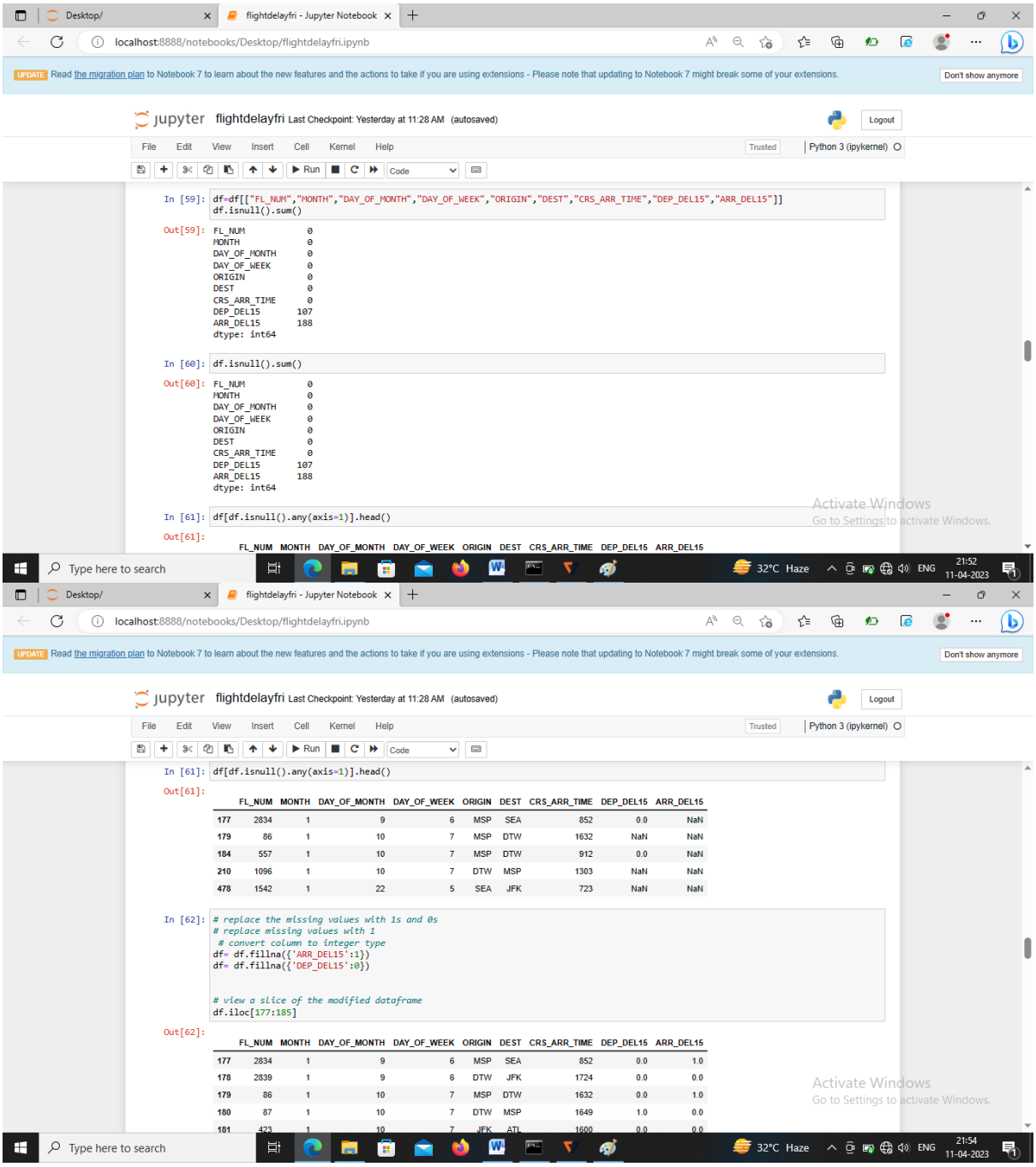

Desktop/
flightdelayfri - Jupyter Notebook
localhost:8888/notebooks/Desktop/flightdelayfri.ipynb
UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions.
Don't show anymore
jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Trusted
Python 3 (ipykernel)
Run
Code
import numpy as np
import pickle
import matplotlib.pyplot as plt
import seaborn as sns
import sklearn
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import GradientBoostingClassifier, RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import RandomizedSearchCV
import imblearn
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix,f1_score
In [54]:
df=pd.read_csv("flightdata (1).csv")
df.head()
Out[54]:
YEAR QUARTER MONTH DAY_OF_MONTH DAY_OF_WEEK UNIQUE_CARRIER TAIL_NUM FL_NUM ORIGIN_AIRPORT_ID ORIGIN ... CRS_ARR_TIME
0 2016 1 1 1 5 DL N836DN 1399 10397 ATL ... 2143
1 2016 1 1 1 5 DL N964DN 1476 11433 DTW ... 1435
2 2016 1 1 1 5 DL N813DN 1597 10397 ATL ... 1215
3 2016 1 1 1 5 DL N587NW 1768 14747 SEA ... 1335
4 2016 1 1 1 5 DL N836DN 1823 14747 SEA ... 607
5 rows × 26 columns
Activate Windows
Go to Settings to activate Windows.
Type here to search
32°C Haze
ENG
21:46
11-04-2023

Desktop/
flightdelayfri - Jupyter Notebook
localhost:8888/notebooks/Desktop/flightdelayfri.ipynb
UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions.
Don't show anymore
jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Trusted
Python 3 (ipykernel)
Run
Code
5 rows × 26 columns
In [55]:
df.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
12 CRS_DEP_TIME 11231 non-null int64
13 DEP_TIME 11124 non-null float64
14 DEP_DELAY 11124 non-null float64
15 DEP_DEL15 11124 non-null float64
16 CRS_ARR_TIME 11231 non-null int64
17 ARR_TIME 11116 non-null float64
18 ARR_DELAY 11043 non-null float64
19 ARR_DEL15 11043 non-null float64
20 CANCELLED 11231 non-null float64
Activate Windows
Go to Settings to activate Windows.
Type here to search
32°C Haze
ENG
21:47
11-04-2023


Desktop/
flightdelayfri - Jupyter Notebook
localhost:8888/notebooks/Desktop/flightdelayfri.ipynb
UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions.
Don't show anymore
jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Trusted
Python 3 (ipykernel)
Run
Code
22 CRS_ELAPSED_TIME 11231 non-null float64
23 ACTUAL_ELAPSED_TIME 11043 non-null float64
24 DISTANCE 11231 non-null float64
25 Unnamed: 25 0 non-null float64
dtypes: float64(12), int64(10), object(4)
memory usage: 2.2+ MB
In [56]:
df=df.drop('Unnamed: 25',axis=1)
df.isnull().sum()
Out[56]:
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
DEP_DELAY 107
DEP_DEL15 107
CRS_ARR_TIME 0
ARR_TIME 115
ARR_DELAY 188
ARR_DEL15 188
CANCELLED 0
Activate Windows
Go to Settings to activate Windows.
Type here to search
32°C Haze
ENG
21:48
11-04-2023

Desktop/
flightdelayfri - Jupyter Notebook
localhost:8888/notebooks/Desktop/flightdelayfri.ipynb
UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions.
Don't show anymore
jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Trusted
Python 3 (ipykernel)
Run
Code
For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751
sns.distplot(df.MONTH)
Out[57]: <AxesSubplot: xlabel='MONTH', ylabel='Density'>
Density
MONTH
0.20
0.15
0.10
0.05
0.00
0
2
4
6
8
10
12
14
Activate Windows
Go to Settings to activate Windows.
Type here to search
32°C Haze
ENG
21:49
11-04-2023


Desktop/
flightdelayfri - Jupyter Notebook
localhost:8888/notebooks/Desktop/flightdelayfri.ipynb
UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions.
Don't show anymore
jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Trusted
Python 3 (ipykernel)
Run
Code
In [58]: #Bivariate analysis
sns.scatterplot(x='ARR_DELAY',y='ARR_DEL15',data=df)
Out[58]: <AxesSubplot: xlabel='ARR_DELAY', ylabel='ARR_DEL15'>
ARR_DEL15
ARR_DELAY
Activate Windows
Go to Settings to activate Windows.
Type here to search
32°C Haze
ENG
21:49
11-04-2023
Desktop/
flightdelayfri - Jupyter Notebook
localhost:8888/notebooks/Desktop/flightdelayfri.ipynb
UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions.
Don't show anymore
jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Trusted
Python 3 (ipykernel)
Run
Code
Out[17]: <AxesSubplot: >
YEAR
QUARTER
MONTH
DAY_OF_MONTH
DAY_OF_WEEK
FL_NUM
ORIGIN_AIRPORT_ID
DEST_AIRPORT_ID
CRS_DEP_TIME
DEP_TIME
DEP_DELAY
DEP_DEL15
CRS_ARR_TIME
ARR_TIME
ARR_DELAY
ARR_DEL15
CANCELLED
DIVERTED
CRS_ELAPSED_TIME
ACTUAL_ELAPSED_TIME
DISTANCE
Activate Windows
Go to Settings to activate Windows.
localhost:8888/notebooks/Desktop/flightdelayfri.ipynb#
Type here to search
32°C Haze
ENG
21:50
11-04-2023



Desktop/ | flightdelayfri - Jupyter Notebook

localhost:8888/notebooks/Desktop/flightdelayfri.ipynb

UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions. Don't show anymore

jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved) Logout

File Edit View Insert Cell Kernel Help Trusted Python 3 (ipykernel)

Run Code

In [59]:
```
df=df[["FL_NUM","MONTH","DAY_OF_MONTH","DAY_OF_WEEK","ORIGIN","DEST","CRS_ARR_TIME","DEP_DEL15","ARR_DEL15"]]
df.isnull().sum()
```

Out[59]:
```
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
```

In [60]:
```
df.isnull().sum()
```

Out[60]:
```
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
```

In [61]:
```
df[df.isnull().any(axis=1)].head()
```

Out[61]:

	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	NaN
179	86	1	10	7	MSP	DTW	1632	NaN	NaN
184	557	1	10	7	MSP	DTW	912	0.0	NaN
210	1096	1	10	7	DTW	MSP	1303	NaN	NaN
478	1542	1	22	5	SEA	JFK	723	NaN	NaN

Activate Windows
Go to Settings to activate Windows.

Type here to search 32°C Haze ENG 21:52 11-04-2023

Desktop/ | flightdelayfri - Jupyter Notebook

localhost:8888/notebooks/Desktop/flightdelayfri.ipynb

UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions. Don't show anymore

jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved) Logout

File Edit View Insert Cell Kernel Help Trusted Python 3 (ipykernel)

Run Code

In [61]:
```
df[df.isnull().any(axis=1)].head()
```

Out[61]:

	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	NaN
179	86	1	10	7	MSP	DTW	1632	NaN	NaN
184	557	1	10	7	MSP	DTW	912	0.0	NaN
210	1096	1	10	7	DTW	MSP	1303	NaN	NaN
478	1542	1	22	5	SEA	JFK	723	NaN	NaN

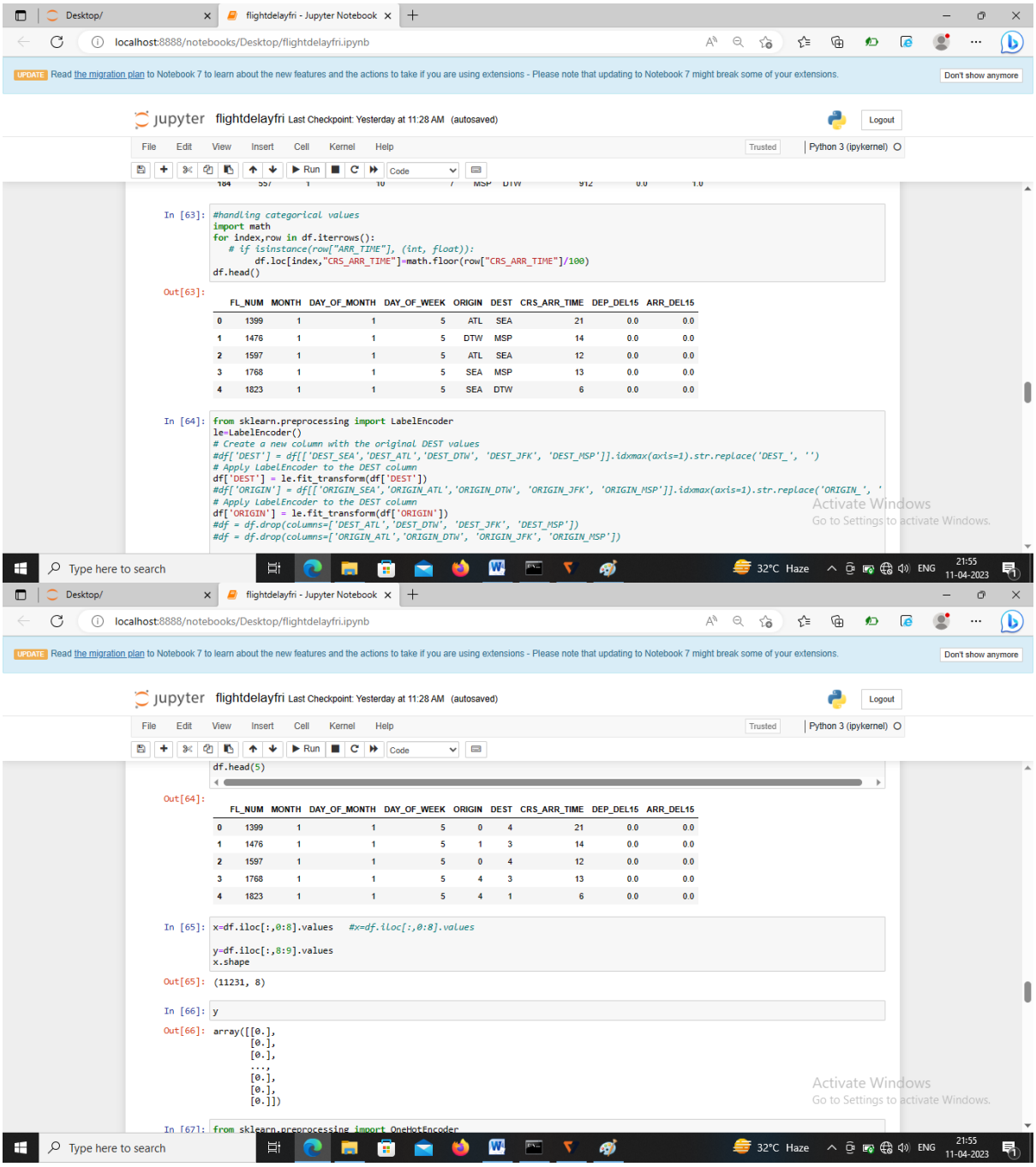
In [62]:
```
# replace the missing values with 1s and 0s
# replace missing values with 1
 # convert column to integer type
df= df.fillna({'ARR_DEL15':1})
df= df.fillna({'DEP_DEL15':0})


# view a slice of the modified dataframe
df.iloc[177:185]
```

Out[62]:

	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	1649	1.0	0.0
181	423	1	10	7	JFK	ATL	1600	0.0	0.0

Activate Windows
Go to Settings to activate Windows.

Type here to search 32°C Haze ENG 21:54 11-04-2023



In [63]:
```
#handling categorical values
import math
for index,row in df.iterrows():
    # if isinstance(row["ARR_TIME"], (int, float)):
        df.loc[index,"CRS_ARR_TIME"]=math.floor(row["CRS_ARR_TIME"]/100)
df.head()
```

Out[63]:

	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 [64]:
```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
# Create a new column with the original DEST values
#df['DEST'] = df[['DEST_SEA','DEST_ATL','DEST_DTW', 'DEST_JFK', 'DEST_MSP']].idxmax(axis=1).str.replace('DEST_', '')
# Apply LabelEncoder to the DEST column
df['DEST'] = le.fit_transform(df['DEST'])
#df['ORIGIN'] = df[['ORIGIN_SEA','ORIGIN_ATL','ORIGIN_DTW', 'ORIGIN_JFK', 'ORIGIN_MSP']].idxmax(axis=1).str.replace('ORIGIN_', '
# Apply LabelEncoder to the DEST column
df['ORIGIN'] = le.fit_transform(df['ORIGIN'])
#df = df.drop(columns=['DEST_ATL','DEST_DTW', 'DEST_JFK', 'DEST_MSP'])
#df = df.drop(columns=['ORIGIN_ATL','ORIGIN_DTW', 'ORIGIN_JFK', 'ORIGIN_MSP'])
```


In [64]:
```
df.head(5)
```

Out[64]:

	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

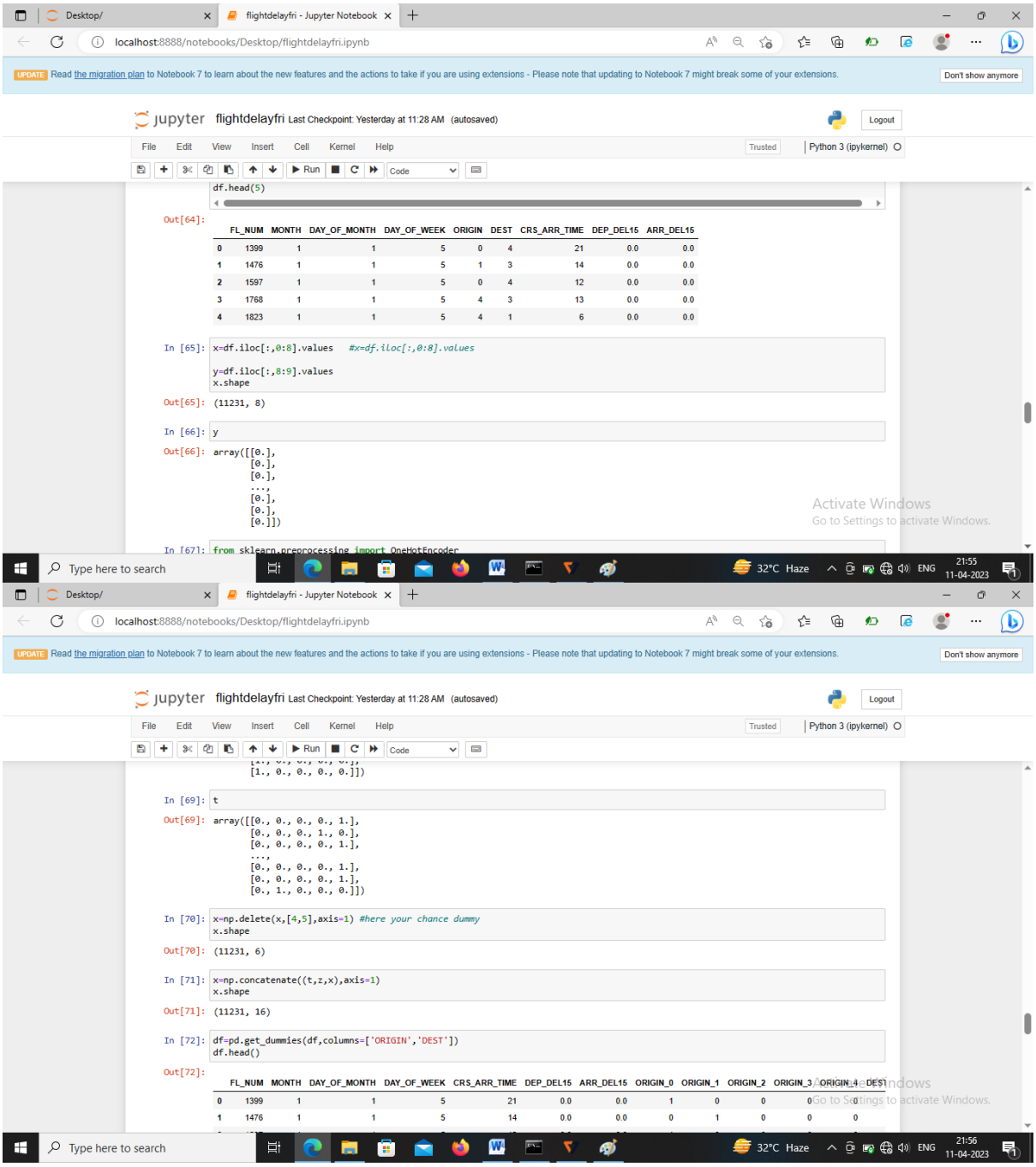
In [65]:
```
x=df.iloc[:,0:8].values   #x=df.iloc[:,0:8].values

y=df.iloc[:,8:9].values
x.shape
```

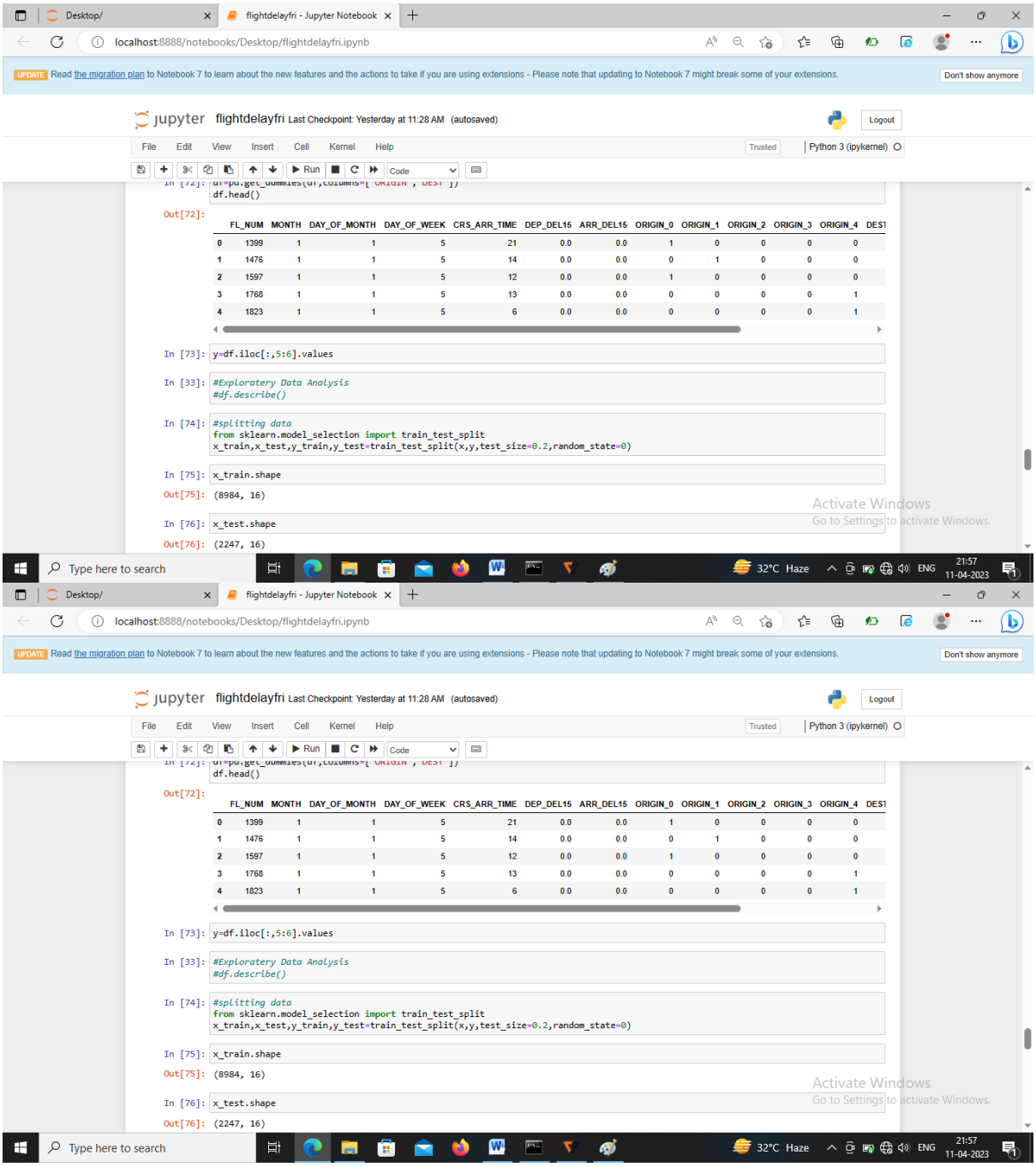
Out[65]: (11231, 8)

In [66]:
```
y
```

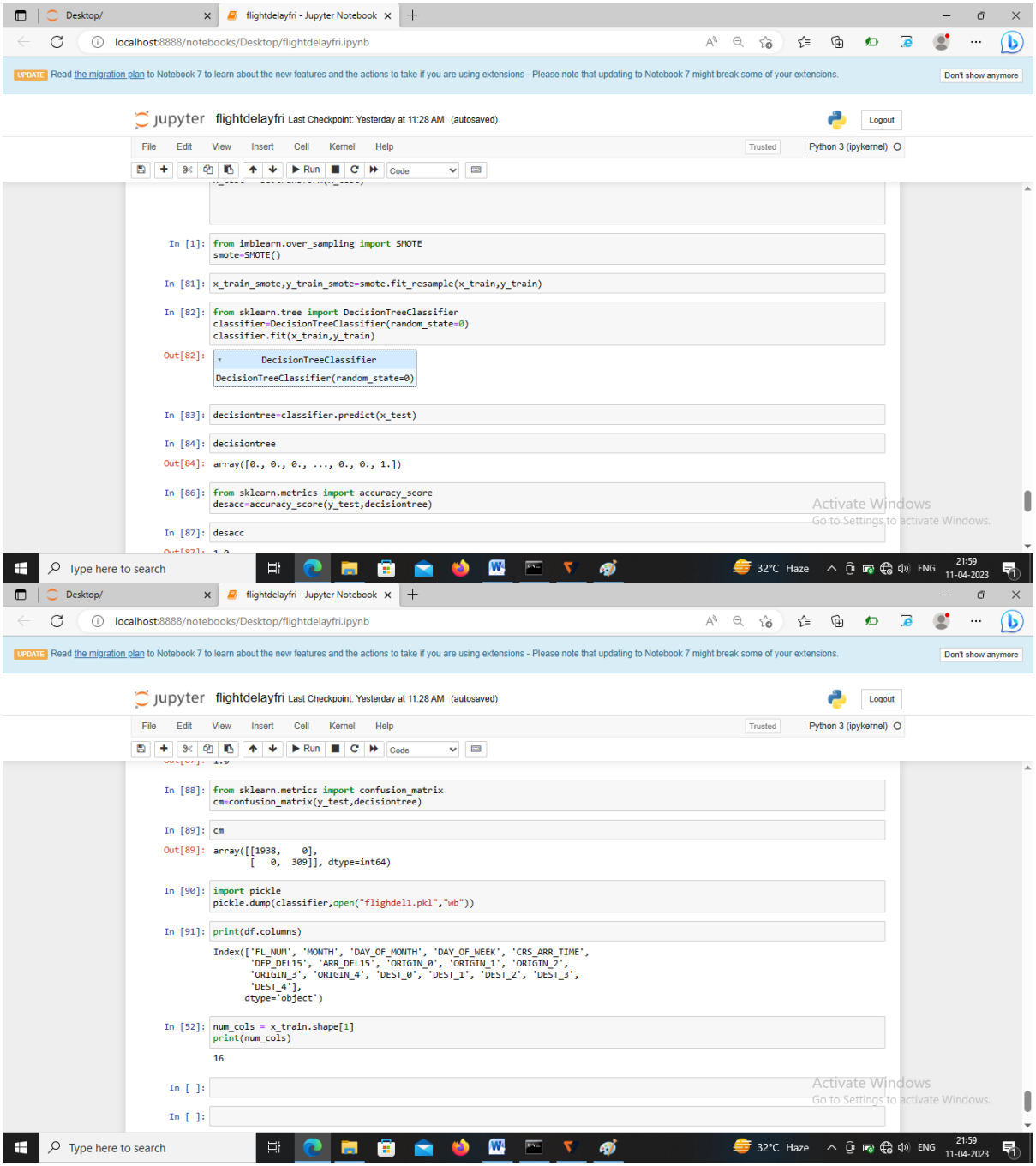
Out[66]:
```
array([[0.],
       [0.],
       [0.],
       ...,
       [0.],
       [0.],
       [0.]])
```

In [67]:
```
from sklearn.preprocessing import OneHotEncoder
```


Desktop/
flightdelayfri - Jupyter Notebook
localhost:8888/notebooks/Desktop/flightdelayfri.ipynb
UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions.
Don't show anymore
jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Trusted
Python 3 (ipykernel)
Run
Code
df.head(5)
Out[64]:
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 [65]:
x=df.iloc[:,0:8].values #x=df.iloc[:,0:8].values
y=df.iloc[:,8:9].values
x.shape
Out[65]: (11231, 8)
In [66]: y
Out[66]: array([[0.],
[0.],
[0.],
...,
[0.],
[0.],
[0.]])
In [67]: from sklearn.preprocessing import OneHotEncoder
Activate Windows
Go to Settings to activate Windows.
Type here to search
32°C Haze
ENG
21:55
11-04-2023
Desktop/
flightdelayfri - Jupyter Notebook
localhost:8888/notebooks/Desktop/flightdelayfri.ipynb
UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions.
Don't show anymore
jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Trusted
Python 3 (ipykernel)
Run
Code
[1., 0., 0., 0.]])
In [69]: t
Out[69]: array([[0., 0., 0., 1.],
[0., 0., 1., 0.],
[0., 0., 0., 1.],
...,
[0., 0., 0., 1.],
[0., 0., 0., 1.],
[0., 1., 0., 0.]])
In [70]: x=np.delete(x,[4,5],axis=1) #here your chance dummy
x.shape
Out[70]: (11231, 6)
In [71]: x=np.concatenate((t,z,x),axis=1)
x.shape
Out[71]: (11231, 16)
In [72]: df=pd.get_dummies(df,columns=['ORIGIN','DEST'])
df.head()
Out[72]:
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 DEST
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
Activate Windows
Go to Settings to activate Windows.
Type here to search
32°C Haze
ENG
21:56
11-04-2023



Desktop/
flightdelayfri - Jupyter Notebook
localhost:8888/notebooks/Desktop/flightdelayfri.ipynb
UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions.
Don't show anymore
jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Trusted
Python 3 (ipykernel)
Run
Code
df.head()
Out[72]:
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 DEST
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 [73]: y=df.iloc[:,5:6].values
In [33]: #Exploratery Data Analysis
#df.describe()
In [74]: #splitting data
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)
In [75]: x_train.shape
Out[75]: (8984, 16)
In [76]: x_test.shape
Out[76]: (2247, 16)
Activate Windows
Go to Settings to activate Windows.
Type here to search
32°C Haze
ENG
21:57
11-04-2023


Desktop/
flightdelayfri - Jupyter Notebook
localhost:8888/notebooks/Desktop/flightdelayfri.ipynb
UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions.
Don't show anymore
jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Trusted
Python 3 (ipykernel)
Run
Code
In [1]: from imblearn.over_sampling import SMOTE
smote=SMOTE()
In [81]: x_train_smote,y_train_smote=smote.fit_resample(x_train,y_train)
In [82]: from sklearn.tree import DecisionTreeClassifier
classifier=DecisionTreeClassifier(random_state=0)
classifier.fit(x_train,y_train)
Out[82]: DecisionTreeClassifier
DecisionTreeClassifier(random_state=0)
In [83]: decisiontree=classifier.predict(x_test)
In [84]: decisiontree
Out[84]: array([0., 0., 0., ..., 0., 0., 1.])
In [86]: from sklearn.metrics import accuracy_score
desacc=accuracy_score(y_test,decisiontree)
In [87]: desacc
Activate Windows
Go to Settings to activate Windows.
Type here to search
32°C Haze
ENG
21:59
11-04-2023
Desktop/
flightdelayfri - Jupyter Notebook
localhost:8888/notebooks/Desktop/flightdelayfri.ipynb
UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions.
Don't show anymore
jupyter flightdelayfri Last Checkpoint: Yesterday at 11:28 AM (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Trusted
Python 3 (ipykernel)
Run
Code
In [88]: from sklearn.metrics import confusion_matrix
cm=confusion_matrix(y_test,decisiontree)
In [89]: cm
Out[89]: array([[1938, 0],
[0, 309]], dtype=int64)
In [90]: import pickle
pickle.dump(classifier,open("flighdel1.pkl","wb"))
In [91]: print(df.columns)
Index(['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', 'DEST_0', 'DEST_1', 'DEST_2', 'DEST_3',
'DEST_4'],
dtype='object')
In [52]: num_cols = x_train.shape[1]
print(num_cols)
16
In [ ]:
In [ ]:
Activate Windows
Go to Settings to activate Windows.
Type here to search
32°C Haze
ENG
21:59
11-04-2023