EDA And Feature Engineering Flight Price Prediction

check the dataset info below https://www.kaggle.com/datasets/shubhambathwal/flight-price-prediction

```
# importing basics libraries
import pandas as  pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

```
df=pd.read_excel('Flight-price.xlsx')
df.head()
```

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total
0	IndiGo	24/03/2019	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	nc
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	

```
df.tail()
```

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration
10678	Air Asia	9/04/2019	Kolkata	Banglore	CCU → BLR	19:55	22:25	2h 30m
10679	Air India	27/04/2019	Kolkata	Banglore	CCU → BLR	20:45	23:20	2h 35m
10680	Jet Airways	27/04/2019	Banglore	Delhi	BLR → DEL	08:20	11:20	3h
10681	Vistara	01/03/2019	Banglore	New Delhi	BLR → DEL	11:30	14:10	2h 40m
10682	Air India	9/05/2019	Delhi	Cochin	DEL → GOI → BOM → COK	10:55	19:15	8h 20m

```
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
```

```
df.describe()
```

	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

```
df.head()
```

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total
0	IndiGo	24/03/2019	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	nc
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	

```
# demo only for practices
# df['split_Date']=df['Date_of_Journey'].str.split()
# df['split_Date_remove(/)']=df['Date_of_Journey'].str.split('/')
# df['Year]=df['Date_of_Journey'].str.split('/').str[2]
```

```
# df['split-Date']=df['Date_of_Journey'].str.split('/')
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]
print(df['Year'])
```

```
0        2019
1        2019
2        2019
3        2019
4        2019
         ...
10678    2019
10679    2019
10680    2019
10681    2019
10682    2019
Name: Year, Length: 10683, dtype: object
```

```
df.head(3)
```

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total
0	IndiGo	24/03/2019	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	nc
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	

```
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
```

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

```
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
```

```
# Drop Date of journey
df.drop('Date_of_Journey',axis=1,inplace=True)
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10683 entries, 0 to 10682
Data columns (total 13 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   Arrival_Time     10683 non-null  object
 6   Duration         10683 non-null  object
 7   Total_Stops      10682 non-null  object
 8   Additional_Info  10683 non-null  object
 9   Price            10683 non-null  int64 
 10  Date             10683 non-null  int64 
 11  Month            10683 non-null  int64 
 12  Year             10683 non-null  int64 
dtypes: int64(4), object(9)
memory usage: 1.1+ MB
```

```
df.head(3)
```

	Airline	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additiona
0	IndiGo	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	non-stop	N
1	Air India	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	N
2	Jet Airways	Delhi	Cochin	DEL → LKO → BOM → COK	09:25	04:25 10 Jun	19h	2 stops	N

```
# use can this technique drop the Date from time
df['Arrival_Time']=df['Arrival_Time'].apply(lambda x:x.split(' ')[0])
print(df['Arrival_Time'])
```

```
0        01:10
1        13:15
2        04:25
3        23:30
4        21:35
         ...
10678    22:25
10679    23:20
10680    11:20
10681    14:10
10682    19:15
Name: Arrival_Time, Length: 10683, dtype: object
```

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

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

	Airline	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional
0	IndiGo	Banglore	New Delhi	BLR → DEL	22:20	01:10	2h 50m	non-stop	N
1	Air India	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	N

```
df['Arrival_hours']=df['Arrival_hours'].astype(int)
df['Arrival_min']=df['Arrival_min'].astype(int)
```

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

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

	Airline	Source	Destination	Route	Dep_Time	Duration	Total_Stops	Additional_Info	Price	I
0	IndiGo	Banglore	New Delhi	BLR → DEL	22:20	2h 50m	non-stop	No info	3897	
1	Air India	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	7h 25m	2 stops	No info	7662	

```
df['Departure_time']=df['Dep_Time'].str.split(':').str[0]
df['Departure_min']=df['Dep_Time'].str.split(':').str[1]
```

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

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

	Airline	Source	Destination	Route	Duration	Total_Stops	Additional_Info	Price	Date	Mont
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	

```
df['Departure_time']=df['Departure_time'].astype(int)
df['Departure_min']=df['Departure_min'].astype(int)
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10683 entries, 0 to 10682
Data columns (total 15 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   Duration         10683 non-null  object
 5   Total_Stops      10682 non-null  object
 6   Additional_Info  10683 non-null  object
 7   Price            10683 non-null  int64 
 8   Date             10683 non-null  int64 
 9   Month            10683 non-null  int64 
 10  Year             10683 non-null  int64 
 11  Arrival_hours    10683 non-null  int64 
 12  Arrival_min      10683 non-null  int64 
 13  Departure_time   10683 non-null  int64 
 14  Departure_min    10683 non-null  int64 
dtypes: int64(8), object(7)
memory usage: 1.2+ MB
```

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

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

```
# df['Total_Stops'].isnull().sum()
df[df['Total_Stops'].isnull()]
```

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

```
df['Total_Stops'].mode()  #mode use for find the maximum value
```

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

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

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

```
df['Total_Stops']=df['Total_Stops'].map({'non-stop':0, '2 stops':2, '1 stop':1, '3 stops':3
```

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

	Airline	Source	Destination	Route	Duration	Total_Stops	Additional_Info	Price	Date	Month

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

	Airline	Source	Destination	Route	Duration	Total_Stops	Additional_Info	Price	Date	Mont
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	

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

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

	Airline	Source	Destination	Duration	Total_Stops	Additional_Info	Price	Date	Month	Year
0	IndiGo	Banglore	New Delhi	2h 50m	0	No info	3897	24	3	2019
1	Air India	Kolkata	Banglore	7h 25m	2	No info	7662	1	5	2019

```
df['Duration-hourse']=df['Duration'].str.split(' ').str[0]
print(df['Duration-hourse'])
```

```
0          2h
1          7h
2         19h
3          5h
4          4h
         ...
10678      2h
10679      2h
10680      3h
10681      2h
10682      8h
Name: Duration-hourse, Length: 10683, dtype: object
```

```
df['Duration-hourse']=df['Duration'].str.split(' ').str[0].str.split('h').str[0]
print(df['Duration-hourse'])
```

```
0         2
1         7
2        19
3         5
4         4
        ..
10678     2
10679     2
10680     3
10681     2
10682     8
Name: Duration-hourse, Length: 10683, dtype: object
```

```
df['Duration-min']=df['Duration'].str.split(" ").str[1].str.split('m').str[0]
print(df['Duration-min'])
```

```
0         50
1         25
2        NaN
3         25
4         45
        ...
10678     30
10679     35
10680    NaN
10681     40
10682     20
Name: Duration-min, Length: 10683, dtype: object
```

```
df['Duration-min'] = df['Duration-min'].fillna(0).astype(int)
```

```
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   Duration         10683 non-null  object
 4   Total_Stops      10683 non-null  int64 
 5   Additional_Info  10683 non-null  object
 6   Price            10683 non-null  int64 
 7   Date             10683 non-null  int64 
 8   Month            10683 non-null  int64 
 9   Year             10683 non-null  int64 
 10  Arrival_hours    10683 non-null  int64 
 11  Arrival_min      10683 non-null  int64 
 12  Departure_time   10683 non-null  int64 
 13  Departure_min    10683 non-null  int64 
 14  Duration-hourse  10683 non-null  object
 15  Duration-min     10683 non-null  int64 
dtypes: int64(10), object(6)
memory usage: 1.3+ MB
```

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

	Airline	Source	Destination	Duration	Total_Stops	Additional_Info	Price	Date	Month	Year
0	IndiGo	Banglore	New Delhi	2h 50m	0	No info	3897	24	3	2019
1	Air India	Kolkata	Banglore	7h 25m	2	No info	7662	1	5	2019

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

```
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)
```

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

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

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

```
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)
```

##from sklearn.preprocessing import OneHotEncoder ##

use to convert categorical data into numerical format for machine learning models.

```
from sklearn.preprocessing import OneHotEncoder
```

```
encoder=OneHotEncoder()
```

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

```
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.]], shape=(10683, 23))
```

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

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

10683 rows × 23 columns