EDA And Feature Engineering Flight Price Prediction

Data Source : https://www.kaggle.com/datasets/shubhambathwal/flight-price-prediction (https://www.kaggle.com/datasets/shubhambathwal/flight-price-prediction)

FEATURES

The various features of the cleaned dataset are explained below:

1. Airline: The name of the airline company is stored in the airline column. It is a categorical feature having 6 different airlines.
2. Flight: Flight stores information regarding the plane's flight code. It is a categorical feature.
3. Source City: City from which the flight takes off. It is a categorical feature having 6 unique cities.
4. Departure Time: This is a derived categorical feature obtained created by grouping time periods into bins. It stores information about the departure time and have 6 unique time labels.
5. Stops: A categorical feature with 3 distinct values that stores the number of stops between the source and destination cities.
6. Arrival Time: This is a derived categorical feature created by grouping time intervals into bins. It has six distinct time labels and keeps information about the arrival time.
7. Destination City: City where the flight will land. It is a categorical feature having 6 unique cities.
8. Class: A categorical feature that contains information on seat class; it has two distinct values: Business and Economy.
9. Duration: A continuous feature that displays the overall amount of time it takes to travel between cities in hours.

10)Days Left: This is a derived characteristic that is calculated by subtracting the trip date by the booking date.

11. Price: Target variable stores information of the ticket price.

In [ ]:

```
#importing all important libararies

import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

In [ ]:

```
df=pd.read_excel("/content/sample_data/flight_price.xlsx")
```

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

Out[3]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration
0	IndiGo	24/03/2019	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m
1	Air India	1/05/2019	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m
2	Jet Airways	9/06/2019	Delhi	Cochin	DEL → LKO → BOM → COK	09:25	04:25 10 Jun	19h
3	IndiGo	12/05/2019	Kolkata	Banglore	CCU → NAG → BLR	18:05	23:30	5h 25m
4	IndiGo	01/03/2019	Banglore	New Delhi	BLR → NAG → DEL	16:50	21:35	4h 45m

In [ ]:
```
#get the basic info about dataset

df.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 [ ]:

```
# get the description of the dataset

df.describe()
```

Out[5]:

	Price
count	10683.000000
mean	9087.064121
std	4611.359167
min	1759.000000
25%	5277.000000
50%	8372.000000
75%	12373.000000
max	79512.000000

In [ ]:

```
#apply the feature engineering to each and every column that are require fo
```

In [ ]:

```
"""take the Date_of_Journey column and split the date in date, month, year""
```

Out[7]:

```
'take the Date_of_Journey column and split the date in date, month, year'
```

In [ ]:

```
df["Date"]=df["Date_of_Journey"].str.split("/").str[0]
df["Month"]=df["Date_of_Journey"].str.split("/").str[1]
df["Year"]=df["Date_of_Journey"].str.split("/").str[2]
```

In [ ]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10683 entries, 0 to 10682
Data columns (total 14 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 
 11  Date             10683 non-null  object
 12  Month            10683 non-null  object
 13  Year             10683 non-null  object
dtypes: int64(1), object(13)
memory usage: 1.1+ MB
```

In [ ]:

```
### change the type of Date, month and year column

df['Date']=df['Date'].astype(int)
df['Month']=df['Month'].astype(int)
df['Year']=df['Year'].astype(int)
```

In [ ]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10683 entries, 0 to 10682
Data columns (total 14 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 
 11  Date             10683 non-null  int64 
 12  Month            10683 non-null  int64 
 13  Year             10683 non-null  int64 
dtypes: int64(4), object(10)
memory usage: 1.1+ MB
```

In [ ]:

```
# Drop the Date_of_Journey column as we don't need anymore
```

In [ ]:

```
df.drop("Date_of_Journey", axis=1, inplace=True)
```

In [ ]:

```
df.head()
```

Out[14]:

	Airline	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Add
0	IndiGo	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	non-stop	
1	Air India	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	
2	Jet Airways	Delhi	Cochin	DEL → LKO → BOM → COK	09:25	04:25 10 Jun	19h	2 stops	
3	IndiGo	Kolkata	Banglore	CCU → NAG → BLR	18:05	23:30	5h 25m	1 stop	
4	IndiGo	Banglore	New Delhi	BLR → NAG → DEL	16:50	21:35	4h 45m	1 stop	

In [ ]:

```
#now take the Arrival_Time column
```

In [ ]:

```
df['Arrival_Time']=df['Arrival_Time'].apply(lambda x:x.split(' ')[0])
```

In [ ]:

```
df["Arrival_hour"]=df["Arrival_Time"].str.split(":").str[0]
df["Arrival_min"]=df["Arrival_Time"].str.split(":").str[1]
```

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

Out[20]:

	Airline	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Add
0	IndiGo	Banglore	New Delhi	BLR → DEL	22:20	01:10	2h 50m	non-stop	
1	Air India	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	
2	Jet Airways	Delhi	Cochin	DEL → LKO → BOM → COK	09:25	04:25	19h	2 stops	
3	IndiGo	Kolkata	Banglore	CCU → NAG → BLR	18:05	23:30	5h 25m	1 stop	
4	IndiGo	Banglore	New Delhi	BLR → NAG → DEL	16:50	21:35	4h 45m	1 stop	

In [ ]:
```
#change the datatype of  arrival hour and arrival min

df["Arrival_hour"]=df["Arrival_hour"].astype(int)
df["Arrival_min"]=df["Arrival_min"].astype(int)
```

In [ ]:
```
# now drop the Arrival_Time column

df.drop('Arrival_Time',axis=1,inplace=True)
```

In [ ]:
```
df.head(2)
```

Out[23]:

	Airline	Source	Destination	Route	Dep_Time	Duration	Total_Stops	Additional_Info	Pr
0	IndiGo	Banglore	New Delhi	BLR → DEL	22:20	2h 50m	non-stop	No info	38
1	Air India	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	7h 25m	2 stops	No info	76

In [ ]:

```
#now take the Dep_Time Column
#split the Dep_time into Departure_hour and Departure_Min
```

In [ ]:

```
df['Departure_hour']=df['Dep_Time'].str.split(':').str[0]
```

In [ ]:

```
df['Departure_min']=df['Dep_Time'].str.split(':').str[1]
```

In [ ]:

```
#now change the type of data

df["Departure_hour"]=df["Departure_hour"].astype(int)
df["Departure_min"]=df["Departure_min"].astype(int)
```

In [ ]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10683 entries, 0 to 10682
Data columns (total 16 columns):
 #   Column           Non-Null Count  Dtype 
---  ------           --------------  ----- 
 0   Airline          10683 non-null  object
 1   Source           10683 non-null  object
 2   Destination      10683 non-null  object
 3   Route            10682 non-null  object
 4   Dep_Time         10683 non-null  object
 5   Duration         10683 non-null  object
 6   Total_Stops      10682 non-null  object
 7   Additional_Info  10683 non-null  object
 8   Price            10683 non-null  int64 
 9   Date             10683 non-null  int64 
 10  Month            10683 non-null  int64 
 11  Year             10683 non-null  int64 
 12  Arrival_hour     10683 non-null  int64 
 13  Arrival_min      10683 non-null  int64 
 14  Departure_hour   10683 non-null  int64 
 15  Departure_min    10683 non-null  int64 
dtypes: int64(8), object(8)
memory usage: 1.3+ MB
```

In [ ]:

```
# now drop the Dep_Time column


df.drop('Dep_Time',axis=1,inplace=True)
```

In [ ]:

```
df.head(2)
```

Out[30]:

	Airline	Source	Destination	Route	Duration	Total_Stops	Additional_Info	Price	Date	I
0	IndiGo	Banglore	New Delhi	BLR → DEL	2h 50m	non-stop	No info	3897	24	
1	Air India	Kolkata	Banglore	CCU → IXR → BBI → BLR	7h 25m	2 stops	No info	7662	1	

In [ ]:

```
# now find the unique values of Total_Stops

df['Total_Stops'].unique()
```

Out[31]:

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

In [ ]:

```
#Find the null value within the Total_Stops

df[df["Total_Stops"].isnull()]
```

Out[33]:

	Airline	Source	Destination	Route	Duration	Total_Stops	Additional_Info	Price	Date
9039	Air India	Delhi	Cochin	NaN	23h 40m	NaN	No info	7480	6

In [ ]:

```
df['Total_Stops'].mode()
```

Out[34]:

```
0    1 stop
Name: Total_Stops, dtype: object
```

In [ ]:

```
df['Total_Stops'].unique()
```

Out[35]:

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

In [ ]:

```
##Now replace all these value with 0,1,2,3,4
#repalce non-stop=0
#replace 1stop=1,
#replace  2stop=2,
#replace  3stop=3,
#replace  4stop=4,
#replace  nan= 1
```

In [ ]:

```
df['Total_Stops']=df['Total_Stops'].map({'non-stop':0,'1 stop':1,'2 stops':
```

In [ ]:

```
df[df['Total_Stops'].isnull()]  #no nan value within the Total_Stops column
```

Out[38]:

	Airline	Source	Destination	Route	Duration	Total_Stops	Additional_Info	Price	Date	Mo

In [ ]:
```
df.head(2)
```

Out[39]:

	Airline	Source	Destination	Route	Duration	Total_Stops	Additional_Info	Price	Date	I
0	IndiGo	Banglore	New Delhi	BLR → DEL	2h 50m	0	No info	3897	24	
1	Air India	Kolkata	Banglore	CCU → IXR → BBI → BLR	7h 25m	2	No info	7662	1	

In [ ]:
```
# now we can see that we don't need Route column
```

In [ ]:
```
df.drop('Route',axis=1,inplace=True)
```

In [ ]:
```
df.head(2)
```

Out[42]:

	Airline	Source	Destination	Duration	Total_Stops	Additional_Info	Price	Date	Month
0	IndiGo	Banglore	New Delhi	2h 50m	0	No info	3897	24	3
1	Air India	Kolkata	Banglore	7h 25m	2	No info	7662	1	5

In [ ]:
```
# now take the Duration Column

df['Duration_hour']=df['Duration'].str.split('h').str[0]
df['Duration_min']=df['Duration'].str.split('m').str[1]
```

In [ ]:
```
df.head(2)
```

Out[44]:

	Airline	Source	Destination	Duration	Total_Stops	Additional_Info	Price	Date	Month
0	IndiGo	Banglore	New Delhi	2h 50m	0	No info	3897	24	3
1	Air India	Kolkata	Banglore	7h 25m	2	No info	7662	1	5

In [ ]:
```
# drop the Duration Column

df.drop("Duration", axis=1, inplace=True)
```

In [ ]:
```
df.head(2)
```

Out[46]:

	Airline	Source	Destination	Total_Stops	Additional_Info	Price	Date	Month	Year	Arri
0	IndiGo	Banglore	New Delhi	0	No info	3897	24	3	2019	
1	Air India	Kolkata	Banglore	2	No info	7662	1	5	2019	

In [ ]:
```
#now find the Unique  values within the airline, Source, Additional_Info co

df["Airline"].unique()
```

Out[47]:
```
array(['IndiGo', 'Air India', 'Jet Airways', 'SpiceJet',
       'Multiple carriers', 'GoAir', 'Vistara', 'Air Asia',
       'Vistara Premium economy', 'Jet Airways Business',
       'Multiple carriers Premium economy', 'Trujet'], dtype=object)
```

In [ ]:
```
df['Source'].unique()
```

Out[48]:
```
array(['Banglore', 'Kolkata', 'Delhi', 'Chennai', 'Mumbai'], dtype=object)
```

In [ ]:
```
df['Additional_Info'].unique()
```

Out[49]:
```
array(['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'], dtype=object)
```

In [ ]:
```
df['Destination'].unique()
```

Out[50]:
```
array(['New Delhi', 'Banglore', 'Cochin', 'Kolkata', 'Delhi', 'Hyderaba
d'],
      dtype=object)
```

In [ ]:
```
# now we can procced with the categorical columns

#apply the OneHotEncoder technique
```

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

In [ ]:
```
encoder=OneHotEncoder()
```

In [ ]:
```
encoder.fit_transform(df[['Airline','Source','Destination']]).toarray()
```

Out[54]:
```
array([[0., 0., 0., ..., 0., 0., 1.],
       [0., 1., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 1.],
       [0., 1., 0., ..., 0., 0., 0.]])
```

In [ ]:

```
pd.DataFrame(encoder.fit_transform(df[['Airline','Source','Destination']]).
```

Out[55]:

	Airline_Air Asia	Airline_Air India	Airline_GoAir	Airline_IndiGo	Airline_Jet Airways	Airline_Jet Airways Business	Airline_M c
0	0.0	0.0	0.0	1.0	0.0	0.0	
1	0.0	1.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	0.0	0.0	1.0	0.0	
3	0.0	0.0	0.0	1.0	0.0	0.0	
4	0.0	0.0	0.0	1.0	0.0	0.0	
...	...	...	...	...	...	...	
10678	1.0	0.0	0.0	0.0	0.0	0.0	
10679	0.0	1.0	0.0	0.0	0.0	0.0	
10680	0.0	0.0	0.0	0.0	1.0	0.0	
10681	0.0	0.0	0.0	0.0	0.0	0.0	
10682	0.0	1.0	0.0	0.0	0.0	0.0	

10683 rows × 23 columns

In [ ]:

```
df.head()
```

Out[56]:

	Airline	Source	Destination	Total_Stops	Additional_Info	Price	Date	Month	Year	Ar
0	IndiGo	Banglore	New Delhi	0	No info	3897	24	3	2019	
1	Air India	Kolkata	Banglore	2	No info	7662	1	5	2019	
2	Jet Airways	Delhi	Cochin	2	No info	13882	9	6	2019	
3	IndiGo	Kolkata	Banglore	1	No info	6218	12	5	2019	
4	IndiGo	Banglore	New Delhi	1	No info	13302	1	3	2019	

In [ ]: