FLIGHT DELAY PREDICTION FOR AVIATION INDUSTRY USING MACHINE LEARNING

Submitted in partial fulfillment of requirement for the award of the Degree

Bachelor of Computer Science

In the faculty of Computer Science of Bharathiar University, Coimbatore

Submitted by

TEAM ID: NM2023TMID20800

T.POOVIZHI

N.DHARSHINI

M.P.NIVITHA SANKARI

S.SURUTHI

Under the guidance of

Mrs.V.PUNITHAVATHI MCA., M.Phil.,DOA.,

Guest Lecturer, Department of Computer Science

Department of Computer Science

L.R.G GOVERNMENT ARTS COLLEGE FOR WOMEN

(Affiliated To Bharathiar University)

TIRUPUR-4

APRIL-2023

LRG GOVERNMENT ARTS COLLEGE

NAAN MUDHALVAN PROJECT

(AFFILIATED TO BHARATHIAR UNIVERSITY)

TIRUPUR-641604

TITLE: Flight Delay Prediction For Aviation Industry Using Machine Learning

This is to certify that this is a bonafide record of work done by the above students of III B.Sc (CS) Degree **NAAN MUDHALVAN PROJECT** during the year________.

Submitted for the Naan Mudhalvan project work held

on ___________20

CLASS TUTOR **HEAD OF DEPARTMENT**

TABLE OF CONTENTS

1. INTRODUCTION

2. OVERVIEW

In the present world, the major components of any transportation system include passenger airline, cargo airline, and air traffic control system. With the passage of time, nations around the world have tried to evolve numerous techniques of improving the airline transportation system. This has brought drastic change in the airline operations. Flight delays occasionally cause inconvenience to the modern passengers. Every year approximately 20% of airline flights are cancelled or delayed, costing passengers more than 20 billion dollars in money and their time.

Air ways is one of the crucial modes of transportation in our modern words, and with the increasing number of air vehicles its leading to simultaneous increase in the air traffic. So its important to maintain a flexible system. The Corporate travels and tourism are the two major contributors to flight transportation which is expected to be doubled by 2030, As a result the air traffic is also expected to increase in the same multiple .If we consider the US , where the airlines are handled by federal aviation administration ,they handle about 16,405,000 flights every year and handling the air traffic became a crucial part for safe movement. The airtraffic authorities continuously try to disparage the delay in departure and arrival of the flights.

Despite their best efforts , the outcome is undesirable as sometimes the delays are hours causing chaos for the days schedule. Some of the important parameters that cause delay include weather, carrier, maintenance, security. These delays causes congestion in the air traffic. One of the solution is to minimize the air traffic congestion is to construct new airports, but the complexity increases .we could improvise the existing airports but considering the limited availability of land resources, the ultimate logical solution would be predicting the delay of the flights. Delay basically represents the period by which the aircraft is late or has been cancelled. The delay results in complexity in air traffic and dissatisfaction of customers and increase in costs for the company .If a flight is delayed by 10 minutes the flight is considered delay. In the USA from march 2021 to march 2022 itself the delay in flights is about 20.29 % of which air carrier delay is 7.04%, delay due weather is 0.72% , delay due to navigation system is 4.26 and delay due to aircraft arriving late and security delay is 6.12% .we cannot exactly predict the reason for the delay but after the arrival we can predict the delay time for reaching the destination.

3. RESEARCH MOTIVATION

Average aircraft delay is regularly referred to as an indication of airport capacity. Flight delay is a prevailing problem in this world. It’s very tough to explain the reason for a delay. A few factors responsible for the flight delays like runway construction to excessive traffic are rare, but bad weather seems to be a common cause. Some flights are delayed because of the reactionary delays, due to the late arrival of the previous flight. It hurts airports, airlines, and affects a company’s marketing strategies as companies rely on customer loyalty to support their frequent flying programs.

4. PROBLEM STATEMENT

My case study was about LaGuardia Airport in New York, Logan International Airport in Boston, San Francisco International Airport in San Francisco, and O’Hare International Airport in Chicago, which are four major airports in the United States of America. But we focused the idea and research on LaGuardia International Airport. Compared with the data produced by all airports in USA, the data which we gathered was very limited, but it gave us a great direction on how weather plays a part in flight delays. In this project, the goal is to use exploratory analysis and to build machine learning models to predict airline departure and arrival delays.

5. PURPOSE

Predicting flight delays can improve airline operations and passenger satisfaction, which will result in a positive impact on the economy. The main goal is to compare the performance of machine learning classification algorithms when predicting flight delays. The proposed model made use of several algorithms, and their predictions were evaluated using a number of measures. Analysis of flight delay, therefore, has become a popular research area. Various researchers used different techniques of machine learning and data mining to conduct the investigation. They were interested in different aspects such as airport facility location, weather condition, and airport capacity. Using machine learning allows researchers to handle large quantities of flight data for storing and processing.

2. PROBLEM SELECTION

Predicting thyroid disease using machine learning is an interesting and important project that can have significant implications in the field of healthcare. Here are some steps you can follow to select a suitable project:

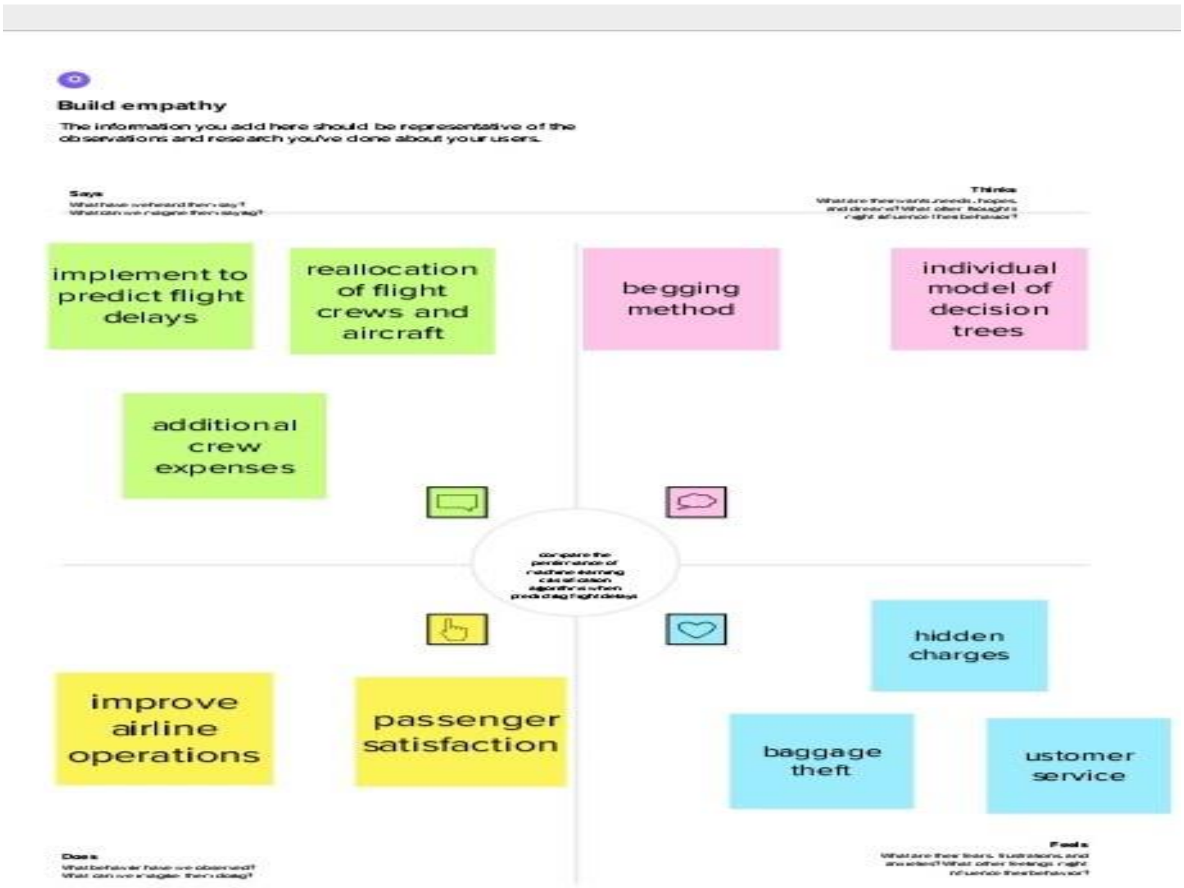
1. Research existing work: Before starting any project, it's important to research existing work in the field. Look for similar projects that have been done before and see what approaches were used. This will help you understand the current state-of-the-art and identify any gaps in the literature that your project can address.
2. Define the problem: Once you have a good understanding of the current research, define the problem you want to solve. For example, you could focus on predicting the risk of developing thyroid disease in a particular population or predicting the progression of the disease in patients who have already been diagnosed.
3. Gather data: In order to train a machine learning model, you will need a dataset. Look for publicly available datasets or consider collecting your own data. Make sure that the data you use is of high quality and is representative of the problem you are trying to solve.
4. Train and evaluate the model: Once you have chosen an algorithm, train the model on your dataset and evaluate its performance using appropriate metrics such as accuracy, precision, recall, and F1 score. If the model's performance is not satisfactory, consider tweaking the algorithm or using a different one.
5. Deploy the model: Once you have a model that performs well, deploy it in a real-world setting. This could involve integrating it into an electronic health.

3. PROBLEM DEFENITION AND DESIGN THINKING

3.1 EMPATHY MAP

Empathy maps are an efficient tool used by designers to not only understand user 7ehaviour, but also visually communicate those findings to colleagues, uniting the team under one shared understanding of the user. Originally invented by Dave Gray at Xplane, the empathy map was made in an attempt to limit miscommunication and misunderstanding about target audiences, including customers and users.

An empathy map is a square divided into four quadrants with the user or client in the middle. Each of the four quadrants comprises a category that helps us delve into the mind of the user. The four empathy map quadrants look at what the user **says**, **thinks**, **feels**, and **does**.



3.2 IDEATION

Ideation is often the most exciting stage in a Design Thinking project, because during Ideation, the aim is to generate a large quantity of ideas that the team can then filter and cut down into the best, most practical or most innovative ones in order to inspire new and better design solutions and products.

1. Collect Data: Collect data from various sources, including medical records, patient information, and clinical studies. Ensure that the data is in a structured format and that it contains all the necessary features for building a model.
2. Preprocess the Data: Once you have the data, preprocess it by cleaning, normalizing, and transforming it into a format that can be used by a machine learning model.
3. Feature Selection: Choose the relevant features that will be used to predict the thyroid disease. These features can include patient demographics, lab test results, and medical history.
4. Choose a Model: Choose an appropriate machine learning algorithm that will be used for prediction. There are various algorithms available, including decision trees, random forests, and support vector machines.
5. Train the Model: Use the preprocessed data and the chosen algorithm to train the model. Split the data into training and testing sets to evaluate the model's accuracy.
6. Evaluate the Model: Evaluate the model's performance using different metrics such as accuracy, precision, recall, and F1-score.
7. Deploy the Model: Finally, deploy the model into a production environment where it can be used to predict the thyroid disease of new patients.

4. REQUIREMENT ANALYSIS

1. **Data collection:**

   Gather a large dataset of patient information, including demographics, medical history, symptoms, lab test results, and imaging data (if available). The dataset should include both positive and negative cases of thyroid disease to ensure the model is trained on a balanced dataset.

2. **Data pre-processing:**

   Clean the data, remove any missing or irrelevant information, and encode categorical variables.

3. **Feature select:**

   Identify the most important features that contribute to the prediction of thyroid disease.

4. **Algorithm select:**

   Select the appropriate machine learning algorithm for the task, such as logistic regression, decision trees, random forests, or support vector machines.

5. **Model training:**

   Split the dataset into training and validation sets, train the model on the training set, and evaluate the performance on the validation set. Adjust the model hyperparameters to optimize performance.

6. **Model evaluation:**

   Evaluate the model's performance using metrics such as accuracy, precision, recall, and F1-score. Use cross-validation to ensure the model's generalizability to new data.

7. **Model deployment:**

   Deploy the trained model in a user-friendly interface, such as a web application or mobile app, to enable healthcare professionals to make accurate predictions of thyroid disease in their patients.

8. **Maintenance and updates:**

   Regularly update the model with new data and retrain as necessary to ensure the accuracy of the predictions.

5. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- Predicting flight delays can improve airline operations and passenger satisfaction, which will result in a positive impact on the economy
- Prevents financial difficulties and customer dissatisfaction to airline companies.
- Supervised machine learning models were implemented to predict the flight delays.
- Passengers can get to know the exact time of air boarding, air time and further more after the prediction of delayed flights.

DISADVANTAGE

- Over fitting problem occurs while using limited dataset.
- Noise of input data can decrease the accuracy of large test set.
- Selection of class weights used for iterations.

6. INPUT DESIGN

The input design is the process of entering data to the system. The input design goal is to enter to the computer as accurate as possible. Here inputs are designed effectively so that errors made by the operation are minimized.

The inputs to the system have been designed in such a way that manual forms and the inputs are coordinated where the data elements are common to the sources document and to the input. The input is acceptable and understandable by the users who are using it.

Input design is the process of converting user-originated inputs to a computer-based format input data are collected and organized into group of similar data. Once identified, appropriate input media are selected for processing.

Input design means the physical and performance requirements of a device that are used as a basis for device design. Input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc

6.1 INPUT DESIGN DESCRIPTION

HOME

This page collects the information of the passenger's travel such as

- Month of travel
- Travel day of month
- Scheduled Departure
- Delay in Departure
- Scheduled arrival time
- Was the flight diverted?
- Was the flight cancelled.
- Delay due to air system

7. OUTPUT DESIGN

A design output is a drawing or specification or manufacturing instruction. Design outputs describe all the components, parts, and pieces that go into your device. Design outputs describe all assemblies and subassemblies of product.

Output design is the process of converting data into hard copy that is understood by all. The various outputs have been in such a way that they represent the same format that the office and management used to.

Computer output is the most important and direct source of information to the user. Efficient, intelligible output design should improve the systems relationships with the user and help in decision making. A major form of out is the hardcopy from the printer. Output requirements are designed during system analysis.

7.1 OUTPUT DESIGN DESCRIPTION

OUTPUT FORM

This page inspects the details collected from the passenger and displace the page “Your Flight is Delayed” if the flight is delayed or else “Your Flight is Not Delay” if the flight is not delayed.

8. PROJECT PLANNING PHASE

1. **Define the problem:**

   Define the problem you want to solve. For example, you may want to build a model that can accurately predict the risk of thyroid disease in patients based on certain clinical and demographic features.

2. **Gather data**:

   Identify the data sources that can be used to build the machine learning model. This may involve gathering data from electronic health records, medical imaging, or patient surveys.

3. **Choose a machine learning algorithm:**

   There are several machine learning algorithms that can be used to build a predictive model. You will need to choose an algorithm that is appropriate for your data and problem.

4. **Train the model:**

   Once you have chosen an algorithm, you will need to train the model on the data. This involves dividing the data into training and validation sets, and using the training set to optimize the model's parameters.

5. **Evaluate the model**:

   After training the model, you will need to evaluate its performance on the validation set. This will give you an idea of how well the model will perform on new data.

6. **Deploy the model:**

   Once you have a model that performs well on the validation set, you can deploy it in a clinical setting. This may involve integrating it into an electronic health record system, or creating a standalone application.

9. PROJECT DEVELOPMENT PHASE

1. **Data collection and preparation:**

   This involves gathering relevant data from various sources and preparing it for use in machine learning algorithms. In the case of thyroid disease prediction, this might include medical records, lab results, and patient demographics.

2. **Feature selection and engineering:**

   Once the data has been collected, the next step is to select the most relevant features (i.e., variables or attributes) that are likely to be predictive of thyroid disease. This might involve domain expertise from medical professionals, as well as statistical techniques such as correlation analysis or principal component analysis.

3. **Model selection and training:**

   With the features selected, the next step is to choose an appropriate machine learning algorithm to use for prediction. This might include traditional models such as logistic regression or more advanced methods such as deep learning

4. **Model evaluation and tuning:**

   Once the model has been trained, it is evaluated using various metrics such as accuracy, precision, and recall

5. **Deployment and monitoring:**

   Finally, the trained and validated model can be deployed for use in clinical settings. Ongoing monitoring and evaluation of the model's performance are critical to ensure that it remains accurate and effective over time.

10.CONCLUSION

In this project, we use flight data, weather, and demand data to predict flight departure delay. Our result shows that the Random Forest method yields the best performance compared to the SVM model. Somehow the SVM model is very time consuming and does not necessarily produce better results. In the end, our model correctly predicts 91% of the non-delayed flights. However, the delayed flights are only correctly predicted 41% of time. As a result, there can be additional features related to the causes of flight delay that are not yet discovered using our existing data sources.

In the second part of the project, we can see that it is possible to predict flight delay patterns from just the volume of concurrently published tweets, and their sentiment and objectivity. This is not unreasonable; people tend to post about airport delays on Twitter; it stands to reason that these posts would become more frequent, and more profoundly emotional, as the delays get worse. Without more data, we cannot make a robust model and find out the role of related factors and chance on these results. However, as a proof of concept, there is potential for these results. It may be possible to routinely use tweets to ascertain an understanding of concurrent airline delays and traffic patterns, which could be useful in a variety of circumstances.

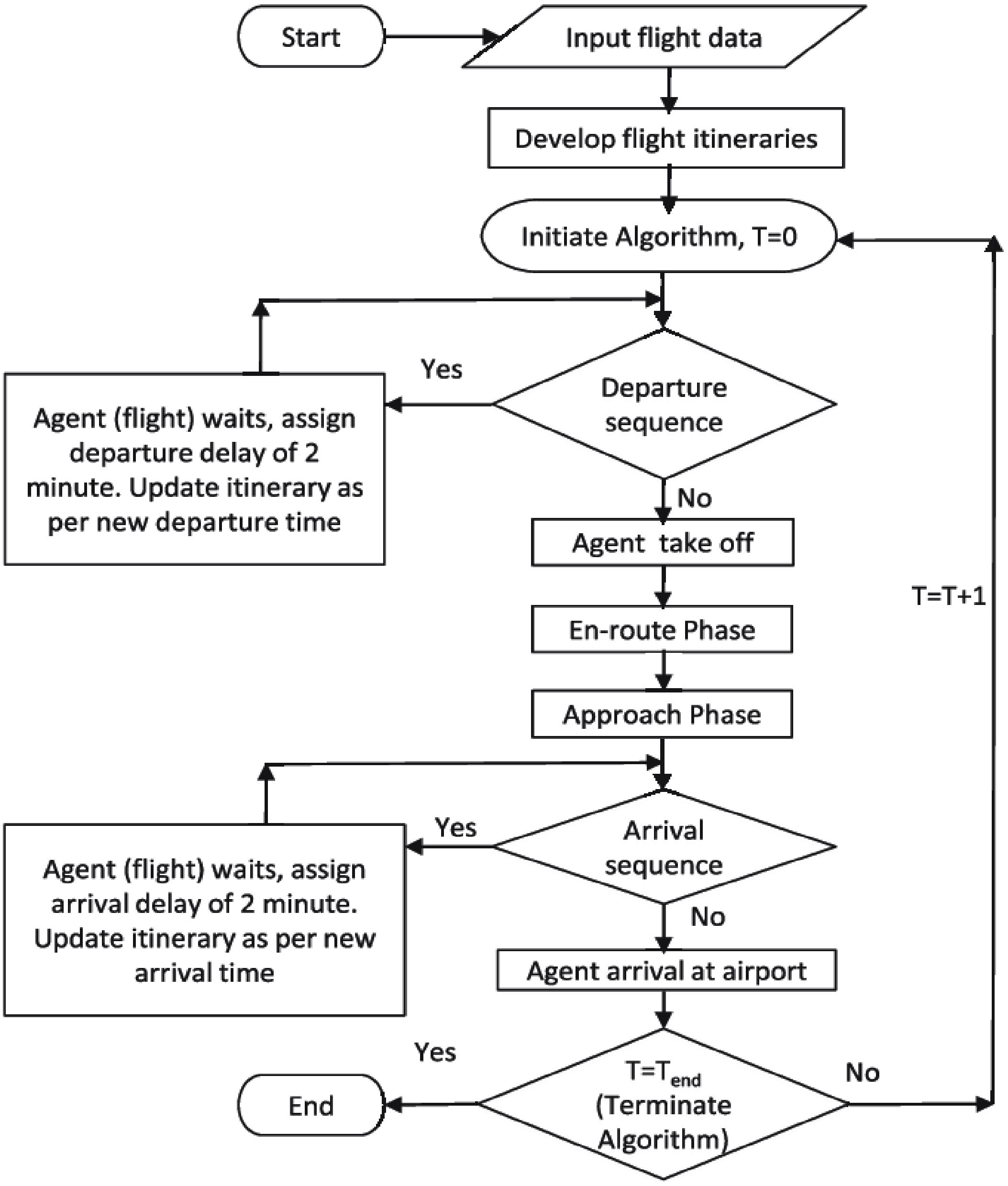
11.FUTURE SCOPE

This project is based on data analysis from year 2008. A large dataset is available from 1987-2008 but handling a bigger dataset requires a great amount of pre-processing and cleaning of the data. Therefore, the future work of this project includes incorporating a larger dataset. There are many different ways to pre-process a larger dataset like running a Spark cluster over a server or using a cloud-based services like AWS and Azure to process the data. With the new advancement in the field of deep learning, we can use Neural Networks algorithm on the flight and weather data. Neural Network works on the pattern matching methodology. It is divided into three basic parts for data modelling that includes feed forward networks, feedback networks, and self-organization network. Feed-forward and feedback networks are generally used in the areas of prediction, pattern recognition, associative memory, and optimization calculation, whereas self-organization networks are generally used in cluster analysis.

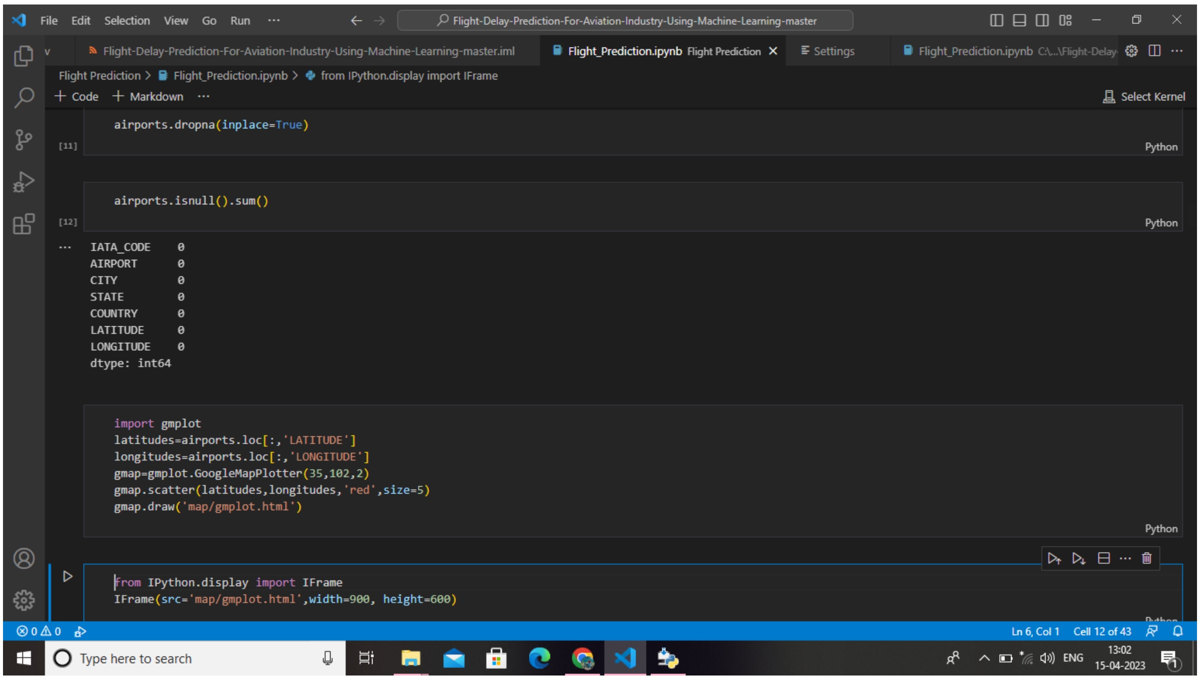
Neural Network offers distributed computer architecture with important learning abilities to represent nonlinear relationships. Also, the scope of this project is very much confined to flight and weather data of United States, but we can include more countries like China, India, and Russia. Expanding the scope of this project, we can also add the flight data from international flights and not just restrict our self to the domestic flights.

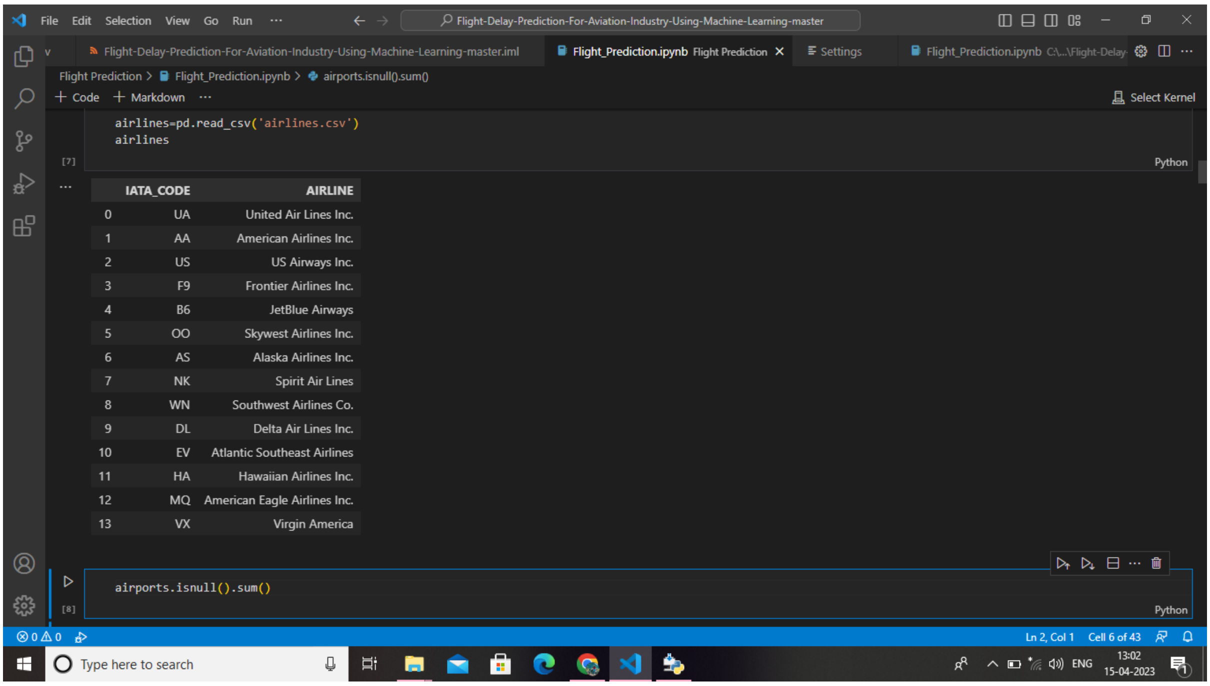
12.APPENDICES

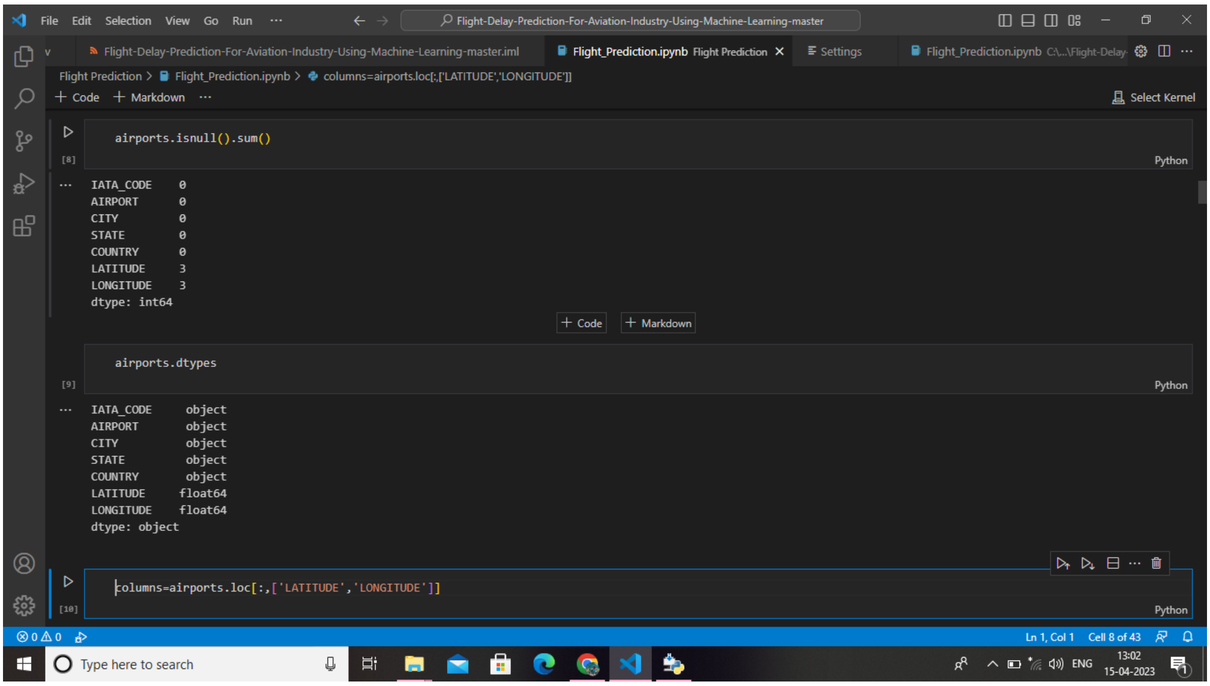
A. FLOW GRAPH

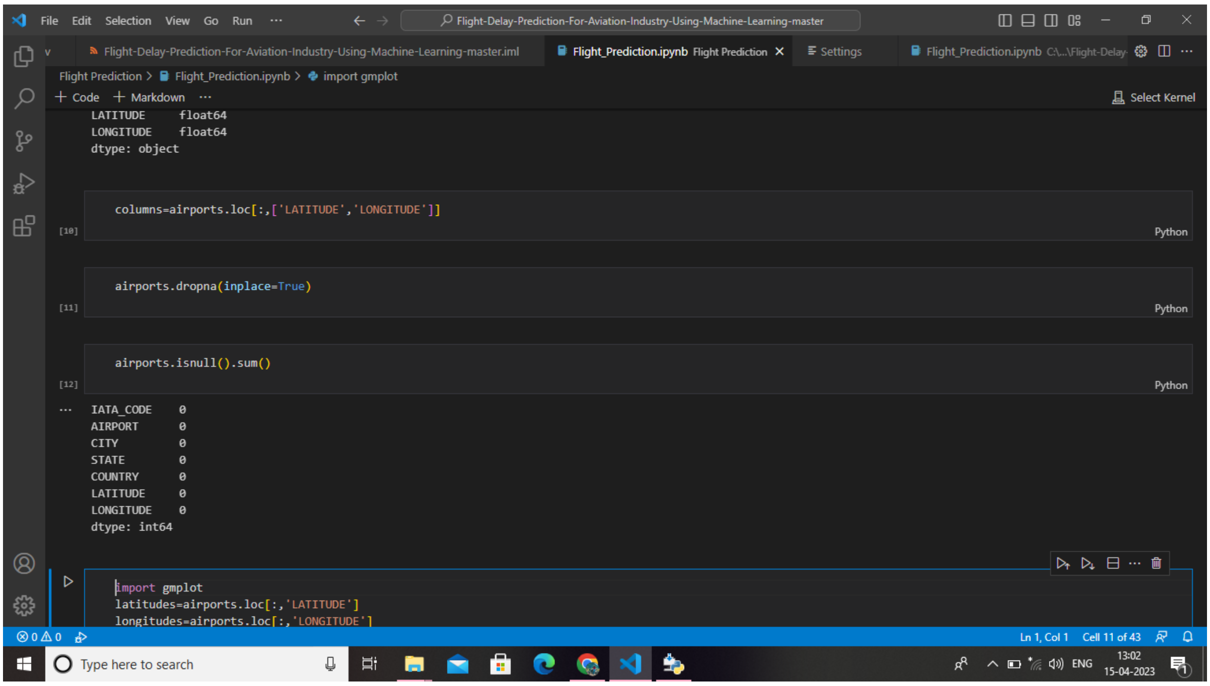


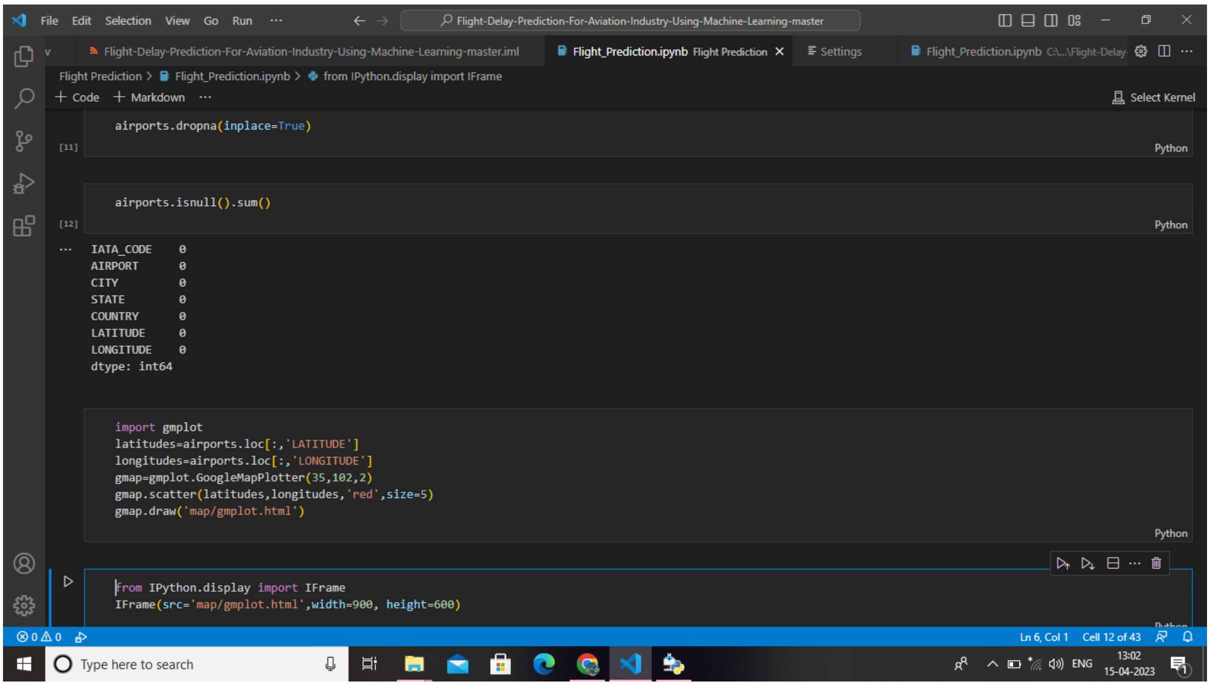
B. SAMPLE CODE

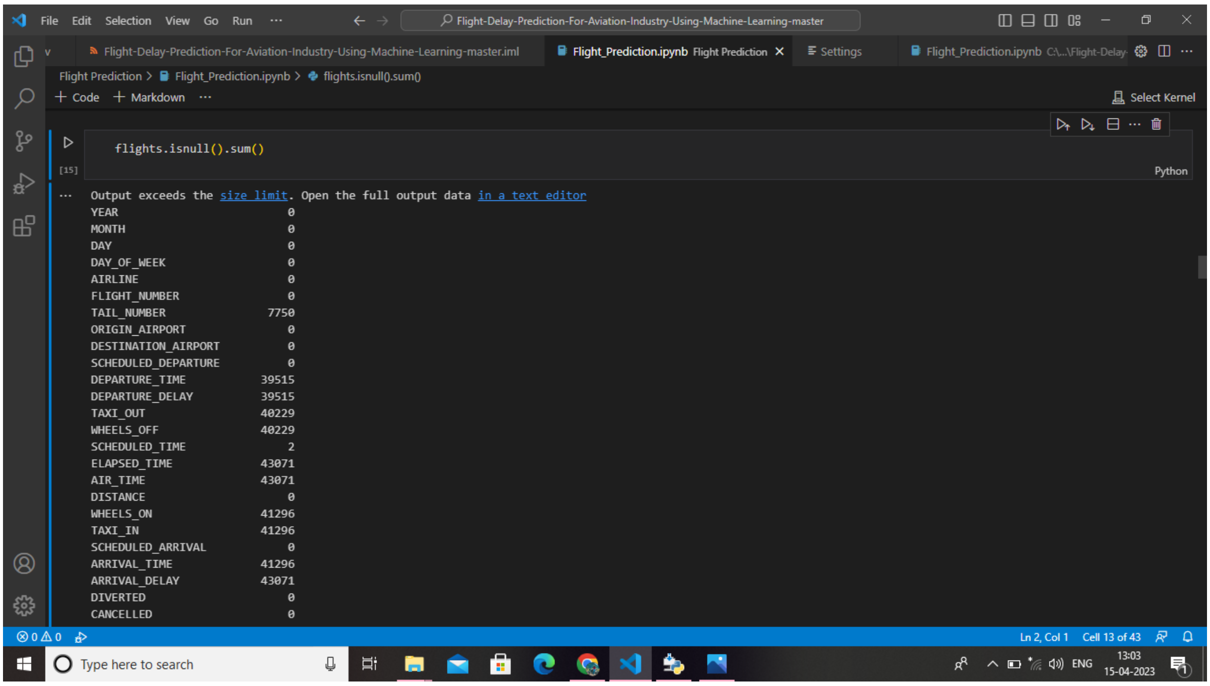

File Edit Selection View Go Run ···
Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master
Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master.iml
Flight_Prediction.ipynb Flight Prediction
Settings
Flight Prediction > Flight_Prediction.ipynb > from IPython.display import IFrame
+ Code + Markdown ···
Select Kernel
airports.dropna(inplace=True)
[11]
Python
airports.isnull().sum()
[12]
Python
IATA_CODE 0
AIRPORT 0
CITY 0
STATE 0
COUNTRY 0
LATITUDE 0
LONGITUDE 0
dtype: int64
import gmplot
latitudes=airports.loc[:,'LATITUDE']
longitudes=airports.loc[:,'LONGITUDE']
gmap=gmplot.GoogleMapPlotter(35,102,2)
gmap.scatter(latitudes,longitudes,'red',size=5)
gmap.draw('map/gmplot.html')
Python
from IPython.display import IFrame
IFrame(src='map/gmplot.html',width=900, height=600)
Ln 6, Col 1 Cell 12 of 43
Type here to search
ENG 13:02 15-04-2023

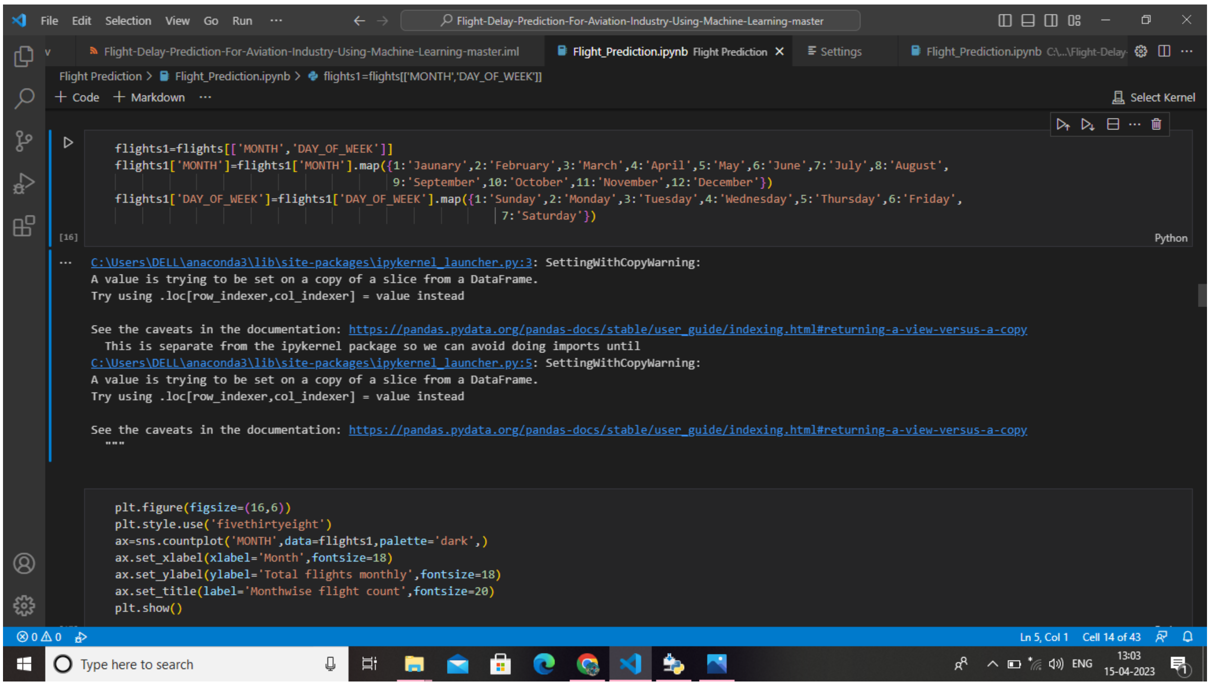

File Edit Selection View Go Run ···
Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master
Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master.iml
Flight_Prediction.ipynb Flight Prediction
Settings
Flight Prediction > Flight_Prediction.ipynb > airports.isnull().sum()
+ Code + Markdown ···
Select Kernel
airlines=pd.read_csv('airlines.csv')
airlines
[7]
Python
IATA_CODE AIRLINE
0 UA United Air Lines Inc.
1 AA American Airlines Inc.
2 US US Airways Inc.
3 F9 Frontier Airlines Inc.
4 B6 JetBlue Airways
5 OO Skywest Airlines Inc.
6 AS Alaska Airlines Inc.
7 NK Spirit Air Lines
8 WN Southwest Airlines Co.
9 DL Delta Air Lines Inc.
10 EV Atlantic Southeast Airlines
11 HA Hawaiian Airlines Inc.
12 MQ American Eagle Airlines Inc.
13 VX Virgin America
airports.isnull().sum()
[8]
Python
Ln 2, Col 1 Cell 6 of 43
Type here to search
ENG 13:02 15-04-2023

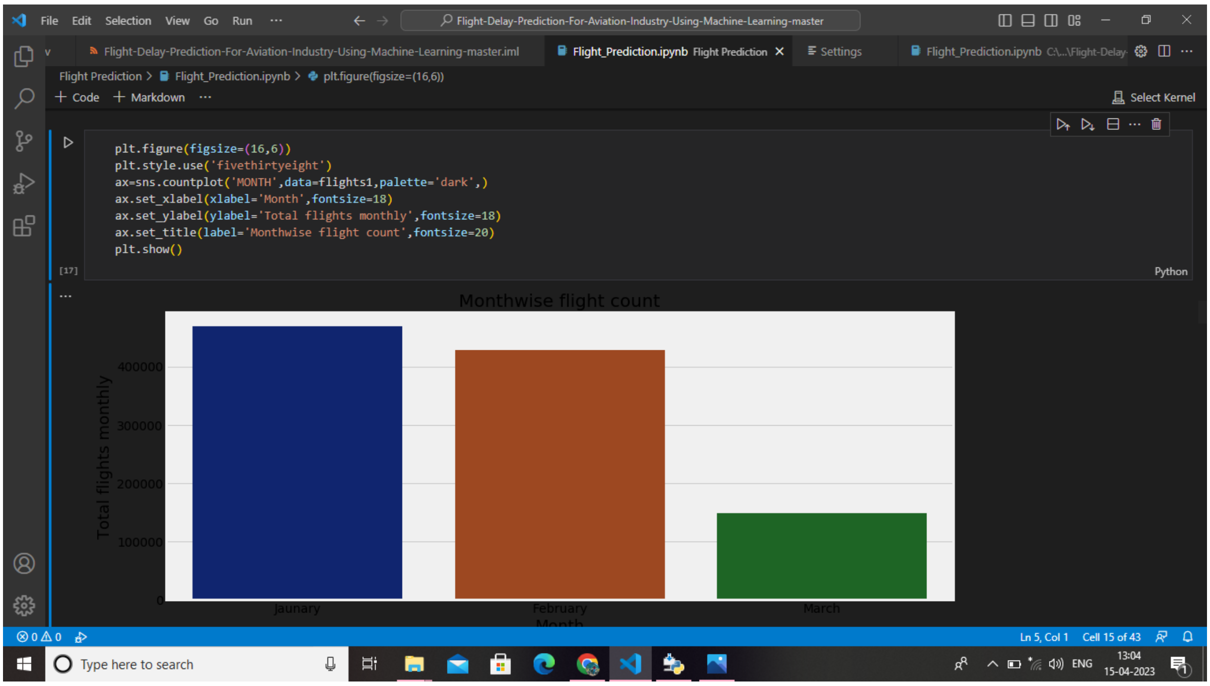

Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master
Flight_Prediction.ipynb
Select Kernel
airports.isnull().sum()
IATA_CODE 0
AIRPORT 0
CITY 0
STATE 0
COUNTRY 0
LATITUDE 3
LONGITUDE 3
dtype: int64
airports.dtypes
IATA_CODE object
AIRPORT object
CITY object
STATE object
COUNTRY object
LATITUDE float64
LONGITUDE float64
dtype: object
columns=airports.loc[:,['LATITUDE','LONGITUDE']]
Ln 1, Col 1 Cell 8 of 43
Type here to search
13:02
15-04-2023

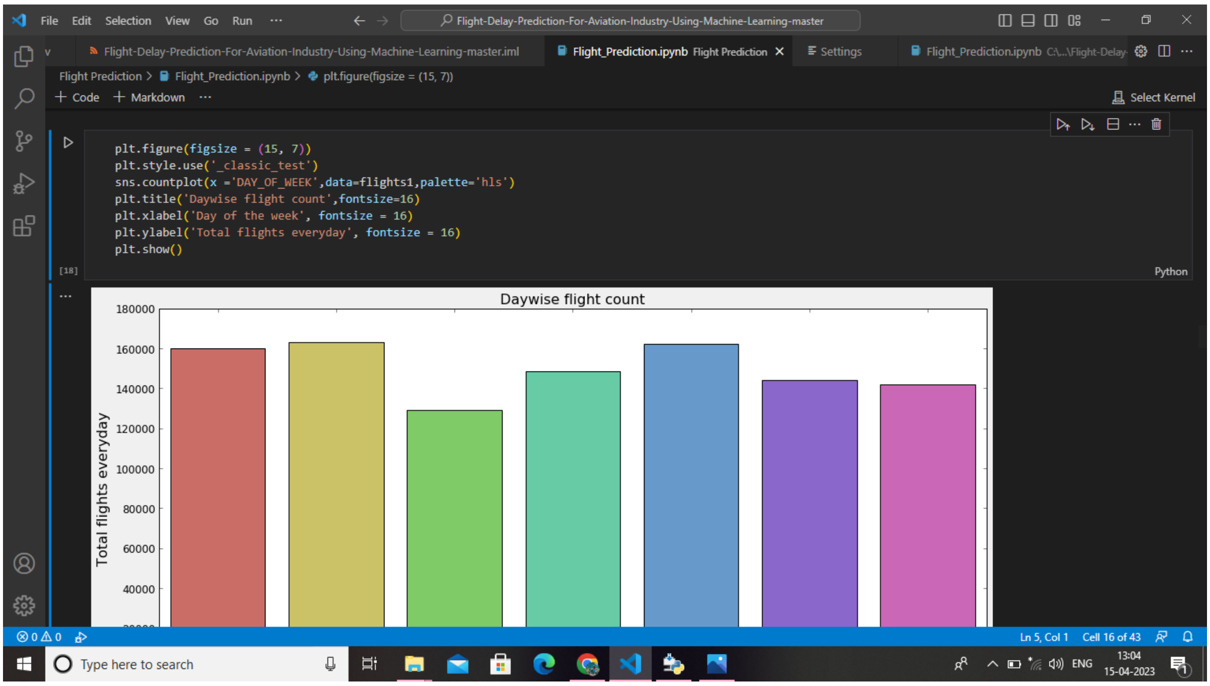

Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master
Flight_Prediction.ipynb
Select Kernel
LATITUDE float64
LONGITUDE float64
dtype: object
columns=airports.loc[:,['LATITUDE','LONGITUDE']]
airports.dropna(inplace=True)
airports.isnull().sum()
IATA_CODE 0
AIRPORT 0
CITY 0
STATE 0
COUNTRY 0
LATITUDE 0
LONGITUDE 0
dtype: int64
import gmplot
latitudes=airports.loc[:,'LATITUDE']
longitudes=airports.loc[:,'LONGITUDE']
Ln 1, Col 1 Cell 11 of 43
Type here to search
13:02
15-04-2023

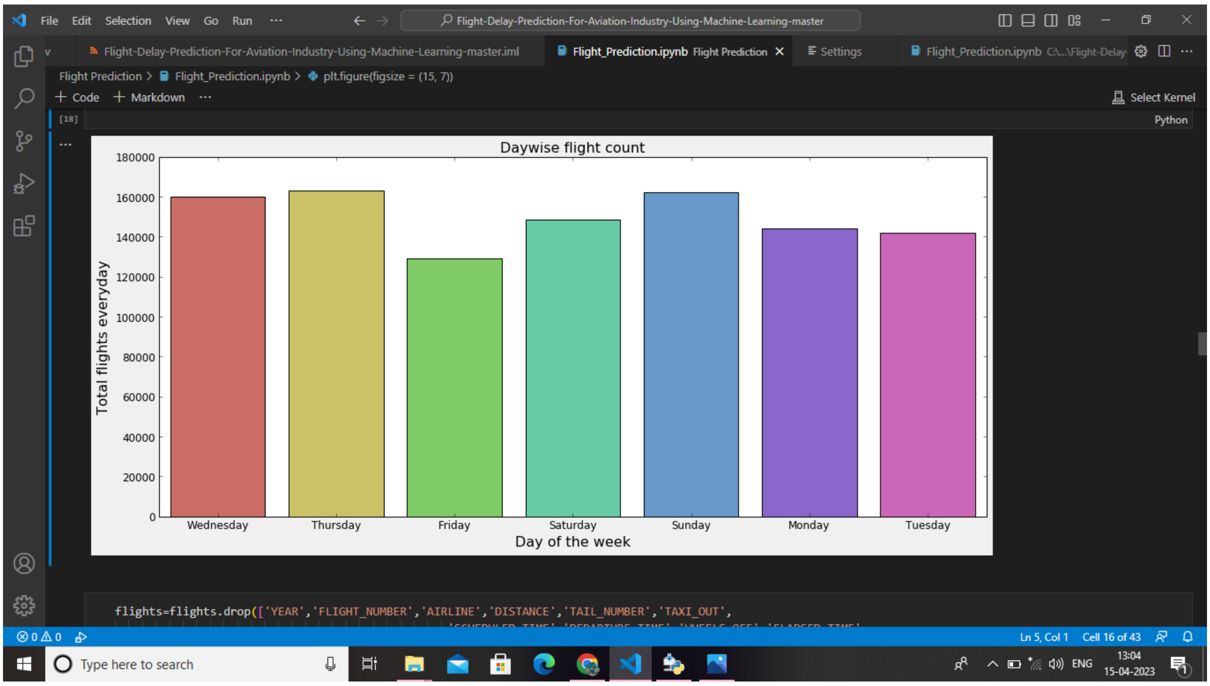

File Edit Selection View Go Run
Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master
Flight_Prediction.ipynb Flight Prediction
Settings
Flight Prediction > Flight_Prediction.ipynb > from IPython.display import IFrame
+ Code + Markdown
Select Kernel
airports.dropna(inplace=True)
[11]
Python
airports.isnull().sum()
[12]
Python
IATA_CODE 0
AIRPORT 0
CITY 0
STATE 0
COUNTRY 0
LATITUDE 0
LONGITUDE 0
dtype: int64
import gmplot
latitudes=airports.loc[:,'LATITUDE']
longitudes=airports.loc[:,'LONGITUDE']
gmap=gmplot.GoogleMapPlotter(35,102,2)
gmap.scatter(latitudes,longitudes,'red',size=5)
gmap.draw('map/gmplot.html')
Python
from IPython.display import IFrame
IFrame(src='map/gmplot.html',width=900, height=600)
Ln 6, Col 1 Cell 12 of 43
Type here to search
ENG
13:02
15-04-2023

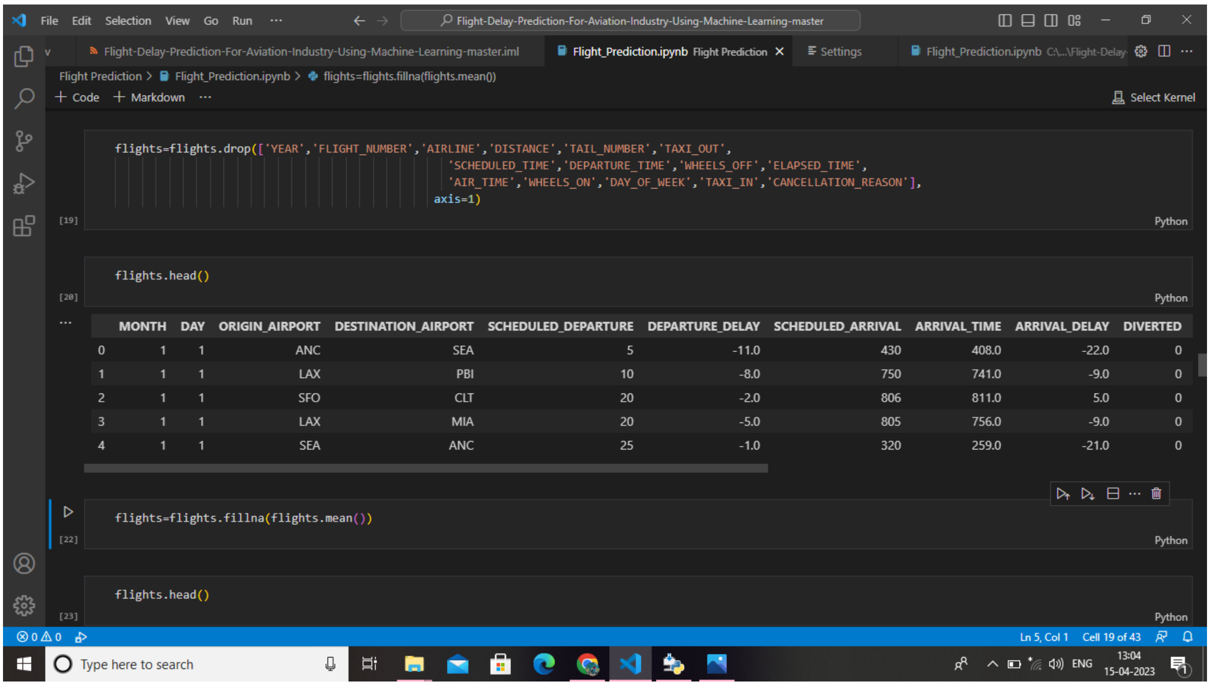

File Edit Selection View Go Run
Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master
Flight_Prediction.ipynb Flight Prediction
Settings
Flight Prediction > Flight_Prediction.ipynb > flights.isnull().sum()
+ Code + Markdown
Select Kernel
flights.isnull().sum()
[15]
Python
Output exceeds the size limit. Open the full output data in a text editor
YEAR 0
MONTH 0
DAY 0
DAY_OF_WEEK 0
AIRLINE 0
FLIGHT_NUMBER 0
TAIL_NUMBER 7750
ORIGIN_AIRPORT 0
DESTINATION_AIRPORT 0
SCHEDULED_DEPARTURE 0
DEPARTURE_TIME 39515
DEPARTURE_DELAY 39515
TAXI_OUT 40229
WHEELS_OFF 40229
SCHEDULED_TIME 2
ELAPSED_TIME 43071
AIR_TIME 43071
DISTANCE 0
WHEELS_ON 41296
TAXI_IN 41296
SCHEDULED_ARRIVAL 0
ARRIVAL_TIME 41296
ARRIVAL_DELAY 43071
DIVERTED 0
CANCELLED 0
Ln 2, Col 1 Cell 13 of 43
Type here to search
ENG
13:03
15-04-2023


File Edit Selection View Go Run ···
Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master
Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master.iml
Flight_Prediction.ipynb Flight Prediction
Settings
Flight Prediction > Flight_Prediction.ipynb > flights1=flights[['MONTH','DAY_OF_WEEK']]
+ Code + Markdown ···
Select Kernel
flights1=flights[['MONTH','DAY_OF_WEEK']]
flights1['MONTH']=flights1['MONTH'].map({1:'Jaunary',2:'February',3:'March',4:'April',5:'May',6:'June',7:'July',8:'August',
9:'September',10:'October',11:'November',12:'December'})
flights1['DAY_OF_WEEK']=flights1['DAY_OF_WEEK'].map({1:'Sunday',2:'Monday',3:'Tuesday',4:'Wednesday',5:'Thursday',6:'Friday',
7:'Saturday'})
[16]
Python
C:\Users\DELL\anaconda3\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
This is separate from the ipykernel package so we can avoid doing imports until
C:\Users\DELL\anaconda3\lib\site-packages\ipykernel_launcher.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
"""
plt.figure(figsize=(16,6))
plt.style.use('fivethirtyeight')
ax=sns.countplot('MONTH',data=flights1,palette='dark',)
ax.set_xlabel(xlabel='Month',fontsize=18)
ax.set_ylabel(ylabel='Total flights monthly',fontsize=18)
ax.set_title(label='Monthwise flight count',fontsize=20)
plt.show()
Ln 5, Col 1 Cell 14 of 43
Type here to search
ENG 13:03 15-04-2023


File Edit Selection View Go Run ···
Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master
Flight Prediction > Flight_Prediction.ipynb > plt.figure(figsize=(16,6))
+ Code + Markdown ···
Select Kernel
plt.figure(figsize=(16,6))
plt.style.use('fivethirtyeight')
ax=sns.countplot('MONTH',data=flights1,palette='dark',)
ax.set_xlabel(xlabel='Month',fontsize=18)
ax.set_ylabel(ylabel='Total flights monthly',fontsize=18)
ax.set_title(label='Monthwise flight count',fontsize=20)
plt.show()
[17]
Python
Monthwise flight count
Total flights monthly
400000
300000
200000
100000
0
Jaunary
February
March
Month
Ln 5, Col 1 Cell 15 of 43
Type here to search
ENG 13:04 15-04-2023


File Edit Selection View Go Run
Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master
Flight_Prediction.ipynb Flight Prediction
Settings
Flight Prediction > Flight_Prediction.ipynb > plt.figure(figsize = (15, 7))
+ Code + Markdown
Select Kernel
plt.figure(figsize = (15, 7))
plt.style.use('_classic_test')
sns.countplot(x ='DAY_OF_WEEK',data=flights1,palette='hls')
plt.title('Daywise flight count',fontsize=16)
plt.xlabel('Day of the week', fontsize = 16)
plt.ylabel('Total flights everyday', fontsize = 16)
plt.show()
[18]
Python
Daywise flight count
Total flights everyday
180000
160000
140000
120000
100000
80000
60000
40000
Ln 5, Col 1 Cell 16 of 43
Type here to search
13:04
15-04-2023


File Edit Selection View Go Run
Flight-Delay-Prediction-For-Aviation-Industry-Using-Machine-Learning-master
Flight_Prediction.ipynb Flight Prediction
Settings
Flight Prediction > Flight_Prediction.ipynb > plt.figure(figsize = (15, 7))
+ Code + Markdown
Select Kernel
[18]
Python
Daywise flight count
Total flights everyday
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
Wednesday
Thursday
Friday
Saturday
Sunday
Monday
Tuesday
Day of the week
flights=flights.drop(['YEAR','FLIGHT_NUMBER','AIRLINE','DISTANCE','TAIL_NUMBER','TAXI_OUT',
Ln 5, Col 1 Cell 16 of 43
Type here to search
13:04
15-04-2023



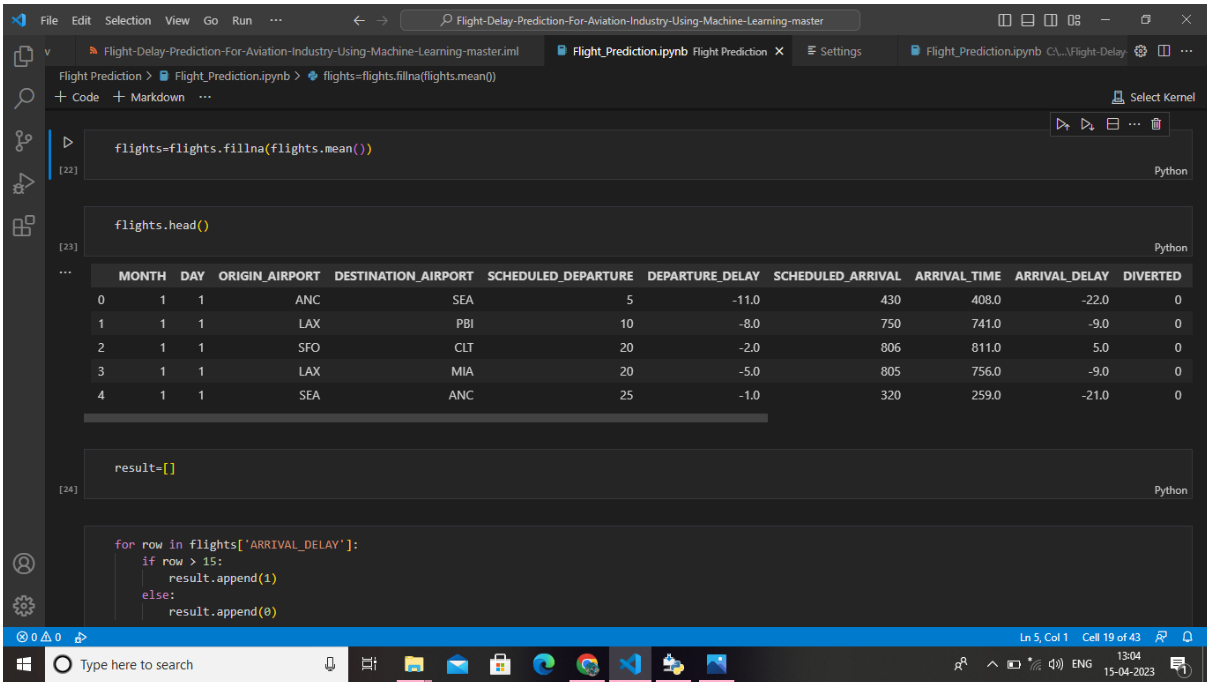
```python
flights=flights.drop(['YEAR','FLIGHT_NUMBER','AIRLINE','DISTANCE','TAIL_NUMBER','TAXI_OUT',
                      'SCHEDULED_TIME','DEPARTURE_TIME','WHEELS_OFF','ELAPSED_TIME',
                      'AIR_TIME','WHEELS_ON','DAY_OF_WEEK','TAXI_IN','CANCELLATION_REASON'],
                     axis=1)
```

```python
flights.head()
```

	MONTH	DAY	ORIGIN_AIRPORT	DESTINATION_AIRPORT	SCHEDULED_DEPARTURE	DEPARTURE_DELAY	SCHEDULED_ARRIVAL	ARRIVAL_TIME	ARRIVAL_DELAY	DIVERTED
0	1	1	ANC	SEA	5	-11.0	430	408.0	-22.0	0
1	1	1	LAX	PBI	10	-8.0	750	741.0	-9.0	0
2	1	1	SFO	CLT	20	-2.0	806	811.0	5.0	0
3	1	1	LAX	MIA	20	-5.0	805	756.0	-9.0	0
4	1	1	SEA	ANC	25	-1.0	320	259.0	-21.0	0

```python
flights=flights.fillna(flights.mean())
```

```python
flights.head()
```



```python
flights=flights.fillna(flights.mean())
```

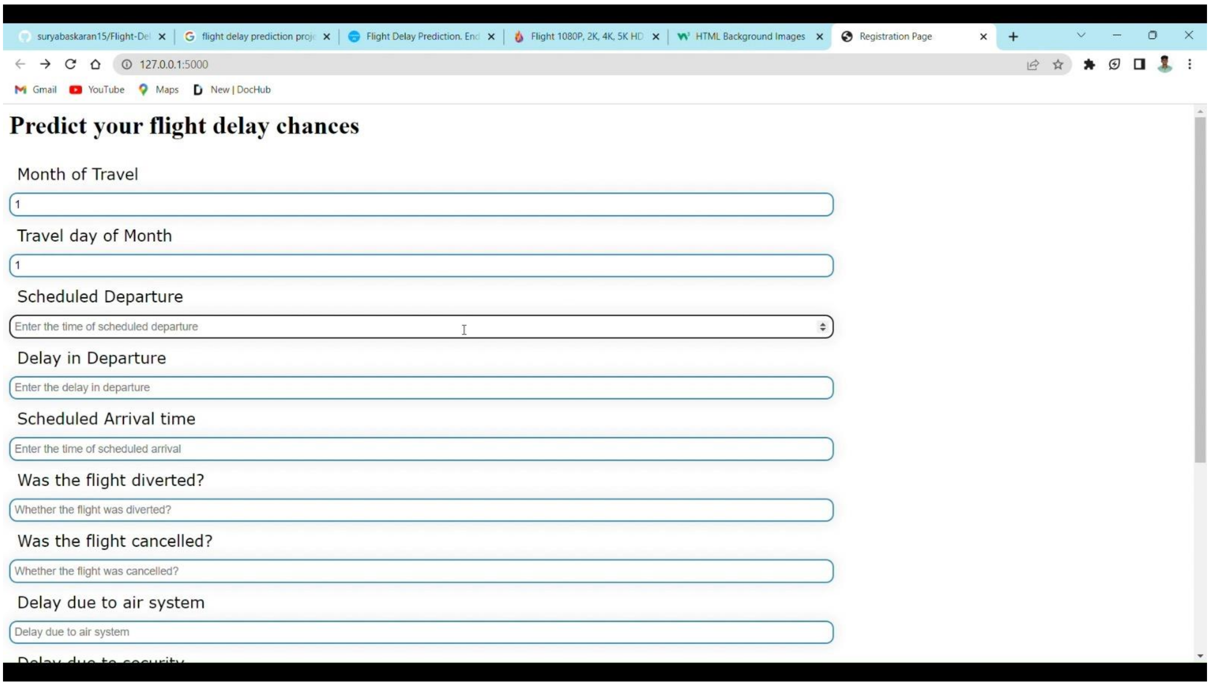
```python
flights.head()
```

	MONTH	DAY	ORIGIN_AIRPORT	DESTINATION_AIRPORT	SCHEDULED_DEPARTURE	DEPARTURE_DELAY	SCHEDULED_ARRIVAL	ARRIVAL_TIME	ARRIVAL_DELAY	DIVERTED
0	1	1	ANC	SEA	5	-11.0	430	408.0	-22.0	0
1	1	1	LAX	PBI	10	-8.0	750	741.0	-9.0	0
2	1	1	SFO	CLT	20	-2.0	806	811.0	5.0	0
3	1	1	LAX	MIA	20	-5.0	805	756.0	-9.0	0
4	1	1	SEA	ANC	25	-1.0	320	259.0	-21.0	0

```python
result=[]
```

```python
for row in flights['ARRIVAL_DELAY']:
    if row > 15:
        result.append(1)
    else:
        result.append(0)
```

C. SAMPLE INPUT



D. SAMPLE OUTPUT