

# flight

March 21, 2025

Importing Libraries

```
[1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.preprocessing import LabelEncoder
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score, mean_squared_error, mean_absolute_error
from sklearn.model_selection import train_test_split
```

Loading Data

```
[14]: fp=pd.read_excel(r"C:\Users\Megha I Angadi\Downloads\archive\Data_Train.xlsx")
```

```
[15]: fp.head()
```

```
[15]:
```

	Airline	Date_of_Journey	Source	Destination	Route	\
0	IndiGo	24/03/2019	Banglore	New Delhi	BLR → DEL	
1	Air India	1/05/2019	Kolkata	Banglore	CCU → IXR → BBI → BLR	
2	Jet Airways	9/06/2019	Delhi	Cochin	DEL → LKO → BOM → COK	
3	IndiGo	12/05/2019	Kolkata	Banglore	CCU → NAG → BLR	
4	IndiGo	01/03/2019	Banglore	New Delhi	BLR → NAG → DEL	

	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
0	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897
1	05:50	13:15	7h 25m	2 stops	No info	7662
2	09:25	04:25 10 Jun	19h	2 stops	No info	13882
3	18:05	23:30	5h 25m	1 stop	No info	6218
4	16:50	21:35	4h 45m	1 stop	No info	13302

Clean Data

```
[16]: fp.isnull().sum()
```

```
[16]: 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

```

```
[17]: fp.dropna(inplace=True)
```

```
[18]: fp.isnull().sum()
```

```

[18]: 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

```

## Data Preprocessing

```

[22]: En=LabelEncoder()
fp["Airline_encode"]=En.fit_transform(fp["Airline"])
fp.head()

```

```

[22]:   Airline Date_of_Journey  Source Destination      Route \
0    IndiGo    24/03/2019  Bangalore   New Delhi      BLR → DEL
1  Air India    1/05/2019  Kolkata    Bangalore  CCU → IXR → BBI → BLR
2  Jet Airways  9/06/2019    Delhi      Cochin   DEL → LKO → BOM → COK
3    IndiGo   12/05/2019  Kolkata    Bangalore  CCU → NAG → BLR
4    IndiGo    01/03/2019  Bangalore   New Delhi  BLR → NAG → DEL

```

```

   Dep_Time  Arrival_Time  Duration  Total_Stops  Additional_Info  Price \
0    22:20    01:10 22 Mar    2h 50m    non-stop          No info   3897
1    05:50           13:15    7h 25m      2 stops          No info   7662
2    09:25    04:25 10 Jun      19h      2 stops          No info  13882
3    18:05           23:30    5h 25m      1 stop          No info   6218
4    16:50           21:35    4h 45m      1 stop          No info  13302

```

```

   Airline_encode
0                3

```

1	1
2	4
3	3
4	3

```
[23]: En1=LabelEncoder()
fp["Source_encode"]=En1.fit_transform(fp["Source"])
fp.head()
```

```
[23]:      Airline Date_of_Journey  Source Destination      Route \
0      IndiGo      24/03/2019  Bangalore  New Delhi      BLR → DEL
1      Air India      1/05/2019  Kolkata    Bangalore  CCU → IXR → BBI → BLR
2      Jet Airways      9/06/2019    Delhi      Cochin  DEL → LKO → BOM → COK
3      IndiGo      12/05/2019  Kolkata    Bangalore      CCU → NAG → BLR
4      IndiGo      01/03/2019  Bangalore  New Delhi      BLR → NAG → DEL
```

	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price	\
0	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897	
1	05:50	13:15	7h 25m	2 stops	No info	7662	
2	09:25	04:25 10 Jun	19h	2 stops	No info	13882	
3	18:05	23:30	5h 25m	1 stop	No info	6218	
4	16:50	21:35	4h 45m	1 stop	No info	13302	

	Airline_encode	Source_encode
0	3	0
1	1	3
2	4	2
3	3	3
4	3	0

```
[24]: En2=LabelEncoder()
fp["Destination_encode"]=En2.fit_transform(fp["Destination"])
fp.head()
```

```
[24]:      Airline Date_of_Journey  Source Destination      Route \
0      IndiGo      24/03/2019  Bangalore  New Delhi      BLR → DEL
1      Air India      1/05/2019  Kolkata    Bangalore  CCU → IXR → BBI → BLR
2      Jet Airways      9/06/2019    Delhi      Cochin  DEL → LKO → BOM → COK
3      IndiGo      12/05/2019  Kolkata    Bangalore      CCU → NAG → BLR
4      IndiGo      01/03/2019  Bangalore  New Delhi      BLR → NAG → DEL
```

	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price	\
0	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897	
1	05:50	13:15	7h 25m	2 stops	No info	7662	
2	09:25	04:25 10 Jun	19h	2 stops	No info	13882	
3	18:05	23:30	5h 25m	1 stop	No info	6218	
4	16:50	21:35	4h 45m	1 stop	No info	13302	

	Airline_encode	Source_encode	Destination_encode
0	3	0	5
1	1	3	0
2	4	2	1
3	3	3	0
4	3	0	5

split- ind,dep

```
[26]: X=fp[["Airline_encode","Source_encode","Destination_encode"]]
      Y=fp["Price"]
```

Split- train and test

```
[31]: X_train,X_test,Y_train,Y_test=train_test_split(X,Y,test_size=0.
      ↪2,random_state=42)
```

```
[32]: price_m=LinearRegression()
      price_m.fit(X_train,Y_train)
```

```
[32]: LinearRegression()
```

Test

```
[33]: Air=input("Enter the Airline:")
      Source=input("Enter your source:")
      Dest=input("Enter your destination:")
      price=float(input("Enter the price:"))
```

```
Enter the Airline: Air India
Enter your source: Bangalore
Enter your destination: New Delhi
Enter the price: 3897
```

```
[45]: A_e=En.transform([Air])[0]
      S_e=En1.transform([Source])[0]
      D_e=En2.transform([Dest])[0]
      print(A_e,S_e,D_e,price)
```

```
1 0 5 3897.0
```

```
[46]: result=price_m.predict([[A_e,S_e,D_e]])
      print("The predicted price is:",result[0])
```

```
The predicted price is: 8485.826304199465
```

```
C:\Users\Megha I Angadi\anaconda3\Lib\site-packages\sklearn\base.py:493:
UserWarning: X does not have valid feature names, but LinearRegression was
```

```
fitted with feature names
warnings.warn(
```

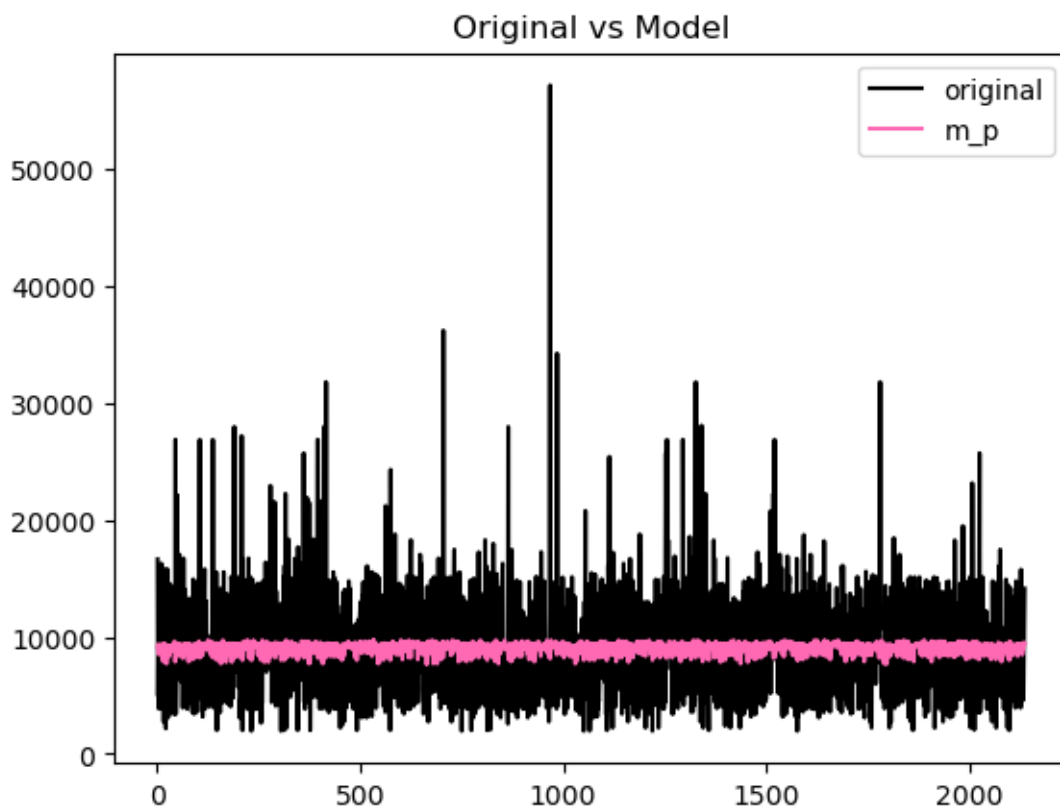
```
Evaluation
```

```
[47]: m_p=price_m.predict(X_test)
```

```
[48]: len(Y_test)
```

```
[48]: 2137
```

```
[55]: plt.plot(np.arange(1,2138),Y_test,color='k',label="original")
plt.plot(np.arange(1,2138),m_p,color="hotpink",label="m_p")
plt.title("Original vs Model")
plt.legend()
plt.show()
```



```
[60]: r2score=r2_score(Y_test,m_p)
print(r2score)
if r2score>0.5:
    print("Model is good fit")
else:
```

```
print("Model is not good fit")
```

0.0019270502637867004

Model is not good fit

MSE

```
[57]: mse=mean_squared_error(Y_test,m_p)
      print(mse)
```

21520505.157834526

MAE

```
[58]: mae=mean_absolute_error(Y_test,m_p)
      print(mae)
```

3627.47729797142

```
[ ]:
```