Flight Price Prediction

**Content:**

1. Introduction
2. Problem Statement



1. Important Libraries used to build the model.



1. Data Analysis
2. EDA (Exploratory Data Analysis)



1. Pre-Processing Pipeline
2. Building Machine Learning Model
3. Conclusion



**Introduction**

Airline industry is one of the most sophisticated in its use of dynamic pricing strategies to maximize revenue, based on proprietary algorithms and hidden variables. That is why the airline companies use complex algorithms to calculate the flight ticket prices.

Nowadays, the number of people using flights has increased significantly. It is difficult for airlines to maintain prices since prices change dynamically due to different conditions. That’s why we will try to use machine learning models to solve this problem. This can help airlines by predicting what prices they can maintain. It can also help customers to predict future flight prices and plan their journey accordingly.

2**. Problem Statement**

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.

The problem statement explains that the target variable is continuous and it’s a **“Regression type problem”** since we need to predict the price of the flight tickets. In this project we will be using many regression models that can help the consumers to make purchasing decisions by predicting how flight ticket prices will evolve in the future.

Attribute Information:

**Airline**: The name of the airline.

**Date\_of\_Journey**: The date of the journey

**Source**: The source from which the service begins.

**Destination**: The destination where the service ends.

**Route**: The route taken by the flight to reach the destination.

**Dep\_Time**: The time when the journey starts from the source.

**Arrival\_Time**: Time of arrival at the destination.

**Duration**: Total duration of the flight.

**Total\_Stops**: Total stops between the source and destination.

**Additional\_Info**: Additional information about the flight

**Price**: The price of the ticket

This is the dataset having 11 attributes with its description where “**Price**” is the target or dependent variable whereas the rest features are independent variables.

3. **Important Libraries used to build the model:**

a. **NumPy** : NumPy can be used to perform a wide variety of mathematical operations on arrays. It adds powerful data structures to Python that guarantee efficient calculations with arrays and matrices and it supplies an enormous library of high-level mathematical functions that operate on these arrays and matrices.

b. **Pandas**: Pandas is a Python library is used to analysis the data in different variation.

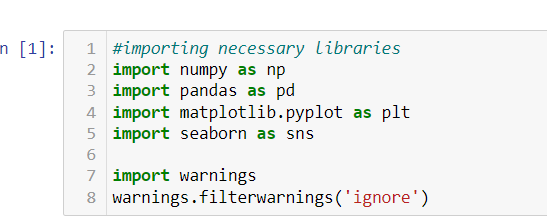
**c. Matplotlib**: Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy.

**d. Seaborn:** Seaborn is a Python data visualization library based on matplotlib which provides a high-level interface for drawing attractive and informative statistical graphics.

Above are the main libraries which is used to analyze the Data.

Later on, for building a model we have used several libraries such as Regression Algorithms**,** Random Forest Regressor, Decision Tree Regressor, Bagging Regressor etc.







# 4. Data Analysis

# Lets, understand the types of Data Sets, There are two Data Sets



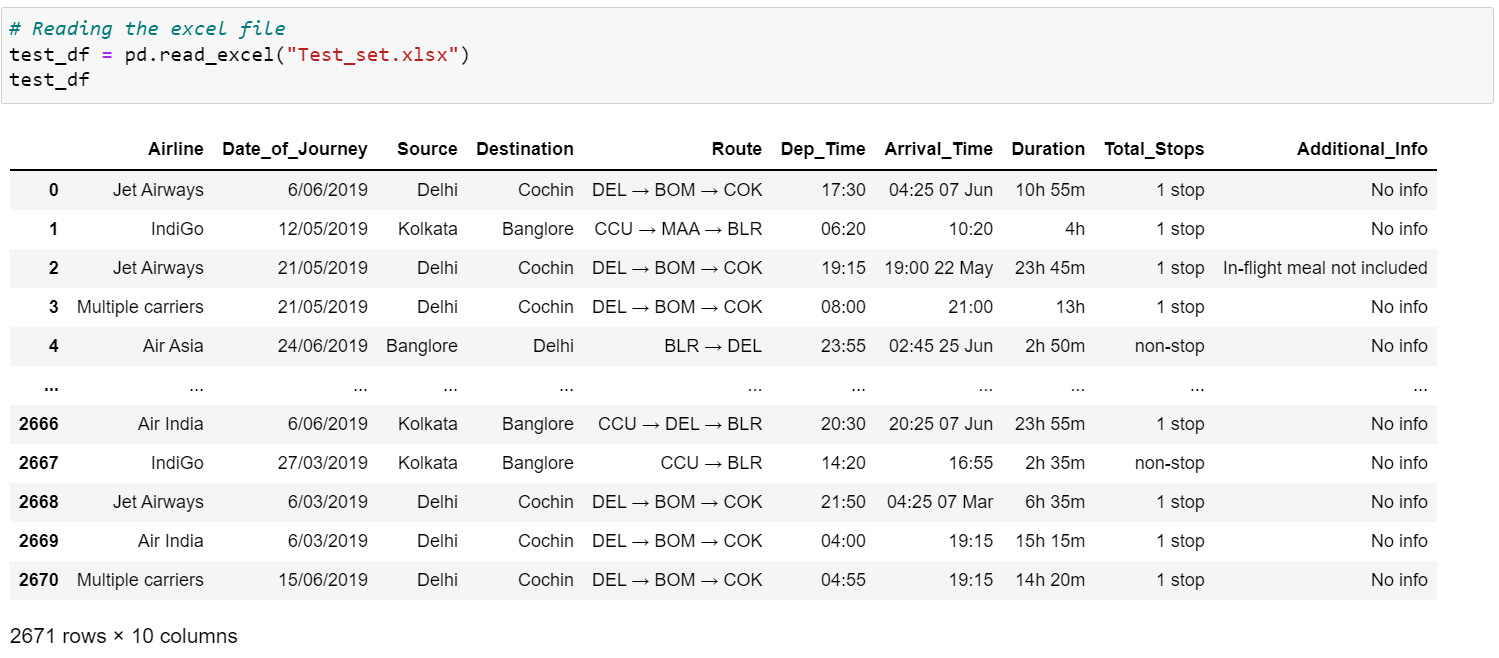
# Train Data sets.

# Test Data sets.

# Train Data set.

****

1. Test Data set.



* **Date\_Of\_Journey:**

Even though the problem statement specifically mentioned about the months, but there are no particular columns for months. So, let’s extract the values of month and day from date of journey column and make a separate column for them to study the prices of flight tickets for various airlines based on month.

* **Arrival\_Time & Dep\_Time:**

The arrival time and the departure columns contain date mentioned with time, so we need to format that too keeping separate columns for that.

* **Duration:**

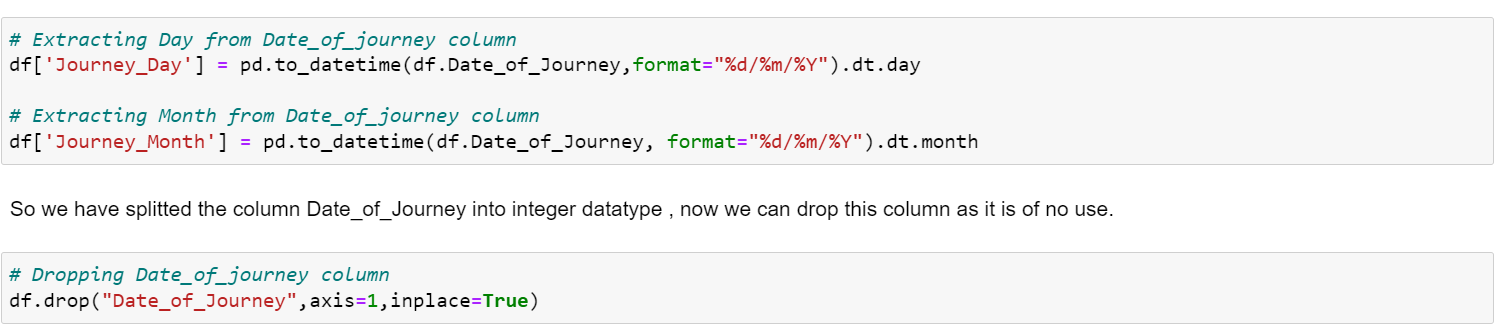
Duration is the difference between the arrival time and departure time. Since the duration column contains both hours and minutes data, we are going to extract the values from this column.



We have converted the object type data into datetime data type. Now let’s extract the values from respective columns**.**

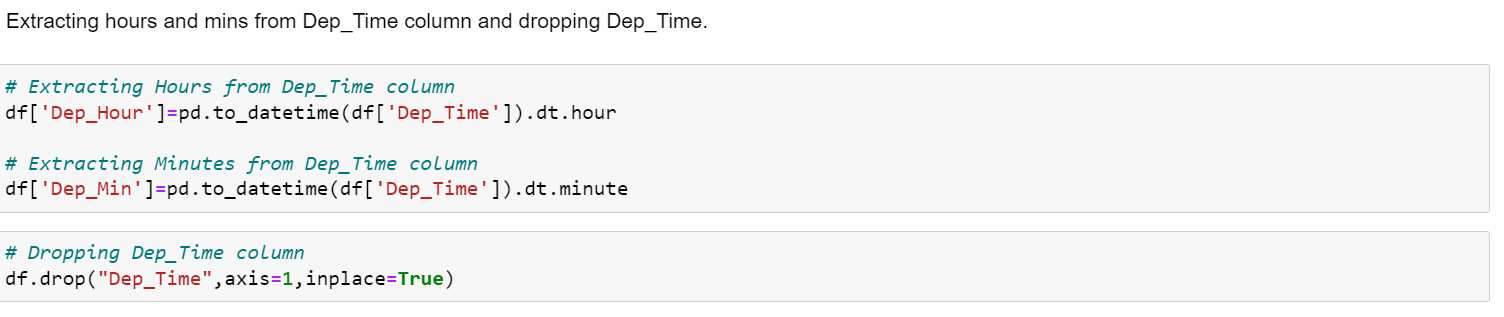
**Date\_of\_Journey:**

Now splitting Date\_of\_journey into Month and Day, and as the dataset contains only 2019-year data so no need to take year column.

****

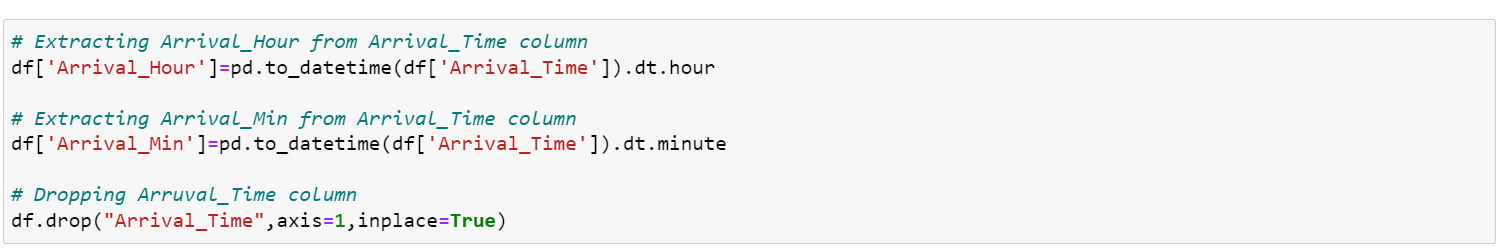
**Dep\_Time:**

Departure time means when a flight leaves the airport and this column contains hours and minutes so we will extract hours and minutes from Dep\_Time and dropping Dep\_Time column.



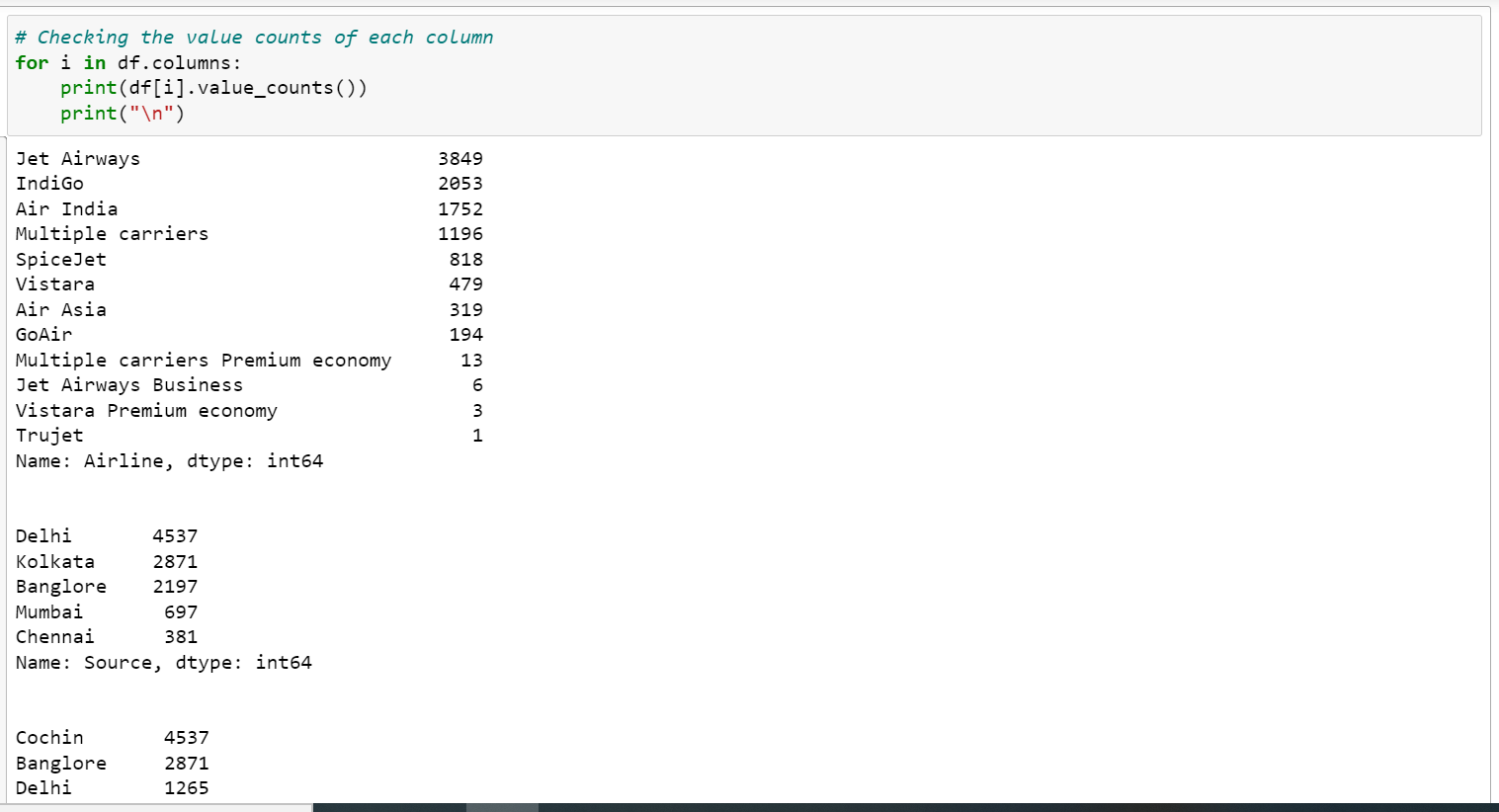
**Arrival\_Time:**

Similarly, we can extract hours and minutes from Arrival\_Time column and accordingly dropping Arrival\_time column.

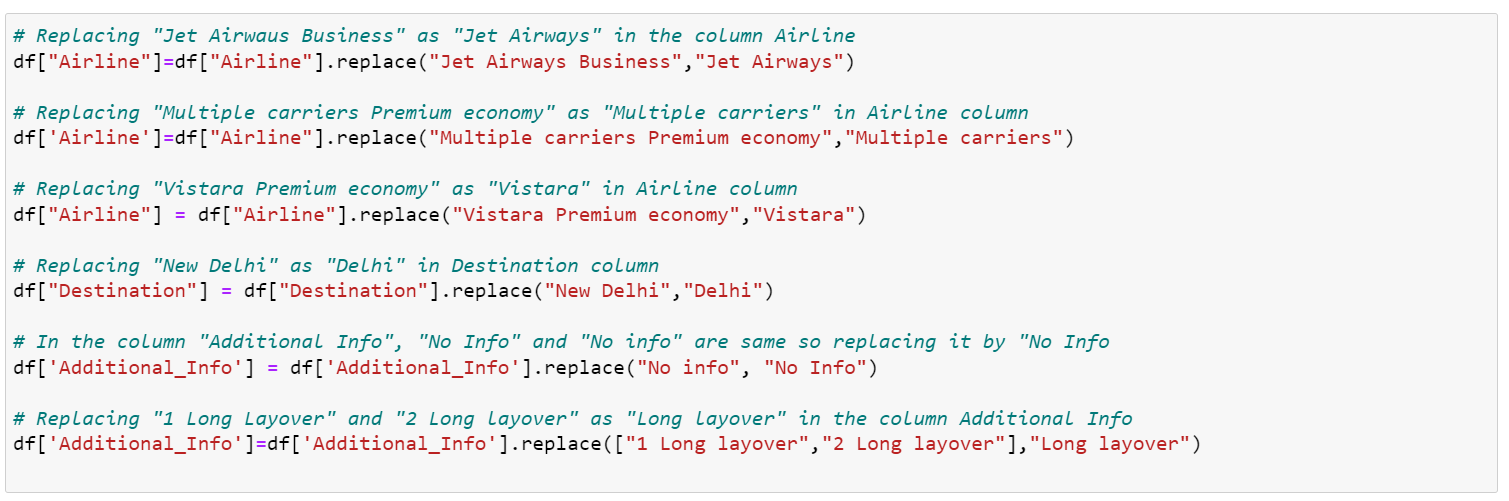


**Duration:**

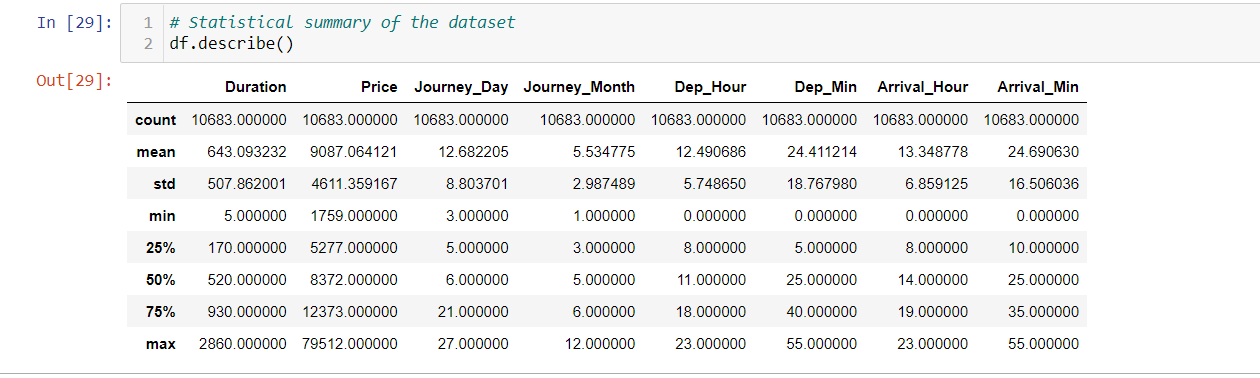
The column Duration has values in terms of minutes and hours. Duration means the time taken by the plane to reach the destination. It is basically the difference between arrival and departure time. Instead of extracting hours and minutes from this column let’s count up the time and format it to numerical data variable.

After dealing with datetime datatype variables, let’s check the other variable for having some repeated categories using value\_counts ()method.  


The columns Airline, Destination and Additional Info found to have some repeated categories. Let’s replace them.

****

After performing feature engineering let’s check the description of dataset.



This gives the statistical information of the dataset. The summary looks perfect since there is no negative/invalid values present. It gives the summary of numerical data.\ From the above description we can observe the following things.

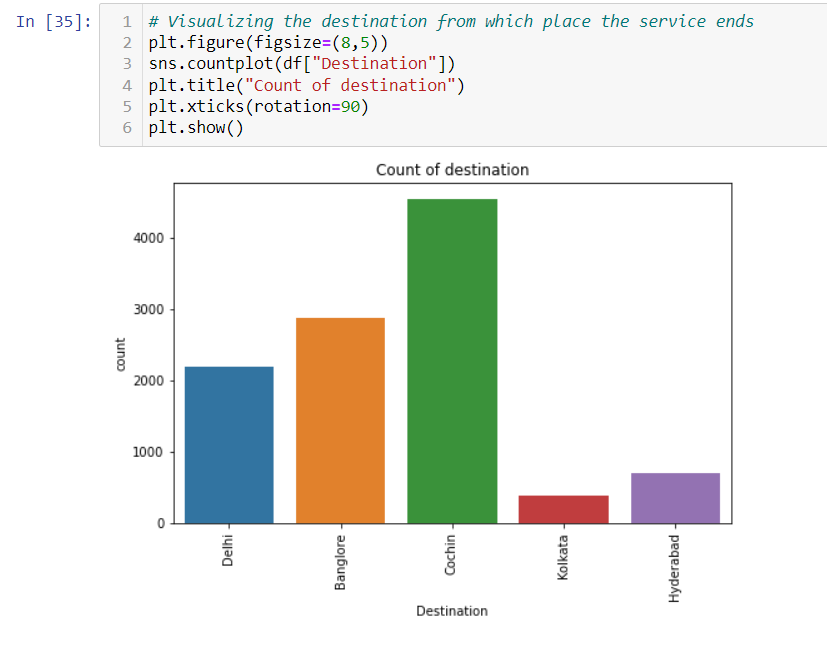
* The counts of all columns are same which means there are no missing values present int he dataset.
* The mean value is greater than the median(50%) in the columns Price,Journey\_Day,Duration and Dep\_Hour so we can say they are skewed to right.
* The median(50%) is bit greater than mean in Dep\_Min,Arrival\_Hour and Arrival\_Min which means they are skewed to left.
* From the description we can say the minimum price of the flight tickets is Rs.1759 and maximum price is Rs.79512 and the mean is 9087.
* Also there is a huge difference in maximum and 75% percentile in the columns Price, Arrival\_Min which leads to outlies in the columns.
* The std of target variable is high which means it has high rate of dispersion.

# Data Visualization:

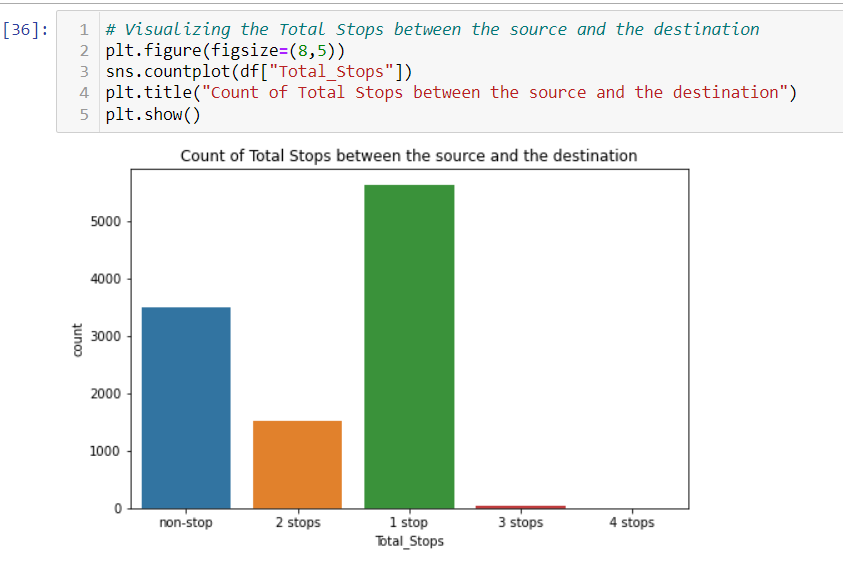
# *Univariate Analysis*:

# 

* Jet Airways flights has high counts whereas Trujet and GoAir has the least counts.
* The majority of Airline source is from Delhi while the least is from Chennai.



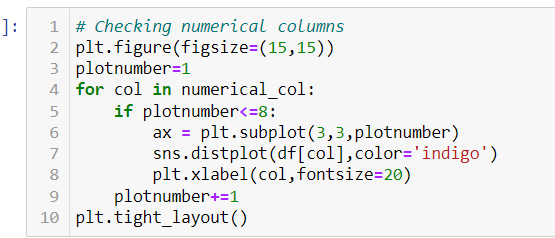
The Cochin destination has highest counts. Most of the flights services ends in Cochin destination.

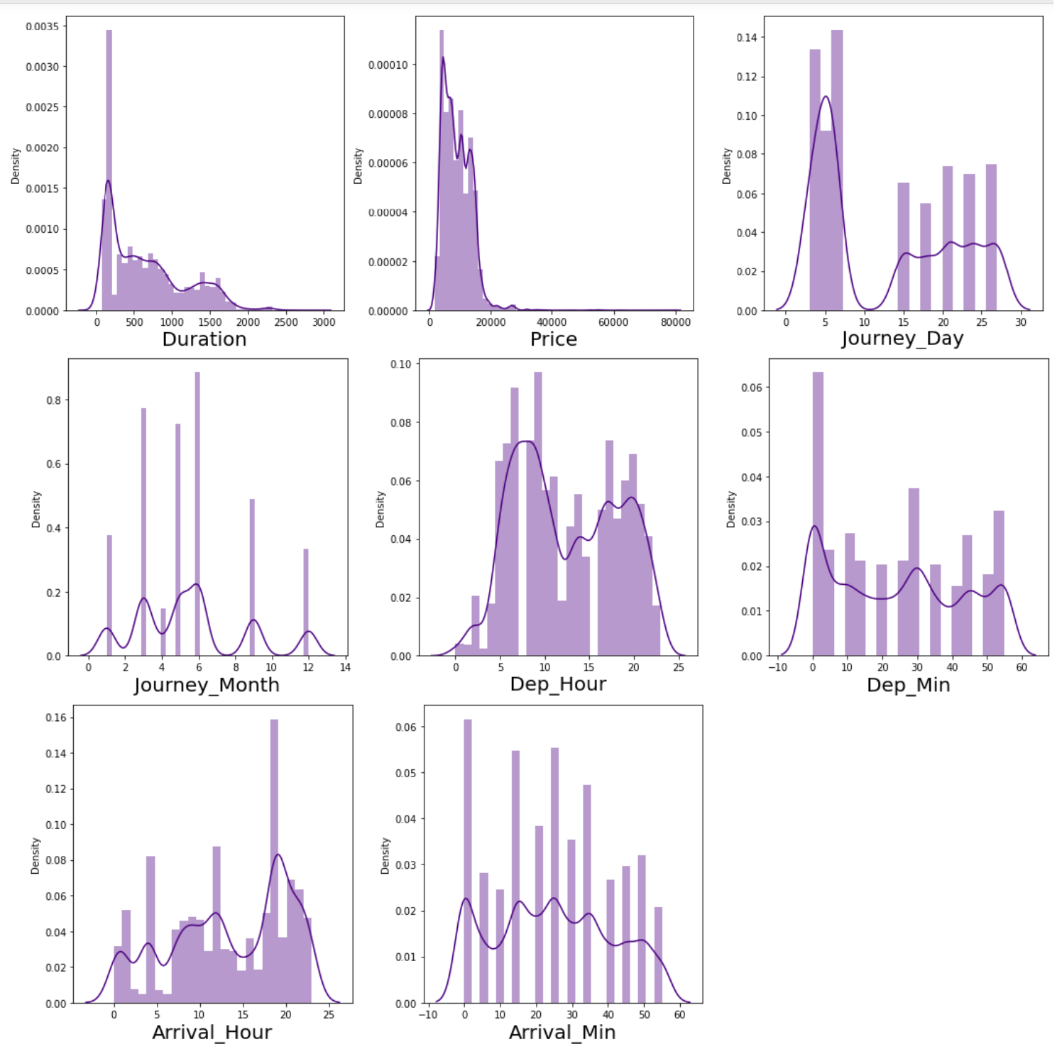


Majority of flights has 1 stop between the source and destination, followed by non-stop. No flights have 4 stops between the source and destination.

**Distribution of skewness**

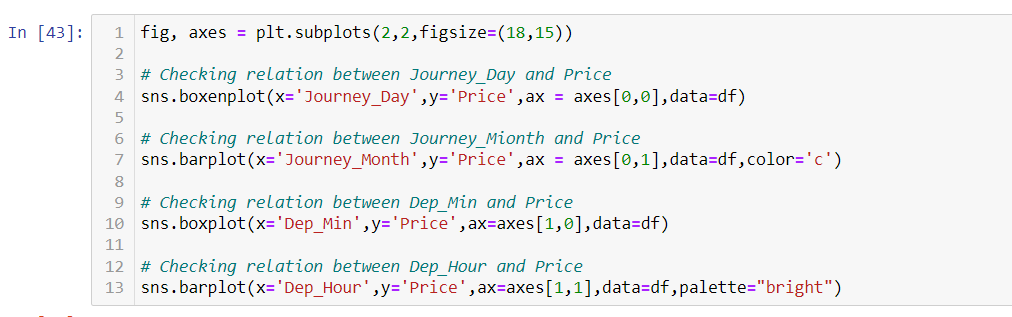
*Plotting Categorical column*:

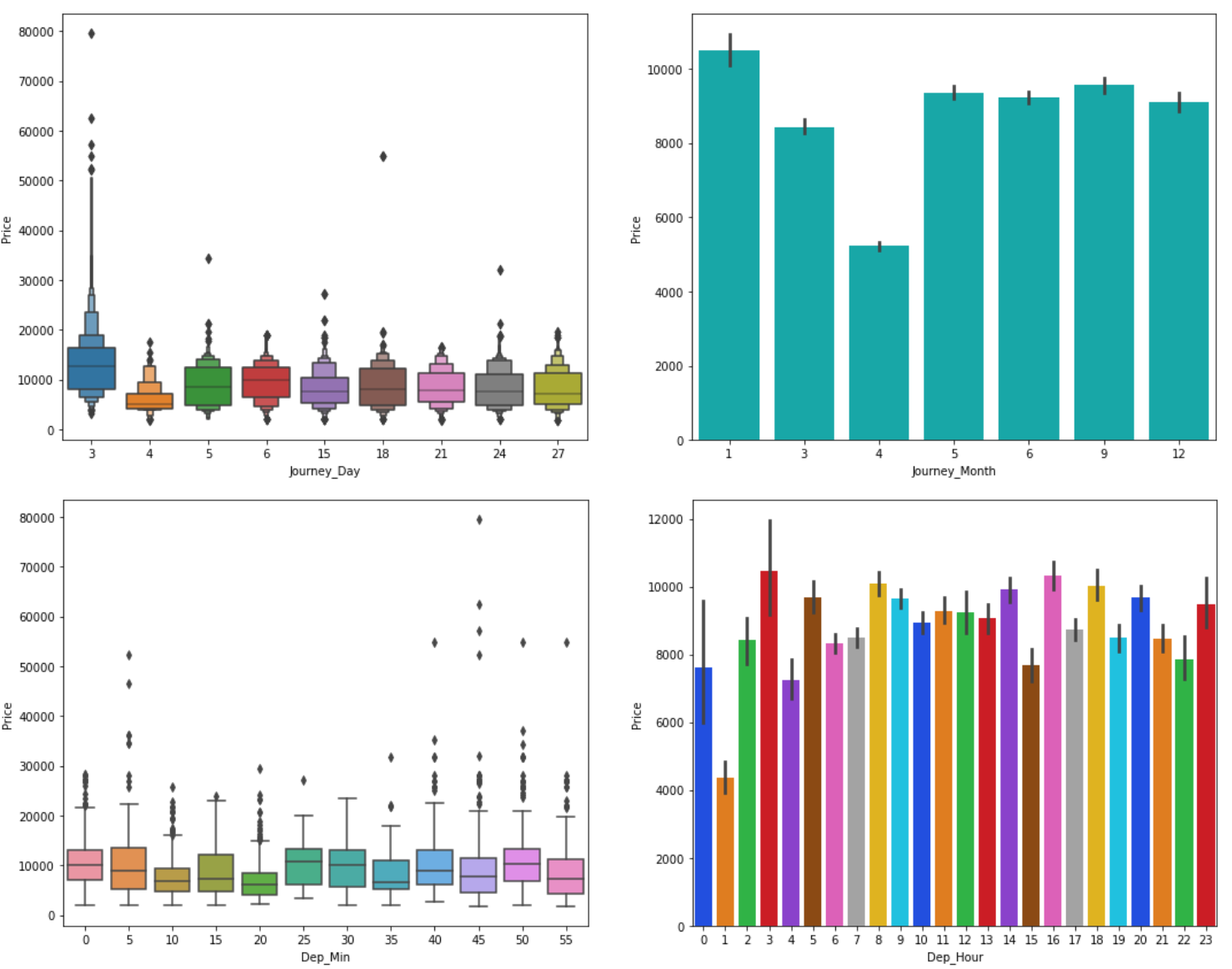




From the distribution plot we can observe that the data i not normally distributed in some columns and some columns are almost normal but have no proper bell shape curve. The Journey\_Month, Duration and Price columns are skewed to right as mean is more than the median.

**Bivariate Analysis**



****

From the above plots we can observe the following

* While comparing Journey Day and Price we can see the price of ticket is high in Day 3 apart from this there is no much impact of day on ticket price.
* While comparing Journey\_Month and Price we can state that the flights travelling in January month are more expensive than others and the flights traveling in April month have very cheap ticket prices.
* There is no significance relation between Dep\_Min and Price of the tickets.
* In the fourth graph also we can say that there is no much impact of Dep\_Hour on Price

# Checking for Outliers:

# 

# 

# 

# 

* The outliers are present in the columns "Duration”, Journey\_Month" and the target variable "Price".
* Since "Price" is the target column so no need to remove outliers from this column.
* We need to remove the outliers from the other two columns using zscore or IQR methods.

# Removing Outliers using Zscore method:

# 

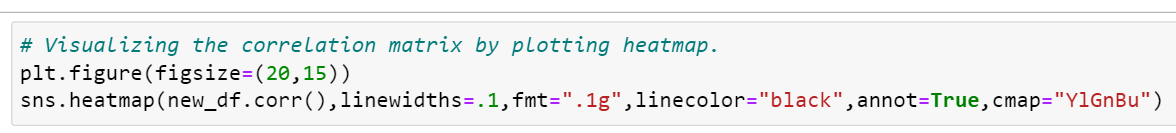
# We have created a new DataFrame which shows the first five Data from the Data Sets.

# 

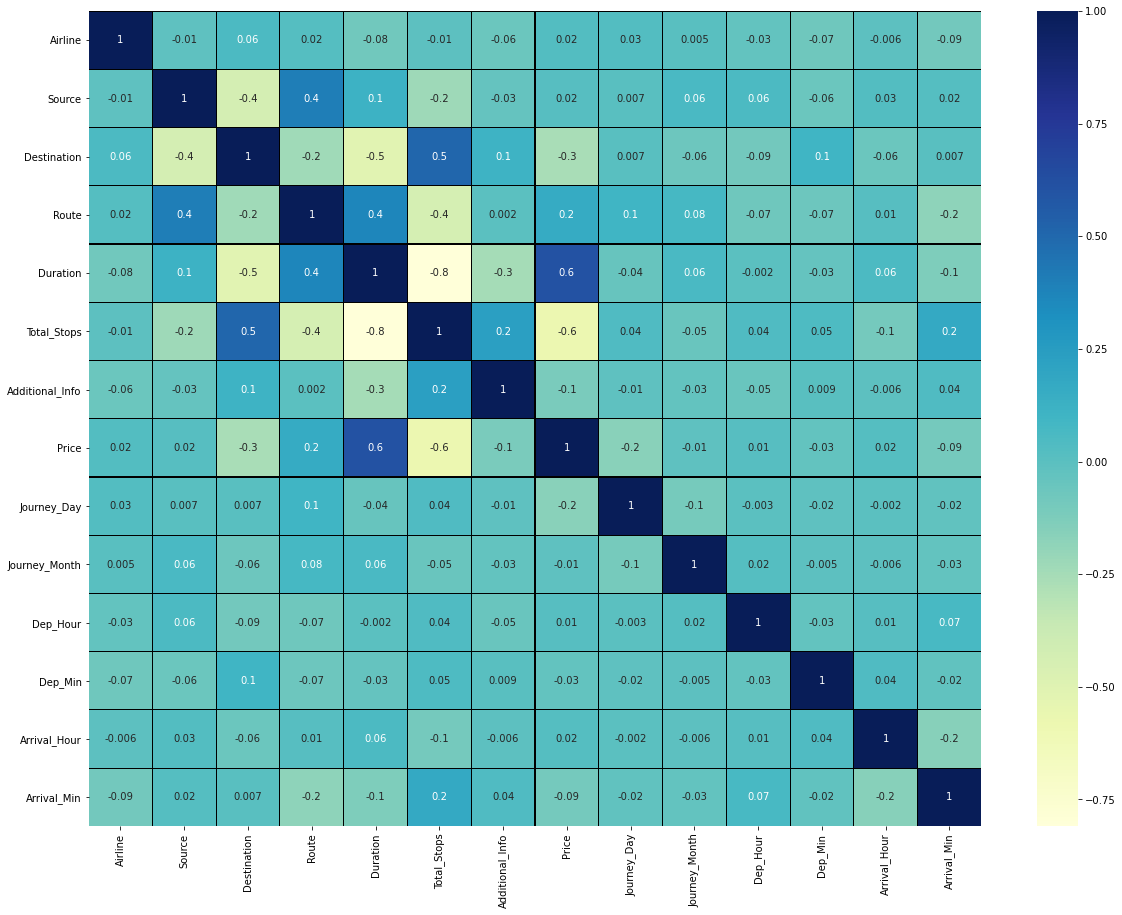
This is the new data frame after removing outliers. Here we have removed the outliers whose zscore is less than 3.

**Checking the Correlation:**

Data Correlation is a way to understand the relationship between multiple variables and attributes in your dataset. Using Correlation, you can get some insights such as: One or multiple attributes depend on another attribute or a cause for another attribute

****

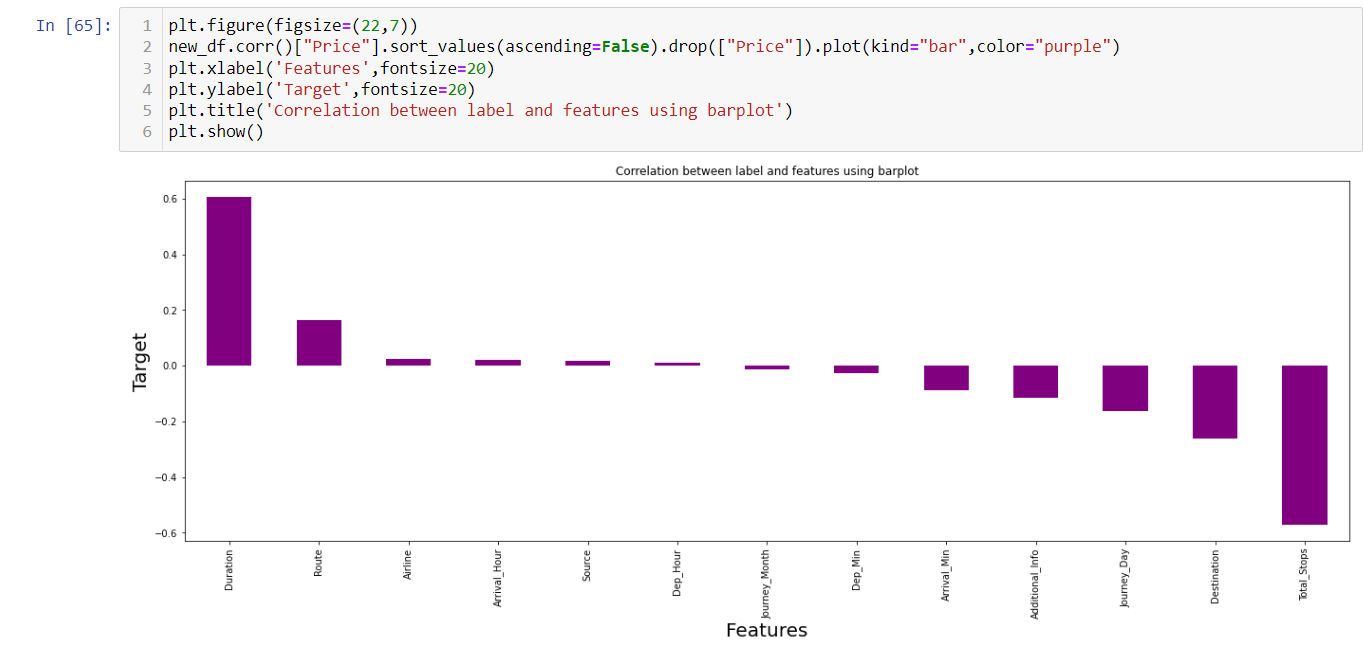
This gives the correlation between the dependent and independent variables. We can visualize this by plotting heatmap.

****

This heatmap shows the correlation matrix by visualizing the data. We can observe the relation between one feature to other.

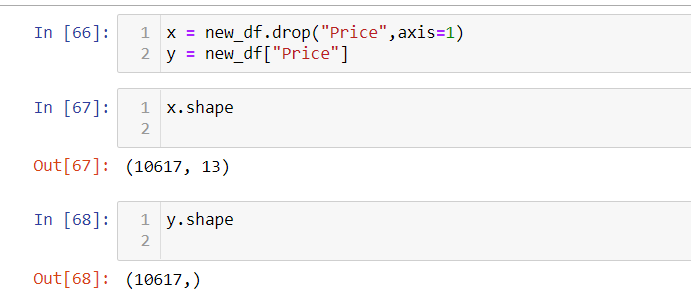
* This heatmap contain both positive and negative correlation.
* The feature Duration is highly positively correlated with the target variable "Price".
* The feature Total\_Stops is highly Negatively correlated with the label.
* The features Duration, Total\_Stops and Destination are highly negatively correlated with each other. This may lead to multicollinearity problem, we will check vif values to avoid this.

# Visualizing the correlation between features and label using bar plot.

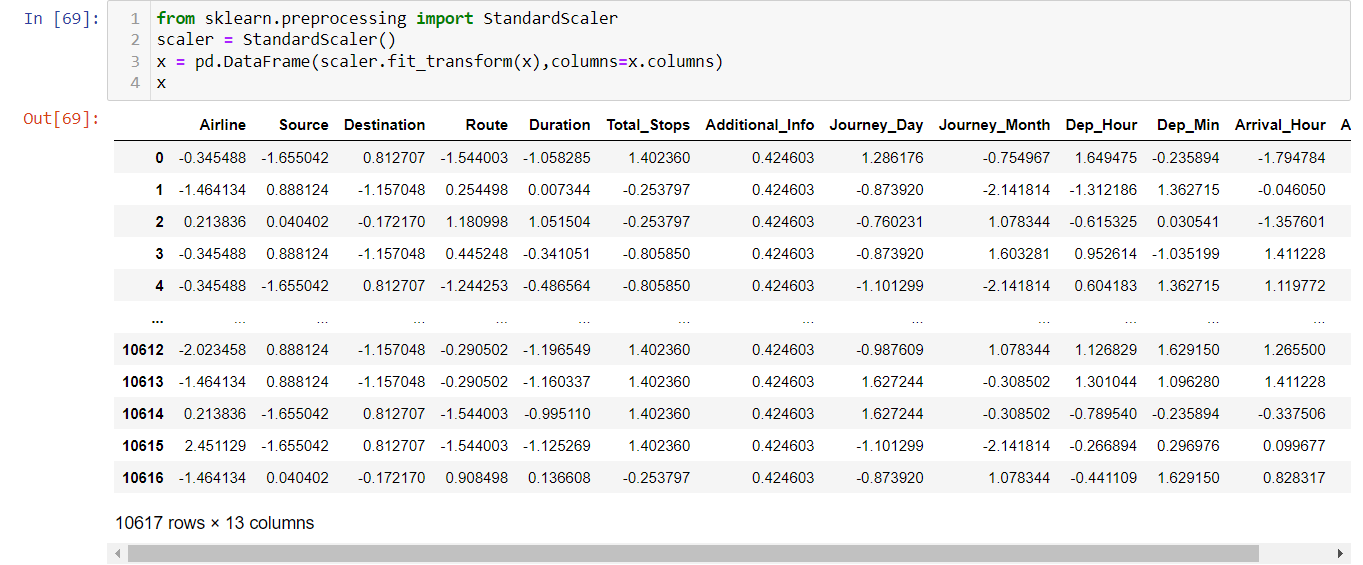


The features Journey\_Month, Source, Arrival\_Hour,Dep\_Hour and Airline have very less correlation with the label so we can drop thee columns.

# Separating the features and label.



# Feature Scaling using Standard Scalarization



We have scaled the data using Standard Scalarization method to overcome the problem of biasness.

# Checking variance inflation factor(VIF).

# 

# As we can notice the vif values is less than 10 in all the columns, there is no multicollinearity exists. We can move ahead for model building.

**Model Building:**

Building machine learning models that have the ability to generalize well on future data requires thoughtful consideration of the data at hand and of assumptions about various available training algorithms.

# The various Regression algorithms I used to build model in this dataset are:

# Random Forest Regressor

# Decision Tree Regressor

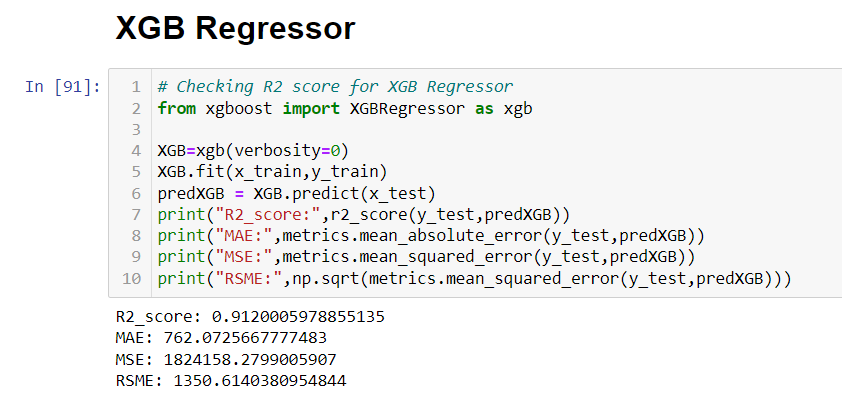
# Gradient Boosting Regressor

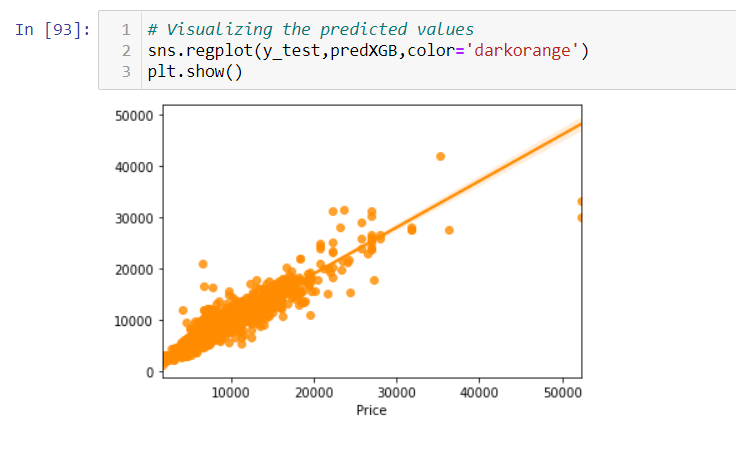
# Bagging Regressor

# Extra Trees Regressor

# XGB Regressor

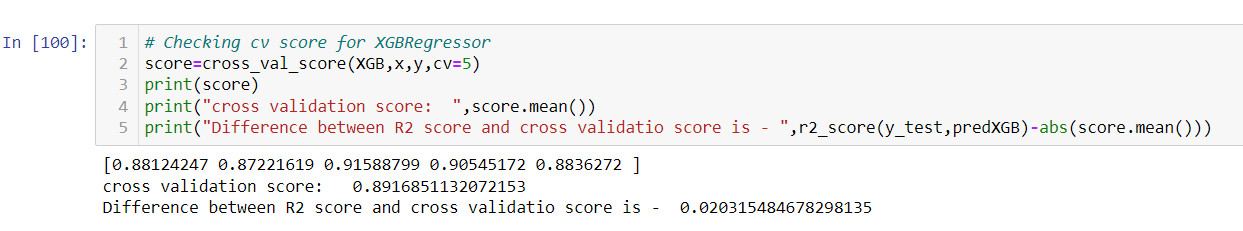
Among all those regression models we have analysed and found that XGB Regressor have highest accuracy i.e. 91.20%% with 89.16% cross validation score which is good and the difference is too less shown in fig below.





The predicted R2 score using XGB Regressor is 91.20% which is highest among all the regression models.

# Checking the cross validation score:



XGB Regressor model have highest accuracy i.e. 91.20%% with 89.16% cross validation score which is good and the difference is too less.

Model selection refers to the process of selecting the right model that fits the data. This is done using test evaluation matrices. The results from the test data are passed back to the hyper-parameter tuner to get the **most optimal hyperparameters**.

# Hyperparameter Tuning:

# 

# After tuning, the best R2 score is 91.20%.

# We have perform hyper parameter tuning to get the best regression model which fits into the data.

# Saving the model:

# 

# Predicting the saved model:

# 

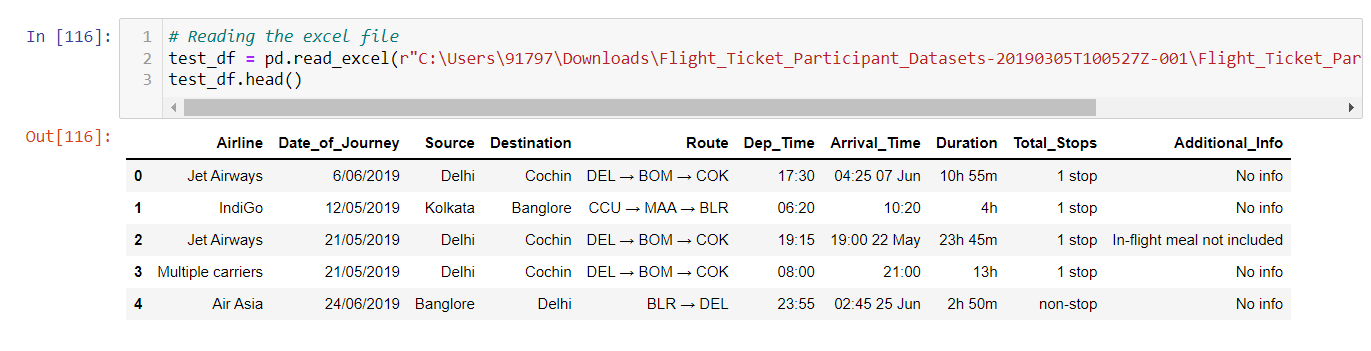
# Prediction Visualization:

# 

# 

As we have already loaded the saved model after saving the final model. Now we can predict the flight price using test data.

**Imported the Test Data**.

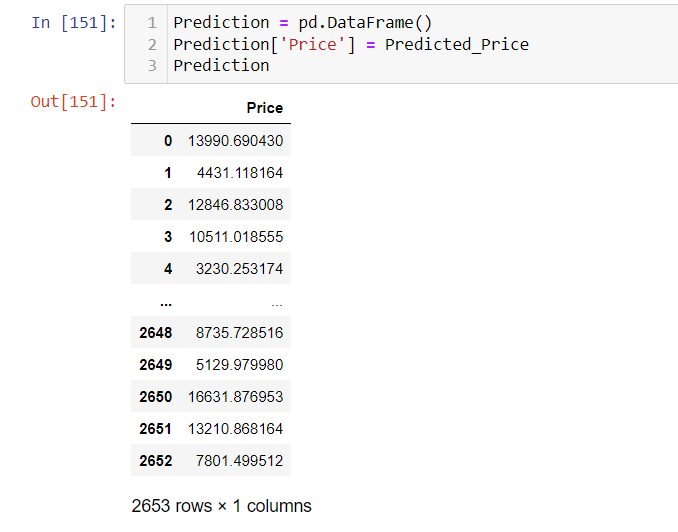


# Prediction Result:

# 

# Here we are using predicted final model which is saved as Model to predict the price of the test dataset.

# Creating Data Frame and Saving the Predictions:



This is the final Predicted price of various flight.

**Conclusion:**

Prediction of flight ticket price hinges on several influences which we visualized as features.

The impactful inferences which assisted for the predictions of flight prices are as follows:

* “**Duration**” is the influencing feature and highly correlated for predicting the flight ticket prices.
* **Jet Airways** is most affluent airline followed by Multiple carriers and Air India.
* Flights with **4 stops** have highest price followed by flights having 3 stops and the flights which have no stops costs very less ticket price compared to others.
* The “**Business class**” flights are more luxurious compared to others and the flights having the class “No check-in baggage” included has very least ticket price.

*Git Hub Link*: <https://github.com/ShubhMeshra/Evaluation-Projects/tree/main/Flight%20Price%20Prediction%20Blog>

**THANK YOU**

**-Shubham Meshram**