2021

Sonali Warkhade

Data Trained

4/18/2021

Flight Price Prediction

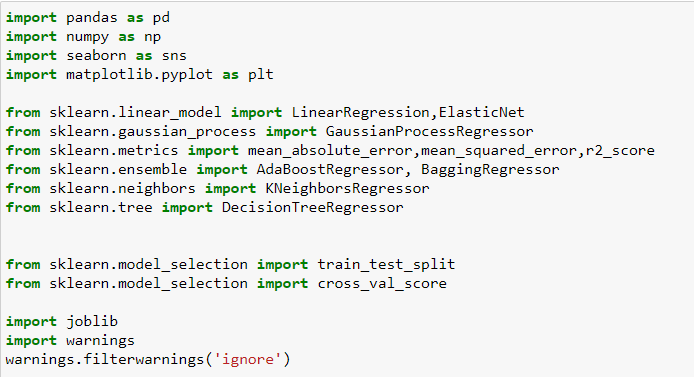


In this article, I will go through the whole process of creating a machine learning model on the Flight Price Prediction dataset. It provides information on the prices of flight tickets, summarized according to Airline, Date of Journey, Source, Destination, Duration and Total Stops.

Flight Price Prediction

Flight ticket prices can be something hard to guess, today we might see a price, check out the price of the same flight tomorrow, it will be a different story. We might have often heard travellers saying that flight ticket prices are so unpredictable. Here you will be provided with prices of flight tickets for various airlines between the months of March and June of 2019 and between various cities.

**Imported Libraries:**



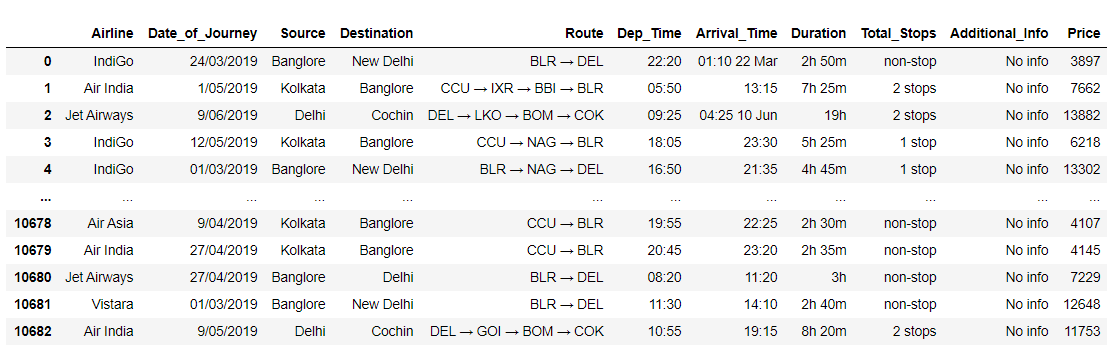
**Let’s import excel train dataset:**

train=pd.read\_excel(r"C:\Users\SAGAR KADAM\Downloads\Flight\_Ticket\_Participant\_Datasets-20190305T100527Z-001\Flight\_Ticket\_Participant\_Datasets\Data\_Train.xlsx")

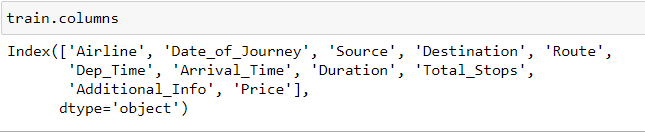
train=pd.DataFrame(data=train)

train

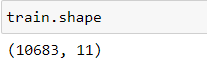
Below is the screenshot which gives us glance of our dataset:



Let’s see the list of all columns:

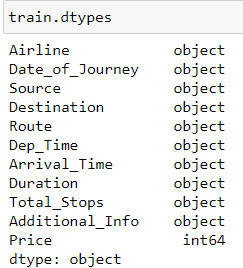


Now let’s see the total number of rows and columns:



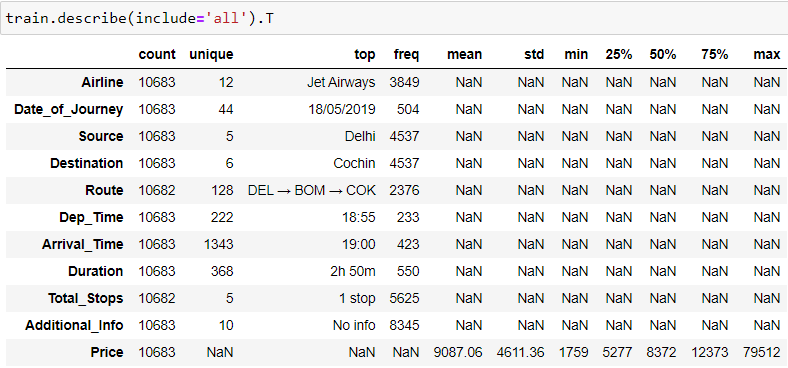
Our train dataset has 10683 rows and 11 columns.

Let’s see datatype of all variables:



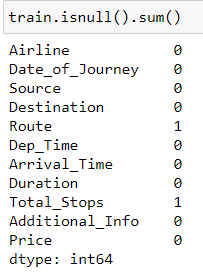
All columns of our dataset are with object datatype except our target column ’Price’.

Now I will give you the overview of the train dataset.

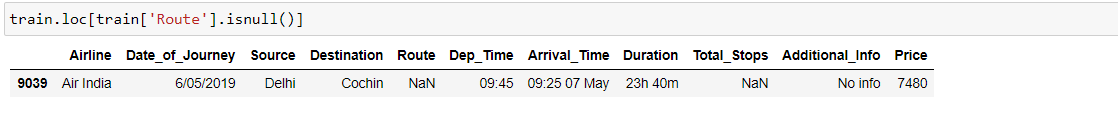


In our dataset we have 12 different airlines, journey happened on 44 different dates, 5 source place and 6 destination places, flights have flied from 128 different routes, we also have Departure and arrival time details of each flight with duration of total journey, maximum 5 total number of stops taken by one passenger from source to destination. There is a column named Additional info which does not contain much information as we have 8345 rows with No info. And lastly, in our target column ‘Price’ flight ticket is ranging from 1759 to 79512.

I can see the count of all columns is not the same hence, I checked for null values and worked on it.



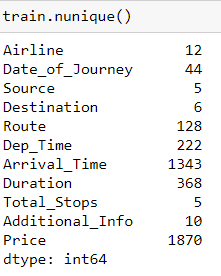
I can see null value in Route and in total stop too hence let’s see the particular column first.



As I have only one row with null values hence, I can drop that row.



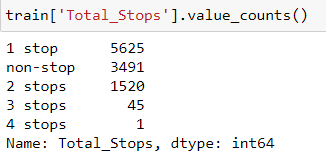
I worked on the data first as further analysis data should be more clear. I have checked for the unique values and dropped some unwanted columns.



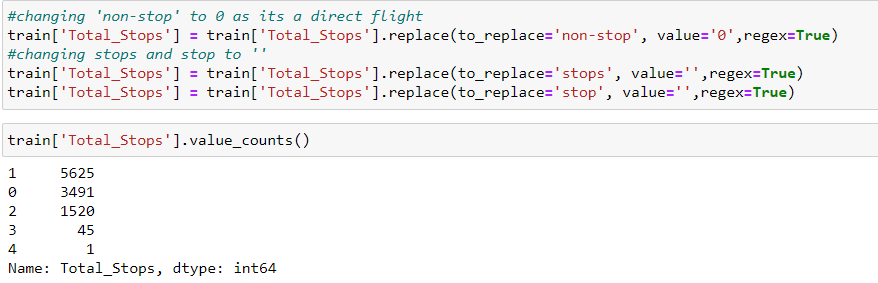
Route column and Total stops column is giving us the same info in different mediums as the route column tells us that flight changed/ stopped at all locations at the same time total stops is also giving us the information that how many stops/halts taken by a particular passenger. We can make changes to total columns more conveniently as it has only 5 unique values and route has 128 hence, I am keeping total stops column and dropping routes columns. In the additional info column for 8345 rows, we don't have any more information hence that column is also not much relevant. I have dropped both the columns.



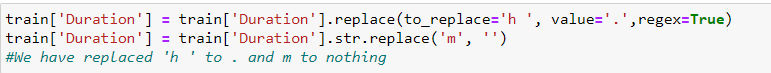
Then I worked on the Total stops column. We had 5 unique values, let’s see what are those:

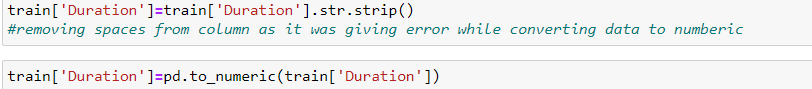


I have removed words stop, stops and non-stop. Stop & stops to ‘’(blank or nothing), for non-stop I used 0(Zero) as it’s a direct flight.

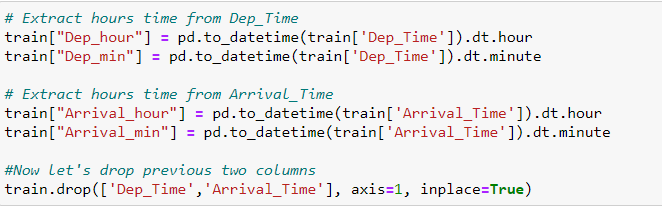


Then I worked on Duration which is giving us timing of total journey hours. I haven’t used minutes and hours function as it was leading to multiple column creation. I used a different approach by replacing alphabets to .(dot) and ‘’(blank or nothing). I have also changed the data type to numeric.

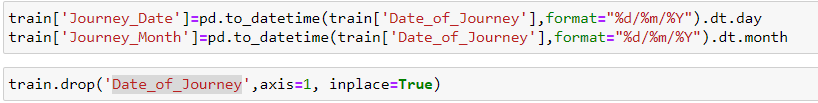




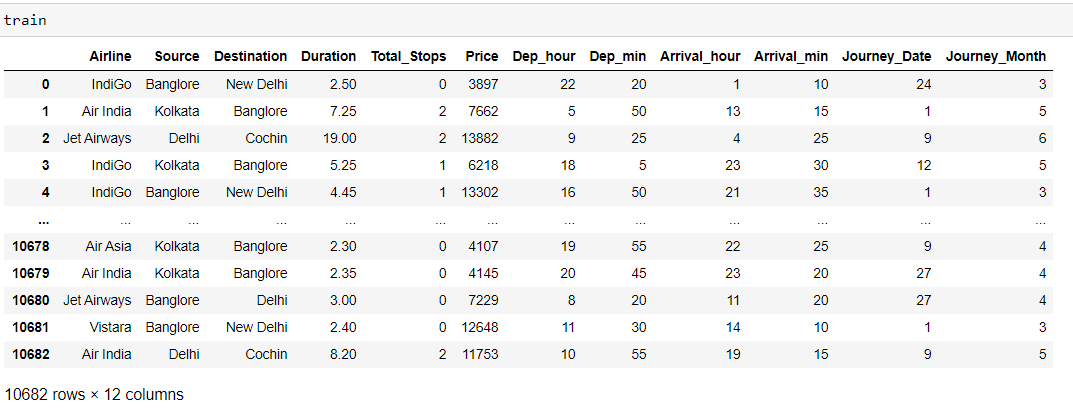
I have then changed data for columns departure time and arrival time. Extracted hours and minutes of departure and arrival both in different columns. After that, I dropped the previous 2 columns.



Then I have extracted the month and journey date in 2 columns, The year is already the same for all the rows. Then I dropped the Date of Journey column.



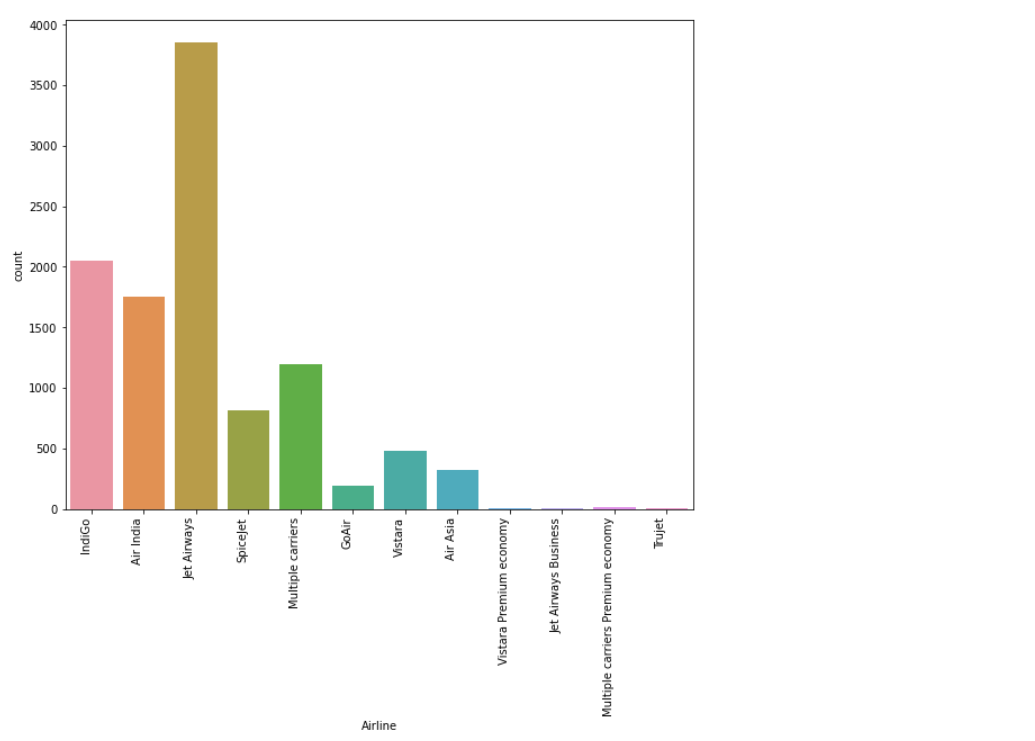
Now dataset looks like this:



**Visualization:**

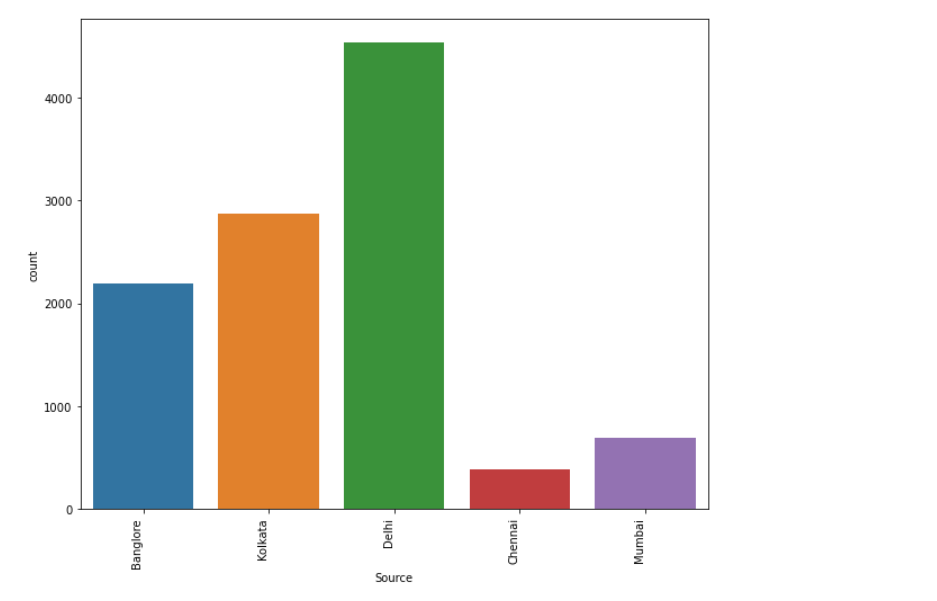
I have first created a list of all object type columns for visualization and the columns are 'Airline', 'Source', 'Destination', 'Total\_Stops'. First, I used countplot to view counts of unique values of each column.

**Airlines:**

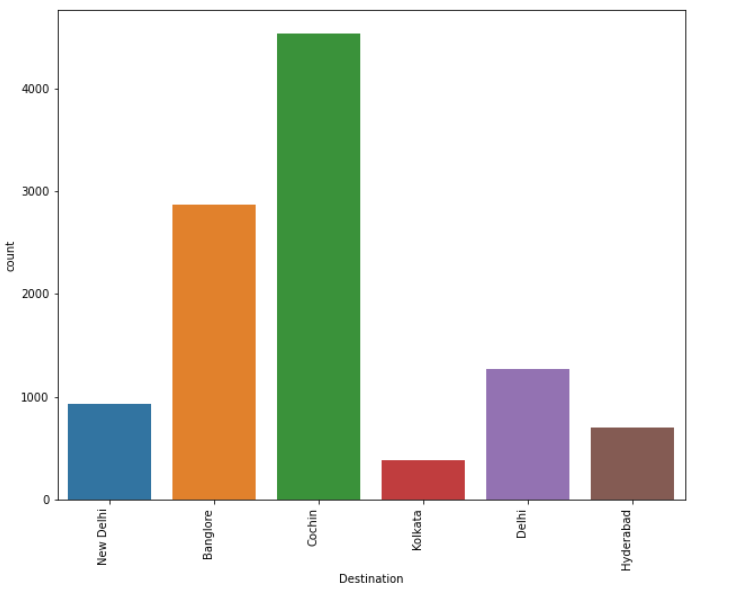


Above plot tells us that Maximum flights are of Jet Airways then we have IndiGo, Air India and Multiple carriers.

**Source:**

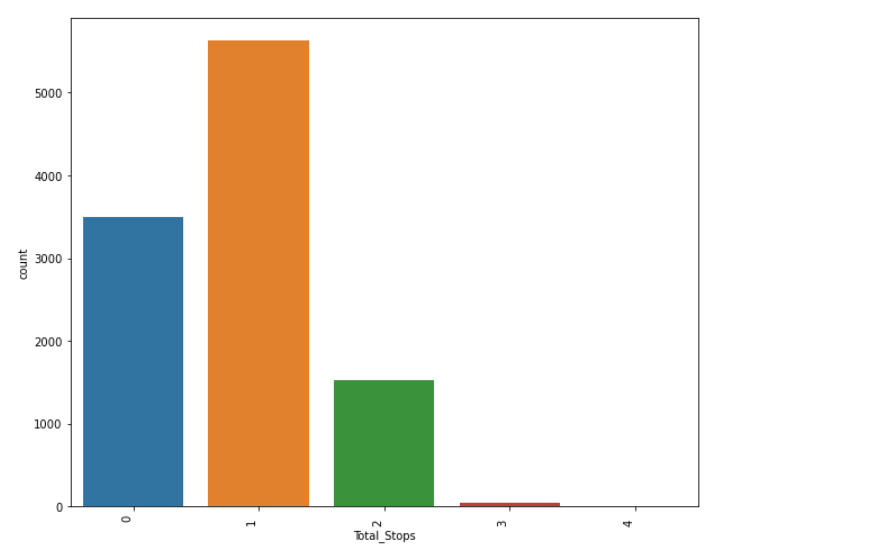
 Above plot tells us that 4536 flights flew from Delhi, 2871 flew from Kolkata, 2197 flew from Bangalore, From Mumbai 697 and lastly from Chennai 381.

**Destination:**



4536 flights arrived at Cochin, 2871 arrived at Bangalore, 1295 flights arrived at Delhi, 932 arrived at New Delhi, 697 arrived at Hyderabad and lastly 381 arrived at Kolkata.

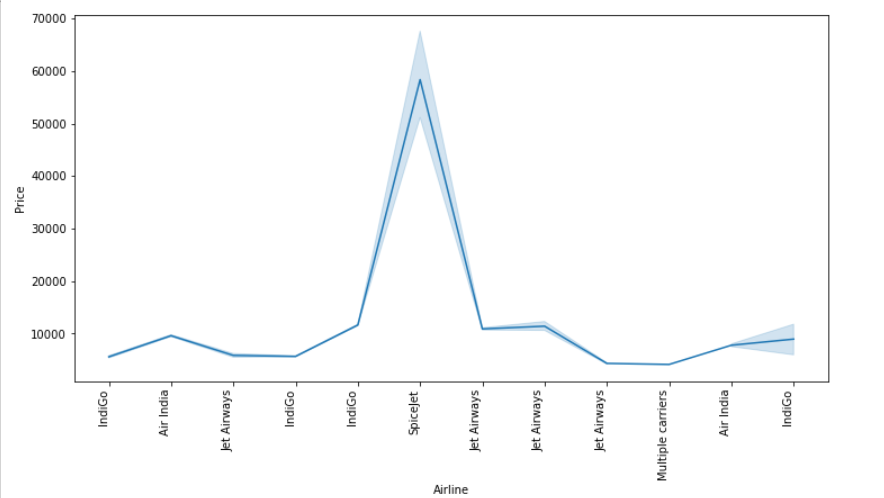
**Total Stops:**



Maximum flights were with one stop i.e., 5625 flights, 3491 flights were direct, 1520 flights were with 2 stops, 45 flights with 3 stops and only one flight had 4 stops.

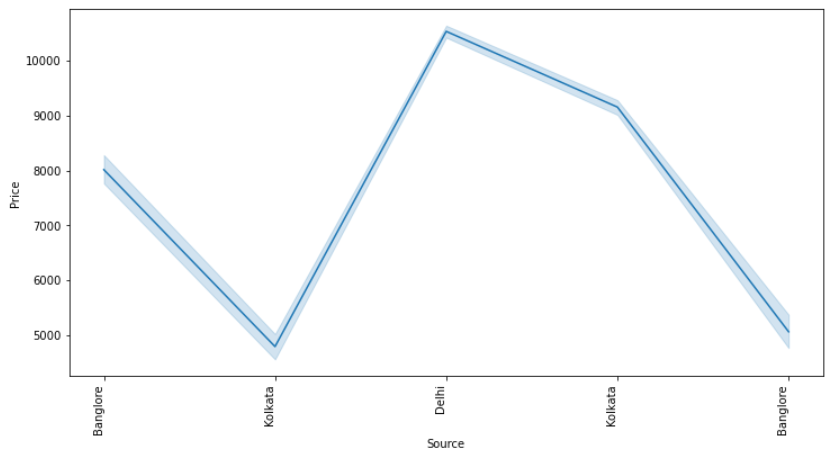
Then I have compared all object type columns with our target column ‘Price’ with the help of **lineplot**.

**Airline vs Price:**



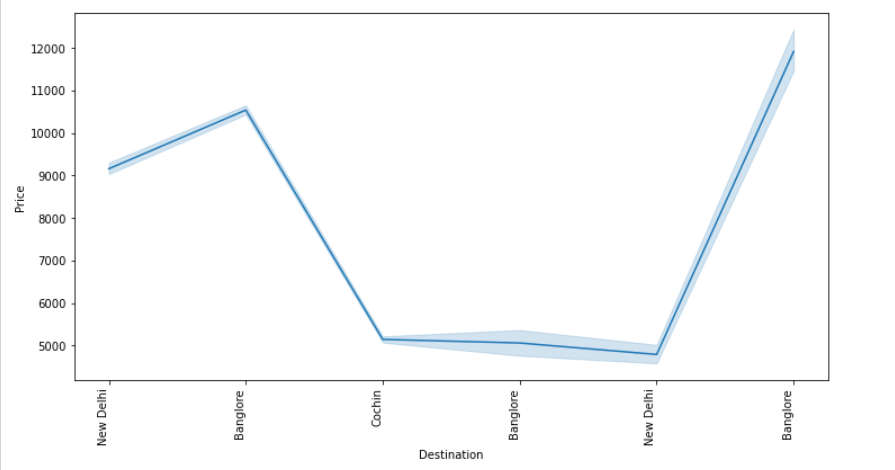
Highest price is for Airline SpiceJet.

Source vs Price:



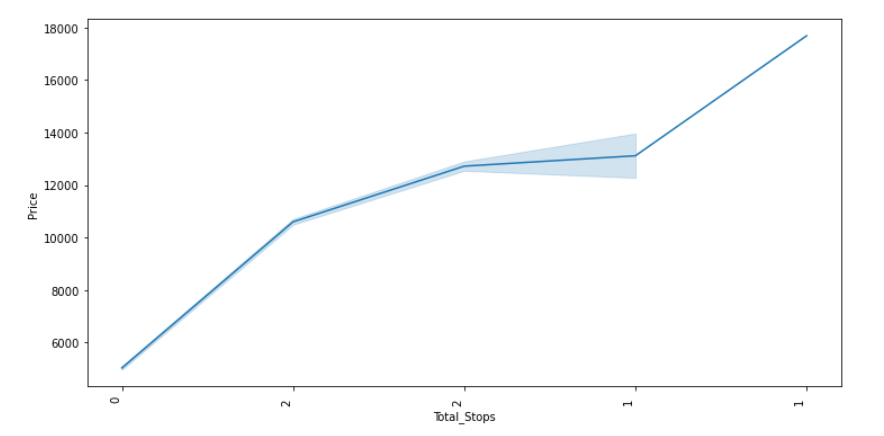
Highest price charged was from source Delhi.

**Destination vs Price:**



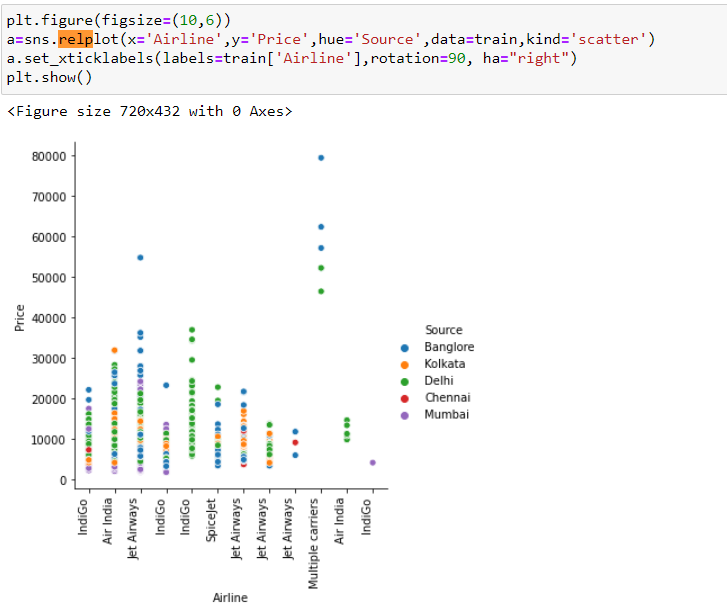
Highest price charged was for destination Bangalore.

**Total Stops vs Price:**



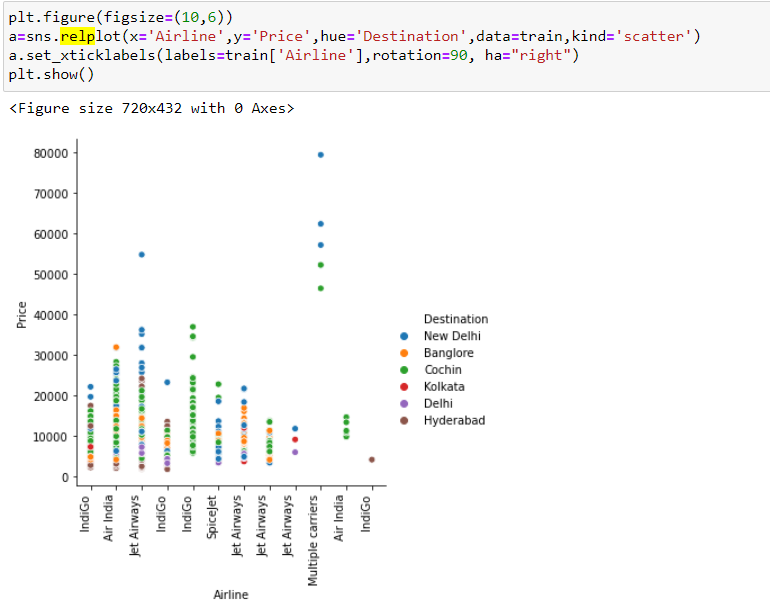
Highest price charged was for 1 stop flights.

Then with the help of **relplot** I have compared prices of airlines with respect to source.



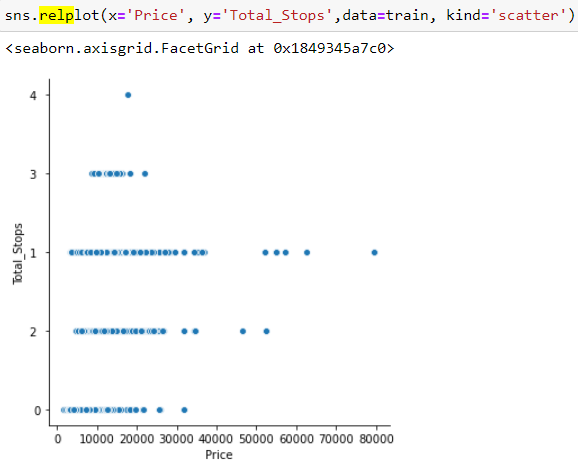
Above plot tells us that Highest price charged was from Bangalore source and with Multiple carriers airlines.

Then compared prices of airlines with respect to destination.



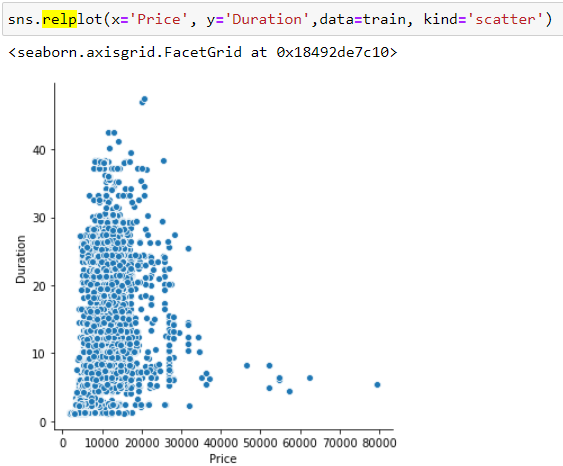
Highest price charged was for destination new delhi and with Multiple carriers airlines.

Then with the help of **regplot** I have compared prices with total stops.



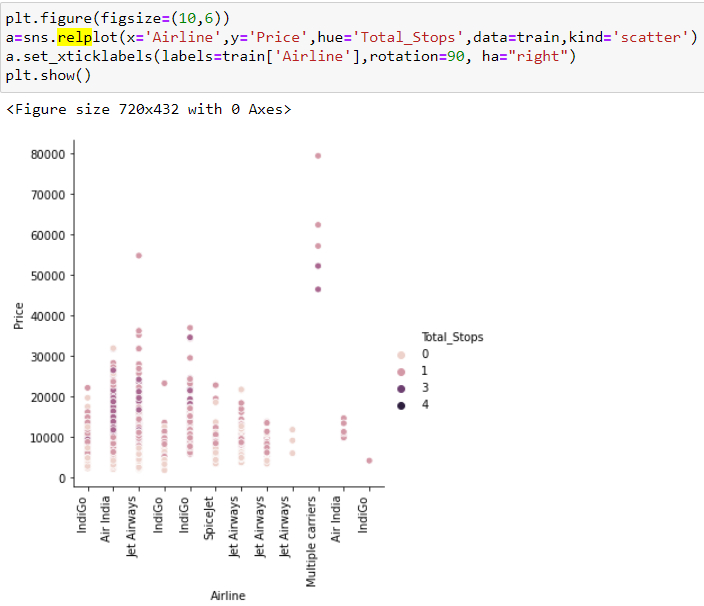
For direct flight, price is ranging till around 33000. For one stop its ranging till 80000, for 2 stops its ranging till around 60000, for 3 stops its ranging till around 25000 and for 4 stops flight cost was around 20000.

Then with the help of **regplot** I have compared prices with Duration.



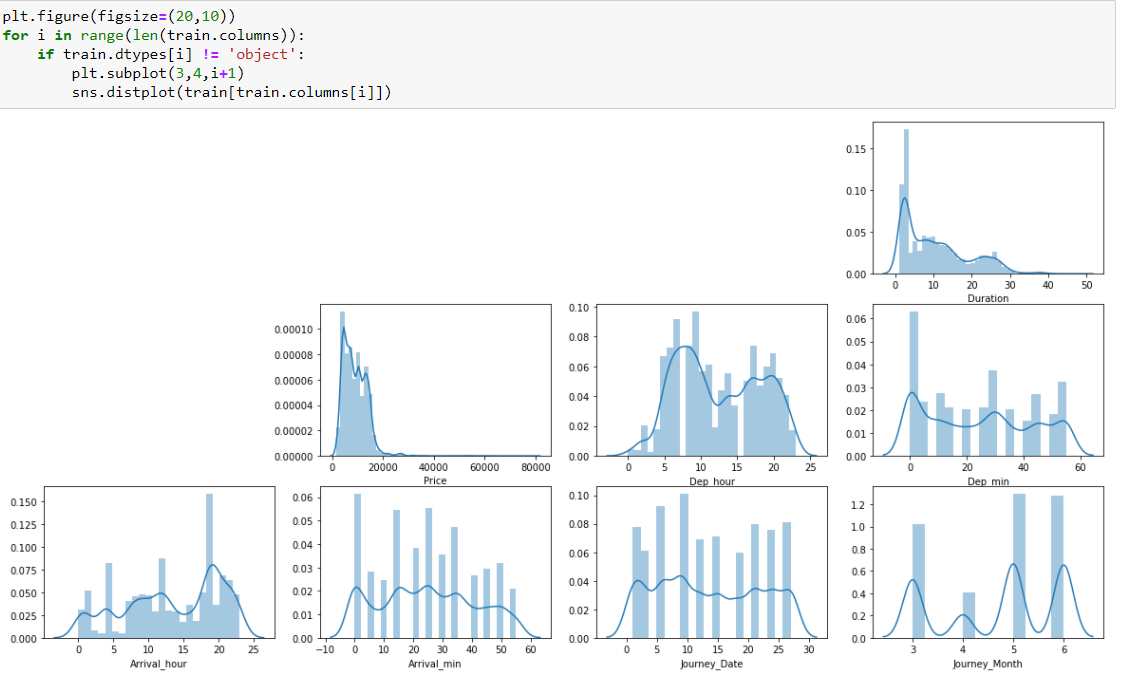
For duration from 0 to 40, usually the price is ranging till 40000.

Then with the help of relplot I have compared prices of airlines with respect to Total stops.

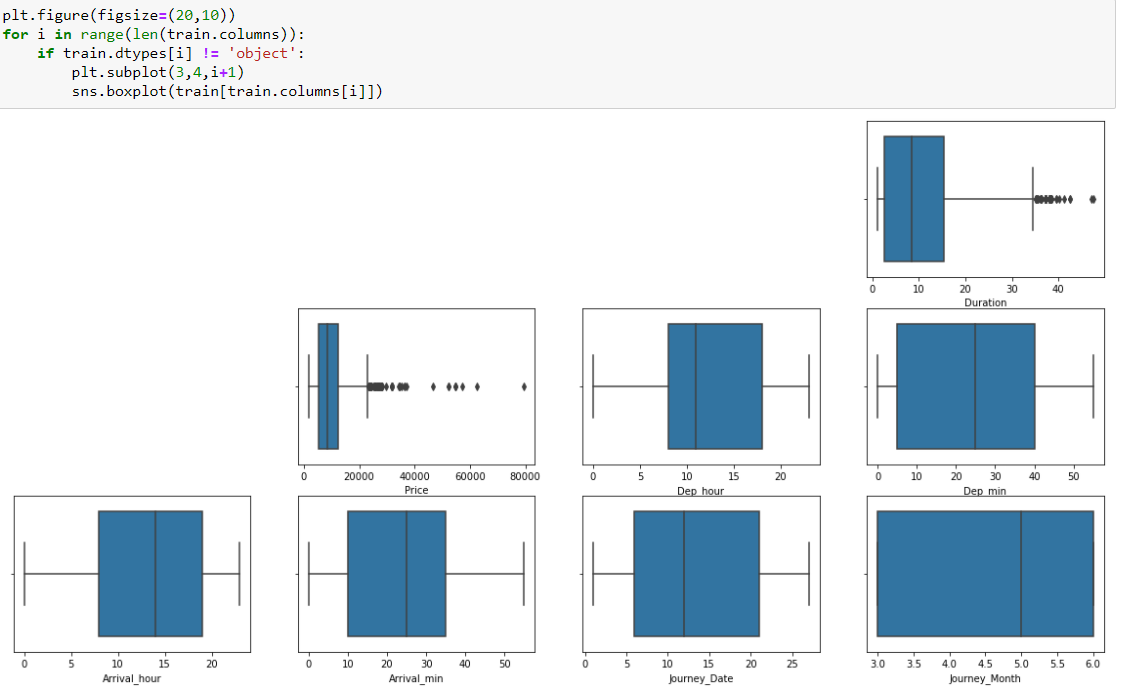


Highest price charged was for one stop flight and with Multiple carriers airlines.

Now let’s look at the numerical columns with the help of distplot and boxplot.

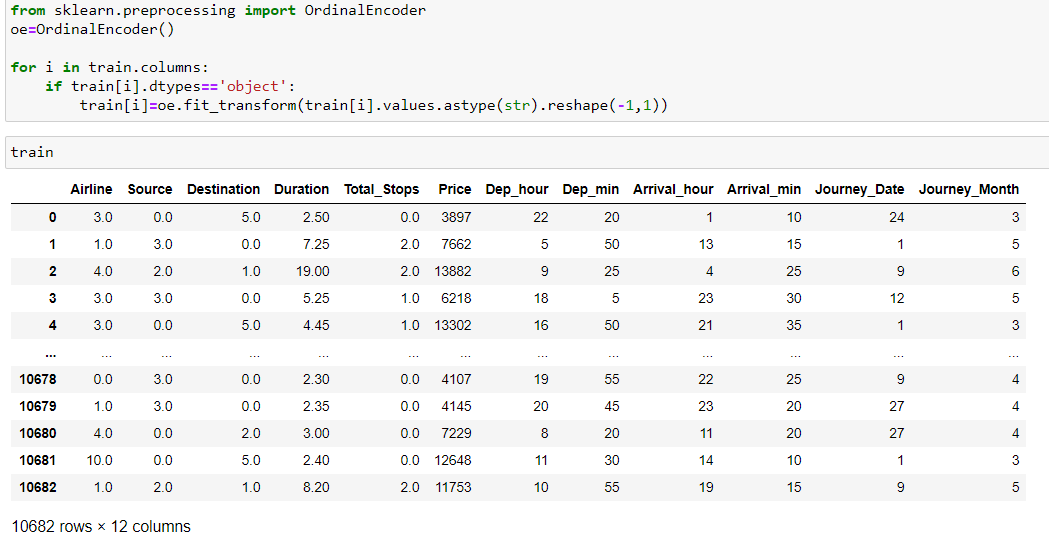


Above plots tell us that all columns are with uneven distribution.



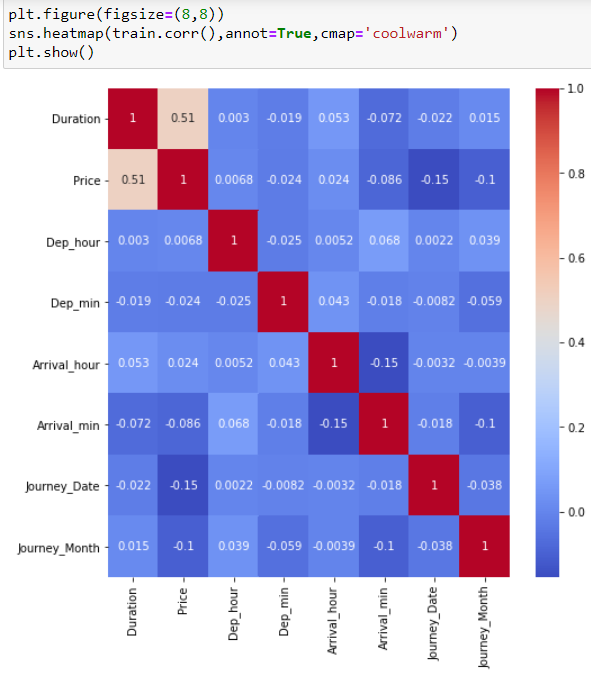
Boxplot tells us the outliers which are there in the price and duration column.

Then I checked the correlation however all columns were not numeric hence I was unable to compare corrections of columns with each other. Hence I encoded all object type columns using Ordinal encoding.

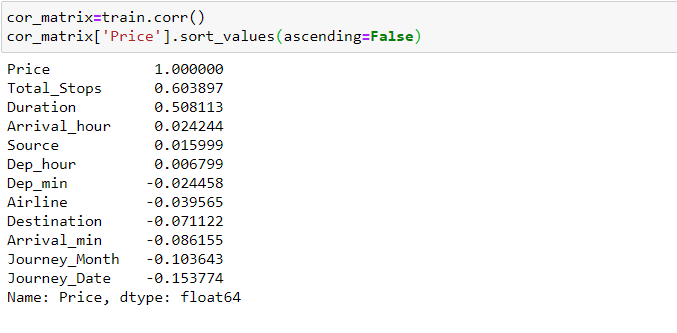


Above table shows us how our data looks after encoding.

Now let’s see the correlation between all columns specially with our target column price.

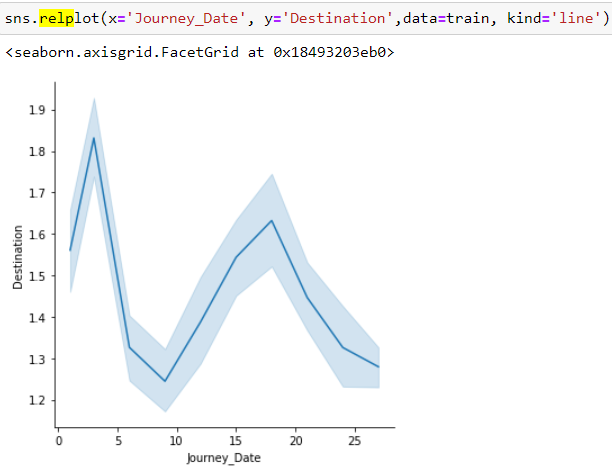


Below code tells us the correlation of column with our target variable ’Price’



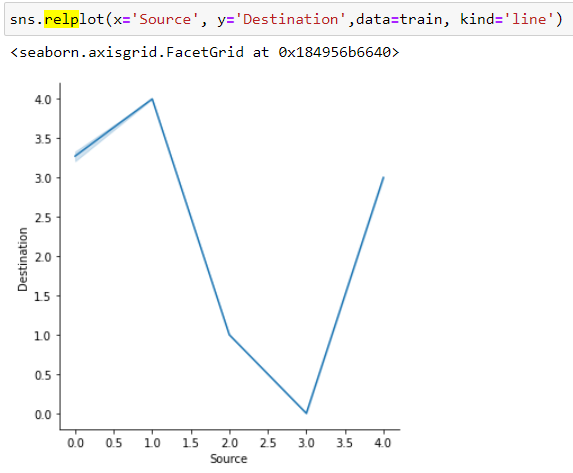
Our target column Price is positively correlated with Total Stops, Duration and has negative correlation with Journey­\_Date, Journey\_Month and Arrival\_min column.

Now with the help of relplot I tried to see the correlation of journey date and destination.



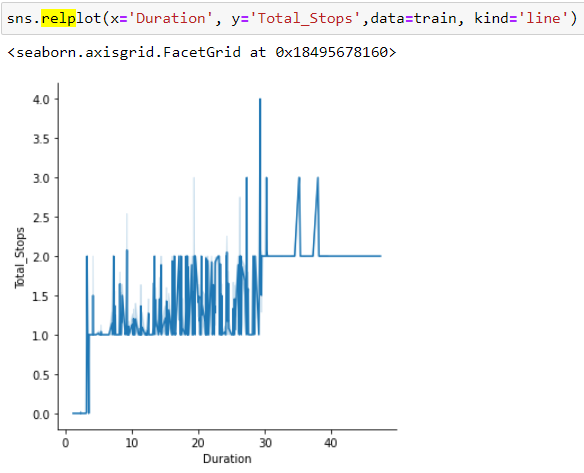
We can see a negative relation between Journey\_Date and Destination.

Then relation between Source and destination.



We can see a negative relation between Source and Destination.

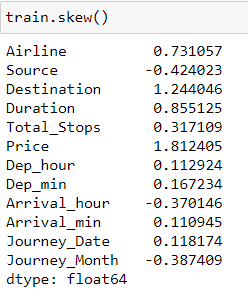
Now let see the relation between duration and total stops.



For duration of maximum 5 hours we have direct flights, for duration of maximum 30 hours we have 2 stops flights and for duration of 30 to 40 hours of flight we have 3 & 4 hours of flights.

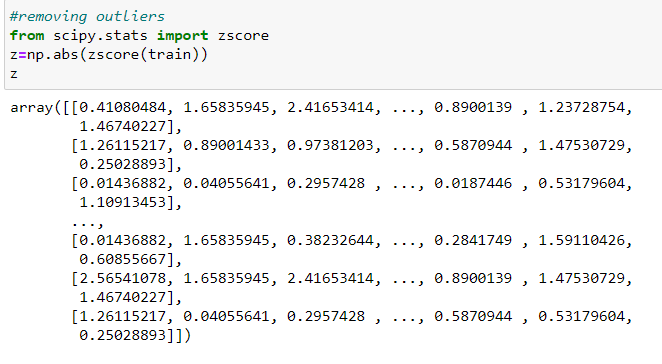
**Skewness and Outliers:**

Before standardizing the data I have checked the skewness and then removed outliers.

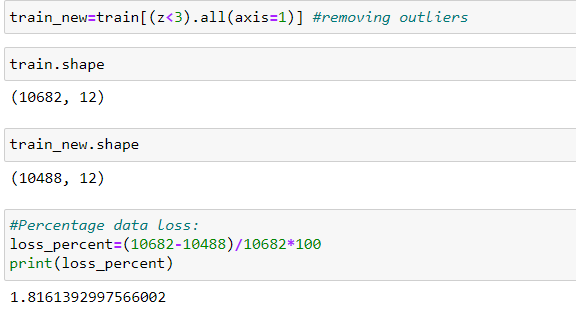


Usually, the skewness range is considered between -0.5 to +0.5. I can see some columns beyond this range and those are Airline, destination, duration and price. Airline and destination are object/categorical type columns then skewness is not considered for them and Price is our target column. Hence skewness is considered for duration column.

I have used the z score method to remove outliers. Another method which one considers is IQR.

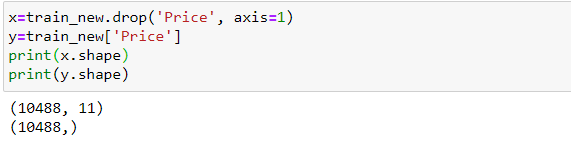


I considered threshold value as 3, now removing outliers.

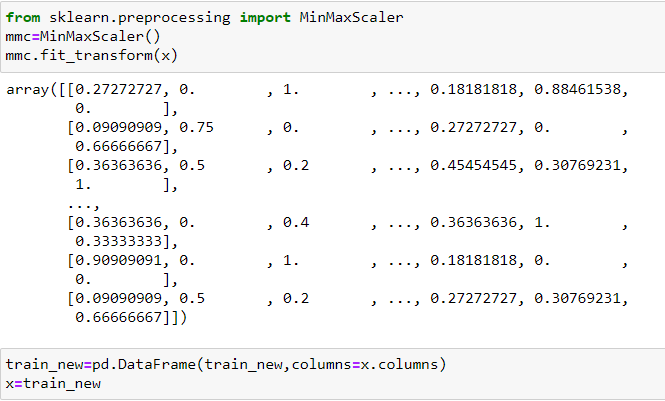


After removing the outlier my new dataset has 10488 rows and 12 columns. Data loss is about 1.82 %.

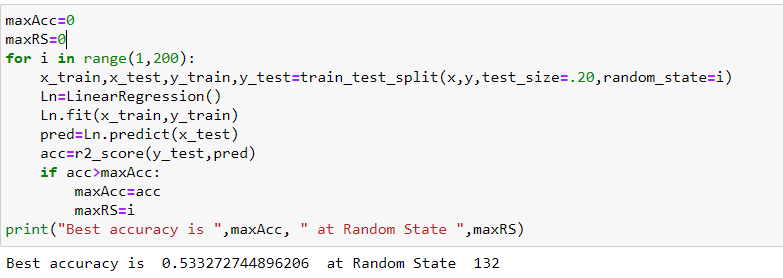
Splitting data between dependent and independent variables i.e. x & y.



Then I have performed standardization. I have a minmax scaler as for maximum column data is within a particular range. I have not standardized our target variable.



As we can see after standardization data is returned in the form of an array hence, I have transformed it into a dataframe so that I can build a model. Before building a model, I have checked the best random state on which are getting best accuracy.

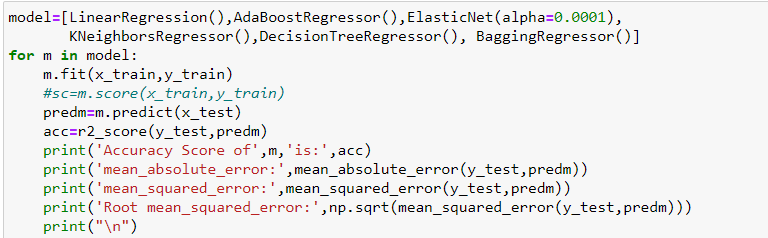


On 132nd random state I have got the best accuracy.

I have then splitted data into training and testing sets. I am using 20% data as test data.



I am trying to find the best accuracy with 6 regression models. I have used a loop rather than repeating the same code again and again.



Below is the accuracy score for all 6 models:

Accuracy for LinearRegression model is 53.33%

Accuracy for AdaBoostRegressor model is 60.35%

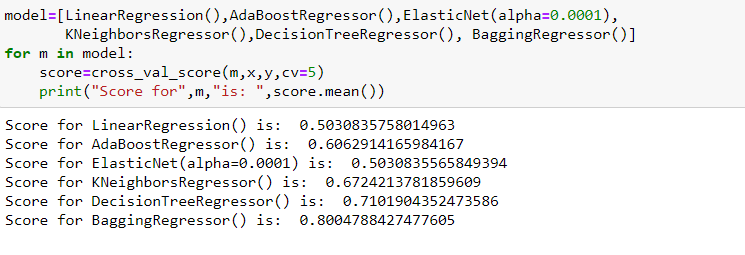
Accuracy for ElasticNet(alpha=0.0001) model is 53.33%

Accuracy for KNeighborsRegressor model is 70.40%

Accuracy for DecisionTreeRegressor model is 72.12%

Accuracy for BaggingRegressor model is 80.17%

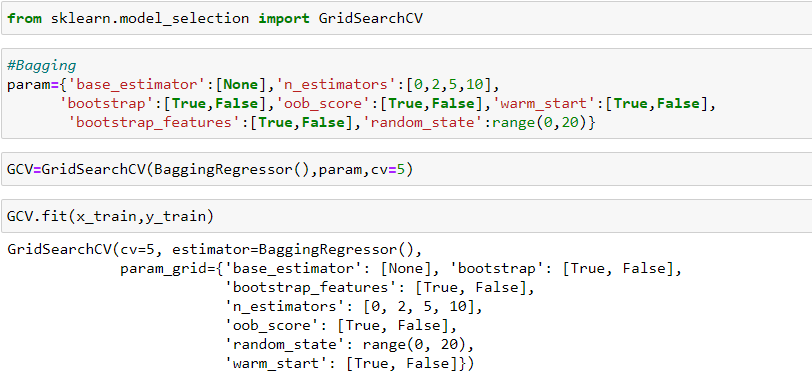
Highest accuracy I have got with BaggingRegressor but before selecting that as the final model let’s check the performance of all models with cross validation technique.



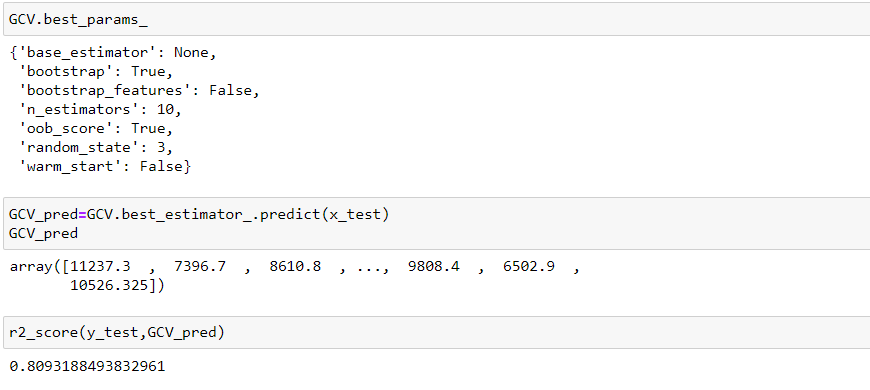
With the score of cross validation technique, I have checked which models actual performance and cross val score is more accurate which means the difference between actual score and cross validation score should be very less than that can be considered as best model. I Have got BaggingRegressor is the best model.

Let’s perform **hyper parameter tuning**:

I have tried to improve model performance with the help of Grid Search CV method. Let’s import the required library and create a dictionary with parameters which are suitable for the BaggingRegressor method.

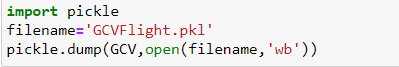


I have then called the model and tried fitting out train set.i.e. x=train (Input variables),y\_train (target variables). Let’s see the best fitted parameters found then predict output for input variables(x\_test)

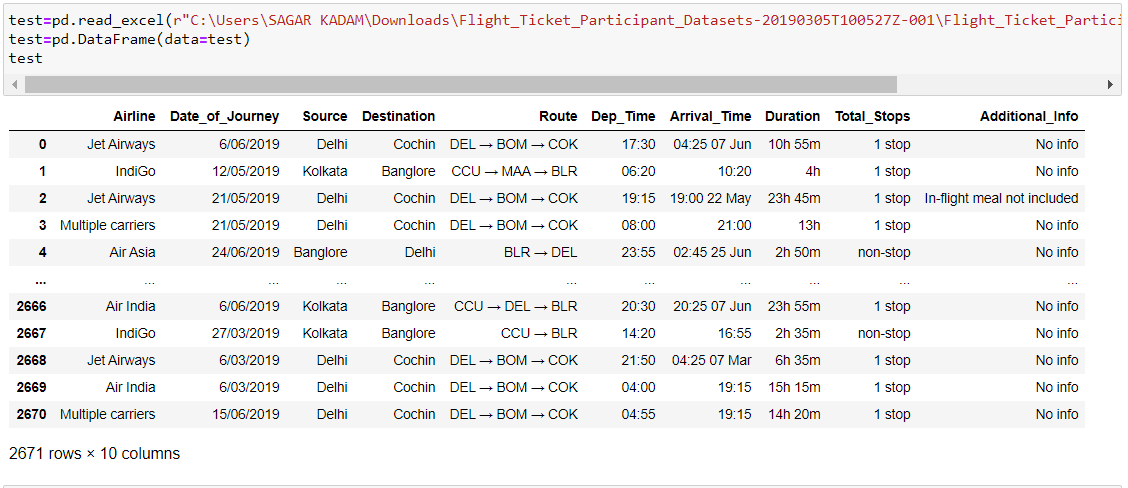


Then i have checked for accuracy again which we found as 80.93%

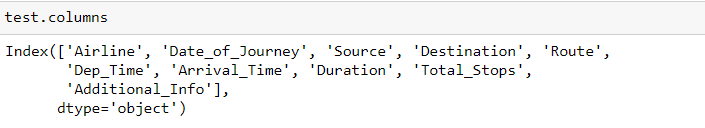
To use this model on future prediction or on a test dataset we need to save the model. I have the model using pickle library.



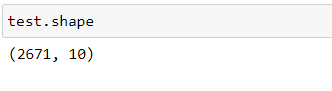
Then I have worked on test the dataset as before predicting prices for the test dataset we need to clear the data and view it. First, we need to import the dataset.



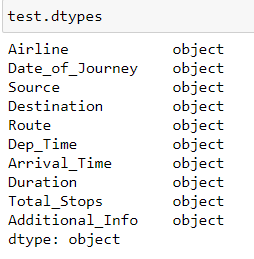
We can see the glance of the test dataset. Now Let’s see the list of all columns:



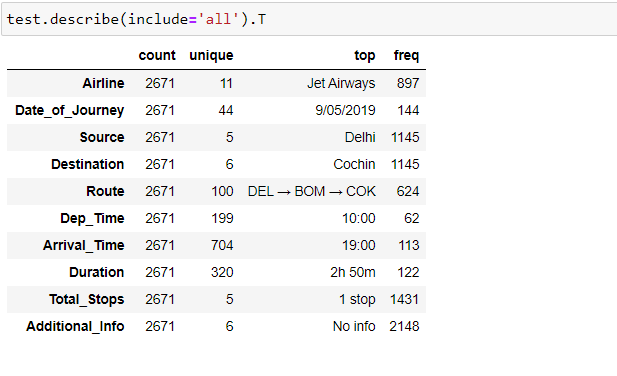
Now let’s see the total number of rows and columns:



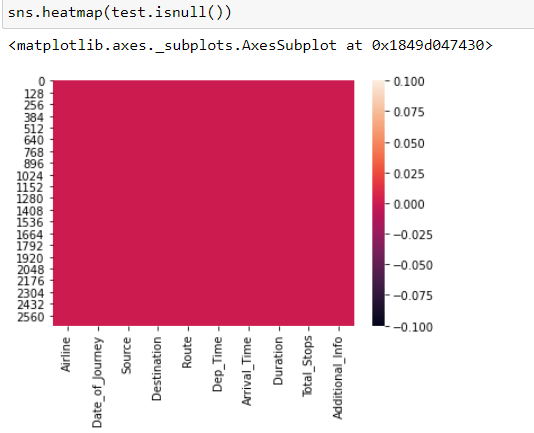
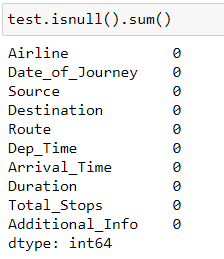
Our test dataset has 2671 rows and 10 columns as we need to predict the prices here. Let’s see datatype of all variables:



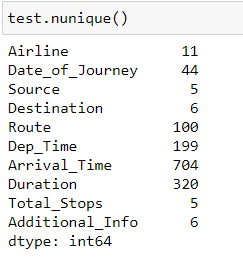
All columns of our dataset are with object datatype. Now I will give you the overview of the test dataset.



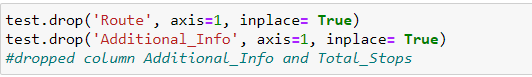
In test dataset we have 11 different airlines, journey happened on 44 different dates, 5 source place and 6 destination places, flights have flew from 100 different routes, we also have Departure and arrival time details of each flight with duration of total journey, maximum 5 total number of stops taken by one passenger from source to destination. There is a column named Additional\_info which does not contain much information as we have 2148 rows with No info. Let’s check null value, I will also show the same thing with the help of a heat map.



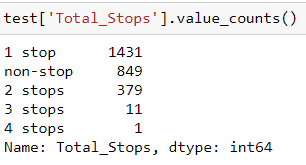
Then I have checked for unique values.



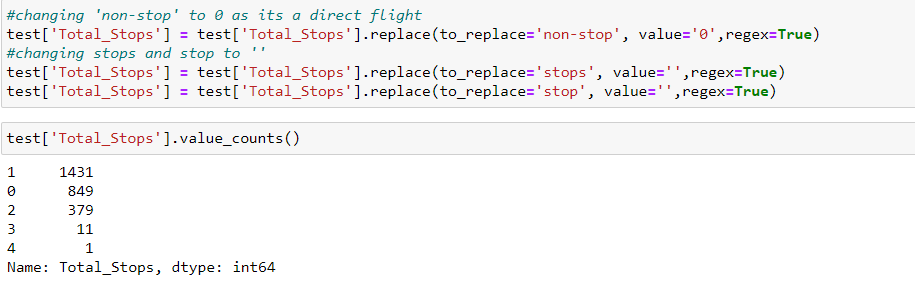
Here also Route column and Total stops column is giving us the same info in different medium as route column tells us that flight changed/stopped at all locations at the same time total stops is also giving us the information that how many stops/halts taken by a particular passenger. We can make change to total columns more conveniently as it has only 5 unique values and route has 100 hence, I am keeping total stops column and dropping routes columns. In the additional info column for 2148 rows, we don’t have any more information hence that column is also not much relevant. I have dropped both the columns.



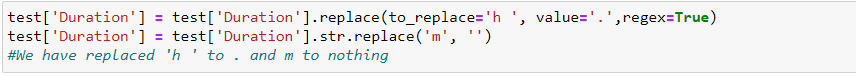
Then I worked on the Total stops column. We had 5 unique values, like train dataset:



I have removed words stop, stops and non-stop. Stop & stops to ‘’(blank or nothing), for non-stop I used 0(Zero) as it’s a direct flight.



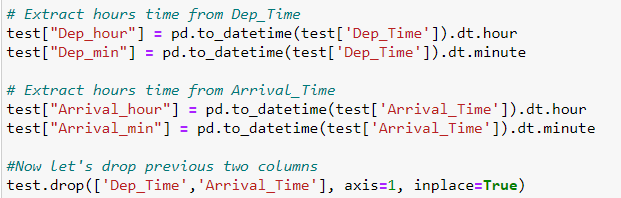
Then I worked on Duration which is giving us timing of total journey hours. I have used the same kind of process as I used in the train dataset.



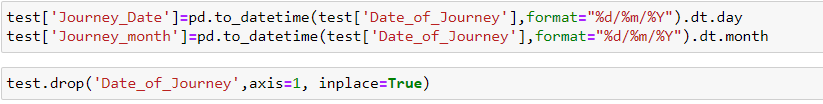




I have then changed data for columns departure time and arrival time. Extracted hours and minutes of departure and arrival both in different columns. After that, I dropped the previous 2 columns.



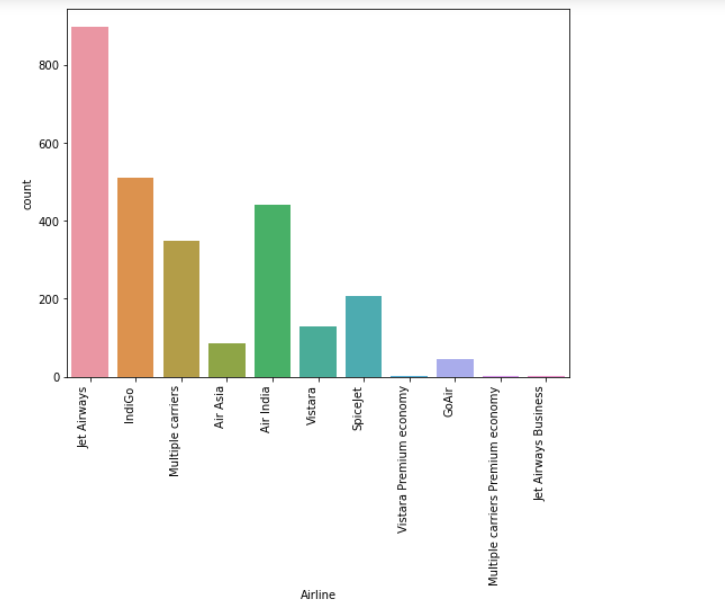
Then I have extracted the month and journey date in 2 columns, the year is already the same for all the rows. Then I dropped the Date of Journey column.



**Visualization:**

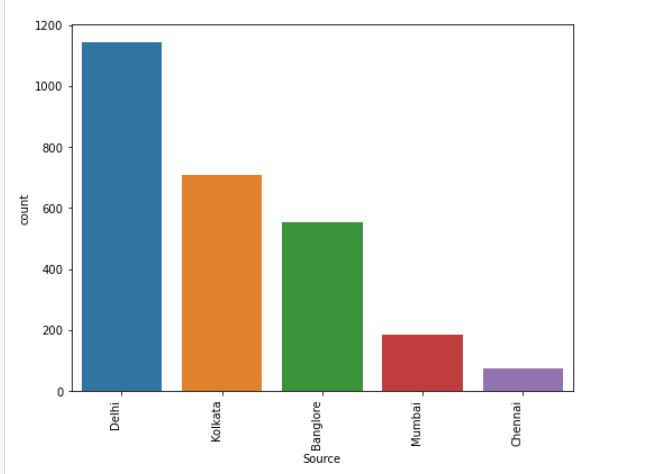
Here also I have first created a list of all object type columns for visualization and the columns are 'Airline', 'Source', 'Destination', 'Total\_Stops'. First, I used countplot to view counts of unique values of each column.

**Airlines:**



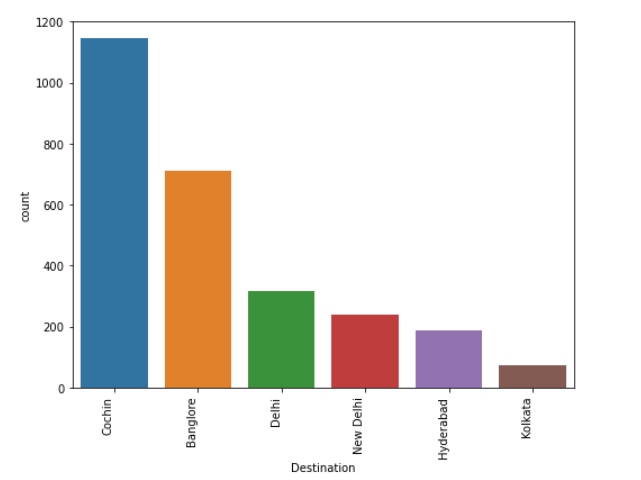
Maximum flights are of Jet Airways then we have IndiGo, Air India and Multiple carriers.

**Source:**



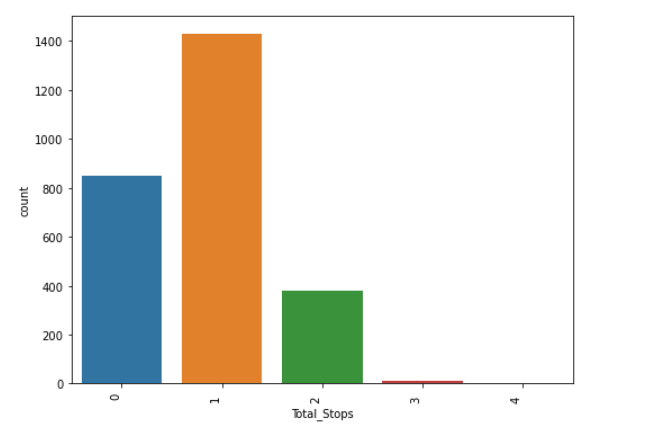
1145 flights flied from Delhi, 710 flied from Kolkata, 555 flied from Banglore, From Mumbai 186 and lastly from Chennai 75.

**Destination:**



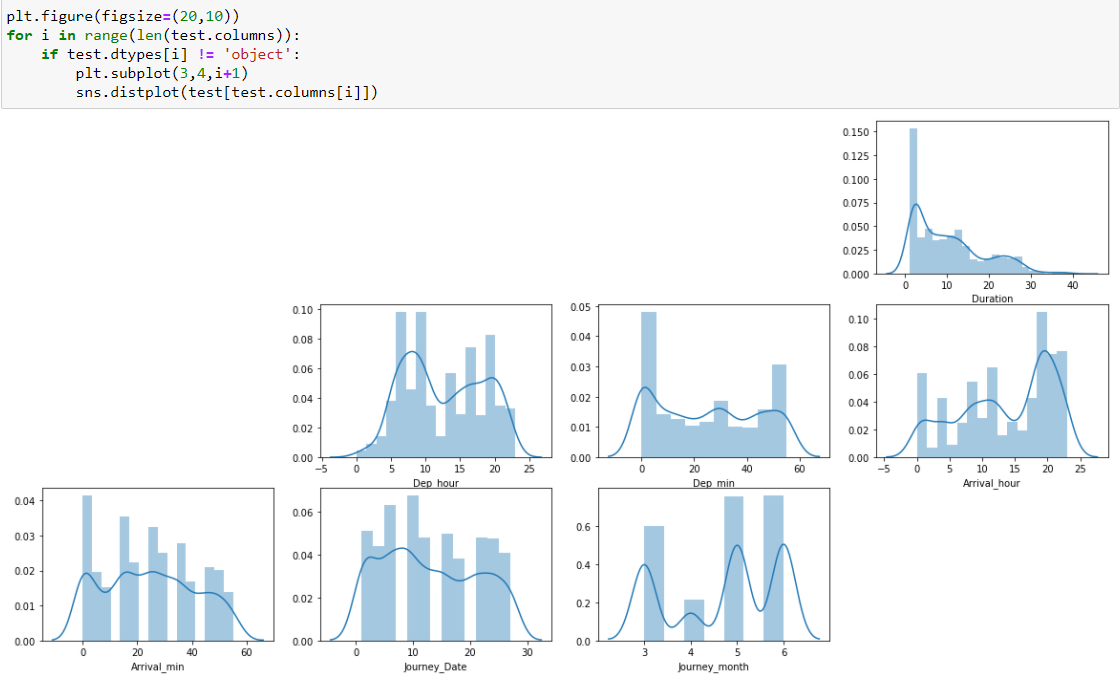
1145 flights arrived at Cochin, 710 arrived at Banglore, 317 flights arrived at Delhi, 238 arrived at New Delhi, 186 arrived at Hyderabad and lastly 75 arrived at Kolkata.

**Total\_Stops:**

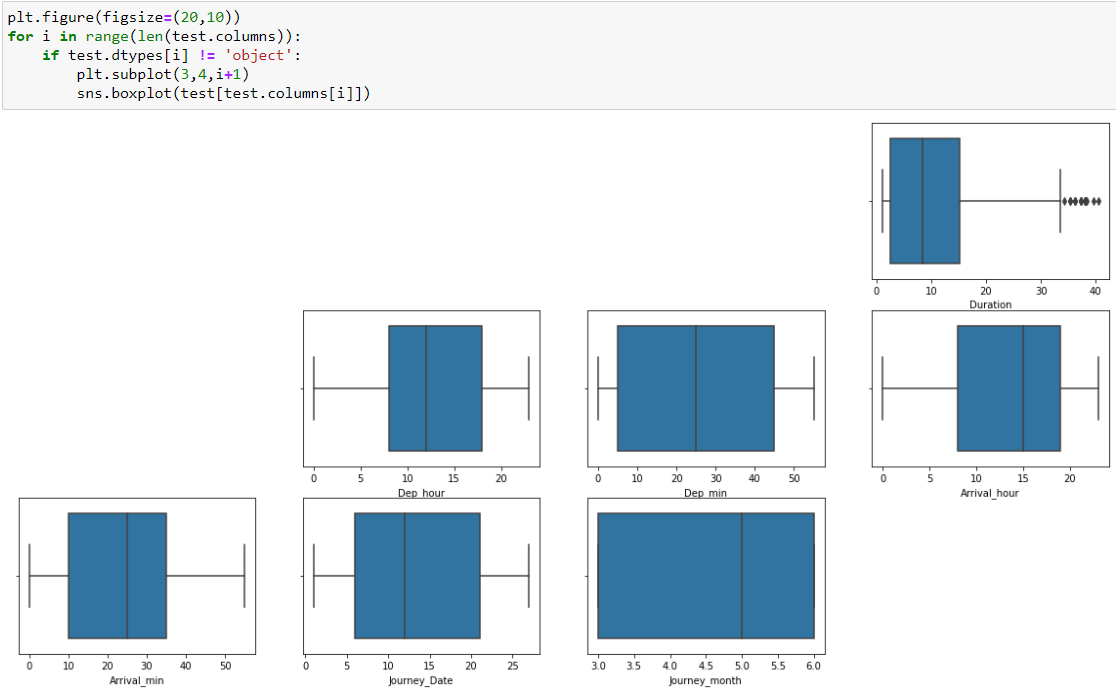


Maximum flights were with one stop i.e., 1431 flights, 849 flights were direct, 379 flights were with 2 stops, 11 flights with 3 stops and only one flight had 4 stops.

Then I checked the **distplot** and **boxplot** for numerical columns.

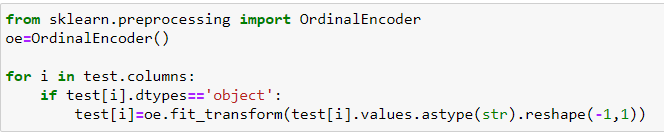


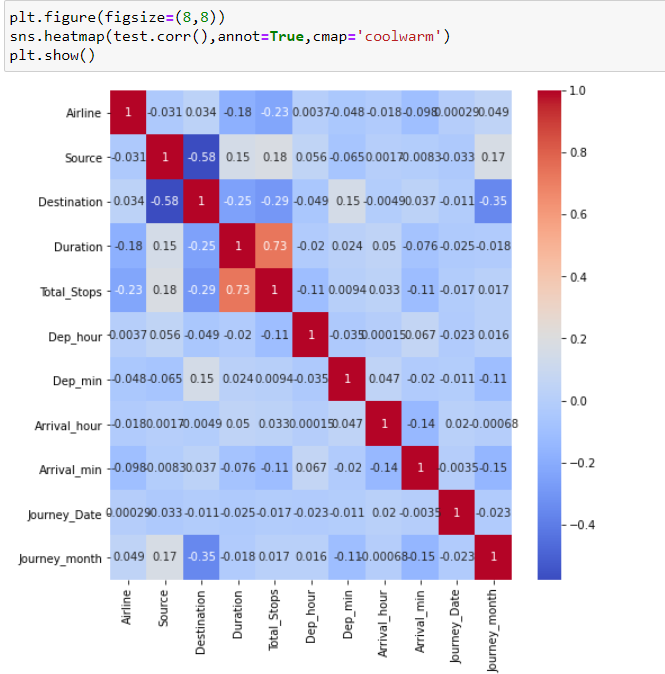
We can see uneven distribution in all columns.

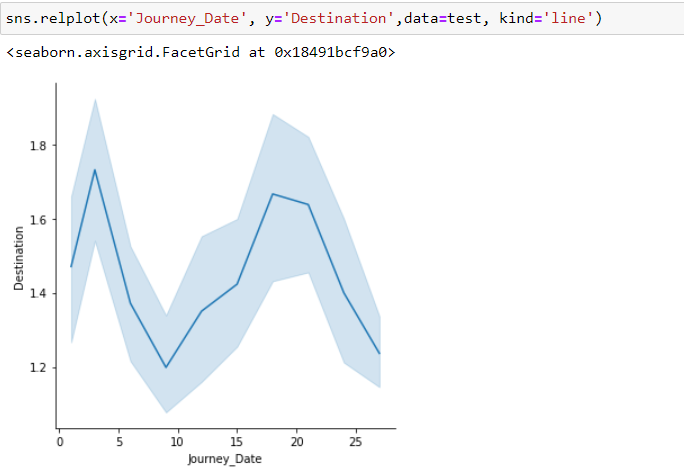


We have outliers in the Duration column.

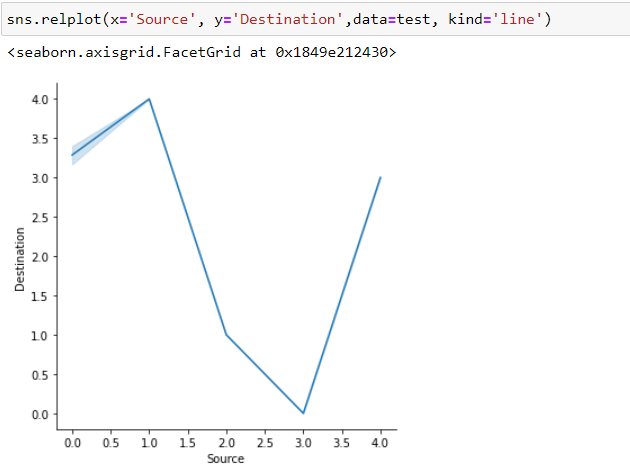
**Encoding:**



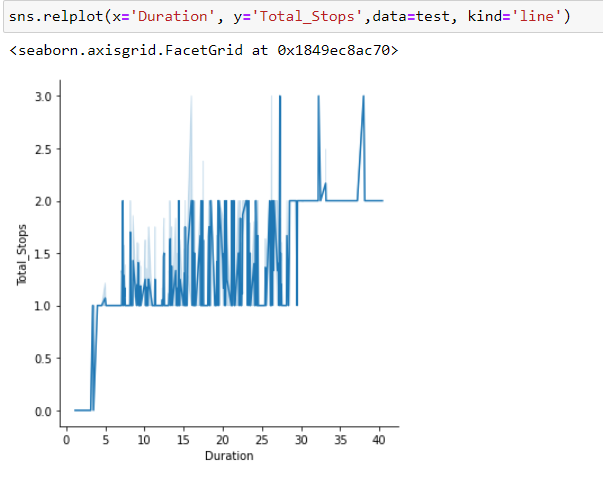




We can see a negative correlation between Journey data and Destination column.

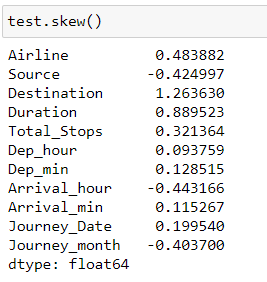


We can see a negative correlation between Source and Destination column.



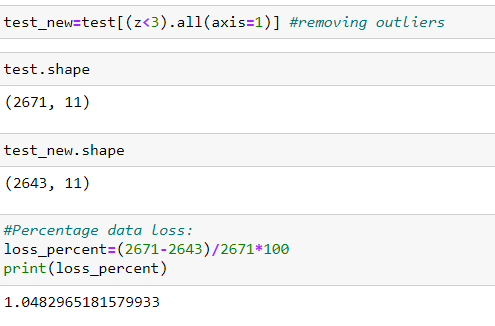
For duration of maximum 5 hours we have direct flights, for duration of maximum 30 hours we have 2 stops flights and for duration of 30 to 40 hours of flight we have 3 & 4 hours of flights.

**Skewness and Outliers removal:**



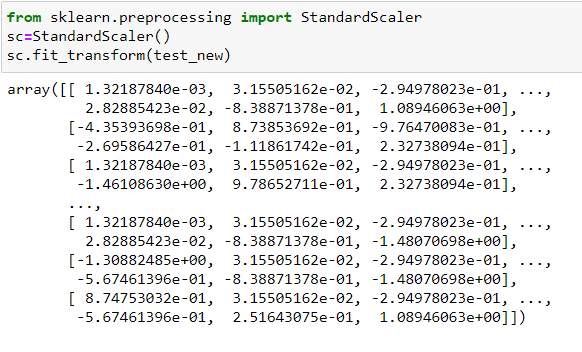
Keeping +/- 0.5 as the range for skewness however Except our target column all other columns were object type hence we will not consider the skewness.

I have used zscore method to remove outliers. I considered threshold value as 3, now removing outliers.

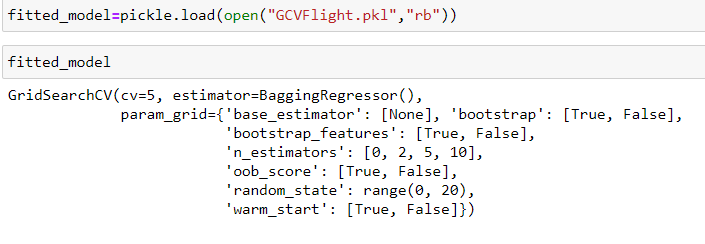


After removing the outlier my new dataset has 2643 rows and 11 columns. Data loss is about 1.05 %.

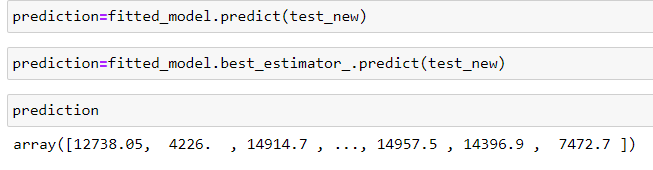
Then I have performed standardization. I have now used the **Standard scaler** method.



Then I have loaded the saved model with read binary mode.



Now I am trying to predict the flight price over a test dataset. Below is the output after prediction.



Summary

We started with the data exploration where we got a feeling for the dataset, checked about missing data and learned which features are important. During this process we used seaborn and matplotlib to do the visualizations. During the data preprocessing part, we computed missing values, converted features into numeric ones, and created a few new features for hours and minute of Departure and arrival both. Afterwards we started training 6 different machine learning models, picked one of them (BaggingRegressor) and applied cross validation on it. Then we looked at how BaggingRegressor performs, look a look at the importance it assigns to the different features and tuned it’s performace through optimizing it’s hyperparameter values. we have saved the model. Then I have worker on train dataset and did the data preprocessing part, we computed missing values, converted features into numeric ones, and created a few new features for hours and minute of Departure and arrival both. Lastly, then recall the model and then predict the data over test dataset.