## Overview

In this article, we will be analyzing the flight fare prediction using Machine Learning dataset using essential exploratory data analysis techniques then will draw some predictions about the price of the flight based on some features such as what type of airline it is, what is the arrival time, what is the departure time, what is the duration of the flight, source, destination and more.



## Takeaways from the blog

In this article, we do prediction using machine learning which leads to below takeaways:

1. EDA: Learn the complete process of EDA
2. Data analysis: Learn to withdraw some insights from the dataset both mathematically and visualize it.
3. Data visualization: Visualizing the data to get better insight from it.
4. Feature engineering: We will also see what kind of stuff we can do in the feature engineering part.

## 

## 

## 

## 

## About the dataset

1. Airline: So this column will have all the types of airlines like Indigo, Jet Airways, Air India, and many more.
2. Date\_of\_Journey: This column will let us know about the date on which the passenger’s journey will start.
3. Source: This column holds the name of the place from where the passenger’s journey will start.
4. Destination: This column holds the name of the place to where passengers wanted to travel.
5. Route: Here we can know about that what is the route through which passengers have opted to travel from his/her source to their destination.
6. Arrival\_Time: Arrival time is when the passenger will reach his/her destination.
7. Duration: Duration is the whole period that a flight will take to complete its journey from source to destination.
8. Total\_Stops: This will let us know in how many places flights will stop there for the flight in the whole journey.
9. Additional\_Info: In this column, we will get information about food, kind of food, and other amenities.
10. Price: Price of the flight for a complete journey including all the expenses before onboarding.

## Reading the training data of our dataset

df **=** pd.read\_excel('Data\_Train.xlsx', engine**=**'openpyxl')

In [3]:

df

Out[3]:

|  | **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 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **10678** | Air Asia | 9/04/2019 | Kolkata | Banglore | CCU → BLR | 19:55 | 22:25 | 2h 30m | non-stop | No info | 4107 |
| **10679** | Air India | 27/04/2019 | Kolkata | Banglore | CCU → BLR | 20:45 | 23:20 | 2h 35m | non-stop | No info | 4145 |
| **10680** | Jet Airways | 27/04/2019 | Banglore | Delhi | BLR → DEL | 08:20 | 11:20 | 3h | non-stop | No info | 7229 |
| **10681** | Vistara | 01/03/2019 | Banglore | New Delhi | BLR → DEL | 11:30 | 14:10 | 2h 40m | non-stop | No info | 12648 |
| **10682** | Air India | 9/05/2019 | Delhi | Cochin | DEL → GOI → BOM → COK | 10:55 | 19:15 | 8h 20m | 2 stops | No info | 11753 |

10683 rows × 11 columns

## Exploratory Data Analysis (EDA)

Now here we will be looking at the kind of columns our dataset has.

# **EDA**

1.Checking the missing values

2.checking for numerical columns

3.checking for the distribution of numerical variables

4.Checking for categorical variables

5.Types of categorical variables

detecting outliers

# **Checking the missing value**

In [12]:

missing\_value**=**[feature **for** feature **in** df.columns **if** df[feature].isnull().sum()**>**1]

In [13]:

missing\_value

Out[13]:

[]

There is no missing values present in the dataset

# **checking the number of columns**

In [10]:

df.columns

Out[10]:

Index(['Airline', 'Date\_of\_Journey', 'Source', 'Destination', 'Route',

'Dep\_Time', 'Arrival\_Time', 'Duration', 'Total\_Stops',

'Additional\_Info', 'Price'],

dtype='object')

In [11]:

df.nunique()

Out[11]:

Airline 12

Date\_of\_Journey 44

Source 5

Destination 6

Route 128

Dep\_Time 222

Arrival\_Time 1343

Duration 368

Total\_Stops 5

Additional\_Info 10

Price 1870

dtype: int64

In [40]:

df['Airline'].value\_counts()

Out[40]:

Jet Airways 3849

IndiGo 2053

Air India 1752

Multiple carriers 1196

SpiceJet 818

Vistara 479

Air Asia 319

GoAir 194

Multiple carriers Premium economy 13

Jet Airways Business 6

Vistara Premium economy 3

Trujet 1

Name: Airline, dtype: int64

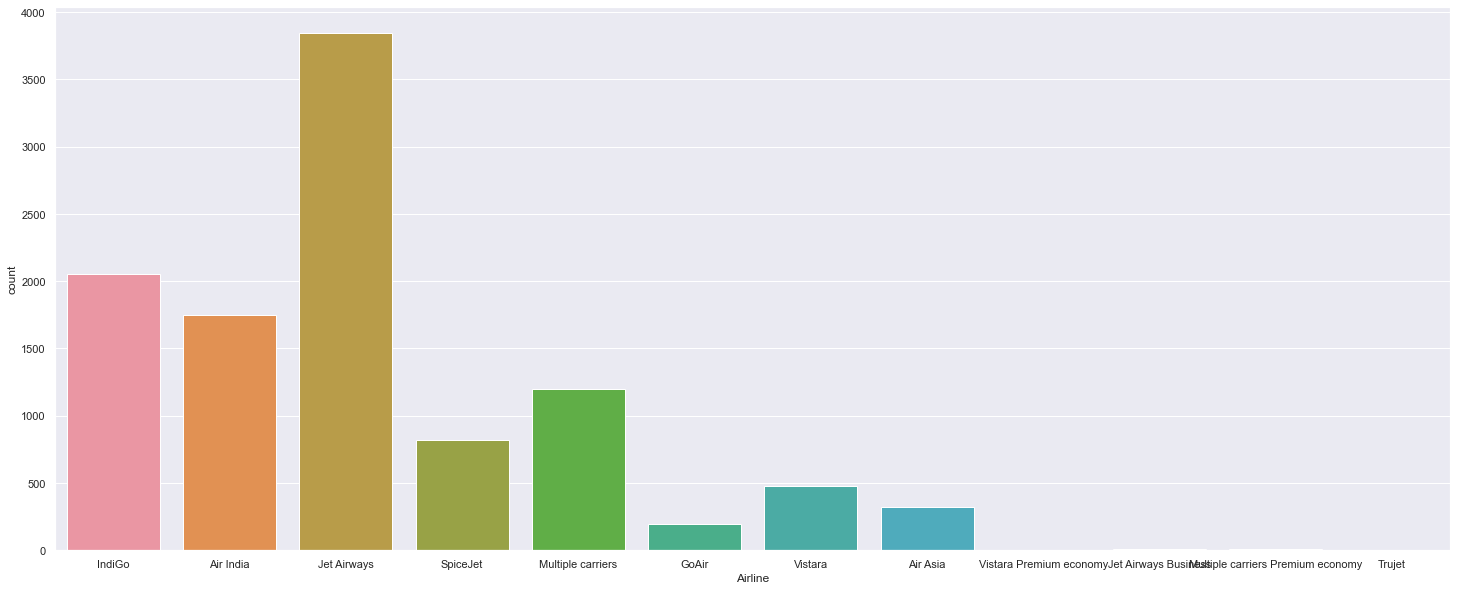
In [41]:

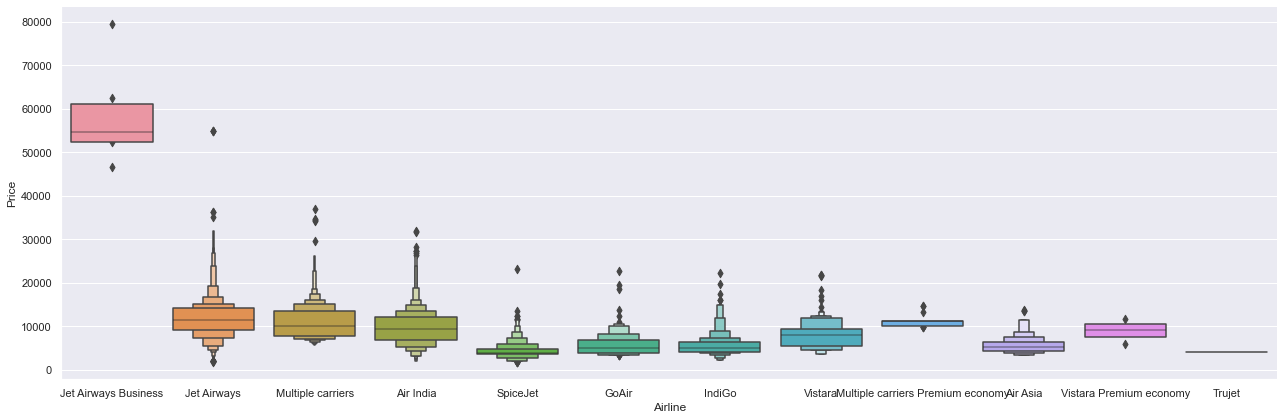
sns.set(rc**=**{'figure.figsize':(25,10)})

sns.countplot(df['Airline'])

Out[41]:

<AxesSubplot:xlabel='Airline', ylabel='count'>

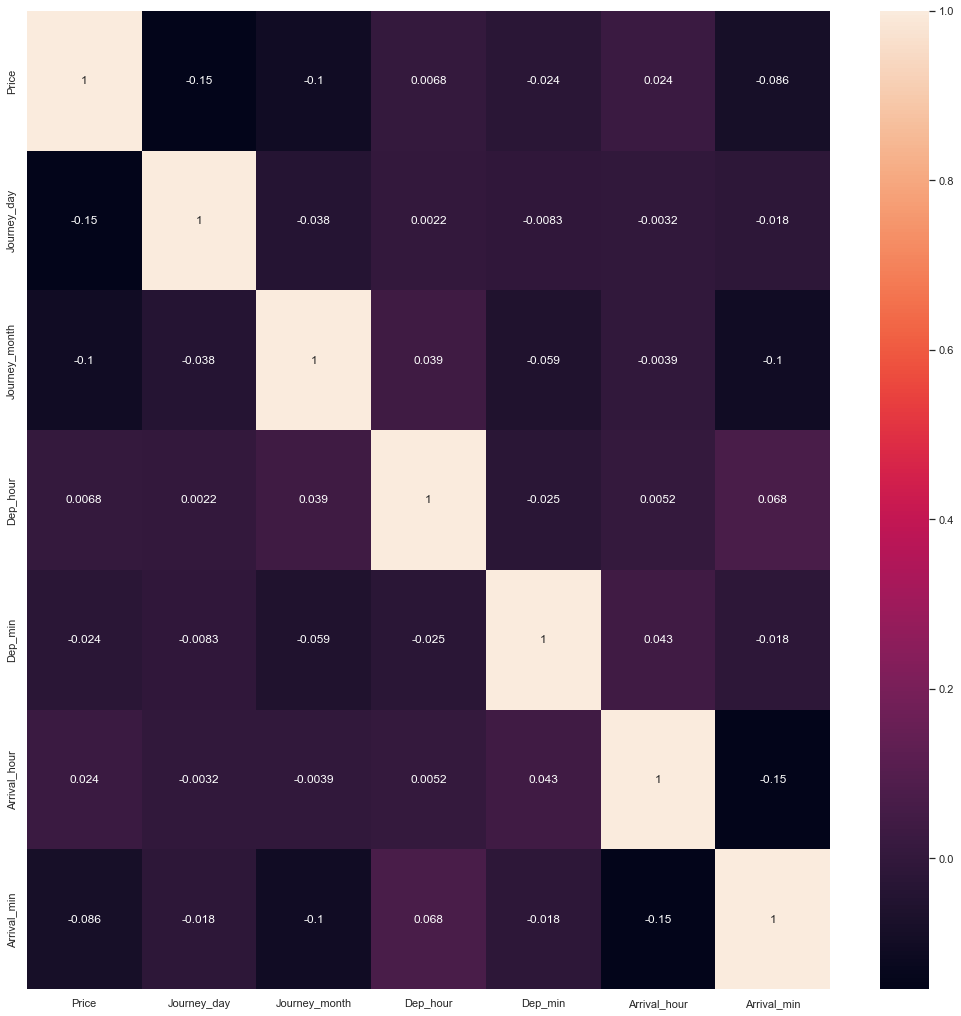




plt.figure(figsize = (18,18))

sns.heatmap(df.corr(),annot= True)

plt.show()



plt.figure(figsize=(20,8))

plt.subplot(1,2,1)

plt.title('Airline Histogram')

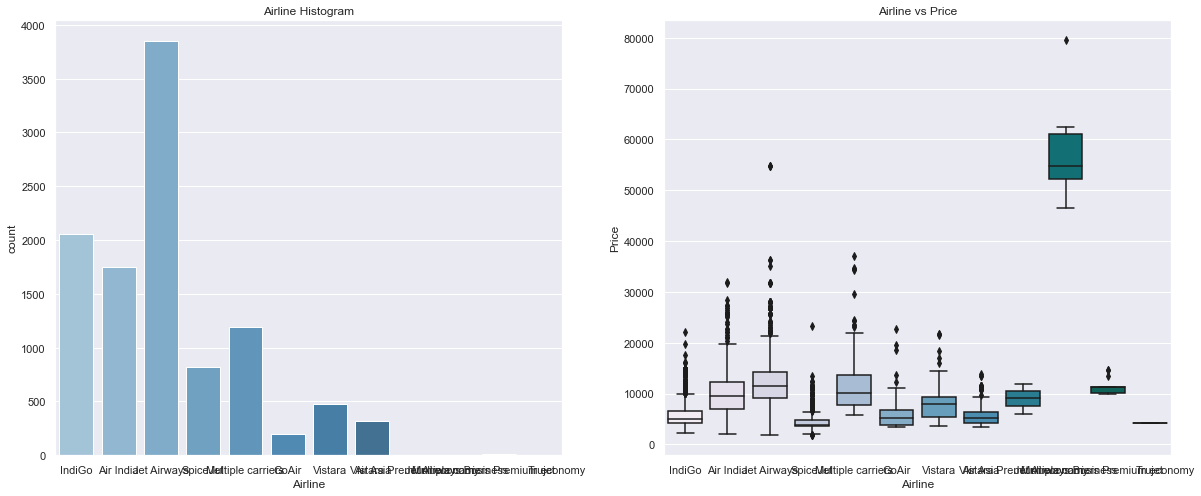
sns.countplot(df.Airline, palette=("Blues\_d"))

plt.subplot(1,2,2)

plt.title('Airline vs Price')

sns.boxplot(x=df.Airline, y=df.Price, palette=("PuBuGn"))

plt.show()



df.Dep\_Time **=** 'dept' **+** df.Dep\_Time

df.Arrival\_Time **=** 'arr' **+** df.Arrival\_Time

​

df.Source **=** 'source' **+** df.Source

df.Destination **=** 'destination' **+** df.Destination

​

In [73]:

df.drop(["Duration"], axis **=** 1, inplace **=** **True**)

df.drop(["Date\_of\_Journey"], axis **=** 1, inplace **=** **True**)

df.drop(["Dep\_Time"], axis **=** 1, inplace **=** **True**)

df.drop(["Arrival\_Time"], axis **=** 1, inplace **=** **True**)

In [74]:

df["Duration\_hours"] **=** duration\_hours

df["Duration\_mins"] **=** duration\_mins

​

In [75]:

Airline **=** df[["Airline"]]

​

Airline **=** pd.get\_dummies(Airline, drop\_first**=** **True**)

​

Airline.head()

Out[75]:

|  | **Airline\_Air India** | **Airline\_GoAir** | **Airline\_IndiGo** | **Airline\_Jet Airways** | **Airline\_Jet Airways Business** | **Airline\_Multiple carriers** | **Airline\_Multiple carriers Premium economy** | **Airline\_SpiceJet** | **Airline\_Trujet** | **Airline\_Vistara** | **Airline\_Vistara Premium economy** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **1** | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **2** | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **3** | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **4** | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# 

# 

# 

# **Detecting Multicollinearity with VIF**

In [103]:

**from** statsmodels.stats.outliers\_influence **import** variance\_inflation\_factor

vif\_data **=** pd.DataFrame()

vif\_data["feature"] **=** X.columns

*# calculating VIF for each feature*

vif\_data["VIF"] **=** [variance\_inflation\_factor(X.values, i)

**for** i **in** range(len(X.columns))]

print(vif\_data)

feature VIF

0 Total\_Stops 6.140181

1 Journey\_day 3.412960

2 Journey\_month 13.545735

3 Dep\_hour 5.589636

4 Dep\_min 2.713153

5 Arrival\_hour 4.834557

6 Arrival\_min 3.324426

7 Duration\_hours 6.192459

8 Duration\_mins 3.855595

9 Airline\_Air India 4.402039

10 Airline\_GoAir 1.359039

11 Airline\_IndiGo 4.141121

12 Airline\_Jet Airways 8.071890

13 Airline\_Jet Airways Business 1.008757

14 Airline\_Multiple carriers 3.070824

15 Airline\_Multiple carriers Premium economy 1.020174

16 Airline\_SpiceJet 2.246288

17 Airline\_Trujet 1.001439

18 Airline\_Vistara 1.866831

19 Airline\_Vistara Premium economy 1.003476

**Saving the Model**

import pickle

# open a file, where you ant to store the data

file = open('flight\_rf.pkl', 'wb')

# dump information to that file

pickle.dump(reg\_rf, file)

## Conclusion

So as we saw that we have done a complete EDA process, getting data insights, feature engineering, and data visualization as well so after all these steps one can go for the prediction using machine learning model-making steps.

Hope you liked my article on flight fare prediction using machine learning. If you have any opinions or questions, then comment below.