OPTIMIZING FLIGHT DECISIONS THROUGH MACHINE LEARNING PRICE PREDICITIONS

Submitted by

Thangadurai

Vallarasu

Gokulraj

Vennila

Susila

1.INTRODUCTION

People who work frequently travel through flight will have better knowledge on best discount and right time to buy the ticket. For the business purpose many airline companies change prices according to the seasons or time duration. They will increase the price when people travel more. Estimating the highest prices of the airlines data for the route is collected with features such as Duration, Source, Destination, Arrival and Departure. Features are taken from chosen dataset and in the price wherein the airline price ticket costs vary overtime.

1.1 OVERVIEW

The average price for an airline ticket in the united states in November 2022 was $280,about 35% higher than November 2021,according to statistics from the U.S.Bureau of Labor statistics .November’s average price was down from may 2022 when the average price for a domestic flight hit an all-time high of $336.

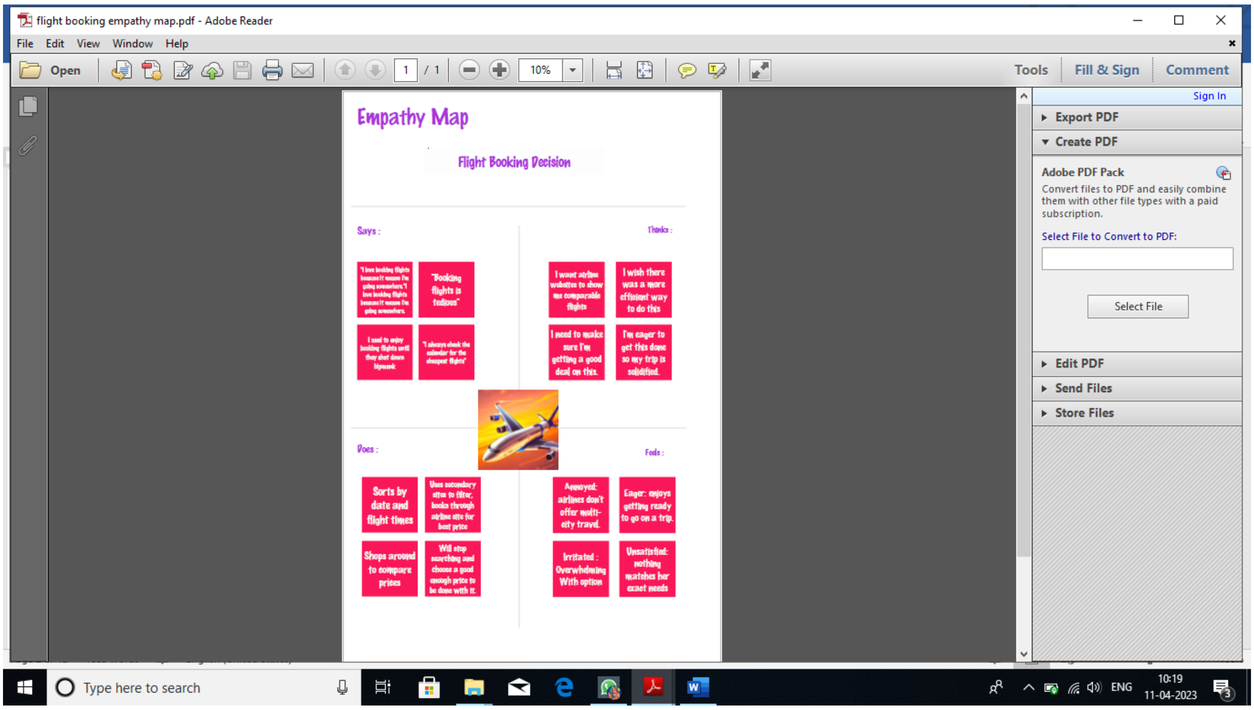
A flighty price prediction application which predicts fares of flight for a particular date based on various parameters like source,destination,stops & airline.

1.2 PURPOSE

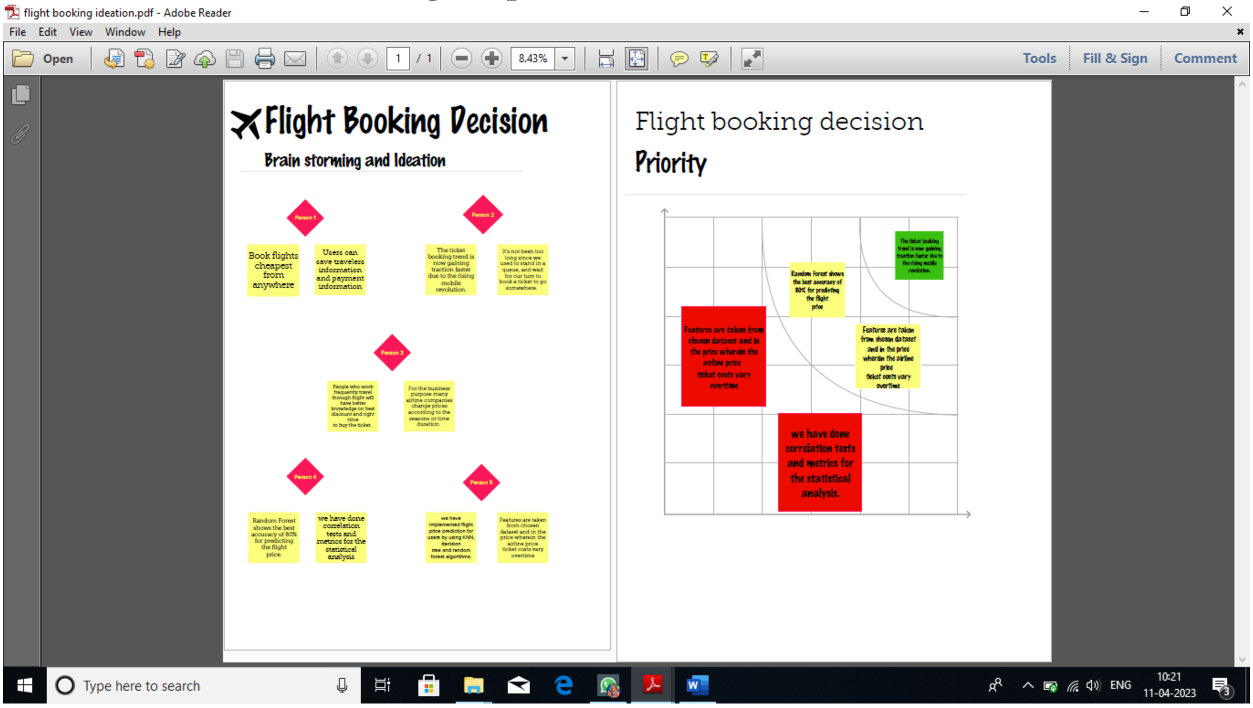
The main objective of the project is , Features are taken from chosen dataset and in the price wherein the airline price ticket costs vary overtime. we have implemented flight price prediction for users by using KNN, decision tree and random forest algorithms. Random Forest shows the best accuracy of 80% for predicting the flight price. also, we have done correlation tests and metrics for the statistical analysis.

2.Problem Definition & Design Thinking

* Empathy map



- Ideation & Brainstroming map



3.THEORITICAL ANALYSIS

This project is to use machine learning techniques to model the behaviour of flight prices over the time and predict the price of the flight-ticket.

To study how airline ticket prices changes over time ,extract the factors that influence these functions and describe how they're correlated.

3.2 HARDWARE / SOFTWARE DESIGNING

The hardware required for the development of this project is:

Processor : AMD
Processor speed : 2.4GHz
RAM Size : 4 GB
System Type : Windows

SOFTWARE DESIGNING:

The software required for the development of this project is:

Desktop GUI : Anaconda Navigator
Operating system : Windows 10
Front end : HTML, VS CODE,
Programming : PYTHON
Cloud Computing Service : IBM Cloud Service

4.EXPERIMENTAL INVESTIGATION

IMPORTING AND READING THE DATASET

Importing the Libraries
First step is usually importing the libraries that will be needed in the program.
Pandas: It is a python library mainly used for data manipulation.
NumPy: This python library is used for numerical analysis.
Matplotlib and Seaborn: Both are the data visualization library used for plotting graph which will help us for understanding the data.
csr_matrix() :A dense matrix stored in a NumPy array can be converted into a sparse matrix using the CSR representation by calling the csr_matrix() function.
Train_test_split: used for splitting data arrays into training data and for testing data.
Pickle: to serialize your machine learning algorithms and save the serialized format to a file.

Reading the Dataset
For this project, we make use of three different datasets (Books_Ratings, Books, Users). We will be selecting the important features from these datasets that will help us in recommending the best results.
The next step is to read the dataset into a data structure that's compatible with pandas. Let's load a .csv data file into pandas. There is a function for it, called **read_csv().**We will need to locate the directory of the CSV file at first (it's more efficient to keep the dataset in the same directory as your program).If the dataset in same directory of your program, you can directly read it, without any path. After the next Steps we made following bellow:

1.Data visualization

2.Collabrative and filtering

3.Creating the Model

4.Test and save the model

5.Buil Python Code

6.Build HTML Code

7.Run the Application
We are the following above sections we did and investigate it.

5.FLOWCHART

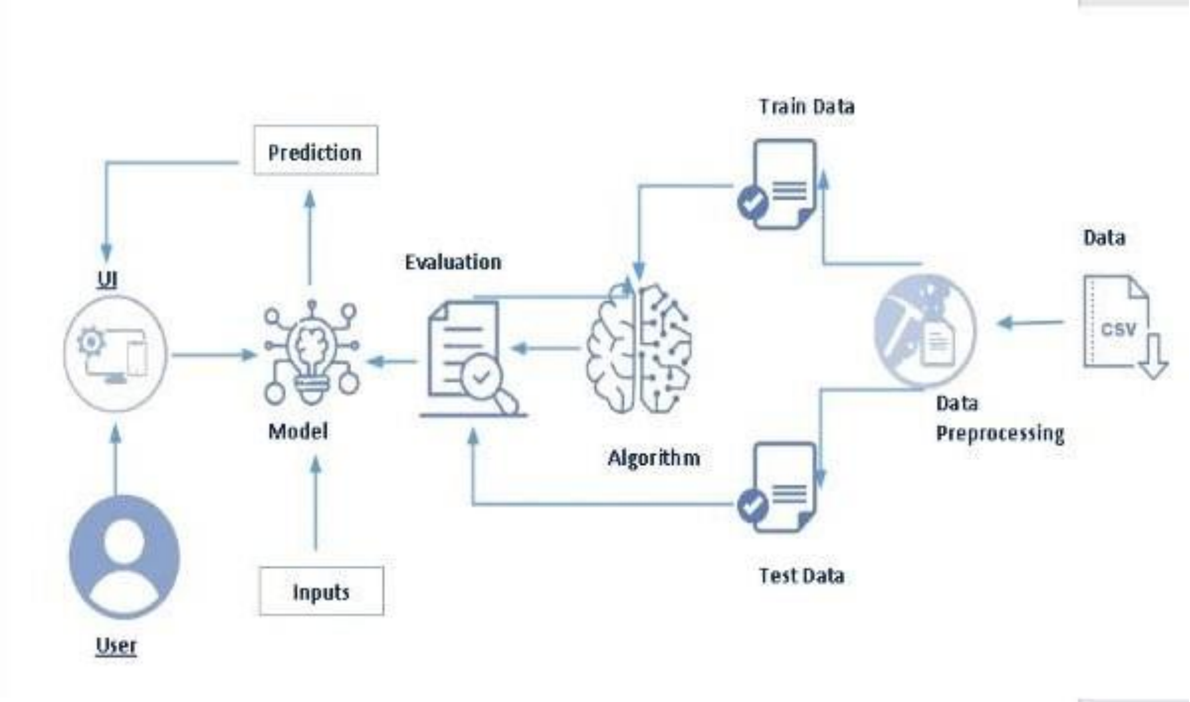


Fig 5.1 Flowchart of the project

Project Flow:

- User interacts with the UI (User Interface) to upload the input features.
- Uploaded features/input is analysed by the model which is integrated.

Once a model analyses the uploaded inputs, the prediction is showcased on the UI.

1. **Data collection**
   - Collect the dataset or create the dataset
   - Visualizing and analyzing data
   - Importing Libraries
   - Read the DataSet

2. **Data pre-processing**
   - Checking for null values
   - Handling outlier
   - Handling categorical data
   - Splitting data into train and test

3. **Model building**
   - Import the model building libraries
   - Initializing the model
   - Training and testing the model
   - Evaluating performance of model
   - Save the model

4. **Application Building**
   - Create an HTML file
   - Build python code

6.RESULT

Fig 6.1 Flask code on Spyder

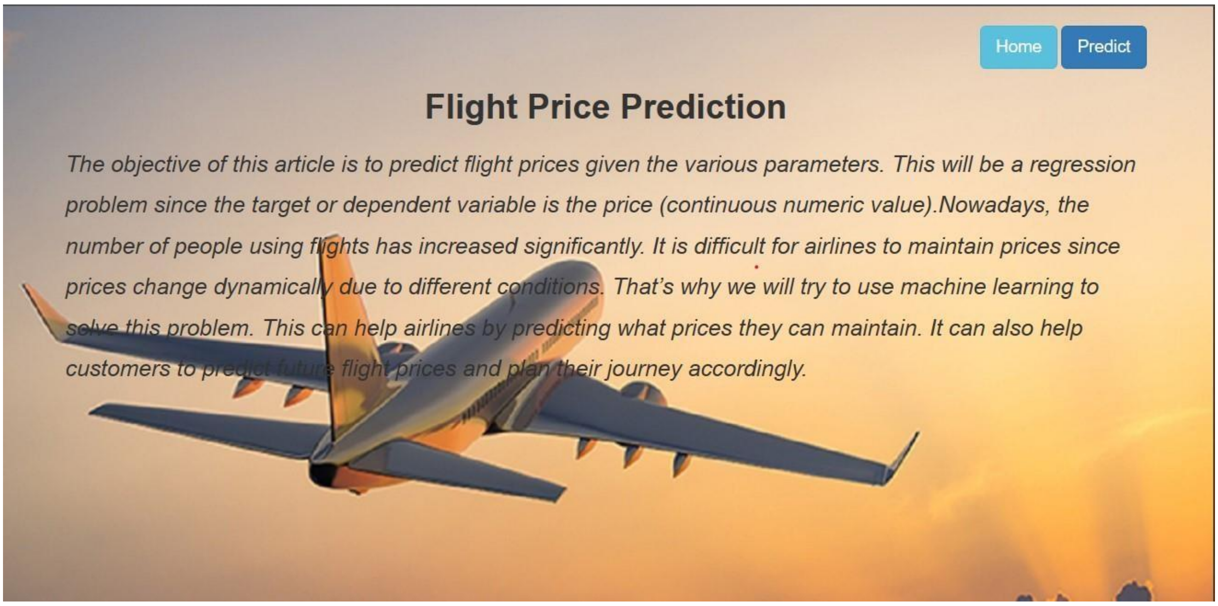


Fig 6.2 Home page for Flight Price Prediction

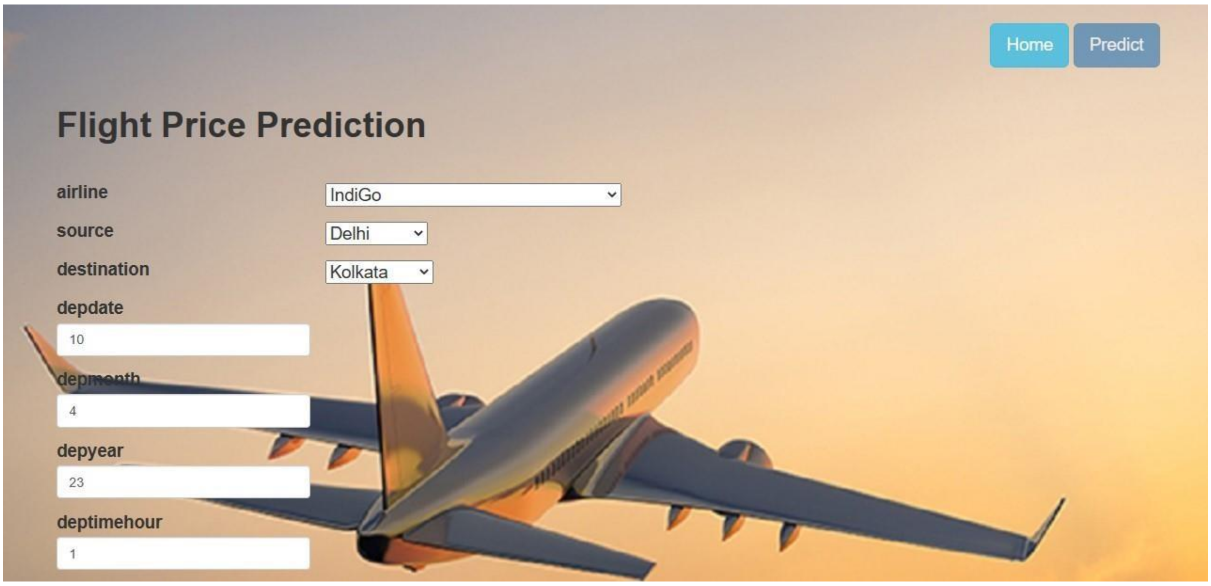


Fig 6.3 Predicting page of Flight Price Prediction

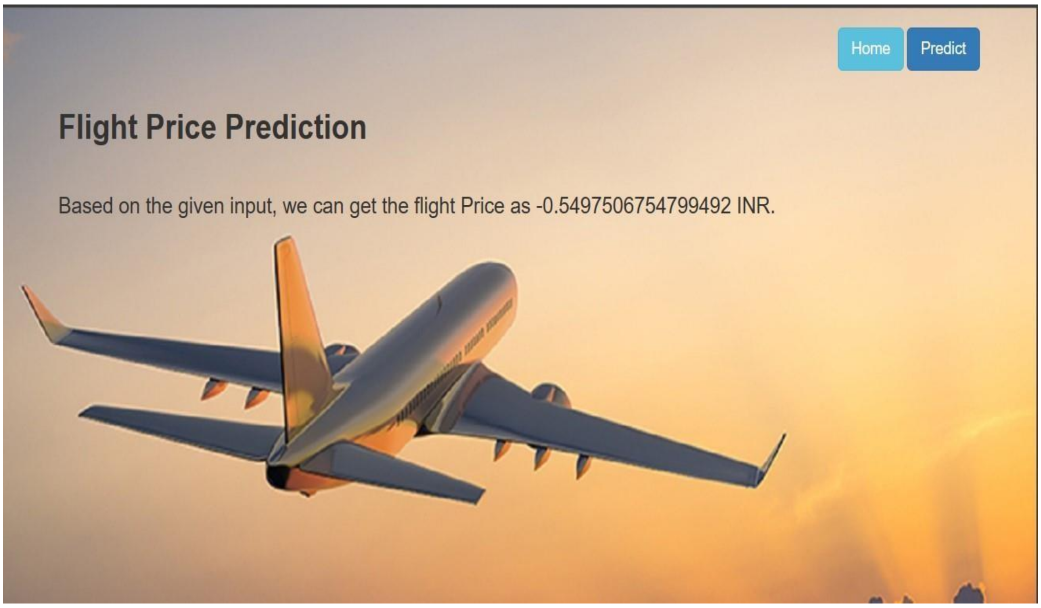


Fig 6.4 Output page of Flight Price Prediction

7. ADVANTAGES AND DISADVANTAGES

ADVANTAGES

- Traveler get the fare prediction handy using which it’s easy to decide the airlines.
- Saves time in searching / deciding for airlines.

DISADVANTAGES

- Improper data will result in incorrect fare predictions.

8. APPLICATIONS

- make traveling easier
- Airfare tracking
- flight search and airfare prediction
- Airfare tracking and hotel booking.

9. CONCLUSION

In this project is to forecast the average flight price at the business segment level. We used training data to train the training data and test data to test it. These records were used to extract a number of characteristics. Our suggested model can estimate the quarterly average flight price using attribute selection strategies.To the highest possible standard, much prior studies into flight price prediction using the large dataset depended on standard statistical approaches, which have their own limitations in terms of underlying issue estimates and hypotheses. To our knowledge, no other research have included statistics from holidays, celebrations, stock market price fluctuations, depression, fuel price, and socioeconomic information to estimate the air transport market sector; nonetheless, there are numerous restrictions.As example, neither of the databases provide precise information about ticket revenue, including such departing and arrival times and days of the week. This framework may be expanded in the future to also include airline tickets payment details, that can offer more detail about each area, such as timestamp of entry and exit, seat placement, covered auxiliary items, and so on. By merging such data, it is feasible to create a more robust and complete daily and even daily flight price forecast model. Furthermore, a huge surge of big commuters triggered by some unique events might alter flight costs in a market sector. Thus, incident data will be gathered from a variety of sources, including social media sites and media organizations, to supplement our forecasting models. We will also examine specific technological Models, such as Deeper Learning methods, meanwhile striving to enhance existing models by modifying their hyper-parameters to get the optimum design for airline price prediction.

10. FUTURESCOPE

- More routes can be added and the same analysis can be expanded to major airports and travel routes in india.
- The analysis can be done by increasing the data points and increasing the historical data used. That will train the model better giving better accuracies and more savings.
- More rules can be added in the rule based learning based on our understanding of the industry, also incorporating the offer periods given by the airlines .
- Developing a more user friendly interface for various routes giving more flexibility to the users.

11. BIBILOGRAPHY

[1] K. Tziridis, T. Kalampokas, G. A. Papakostas and K. I. Diamantaras, Airfare prices prediction using machine learning techniques, 2017 25th European Signal Processing Conference (EUSIPCO), Kos, Greece, 2017, pp. 1036-1039, https://doi.org/10.23919/EUSIPCO.2017.8081365 .

[2] Juhar Ahmed Abdella, Nazar Zaki, Khaled Shuaib, Fahad Khan, Airline ticket price and demand prediction: A survey, Journal of King Saud University - Computer and Information Sciences, 2019, ISSN 1319-1578, https://doi.org/10.1016/j.jksuci.2019.02.001

[3] Martijn Brons, Eric Pels, Peter Nijkamp, Piet Rietveld, Price elasticities of demand for passenger air travel: a meta-analysis, Journal of Air Transport Management, Volume 8, Issue 3, 2002, Pages 165-175, ISSN 0969- 6997, https://doi.org/10.1016/S0969-6997(01)00050-3 .

[4] Silke J. Forbes, The effect of air traffic delays on airline prices, International Journal of Industrial Organization, Volume 26, Issue 5, 2008, Pages 1218-1232, ISSN 0167-7187, https://doi.org/10.1016/j.ijindorg.2007.12.004 .

APPENDIX

A Source Code

flight price prediction (unsaved changes)

```
In [1]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
        from sklearn.model_selection import train_test_split
        from sklearn.ensemble import RandomForestClassifier,GradientBoostingClassifier
        from sklearn.tree import DecisionTreeClassifier
        from sklearn.neighbors import KNeighborsClassifier
        from sklearn.metrics import f1_score
        from sklearn.metrics import classification_report,confusion_matrix
        import warnings
        import pickle
        from scipy import stats
        warnings.filterwarnings('ignore')
        plt.style.use('fivethirtyeight')
```

```
In [2]: data=pd.read_excel('Data_Train.xlsx')
        data.head()
```

Out[2]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
0	IndiGo	24/03/2019	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897
1	Air India	1/05/2019	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	No info	7662
2	Jet Airways	9/06/2019	Delhi	Cochin	DEL → LKO → BOM → COK	09:25	04:25 10 Jun	19h	2 stops	No info	13882
3	IndiGo	12/05/2019	Kolkata	Banglore	CCU → NAG → BLR	18:05	23:30	5h 25m	1 stop	No info	6218
4	IndiGo	01/03/2019	Banglore	New Delhi	BLR → NAG → DEL	16:50	21:35	4h 45m	1 stop	No info	13302

```
In [3]: data.rename(columns={'ï»¿Airline':'Airline'},inplace=True)

In [4]: data.shape
Out[4]: (10683, 11)

In [5]: data.info()
```

flight price prediction (unsaved changes)

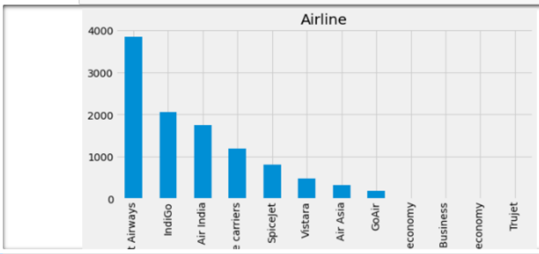

```
In [4]: data.shape
Out[4]: (10683, 11)

In [5]: data.info()
```


```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10683 entries, 0 to 10682
Data columns (total 11 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Airline          10683 non-null  object
 1   Date_of_Journey  10683 non-null  object
 2   Source           10683 non-null  object
 3   Destination      10683 non-null  object
 4   Route            10682 non-null  object
 5   Dep_Time         10683 non-null  object
 6   Arrival_Time     10683 non-null  object
 7   Duration         10683 non-null  object
 8   Total_Stops      10682 non-null  object
 9   Additional_Info  10683 non-null  object
 10  Price            10683 non-null  int64
dtypes: int64(1), object(10)
memory usage: 918.2+ KB
```

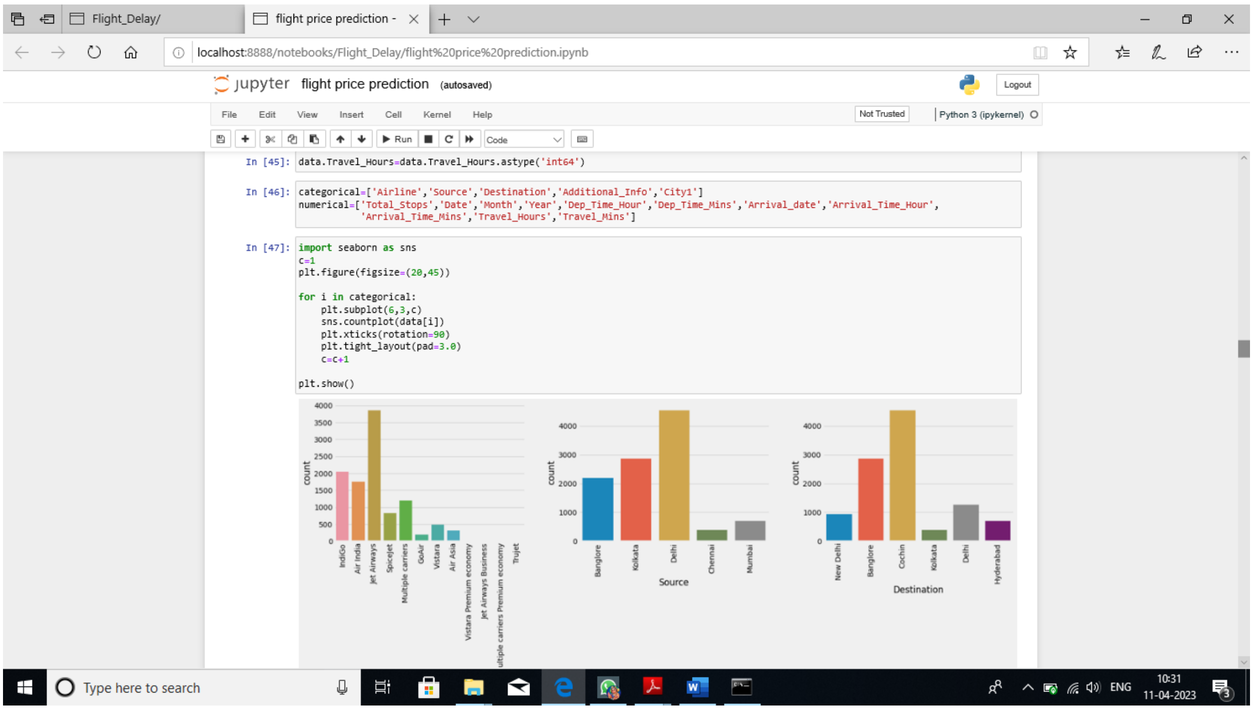
```
In [6]: data.isnull().any()
Out[6]: Airline            False
        Date_of_Journey    False
        Source             False
        Destination        False
        Route               True
        Dep_Time           False
        Arrival_Time       False
        Duration           False
        Total_Stops         True
        Additional_Info    False
        Price              False
        dtype: bool
```

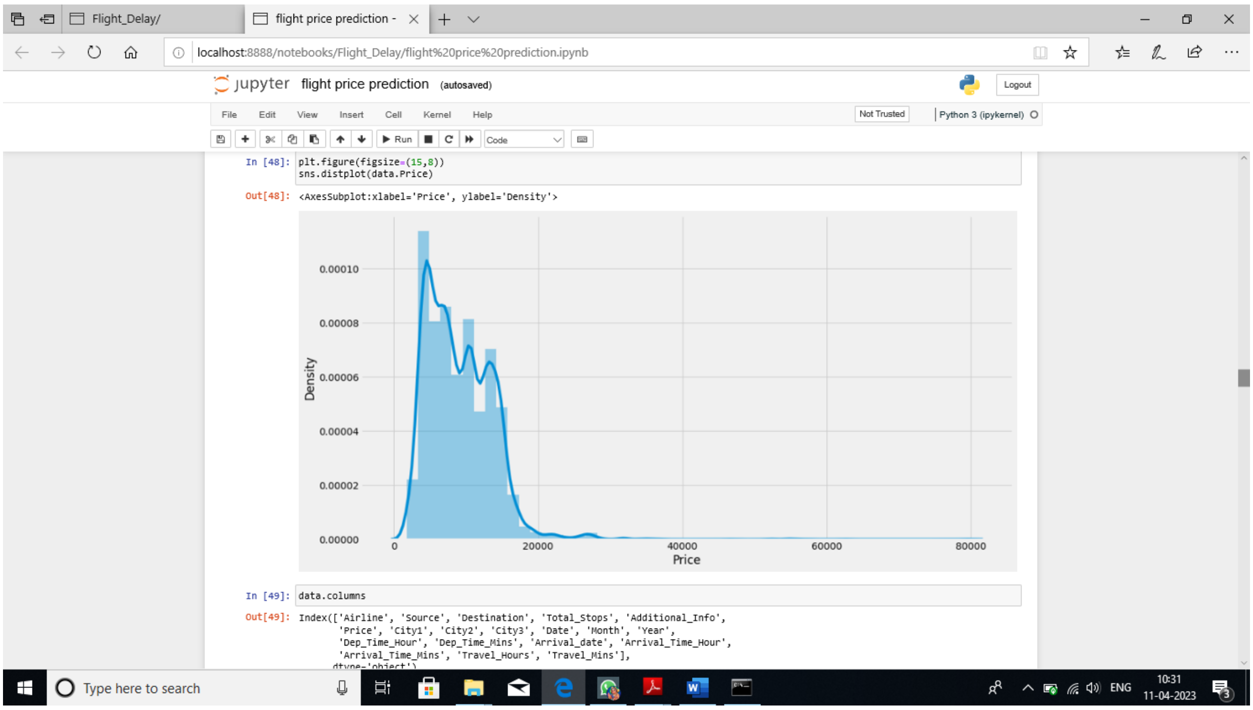
```
dtype: bool

In [7]: data.isnull().sum()

Out[7]: Airline            0
        Date_of_Journey    0
        Source             0
        Destination        0
        Route              1
        Dep_Time           0
        Arrival_Time       0
        Duration           0
        Total_Stops        1
        Additional_Info    0
        Price              0
        dtype: int64

In [8]: category=['Airline','Source','Destination','Additional_Info']

In [9]: category

Out[9]: ['Airline', 'Source', 'Destination', 'Additional_Info']

In [10]: data.dropna(inplace=True)

In [11]: data.isnull().sum()

Out[11]: Airline            0
         Date_of_Journey    0
         Source             0
         Destination        0
         Route              0
         Dep_Time           0
         Arrival_Time       0
         Duration           0
         Total_Stops        0
         Additional_Info    0
         Price              0
         dtype: int64
```

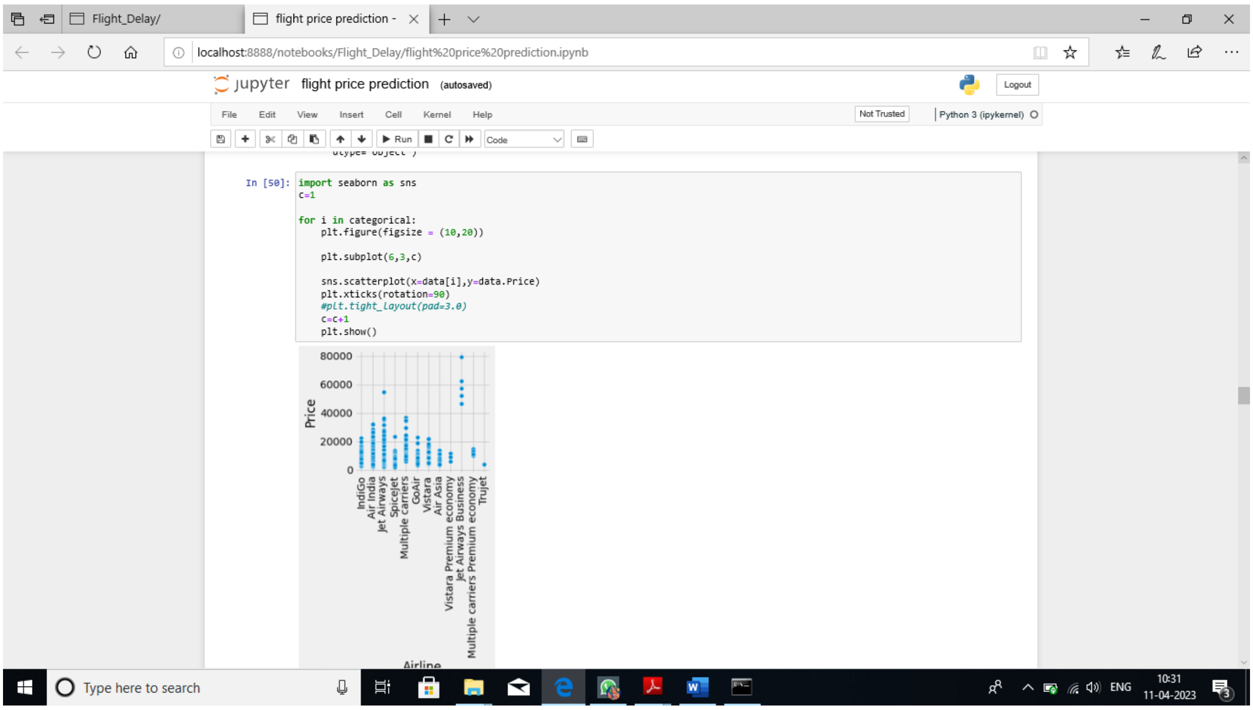
```
    print(i, data[i].unique())

Airline ['IndiGo' 'Air India' 'Jet Airways' 'SpiceJet' 'Multiple carriers' 'GoAir'
 'Vistara' 'Air Asia' 'Vistara Premium economy' 'Jet Airways Business'
 'Multiple carriers Premium economy' 'Trujet']
Source ['Banglore' 'Kolkata' 'Delhi' 'Chennai' 'Mumbai']
Destination ['New Delhi' 'Banglore' 'Cochin' 'Kolkata' 'Delhi' 'Hyderabad']
Additional_Info ['No info' 'In-flight meal not included' 'No check-in baggage included'
 '1 Short layover' 'No Info' '1 Long layover' 'Change airports'
 'Business class' 'Red-eye flight' '2 Long layover']

In [15]: for column in category_cols:
             plt.figure(figsize=(20,4))
             plt.subplot(121)
             data[column].value_counts().plot(kind='bar')
             plt.title(column)
```



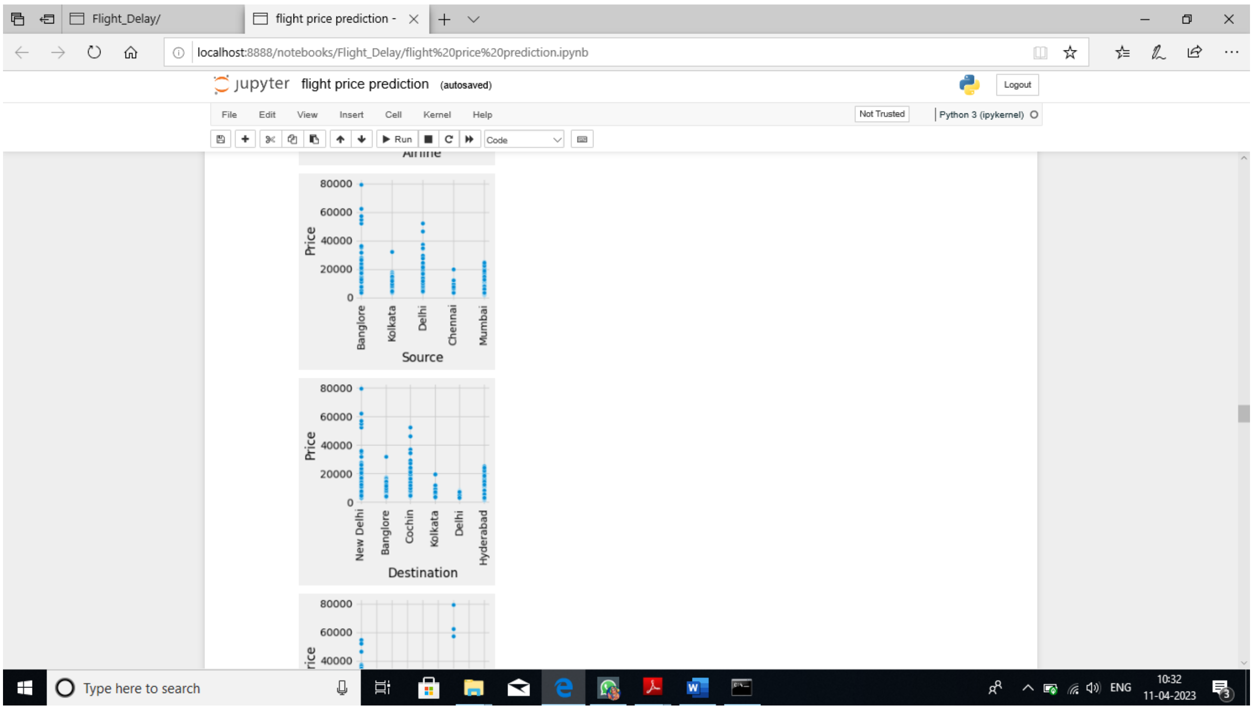

```
In [16]: data.Total_Stops.unique()

Out[16]: array(['non-stop', '2 stops', '1 stop', '3 stops', '4 stops'],
               dtype=object)
```


Flight_Delay/
flight price prediction -
localhost:8888/notebooks/Flight_Delay/flight%20price%20prediction.ipynb
jupyter flight price prediction (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Not Trusted
Python 3 (ipykernel)
Run
Code
In [45]: data.Travel_Hours=data.Travel_Hours.astype('int64')
In [46]: categorical=['Airline','Source','Destination','Additional_Info','City1']
numerical=['Total_Stops','Date','Month','Year','Dep_Time_Hour','Dep_Time_Mins','Arrival_date','Arrival_Time_Hour',
'Arrival_Time_Mins','Travel_Hours','Travel_Mins']
In [47]: import seaborn as sns
c=1
plt.figure(figsize=(20,45))
for i in categorical:
plt.subplot(6,3,c)
sns.countplot(data[i])
plt.xticks(rotation=90)
plt.tight_layout(pad=3.0)
c=c+1
plt.show()
count
IndiGo
Air India
Jet Airways
SpiceJet
Multiple carriers
GoAir
Vistara
Air Asia
Vistara Premium economy
Jet Airways Business
Multiple carriers Premium economy
Trujet
Banglore
Kolkata
Delhi
Chennai
Mumbai
Source
New Delhi
Cochin
Hyderabad
Destination
Type here to search
ENG
10:31
11-04-2023

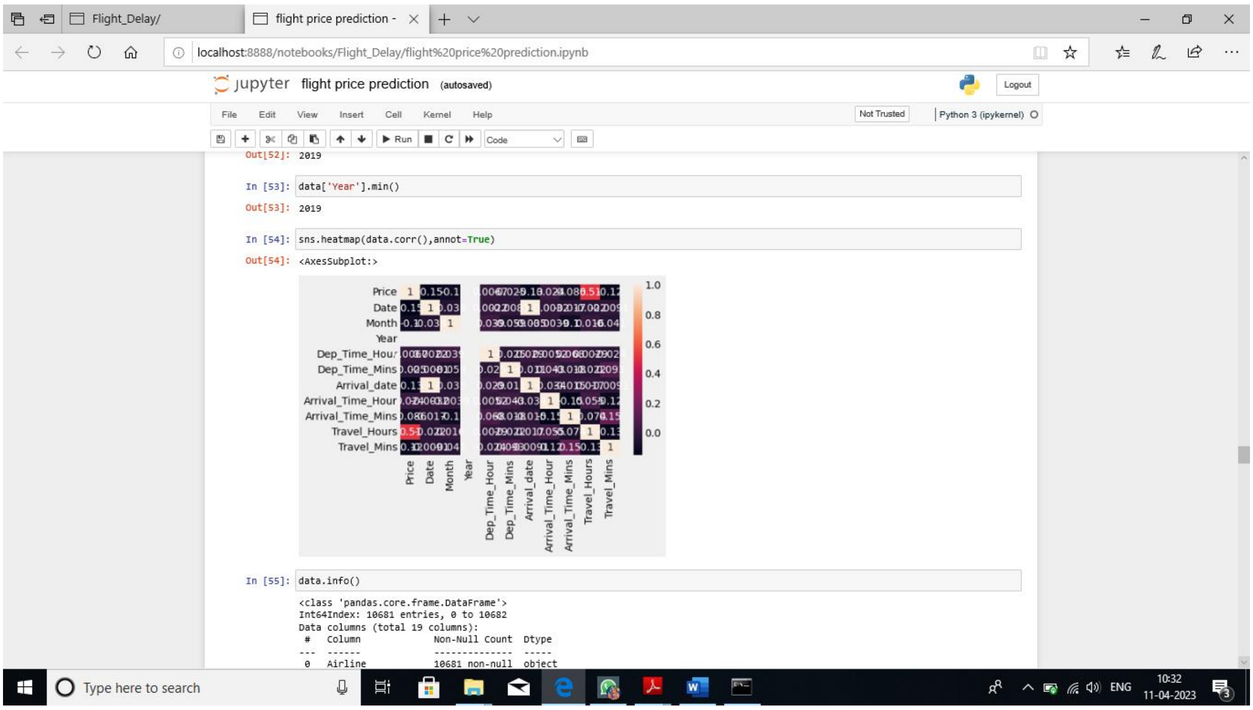

Flight_Delay/
flight price prediction -
localhost:8888/notebooks/Flight_Delay/flight%20price%20prediction.ipynb
jupyter flight price prediction (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Not Trusted
Python 3 (ipykernel)
Run
Code
In [48]: plt.figure(figsize=(15,8))
sns.distplot(data.Price)
Out[48]: <AxesSubplot:xlabel='Price', ylabel='Density'>
Density
0.00010
0.00008
0.00006
0.00004
0.00002
0.00000
0
20000
40000
60000
80000
Price
In [49]: data.columns
Out[49]: Index(['Airline', 'Source', 'Destination', 'Total_Stops', 'Additional_Info',
'Price', 'City1', 'City2', 'City3', 'Date', 'Month', 'Year',
'Dep_Time_Hour', 'Dep_Time_Mins', 'Arrival_date', 'Arrival_Time_Hour',
'Arrival_Time_Mins', 'Travel_Hours', 'Travel_Mins'],
Type here to search
ENG
10:31
11-04-2023

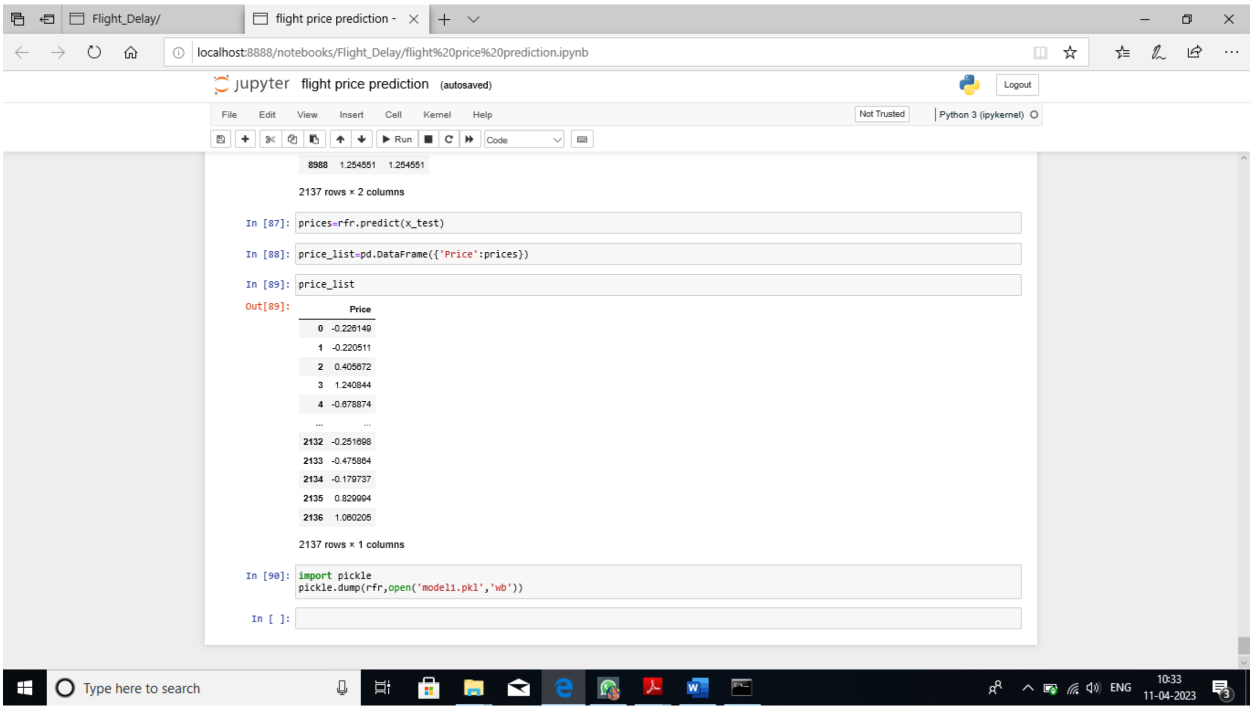


flight price prediction -
localhost:8888/notebooks/Flight_Delay/flight%20price%20prediction.ipynb
jupyter flight price prediction (autosaved)
In [50]:
import seaborn as sns
c=1
for i in categorical:
    plt.figure(figsize = (10,20))
    plt.subplot(6,3,c)
    sns.scatterplot(x=data[i],y=data.Price)
    plt.xticks(rotation=90)
    #plt.tight_layout(pad=3.0)
    c=c+1
    plt.show()
Price
IndiGo
Air India
Jet Airways
SpiceJet
Multiple carriers
GoAir
Vistara
Air Asia
Vistara Premium economy
Jet Airways Business
Multiple carriers Premium economy
Trujet
Airline
Type here to search
10:31
11-04-2023



flight price prediction -
localhost:8888/notebooks/Flight_Delay/flight%20price%20prediction.ipynb
jupyter flight price prediction (autosaved)
Price
Banglore
Kolkata
Delhi
Chennai
Mumbai
Source
New Delhi
Banglore
Cochin
Kolkata
Delhi
Hyderabad
Destination
Type here to search
10:32
11-04-2023





A Source Code of Flask:

from flask import Flask, render_template, request

```
import numpy as np
import pickle
import pandas as pd

model = pickle.load(open(r"D:\flight\flask\model1.pkl",'rb'))

app = Flask(__name__)

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




@app.route("/predict")
def home1():
    return render_template('predict.html')

@app.route("/pred", methods=['POST','GET'])
def predict():
    x = [[int(x) for x in request.form.values()]]
    print(x)

    x = np.array(x)
    print(x.shape)


    print(x)
```

```
    pred = model.predict(x)
    print(pred[0])
    return render_template('submit.html', prediction_text=pred[0])

if__name__== "_main_":
    app.run(debug=False)
```