MINI PROJECT - 2

Problem Statement:Which model is suitable bestfor Flight price Prediction Dataset

In [1]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.linear_model import LogisticRegression
```

Data collection

In [2]:

```
train_df=pd.read_csv(r"C:\Users\91950\Downloads\Data_Train11.csv")
train_df
```

Out[2]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Dura
0	IndiGo	24/03/2019	Banglore	New Delhi	BLR ? DEL	22:20	01:10 22 Mar	2h
1	Air India	1/05/2019	Kolkata	Banglore	CCU ? IXR ? BBI ? BLR	05:50	13:15	7h
2	Jet Airways	9/06/2019	Delhi	Cochin	DEL ? LKO ? BOM ? COK	09:25	04:25 10 Jun	
3	IndiGo	12/05/2019	Kolkata	Banglore	CCU ? NAG ? BLR	18:05	23:30	5h
4	IndiGo	01/03/2019	Banglore	New Delhi	BLR ? NAG ? DEL	16:50	21:35	4h
...	...	...	...	...	...	...	...	
10678	Air Asia	9/04/2019	Kolkata	Banglore	CCU ? BLR	19:55	22:25	2h
10679	Air India	27/04/2019	Kolkata	Banglore	CCU ? BLR	20:45	23:20	2h
10680	Jet Airways	27/04/2019	Banglore	Delhi	BLR ? DEL	08:20	11:20	
10681	Vistara	01/03/2019	Banglore	New Delhi	BLR ? DEL	11:30	14:10	2h
10682	Air India	9/05/2019	Delhi	Cochin	DEL ? GOI ? BOM ? COK	10:55	19:15	8h

10683 rows × 11 columns

In [3]:

```
test_df=pd.read_csv(r"C:\Users\91950\Downloads\Test_set22.csv")
test_df
```

Out[3]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Durat
0	Jet Airways	6/06/2019	Delhi	Cochin	DEL ? BOM ? COK	17:30	04:25 07 Jun	10h 5
1	IndiGo	12/05/2019	Kolkata	Banglore	CCU ? MAA ? BLR	06:20	10:20	
2	Jet Airways	21/05/2019	Delhi	Cochin	DEL ? BOM ? COK	19:15	19:00 22 May	23h 4
3	Multiple carriers	21/05/2019	Delhi	Cochin	DEL ? BOM ? COK	08:00	21:00	
4	Air Asia	24/06/2019	Banglore	Delhi	BLR ? DEL	23:55	02:45 25 Jun	2h 5
...	...	...	...	...	...	...	...	
2666	Air India	6/06/2019	Kolkata	Banglore	CCU ? DEL ? BLR	20:30	20:25 07 Jun	23h 5
2667	IndiGo	27/03/2019	Kolkata	Banglore	CCU ? BLR	14:20	16:55	2h 3
2668	Jet Airways	6/03/2019	Delhi	Cochin	DEL ? BOM ? COK	21:50	04:25 07 Mar	6h 3
2669	Air India	6/03/2019	Delhi	Cochin	DEL ? BOM ? COK	04:00	19:15	15h 1
2670	Multiple carriers	15/06/2019	Delhi	Cochin	DEL ? BOM ? COK	04:55	19:15	14h 2

2671 rows × 10 columns

Data cleaning and preproceesing

In [4]:

```
train_df.shape
```

Out[4]:

```
(10683, 11)
```

In [5]:

```
test_df.shape
```

Out[5]:

```
(2671, 10)
```

In [6]:

```
train_df.describe
```

Out[6]:

```
<bound method NDFrame.describe of            Airline Date_of_Journey    So
urce Destination
0            IndiGo      24/03/2019  Banglore   New Delhi  \
1         Air India       1/05/2019   Kolkata    Banglore
2       Jet Airways       9/06/2019     Delhi      Cochin
3            IndiGo      12/05/2019   Kolkata    Banglore
4            IndiGo      01/03/2019  Banglore   New Delhi
...             ...             ...       ...         ...
10678      Air Asia       9/04/2019   Kolkata    Banglore
10679     Air India      27/04/2019   Kolkata    Banglore
10680   Jet Airways      27/04/2019  Banglore       Delhi
10681       Vistara      01/03/2019  Banglore   New Delhi
10682     Air India       9/05/2019     Delhi      Cochin

                       Route Dep_Time  Arrival_Time Duration Total_Stops
0                BLR ? DEL    22:20  01:10 22 Mar   2h 50m    non-stop
\
1      CCU ? IXR ? BBI ? BLR    05:50         13:15   7h 25m     2 stops
2      DEL ? LKO ? BOM ? COK    09:25  04:25 10 Jun      19h     2 stops
3            CCU ? NAG ? BLR    18:05         23:30   5h 25m      1 stop
4            BLR ? NAG ? DEL    16:50         21:35   4h 45m      1 stop
...                      ...      ...           ...      ...         ...
10678              CCU ? BLR    19:55         22:25   2h 30m    non-stop
10679              CCU ? BLR    20:45         23:20   2h 35m    non-stop
10680              BLR ? DEL    08:20         11:20       3h    non-stop
10681              BLR ? DEL    11:30         14:10   2h 40m    non-stop
10682  DEL ? GOI ? BOM ? COK    10:55         19:15   8h 20m     2 stops

      Additional_Info  Price
0             No info   3897
1             No info   7662
2             No info  13882
3             No info   6218
4             No info  13302
...               ...    ...
10678         No info   4107
10679         No info   4145
10680         No info   7229
10681         No info  12648
10682         No info  11753

[10683 rows x 11 columns]>
```

In [7]:

```
test_df.describe
```

Out[7]:

```
<bound method NDFrame.describe of                 Airline Date_of_Journey    
Source Destination
0           Jet Airways       6/06/2019     Delhi      Cochin  \
1                IndiGo      12/05/2019   Kolkata    Banglore   
2           Jet Airways      21/05/2019     Delhi      Cochin   
3     Multiple carriers      21/05/2019     Delhi      Cochin   
4              Air Asia      24/06/2019  Banglore       Delhi   
...                 ...             ...       ...         ...   
2666          Air India       6/06/2019   Kolkata    Banglore   
2667             IndiGo      27/03/2019   Kolkata    Banglore   
2668        Jet Airways       6/03/2019     Delhi      Cochin   
2669          Air India       6/03/2019     Delhi      Cochin   
2670  Multiple carriers      15/06/2019     Delhi      Cochin   

                Route Dep_Time  Arrival_Time Duration Total_Stops  \
0     DEL ? BOM ? COK    17:30  04:25 07 Jun  10h 55m      1 stop   
1     CCU ? MAA ? BLR    06:20         10:20       4h      1 stop   
2     DEL ? BOM ? COK    19:15  19:00 22 May  23h 45m      1 stop   
3     DEL ? BOM ? COK    08:00         21:00      13h      1 stop   
4           BLR ? DEL    23:55  02:45 25 Jun   2h 50m    non-stop   
...               ...      ...           ...      ...         ...   
2666  CCU ? DEL ? BLR    20:30  20:25 07 Jun  23h 55m      1 stop   
2667        CCU ? BLR    14:20         16:55   2h 35m    non-stop   
2668  DEL ? BOM ? COK    21:50  04:25 07 Mar   6h 35m      1 stop   
2669  DEL ? BOM ? COK    04:00         19:15  15h 15m      1 stop   
2670  DEL ? BOM ? COK    04:55         19:15  14h 20m      1 stop   

                Additional_Info  
0                       No info  
1                       No info  
2     In-flight meal not included  
3                       No info  
4                       No info  
...                         ...  
2666                    No info  
2667                    No info  
2668                    No info  
2669                    No info  
2670                    No info  

[2671 rows x 10 columns]>
```

In [8]:

```
train_df.head
```

Out[8]:

```
<bound method NDFrame.head of            Airline Date_of_Journey    Source 
Destination
0            IndiGo      24/03/2019  Banglore   New Delhi  \
1         Air India       1/05/2019   Kolkata    Banglore
2       Jet Airways       9/06/2019     Delhi      Cochin
3            IndiGo      12/05/2019   Kolkata    Banglore
4            IndiGo      01/03/2019  Banglore   New Delhi
...             ...             ...       ...         ...
10678      Air Asia       9/04/2019   Kolkata    Banglore
10679     Air India      27/04/2019   Kolkata    Banglore
10680   Jet Airways      27/04/2019  Banglore       Delhi
10681       Vistara      01/03/2019  Banglore   New Delhi
10682     Air India       9/05/2019     Delhi      Cochin

                       Route Dep_Time  Arrival_Time Duration Total_Stops 
0                  BLR ? DEL    22:20  01:10 22 Mar   2h 50m    non-stop
\
1      CCU ? IXR ? BBI ? BLR    05:50         13:15   7h 25m     2 stops
2      DEL ? LKO ? BOM ? COK    09:25  04:25 10 Jun      19h     2 stops
3            CCU ? NAG ? BLR    18:05         23:30   5h 25m      1 stop
4            BLR ? NAG ? DEL    16:50         21:35   4h 45m      1 stop
...                      ...      ...           ...      ...         ...
10678              CCU ? BLR    19:55         22:25   2h 30m    non-stop
10679              CCU ? BLR    20:45         23:20   2h 35m    non-stop
10680              BLR ? DEL    08:20         11:20       3h    non-stop
10681              BLR ? DEL    11:30         14:10   2h 40m    non-stop
10682  DEL ? GOI ? BOM ? COK    10:55         19:15   8h 20m     2 stops

      Additional_Info  Price
0             No info   3897
1             No info   7662
2             No info  13882
3             No info   6218
4             No info  13302
...               ...    ...
10678         No info   4107
10679         No info   4145
10680         No info   7229
10681         No info  12648
10682         No info  11753

[10683 rows x 11 columns]>
```

In [9]:

```
test_df.head
```

Out[9]:

```
<bound method NDFrame.head of                  Airline Date_of_Journey    S
ource Destination
0           Jet Airways       6/06/2019     Delhi       Cochin  \
1                IndiGo      12/05/2019   Kolkata     Banglore
2           Jet Airways      21/05/2019     Delhi       Cochin
3     Multiple carriers      21/05/2019     Delhi       Cochin
4              Air Asia      24/06/2019  Banglore        Delhi
...                 ...             ...       ...          ...
2666          Air India       6/06/2019   Kolkata     Banglore
2667             IndiGo      27/03/2019   Kolkata     Banglore
2668        Jet Airways       6/03/2019     Delhi       Cochin
2669          Air India       6/03/2019     Delhi       Cochin
2670  Multiple carriers      15/06/2019     Delhi       Cochin

                Route Dep_Time  Arrival_Time Duration Total_Stops  \
0     DEL ? BOM ? COK    17:30  04:25 07 Jun  10h 55m      1 stop
1     CCU ? MAA ? BLR    06:20         10:20       4h      1 stop
2     DEL ? BOM ? COK    19:15  19:00 22 May  23h 45m      1 stop
3     DEL ? BOM ? COK    08:00         21:00      13h      1 stop
4           BLR ? DEL    23:55  02:45 25 Jun   2h 50m    non-stop
...               ...      ...           ...      ...         ...
2666  CCU ? DEL ? BLR    20:30  20:25 07 Jun  23h 55m      1 stop
2667        CCU ? BLR    14:20         16:55   2h 35m    non-stop
2668  DEL ? BOM ? COK    21:50  04:25 07 Mar   6h 35m      1 stop
2669  DEL ? BOM ? COK    04:00         19:15  15h 15m      1 stop
2670  DEL ? BOM ? COK    04:55         19:15  14h 20m      1 stop

                  Additional_Info
0                         No info
1                         No info
2     In-flight meal not included
3                         No info
4                         No info
...                           ...
2666                      No info
2667                      No info
2668                      No info
2669                      No info
2670                      No info

[2671 rows x 10 columns]>
```

In [10]:

```
train_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 [11]:

```
test_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2671 entries, 0 to 2670
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype 
---  ------           --------------  ----- 
 0   Airline          2671 non-null   object
 1   Date_of_Journey  2671 non-null   object
 2   Source           2671 non-null   object
 3   Destination      2671 non-null   object
 4   Route            2671 non-null   object
 5   Dep_Time         2671 non-null   object
 6   Arrival_Time     2671 non-null   object
 7   Duration         2671 non-null   object
 8   Total_Stops      2671 non-null   object
 9   Additional_Info  2671 non-null   object
dtypes: object(10)
memory usage: 208.8+ KB
```

TO FIND MISSING VALUES

In [12]:

```
train_df.isna().sum()
```

Out[12]:

```
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 [13]:

```
test_df.isna().sum()
```

Out[13]:

```
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
dtype: int64
```

In [14]:

```
train_df.duplicated().sum()
```

Out[14]:

```
220
```

In [15]:

```
test_df.duplicated().sum()
```

Out[15]:

```
26
```

In [16]:

```
train_df.dropna(inplace=True)
```

In [17]:

```
train_df.isna().sum()
```

Out[17]:

```
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
```

In [18]:

```
train_df.shape
```

Out[18]:

```
(10682, 11)
```

Feature selection

In [19]:

```
train_df.columns
```

Out[19]:

```
Index(['Airline', 'Date_of_Journey', 'Source', 'Destination', 'Route',
       'Dep_Time', 'Arrival_Time', 'Duration', 'Total_Stops',
       'Additional_Info', 'Price'],
      dtype='object')
```

In [20]:

```
test_df.columns
```

Out[20]:

```
Index(['Airline', 'Date_of_Journey', 'Source', 'Destination', 'Route',
       'Dep_Time', 'Arrival_Time', 'Duration', 'Total_Stops',
       'Additional_Info'],
      dtype='object')
```

In [21]:

```
train_df['Airline'].value_counts()
```

Out[21]:

```
Airline
Jet Airways                          3849
IndiGo                               2053
Air India                            1751
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: count, dtype: int64
```

In [22]:

```
train_df['Source'].value_counts()
```

Out[22]:

```
Source
Delhi       4536
Kolkata     2871
Banglore    2197
Mumbai       697
Chennai      381
Name: count, dtype: int64
```

In [23]:

```
train_df['Destination'].value_counts()
```

Out[23]:

```
Destination
Cochin       4536
Banglore     2871
Delhi        1265
New Delhi     932
Hyderabad     697
Kolkata       381
Name: count, dtype: int64
```

In [24]:

```
train_df['Total_Stops'].value_counts()
```

Out[24]:

```
Total_Stops
1 stop      5625
non-stop    3491
2 stops     1520
3 stops       45
4 stops        1
Name: count, dtype: int64
```

In [25]:

```
airline={"Airline":{"Jet Airways":0,"IndiGo":1,"Air India":2,"Multiple carriers":3,
"SpiceJet":4,"Vistara":5,"Air Asia":6,"GoAir":7,
"Multiple carriers Premium economy":8,
"Jet Airways Business":9,"Vistara Premium economy":10,"Trujet":11}}
train_df=train_df.replace(airline)
train_df
```

Out[25]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Durat
0	1	24/03/2019	Banglore	New Delhi	BLR ? DEL	22:20	01:10 22 Mar	2h 5
1	2	1/05/2019	Kolkata	Banglore	CCU ? IXR ? BBI ? BLR	05:50	13:15	7h 2
2	0	9/06/2019	Delhi	Cochin	DEL ? LKO ? BOM ? COK	09:25	04:25 10 Jun	
3	1	12/05/2019	Kolkata	Banglore	CCU ? NAG ? BLR	18:05	23:30	5h 2
4	1	01/03/2019	Banglore	New Delhi	BLR ? NAG ? DEL	16:50	21:35	4h 4
...	...	...	...	...	...	...	...	
10678	6	9/04/2019	Kolkata	Banglore	CCU ? BLR	19:55	22:25	2h 3
10679	2	27/04/2019	Kolkata	Banglore	CCU ? BLR	20:45	23:20	2h 3
10680	0	27/04/2019	Banglore	Delhi	BLR ? DEL	08:20	11:20	
10681	5	01/03/2019	Banglore	New Delhi	BLR ? DEL	11:30	14:10	2h 4
10682	2	9/05/2019	Delhi	Cochin	DEL ? GOI ? BOM ? COK	10:55	19:15	8h 2

10682 rows × 11 columns

In [26]:

```
city={"Source":{"Delhi":0,"Kolkata":1,"Banglore":2,
"Mumbai":3,"Chennai":4}}
train_df=train_df.replace(city)
train_df
```

Out[26]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Durati
0	1	24/03/2019	2	New Delhi	BLR ? DEL	22:20	01:10 22 Mar	2h 5
1	2	1/05/2019	1	Banglore	CCU ? IXR ? BBI ? BLR	05:50	13:15	7h 2
2	0	9/06/2019	0	Cochin	DEL ? LKO ? BOM ? COK	09:25	04:25 10 Jun	1
3	1	12/05/2019	1	Banglore	CCU ? NAG ? BLR	18:05	23:30	5h 2
4	1	01/03/2019	2	New Delhi	BLR ? NAG ? DEL	16:50	21:35	4h 4
...	...	...	...	...	...	...	...	
10678	6	9/04/2019	1	Banglore	CCU ? BLR	19:55	22:25	2h 3
10679	2	27/04/2019	1	Banglore	CCU ? BLR	20:45	23:20	2h 3
10680	0	27/04/2019	2	Delhi	BLR ? DEL	08:20	11:20	
10681	5	01/03/2019	2	New Delhi	BLR ? DEL	11:30	14:10	2h 4
10682	2	9/05/2019	0	Cochin	DEL ? GOI ? BOM ? COK	10:55	19:15	8h 2

10682 rows × 11 columns

In [27]:

```
destination={"Destination":{"Cochin":0,"Banglore":1,"Delhi":2,
"New Delhi":3,"Hyderabad":4,"Kolkata":5}}
train_df=train_df.replace(destination)
train_df
```

Out[27]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Durati
0	1	24/03/2019	2	3	BLR ? DEL	22:20	01:10 22 Mar	2h 5
1	2	1/05/2019	1	1	CCU ? IXR ? BBI ? BLR	05:50	13:15	7h 2
2	0	9/06/2019	0	0	DEL ? LKO ? BOM ? COK	09:25	04:25 10 Jun	1
3	1	12/05/2019	1	1	CCU ? NAG ? BLR	18:05	23:30	5h 2
4	1	01/03/2019	2	3	BLR ? NAG ? DEL	16:50	21:35	4h 4
...	...	...	...	...	...	...	...	
10678	6	9/04/2019	1	1	CCU ? BLR	19:55	22:25	2h 3
10679	2	27/04/2019	1	1	CCU ? BLR	20:45	23:20	2h 3
10680	0	27/04/2019	2	2	BLR ? DEL	08:20	11:20	
10681	5	01/03/2019	2	3	BLR ? DEL	11:30	14:10	2h 4
10682	2	9/05/2019	0	0	DEL ? GOI ? BOM ? COK	10:55	19:15	8h 2

10682 rows × 11 columns

In [28]:

```
stops={"Total_Stops":{"non-stop":0,"1 stop":1,"2 stops":2,
"3 stops":3,"4 stops":4}}
train_df=train_df.replace(stops)
train_df
```

Out[28]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Durati
0	1	24/03/2019	2	3	BLR ? DEL	22:20	01:10 22 Mar	2h 5
1	2	1/05/2019	1	1	CCU ? IXR ? BBI ? BLR	05:50	13:15	7h 2
2	0	9/06/2019	0	0	DEL ? LKO ? BOM ? COK	09:25	04:25 10 Jun	1
3	1	12/05/2019	1	1	CCU ? NAG ? BLR	18:05	23:30	5h 2
4	1	01/03/2019	2	3	BLR ? NAG ? DEL	16:50	21:35	4h 4
...	...	...	...	...	...	...	...	
10678	6	9/04/2019	1	1	CCU ? BLR	19:55	22:25	2h 3
10679	2	27/04/2019	1	1	CCU ? BLR	20:45	23:20	2h 3
10680	0	27/04/2019	2	2	BLR ? DEL	08:20	11:20	
10681	5	01/03/2019	2	3	BLR ? DEL	11:30	14:10	2h 4
10682	2	9/05/2019	0	0	DEL ? GOI ? BOM ? COK	10:55	19:15	8h 2

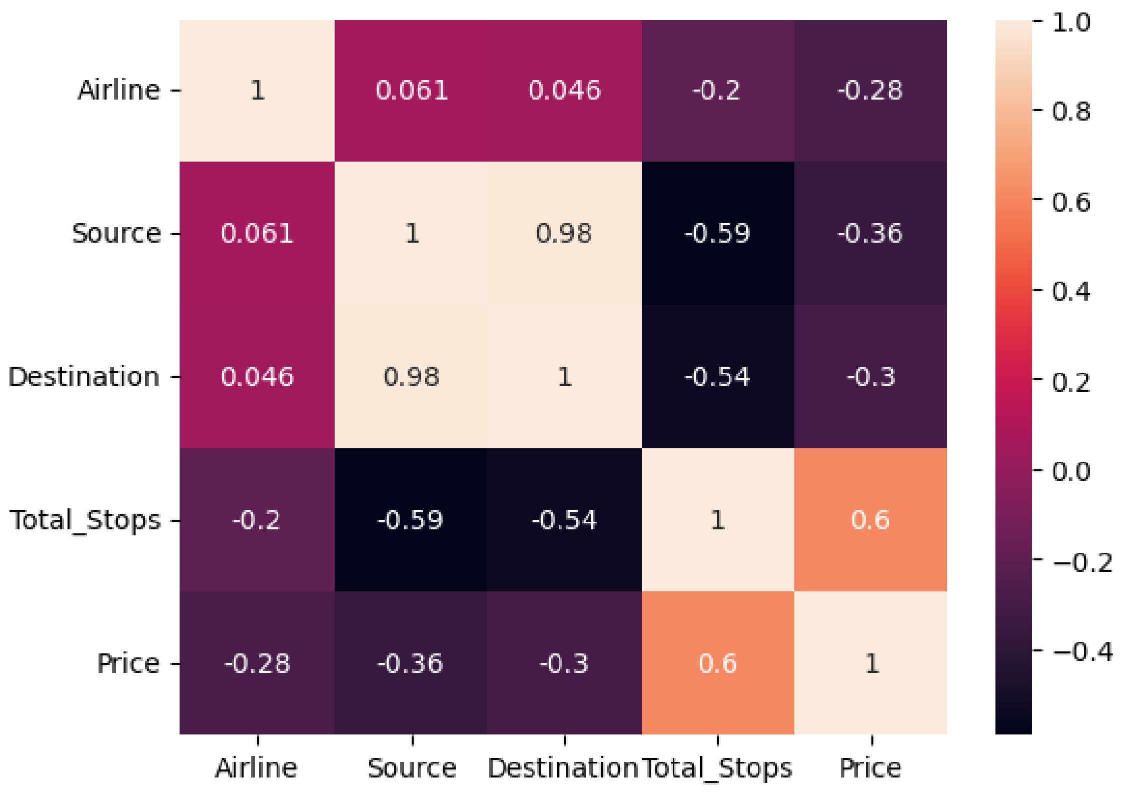
10682 rows × 11 columns

In [29]:

```
fdf=train_df[['Airline','Source','Destination','Total_Stops','Price']]
sns.heatmap(fdf.corr(),annot=True)
```

Out[29]:

```
<Axes: >
```



Linearregression for training data

In [30]:

```
x=np.array(train_df['Source']).reshape(-1,1)
y=np.array(train_df['Destination']).reshape(-1,1)
```

In [31]:

```
x_train,X_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=100)
```

In [32]:

```
from sklearn.linear_model import LinearRegression
regr=LinearRegression()
regr.fit(x_train,y_train)
print(regr.intercept_)
print(regr.coef_)
```

```
[-0.08091813]
[[1.26597749]]
```

In [33]:

```
X_train,X_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
regr.fit(X_train,y_train)
regr.fit(X_train,y_train)
```

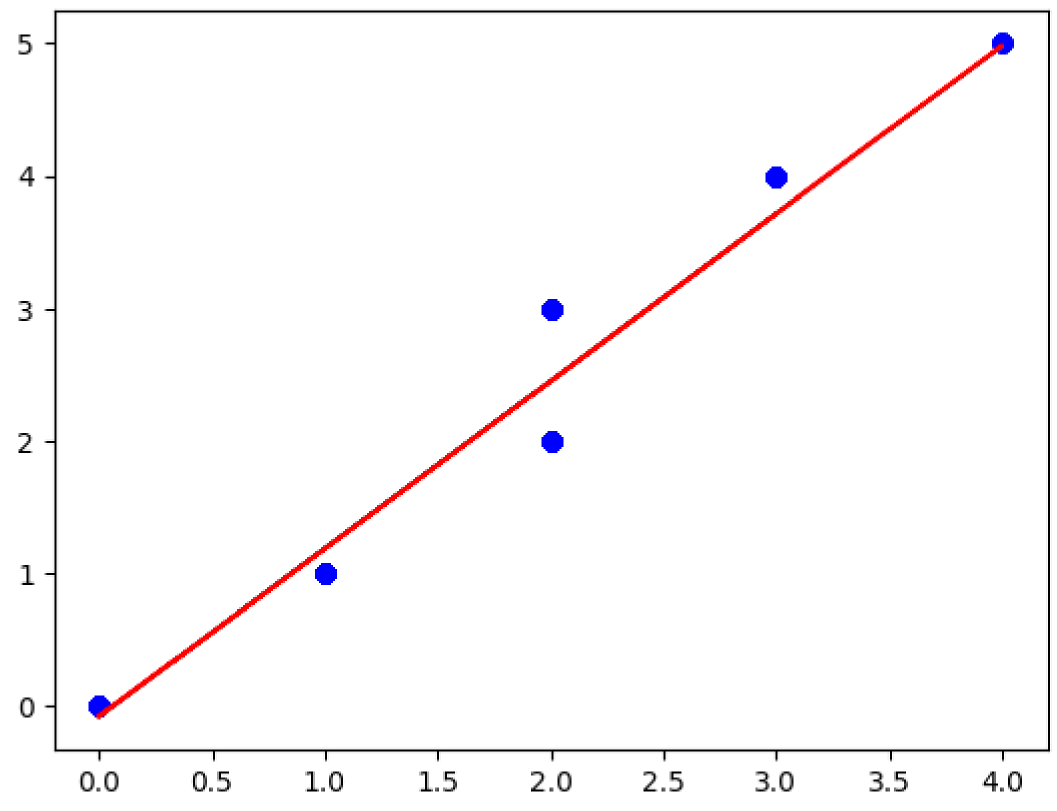
Out[33]:

```
▾ LinearRegression

LinearRegression()
```

In [34]:

```
y_pred=regr.predict(X_test)
plt.scatter(X_test,y_test,color='b')
plt.plot(X_test,y_pred,color='r')
plt.show()
```

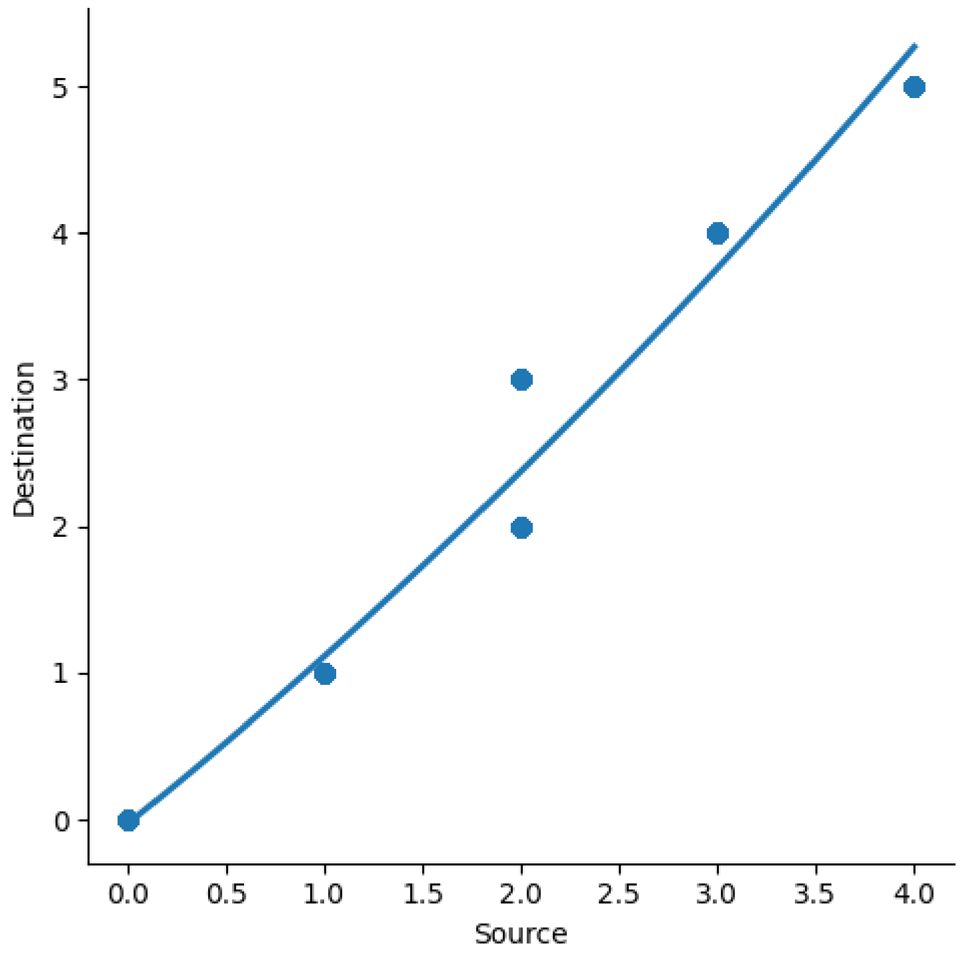


In [35]:

```
sns.lmplot(x="Source",y="Destination",data=train_df,order=2,ci=None)
```

Out[35]:

```
<seaborn.axisgrid.FacetGrid at 0x1b8f036c190>
```



In [36]:

```
score=regr.score(X_test,y_test)
print(score)
```

```
0.9661997366087269
```

ridge

In [37]:

```
from sklearn.linear_model import Ridge, RidgeCV, Lasso
```

In [38]:

```
ridge=Ridge(alpha=2)
ridge.fit(x_train,y_train)
train_score_ridge=ridge.score(x_train,y_train)
test_score_ridge=ridge.score(X_test,y_test)
print("\nLinearRegression\n")
print(train_score_ridge)
print(test_score_ridge)
```

```

LinearRegression

0.0004353748899809107
0.04064037732261583
```

Lasso

In [39]:

```
lasso=Lasso(alpha=100)
lasso=lasso.fit(x_train,y_train)
train_score_lasso=lasso.score(x_train,y_train)
test_score_lasso=lasso.score(X_test,y_test)
print(train_score_lasso)
print(test_score_lasso)
```

```
0.0
-0.0002774640661185046
```

elastic

In [40]:

```
from sklearn.linear_model import ElasticNet
```

In [41]:

```
a=ElasticNet()
a.fit(x,y)
print(a.coef_)
print(a.intercept_)
```

```
[0.6035366]
[0.5919792]
```

In [42]:

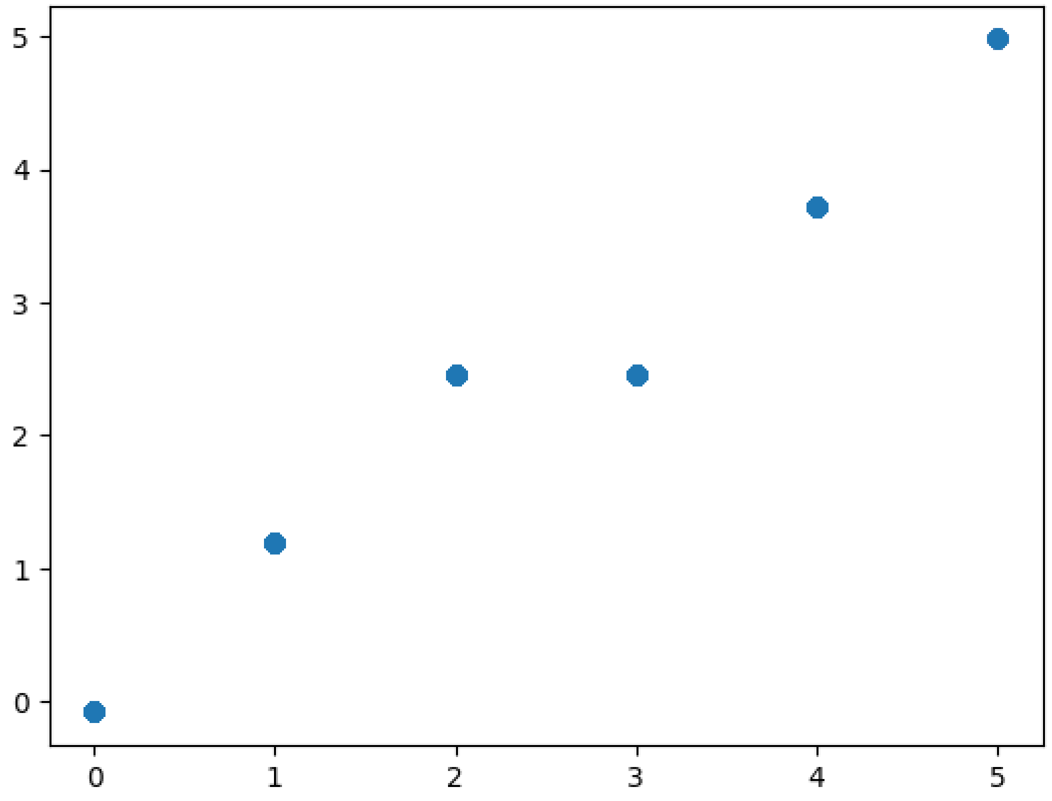
```
predictions=regr.predict(X_test)
```

In [43]:

```
plt.scatter(y_test,predictions)
```

Out[43]:

```
<matplotlib.collections.PathCollection at 0x1b8f4513f50>
```



Logistic regression

In [44]:

```
x=np.array(train_df['Source']).reshape(-1,1)
y=np.array(train_df['Destination']).reshape(-1,1)
train_df.dropna(inplace=True)
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=1)
lr=LogisticRegression(max_iter=10000)
```

In [45]:

```
lr.fit(x_train,y_train)
```

```
C:\Users\91950\AppData\Local\Programs\Python\Python311\Lib\site-packages\s
klearn\utils\validation.py:1143: DataConversionWarning: A column-vector y
was passed when a 1d array was expected. Please change the shape of y to
(n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

Out[45]:

```
▾        LogisticRegression
LogisticRegression(max_iter=10000)
```

In [46]:

```
score=lr.score(x_test,y_test)
print(score)
```

```
0.9110764430577223
```

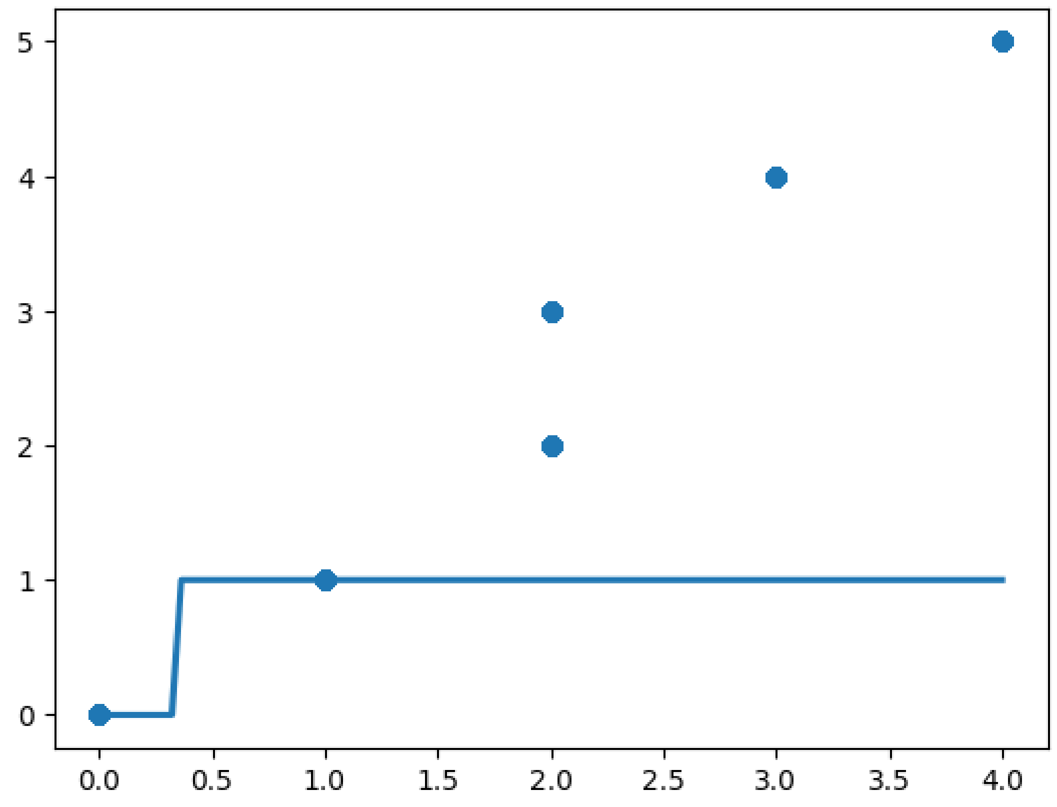
In [47]:

```
sns.regplot(x=x,y=y,data=train_df,logistic=True,ci=None)
```

```
C:\Users\91950\AppData\Local\Programs\Python\Python311\Lib\site-packages\s
tatsmodels\genmod\families\links.py:198: RuntimeWarning: overflow encounte
red in exp
  t = np.exp(-z)
```

Out[47]:

```
<Axes: >
```



Decision tree

In [48]:

```
from sklearn.tree import DecisionTreeClassifier
train_df=DecisionTreeClassifier(random_state=0)
train_df.fit(x_train,y_train)
```

Out[48]:

```
▾ DecisionTreeClassifier
DecisionTreeClassifier(random_state=0)
```

In [49]:

```
score=train_df.score(x_test,y_test)
print(score)
```

```
0.9110764430577223
```

Random classifier

In [104]:

```
from sklearn.ensemble import RandomForestClassifier
rfc=RandomForestClassifier()
rfc.fit(X_train,y_train)
print(rfc.score(x_train,y_train))
print(rfc.score(x_test,y_test))
```

```
0.4218269359368731
0.431201248049922
```

In [105]:

```
rf=RandomForestClassifier()
```

In [106]:

```
params={'max_depth':[2,3,5,10,20],
'min_samples_leaf':[5,10,20,50,100,200],
'n_estimators':[10,25,30,50,100,200]}
```

In [107]:

```
import warnings
warnings.simplefilter(action='ignore')
```

In [108]:

```
from sklearn.model_selection import GridSearchCV
grid_search=GridSearchCV(estimator=rf,param_grid=params,cv=2,scoring="accuracy")
```

In [109]:

```
grid_search.fit(x_train,y_train)
```

Out[109]:

```
▸            GridSearchCV
▸ estimator: RandomForestClassifier
       ▸ RandomForestClassifier
```

In [110]:

```
grid_search.best_score_
```

Out[110]:

```
0.9134679490013939
```

In [111]:

```
rf_best=grid_search.best_estimator_
rf_best
```

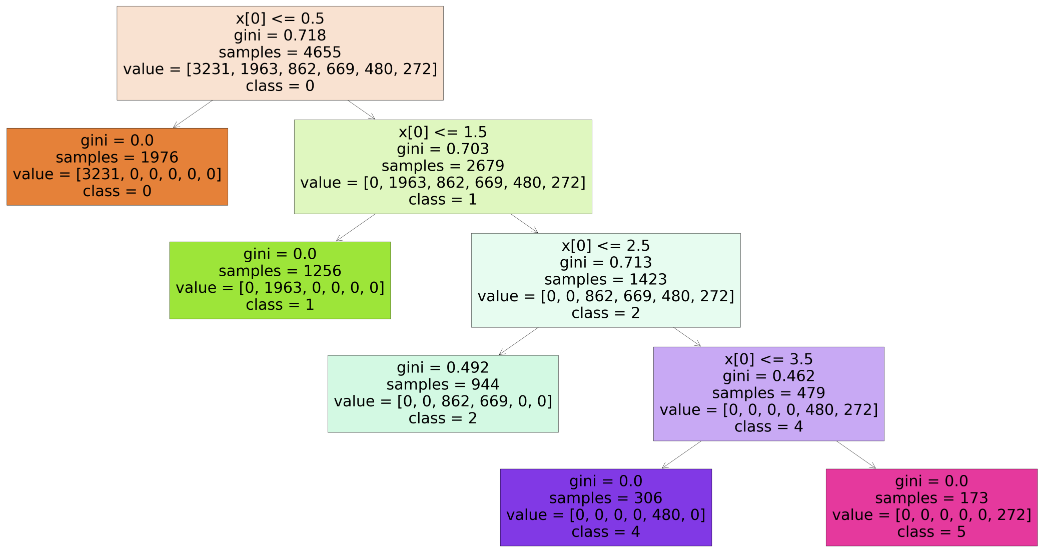
Out[111]:

```
▾                      RandomForestClassifier

RandomForestClassifier(max_depth=5, min_samples_leaf=5, n_estimators=10)
```

In [112]:

```
from sklearn.tree import plot_tree
plt.figure(figsize=(80,40))
plot_tree(rf_best.estimators_[6],class_names=['0','1','2','3','4','5','6'],filled=True);
```



In [113]:

```
score=rfc.score(x_test,y_test)
print(score)
```

```
0.431201248049922
```

conclusion

For the above Dataset we use different Types ofmodels ,for each and every model we getdifferentTypes of Accuracies For linearregression we obtained 96% accuracy, For Logistic regression we obtained 91% accuracy, For decision tree we obtained 91%acuracy, For Random forest we obtained 43% accuracy. From all the observations we canconclude that LinearRegression model is Best fit