**Flight Price Prediction**

A machine learning model is like a computer software designed to recognize patterns or behaviors based on previous experience or data. The learning algorithm discovers patterns within the training data, and it outputs an ML model which captures these patterns and makes predictions on new data.

A picture containing outdoor, sky, plane, airplane

Description automatically generated I am going to build a machine learning model on “**Flight Price Prediction**” dataset.

A dataset in machine learning is simply, **a collection of data pieces that can be treated by a computer as a single unit for analytic and prediction purposes.**

This means that the data collected should be made uniform and understandable for a machine that doesn’t see data the same way as humans do.

**Steps for Preparing Good Training Datasets**

1. Articulate the problem early: The initial step is to pinpoint the set of objectives that you want to achieve through a machine learning application.
2. Establish data collection mechanisms. ...
3. Check your data quality.
4. Format data to make it consistent.
5. Reduce data.
6. Complete data cleaning.
7. Create new features out of existing ones.
8. Join transactional and attribute data
9. Rescale, Discretize data

These steps may involve many internal sub-steps for preparing the data to train using ML.

**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.

* ***Problem Definition:***

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 travelers 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.

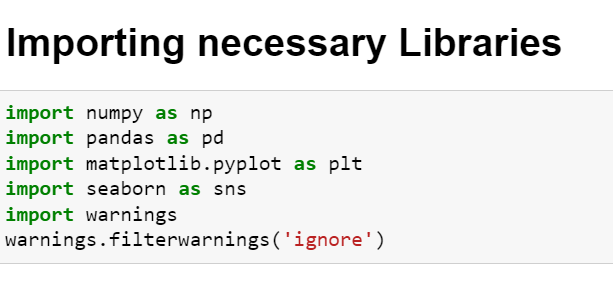
**Additional\_Info**: Additional information about the flight

**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.

* ***Data Analysis:***

Data analysis is the **science of examining a set of data to draw conclusions** about the information to be able to make decisions. Let’s import and examine the data.



This datafile includes two datasets.

* Train dataset: This file will be used to build ML models. It includes 10 independent variables and 1 target variable.

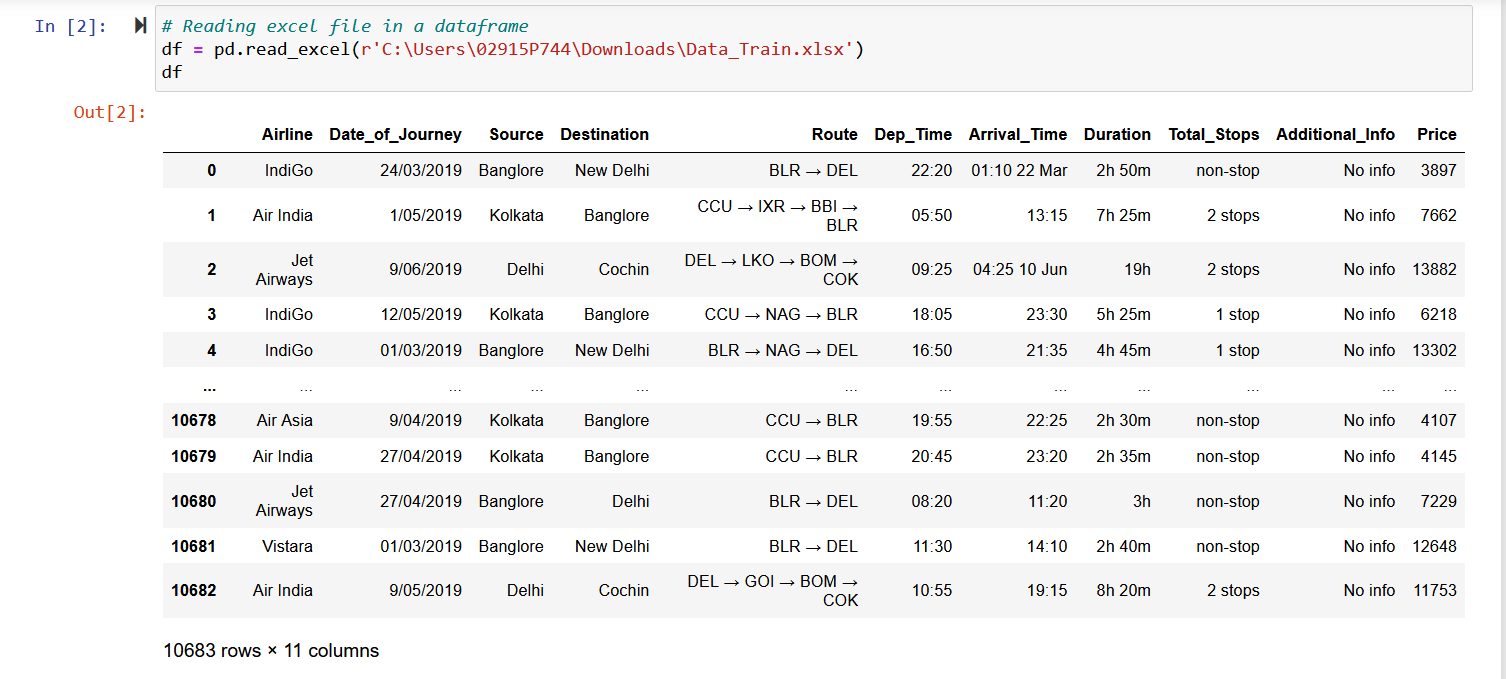
Size of training set :10638 records.

* Test dataset: Test file will be used for getting predictions from the trained model. It includes only independent features.

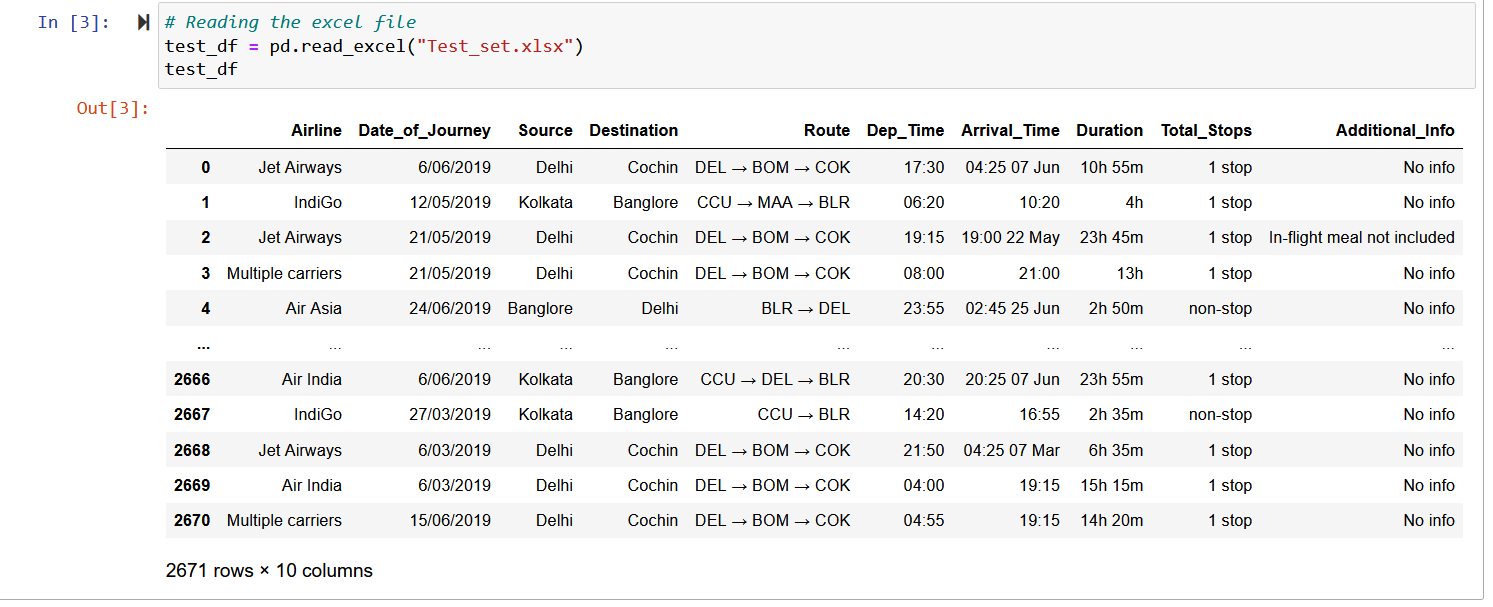
Size of test set: 2671 records.

**Importing Train and Test datasets:**

**Train data**

****

**Test data**

****

**Understanding of Dataset:**

The dataset has both numerical and categorical data values. It also contains special characters which we will handle while data transformation.

* **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:**

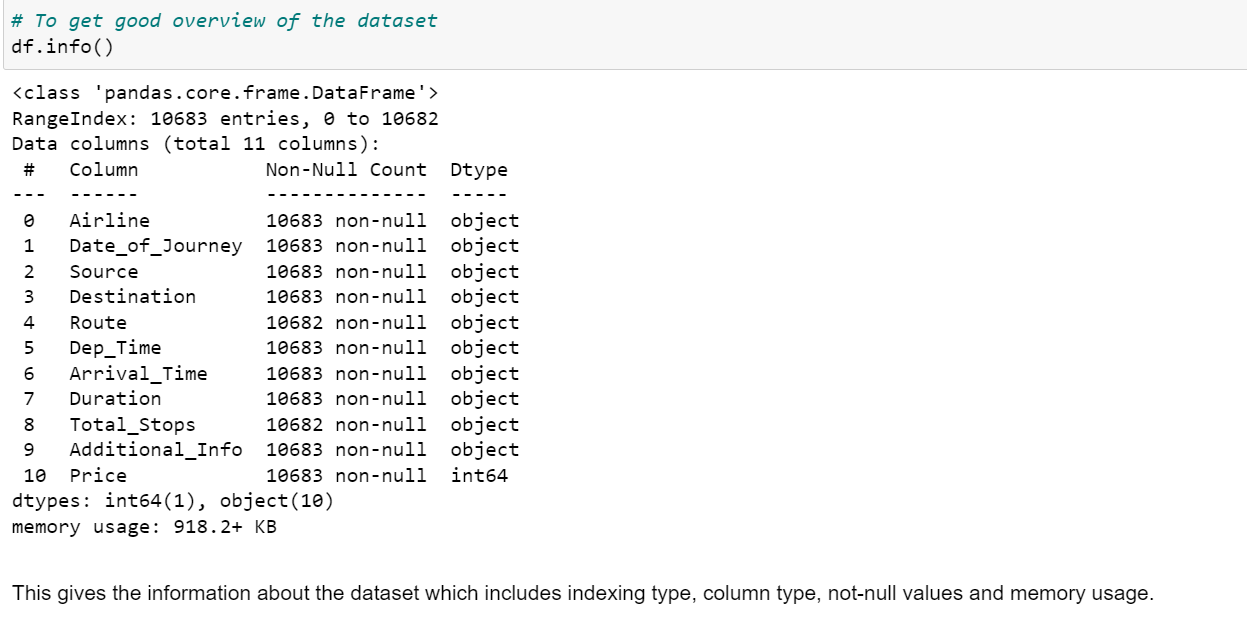
The arrival time and the departure columns contain date mentioned with time, so we need to format that too keeping separate columns for that.

* **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.

* **Exploratory Data Analysis (EDA):**

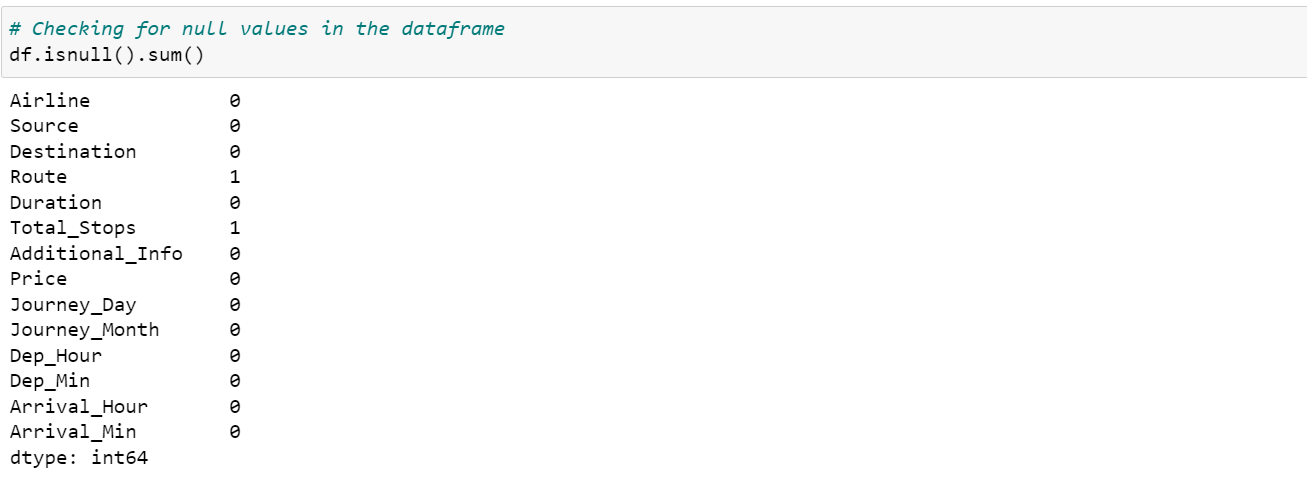
EDA is the first step in data analysis process. It is an approach of analyzing datasets to summarize the main characteristics.



Info () method of Pandas library represents a concise summary of the dataset which includes indexing type, column type, not null values and memory usage.

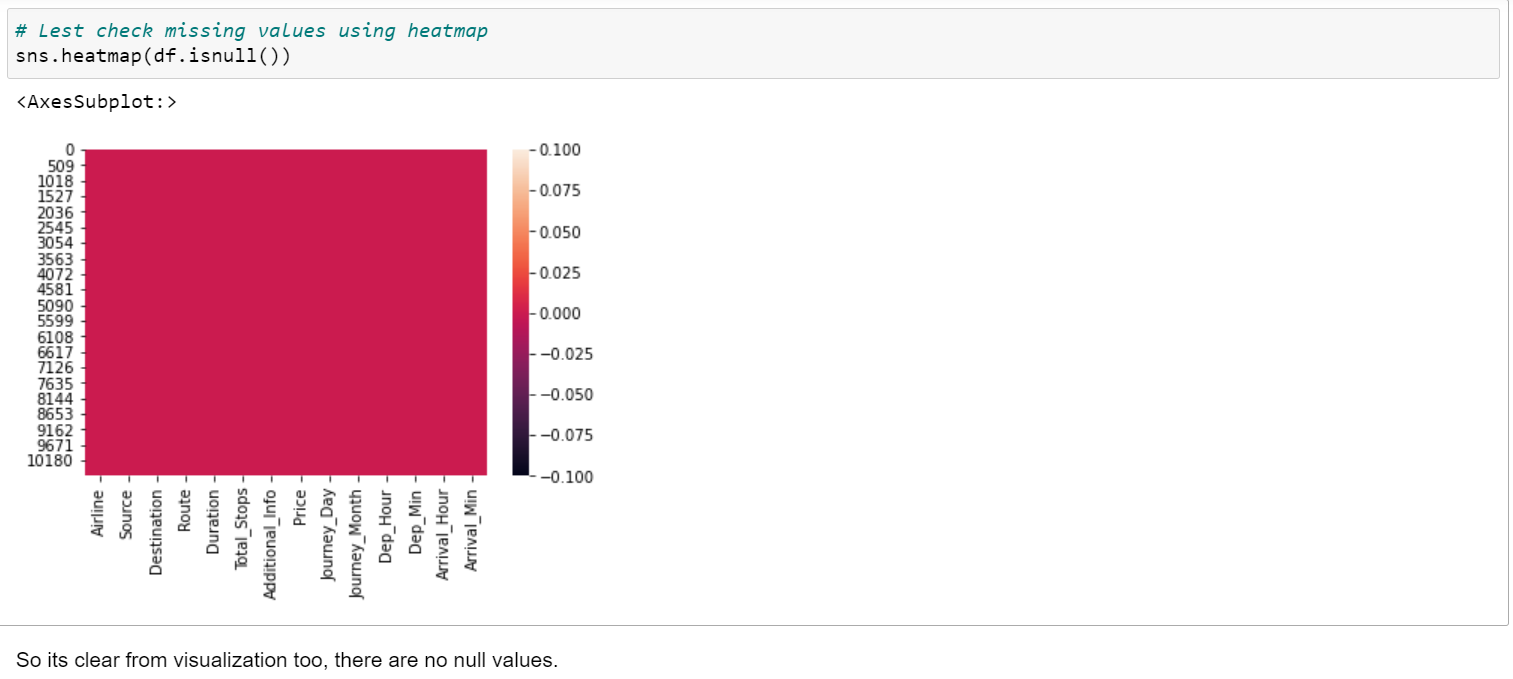
**Missing Values:**

**Missing values** are usually represented in the form of Nan or null or None in the dataset.

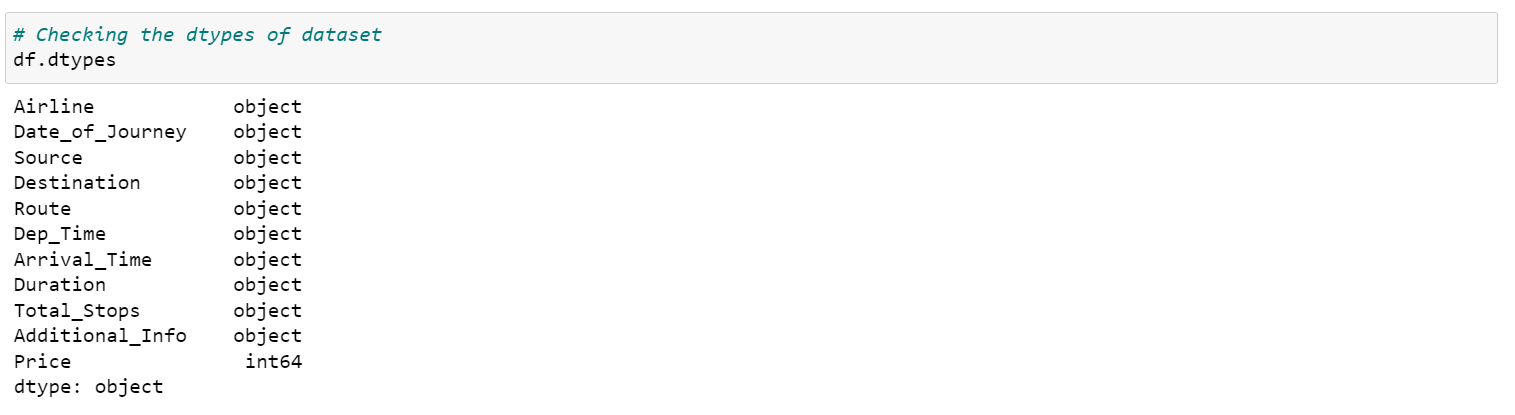


There are multiple ways to deal with nan values. As our dataset is having missing values in categorical columns, we can fill them using mode method

We can visualize it from the below heatmap clearly that no null values are present in the dataset after filling it using mode.



**Checking datatypes:**

****

The dataset contains all the columns having object datatype expect the Price column which is of int64 datatype.

**Feature Engineering:**

Feature engineering refers to the process of using domain knowledge to select and transform the most relevant variables from raw data when creating a predictive model using machine learning or statistical modeling.

Let’s work on all the columns and convert them to numeric.

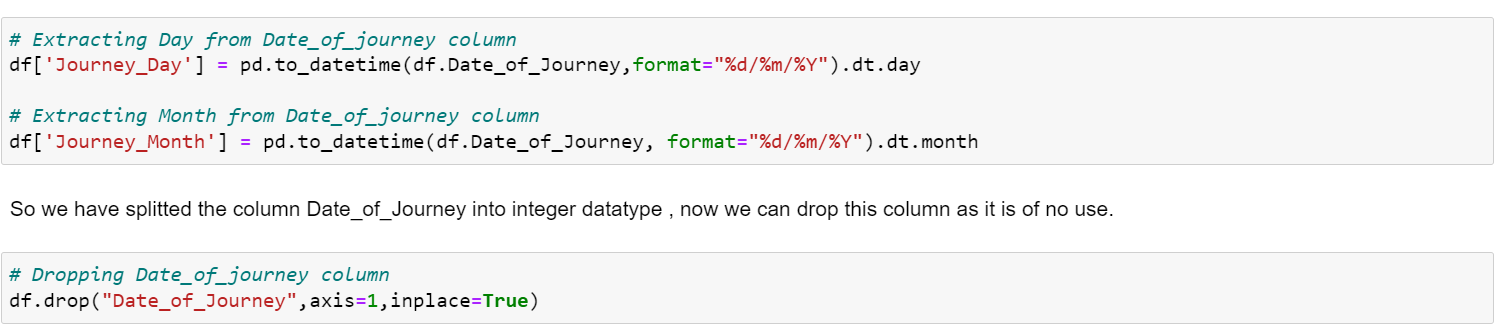
The columns Date\_of\_Journey, Arrival\_Time and Dep\_Time has datetime datatype but its reflecting as object. So, Lets convert this datatype into timestamp to use it properly for prediction.



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

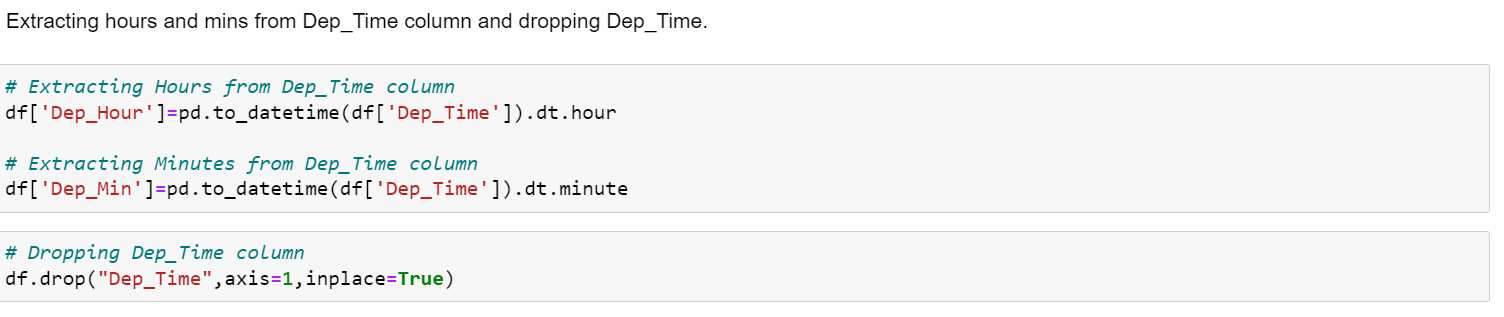
**Date\_of\_Journey:**

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.

****

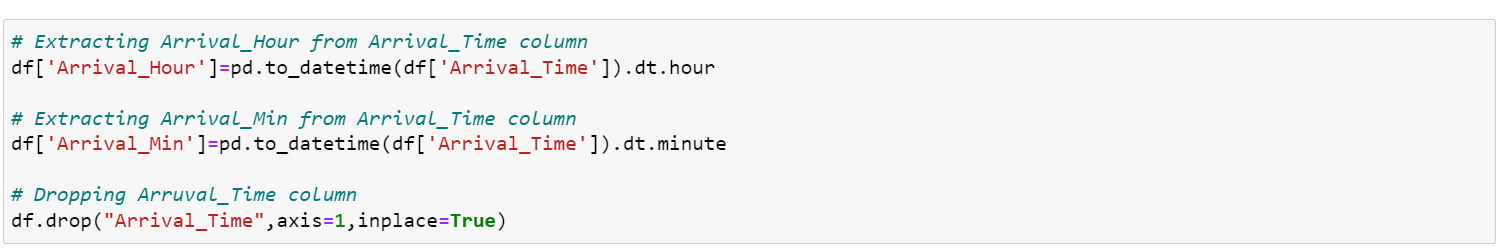
**Dep\_Time:**

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.

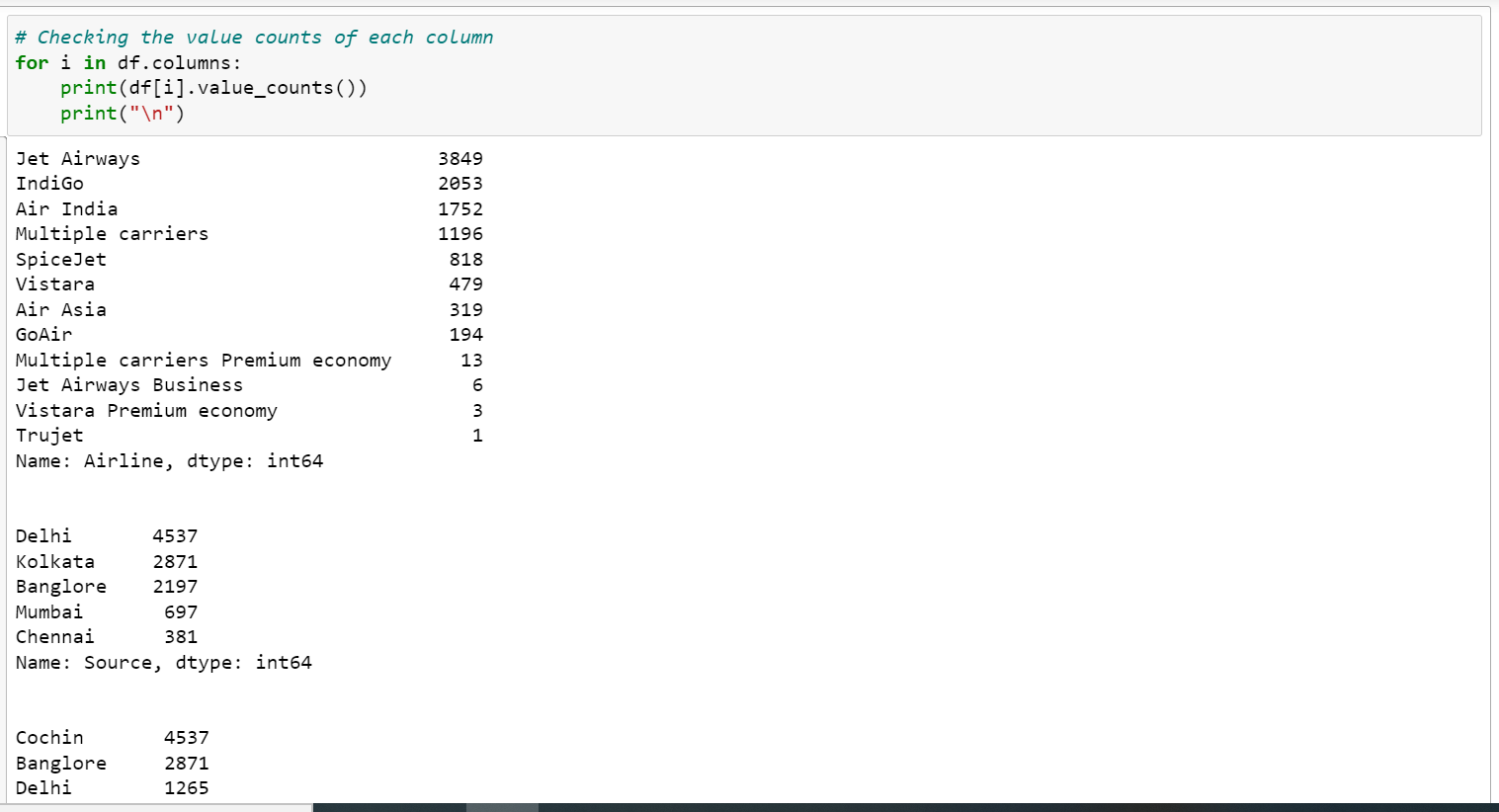


**Duration:**

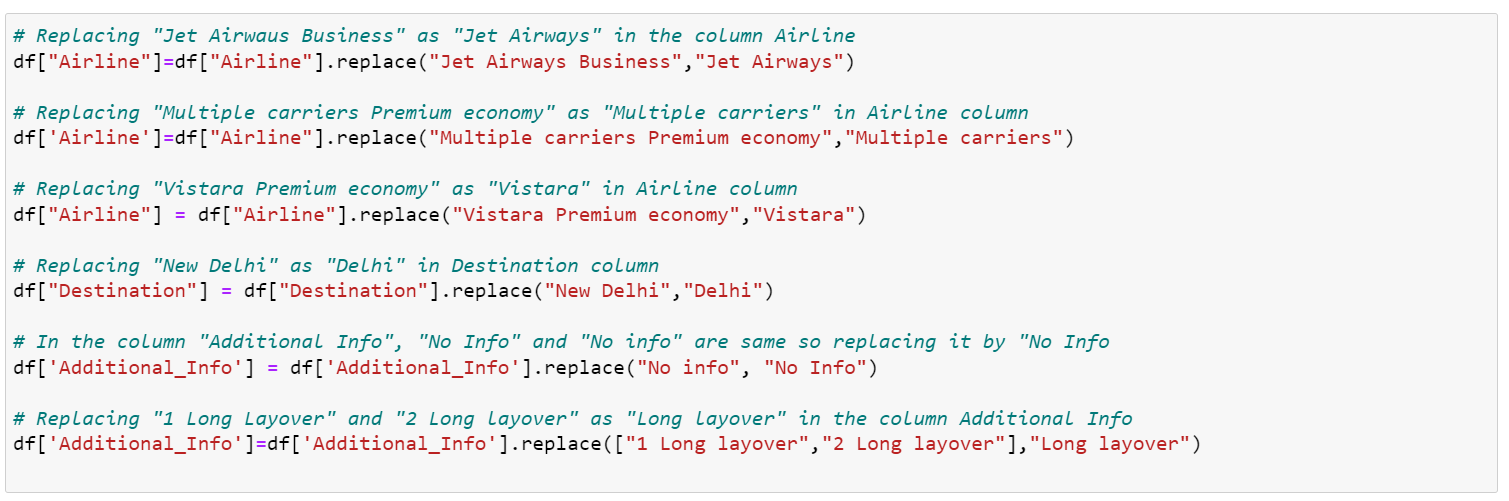
The column Duration has values in terms of minutes and hours. Duration means the time taken by the plane to reach the destination. It is basically the difference between arrival and departure time. Instead of extracting hours and minutes from this column let’s count up the time and format it to numerical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

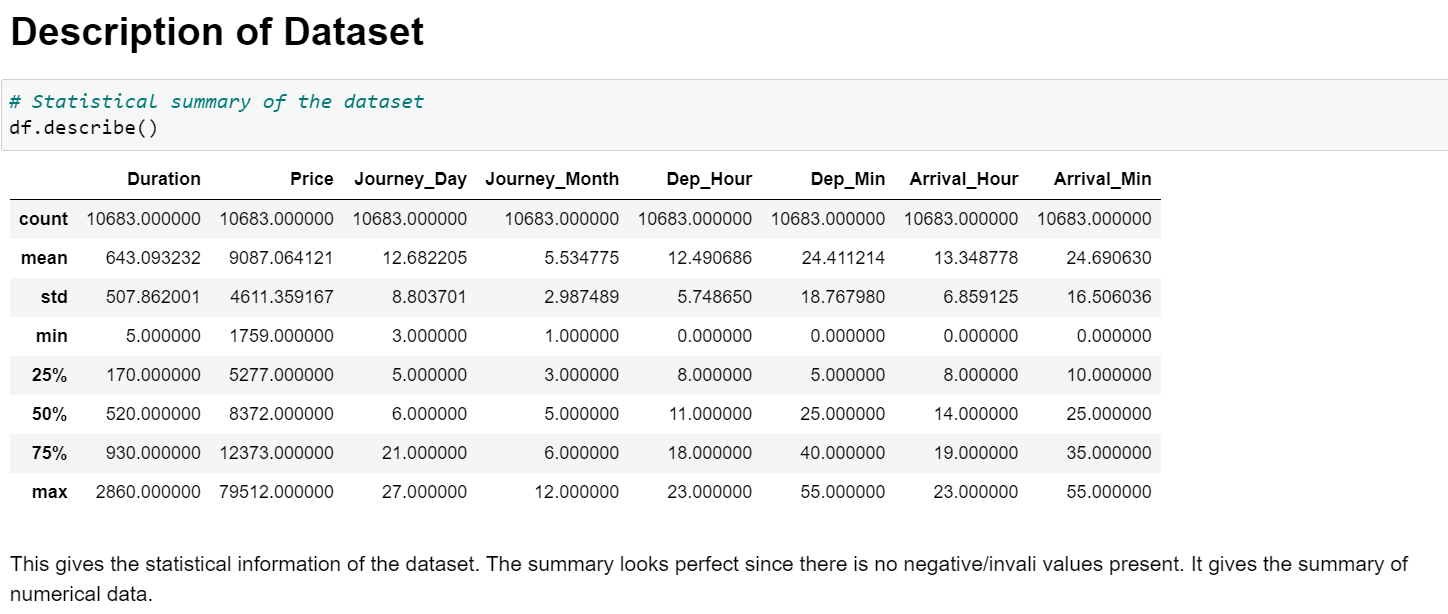


After dealing with **datetime** datatype variables, let’s check the other variables for having some repeated categories using value\_counts () method.



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.

****

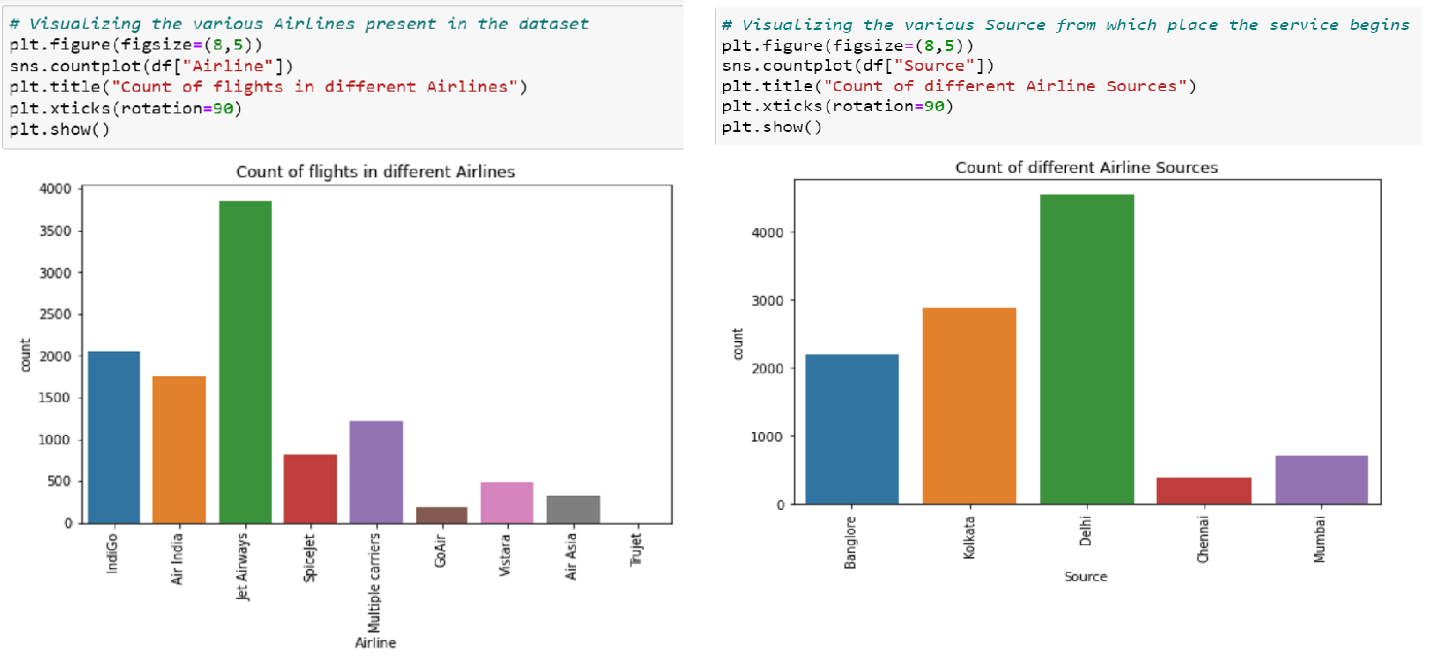
From the above description we can observe the following things

* 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 those columns.
* The std of target variable is high which means it has high rate of dispersion.

**Data Visualization**

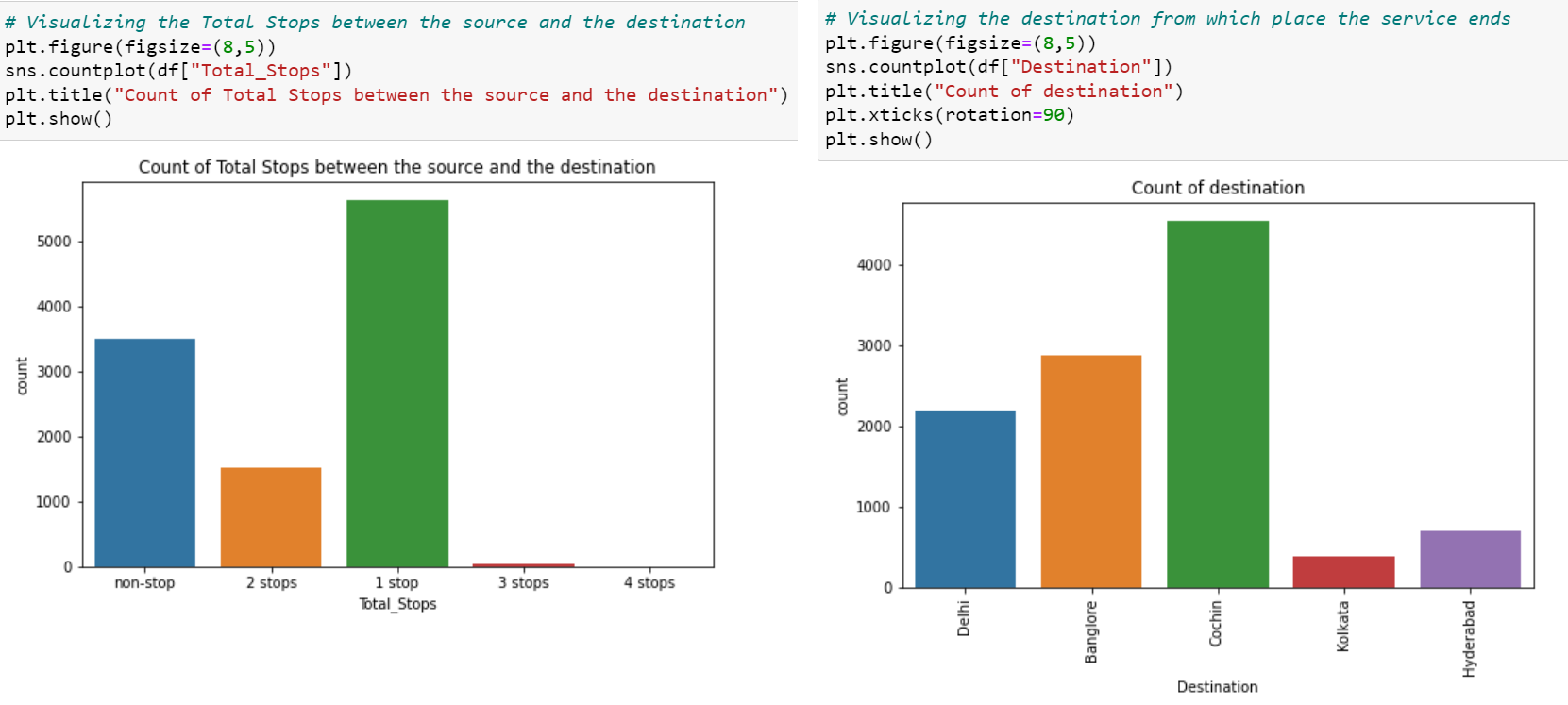
**Plotting categorical columns**

Plotting of **Airlines** and different **Sources** of airlines using countplot of seaborn library

****

The observations are:

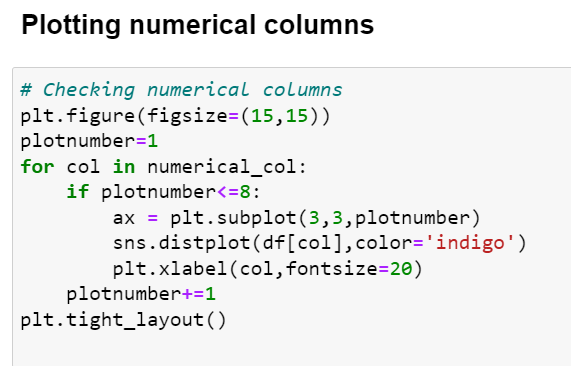
* Jet Airways flights has high counts whereas Trujet and GoAir has the least counts.
* The majority of Airline source is from Delhi while the least is from Chennai.

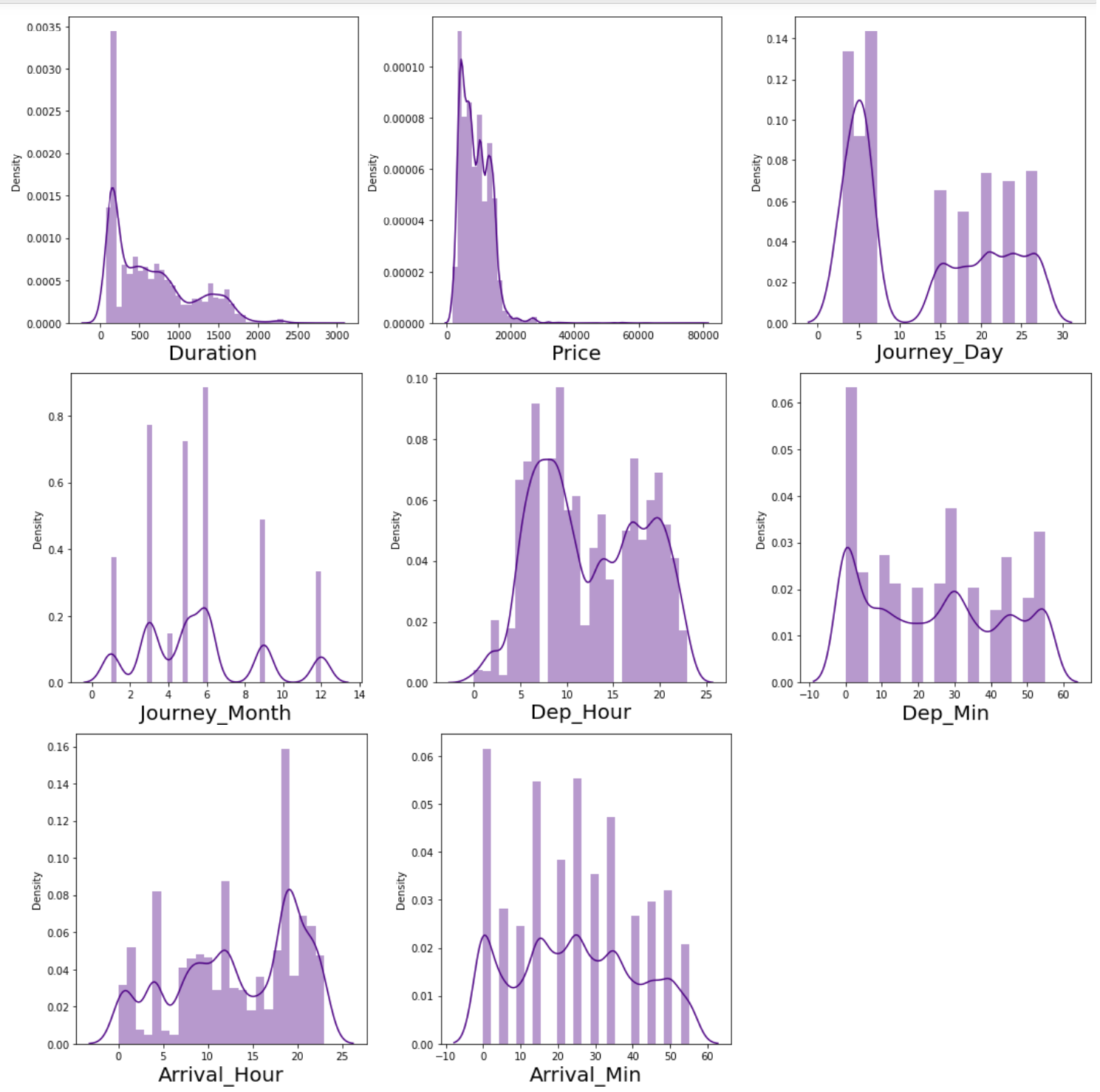


Plotting of **Total\_Stops** and **Destination** varaibles.

And the observations are:

* 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.



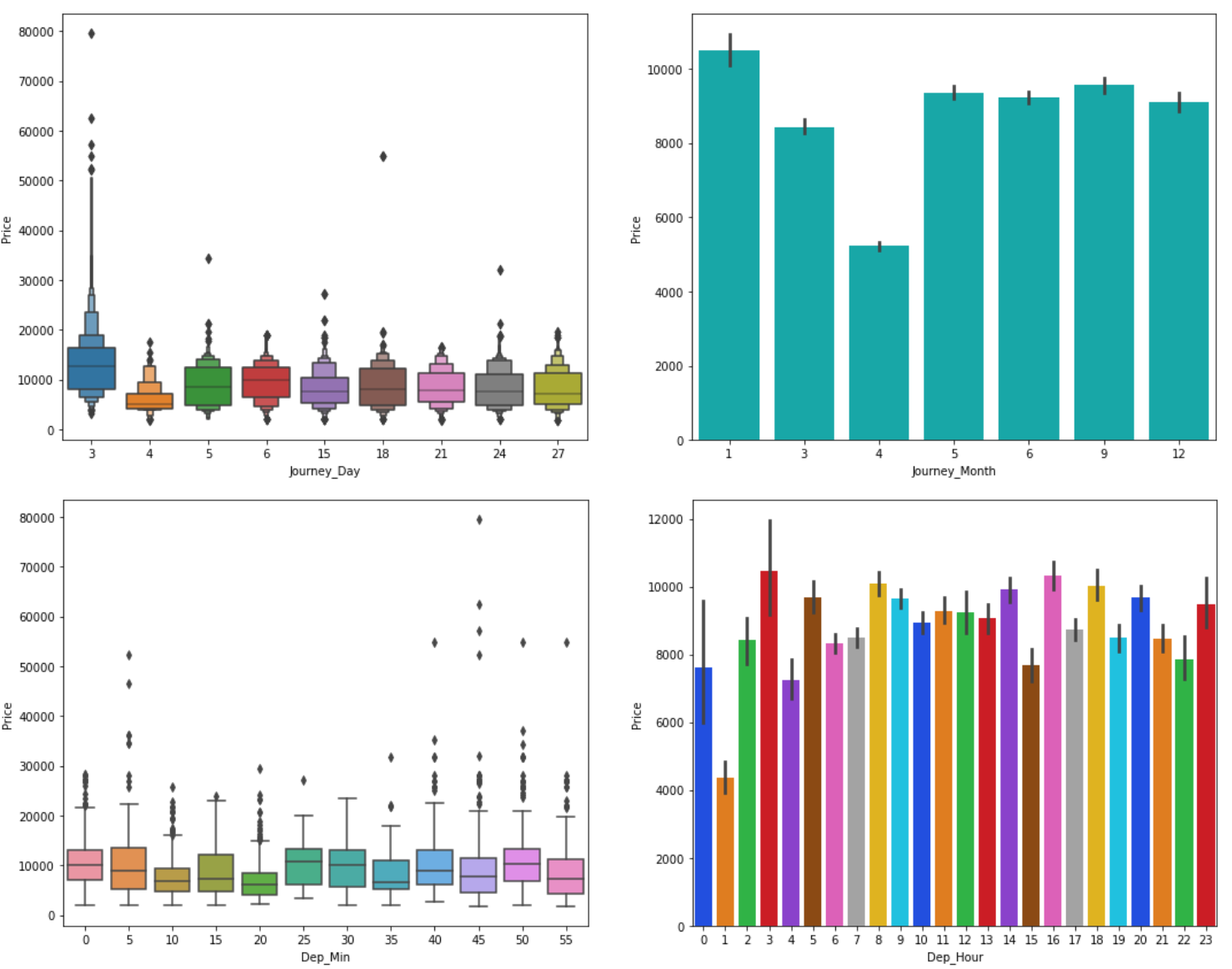


From the distribution plot we can observe that the data is not normally distributed in some columns and some columns are almost normal but have no proper bell shape curve.

The Journey\_Month, Duration and Price columns are skewed to right as mean is more than the median.

**Bivariate Analysis:**

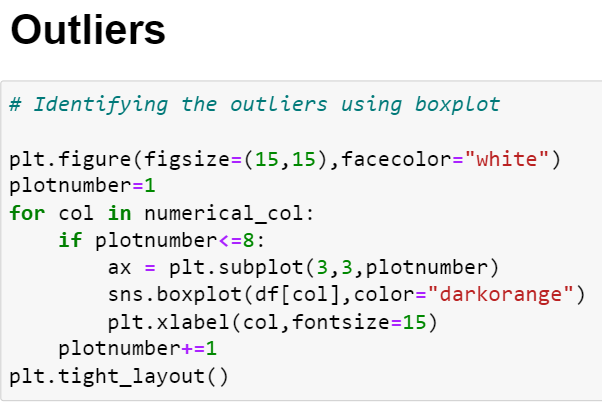


****

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.

In similar manner, we can envisage other columns too and understand the relation between each variable.

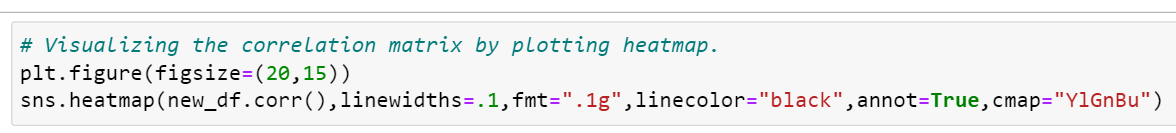


We can check the outliers using boxplot.

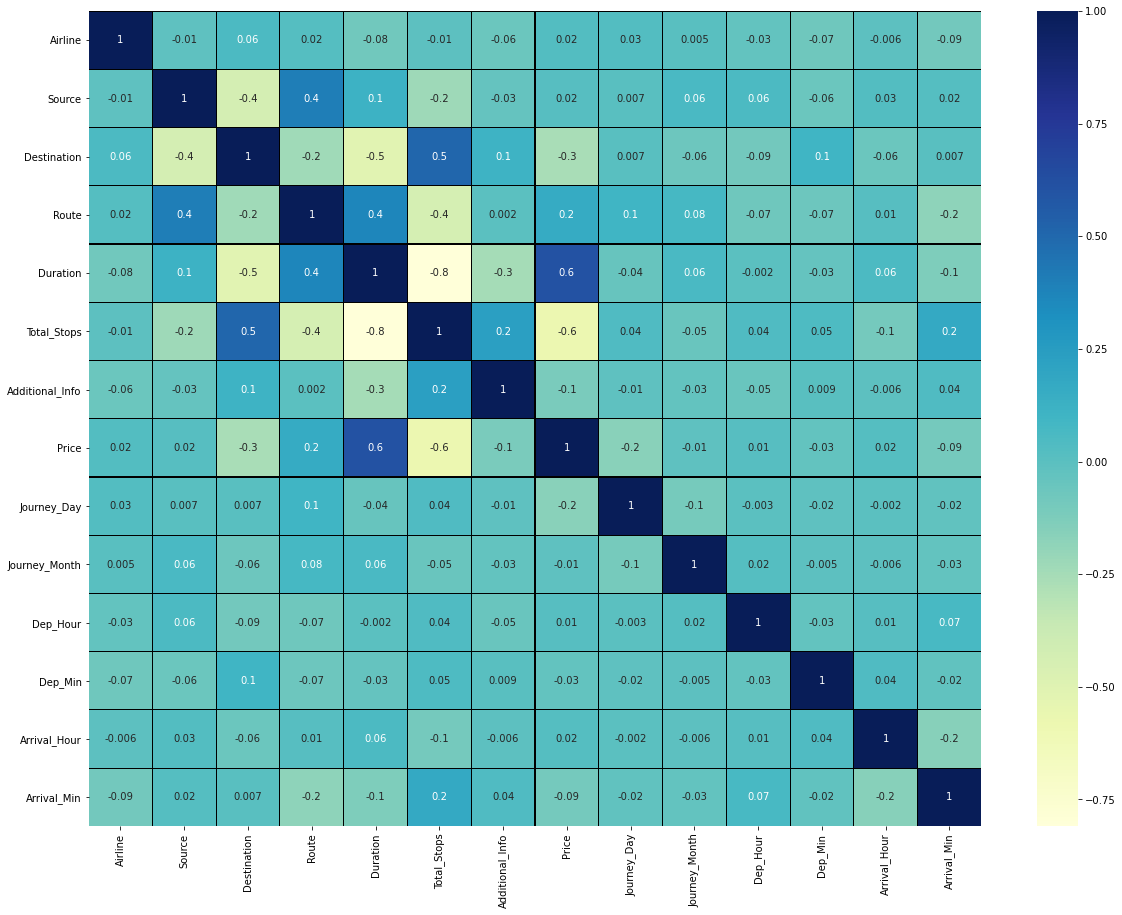
Since the outliers are present in the columns "Duration", "Journey\_Month" and the target variable "Price". We can remove it using Zscore method or IQR method after comparing the data loss in each method.

**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

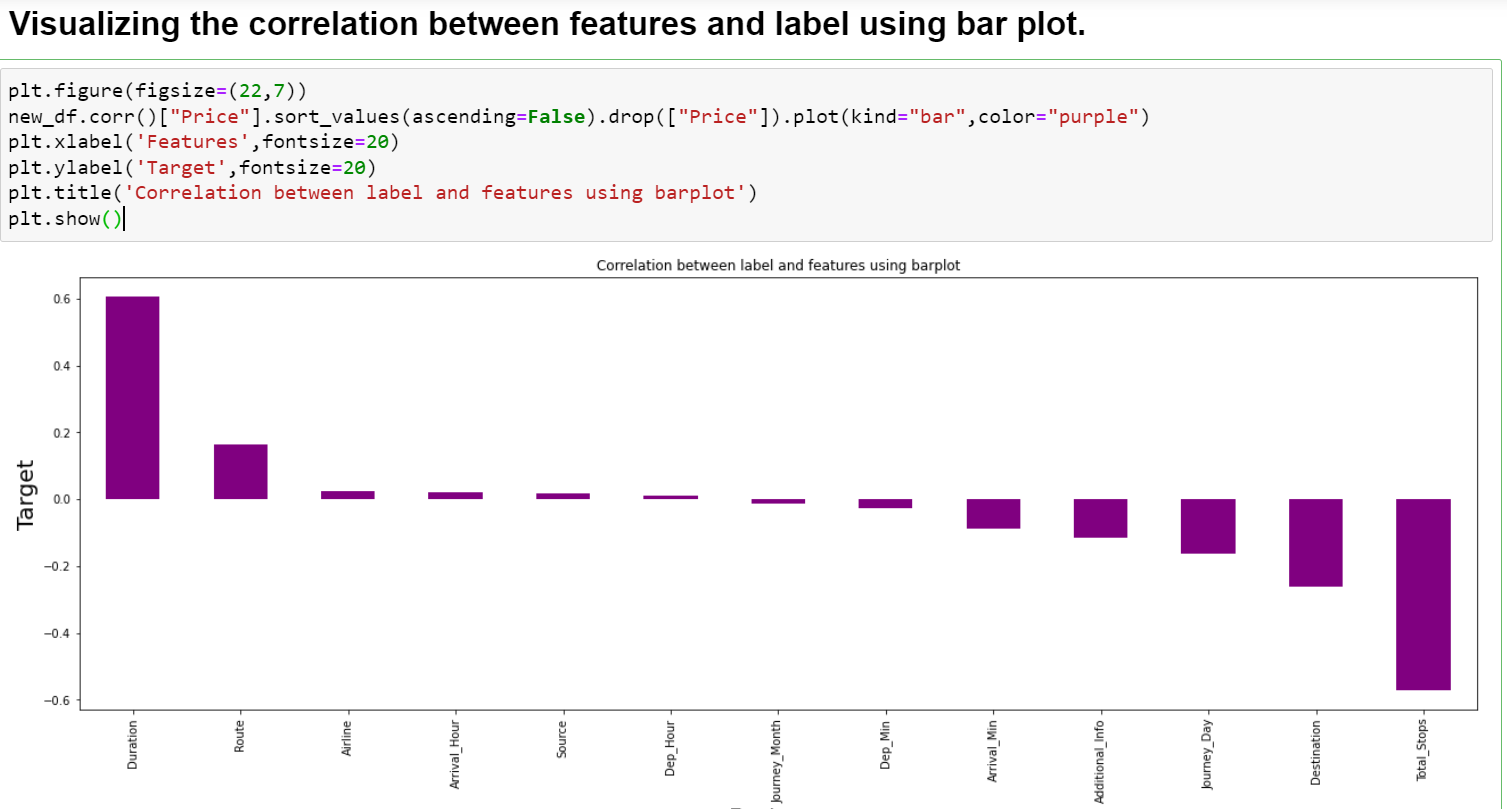
****

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.

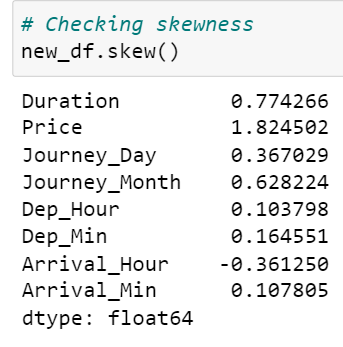


The features Journey\_Month, Source, Arrival\_Hour, Dep\_Hour and Airline have very less correlation with the label “Price”.

**Checking for Skewness:**

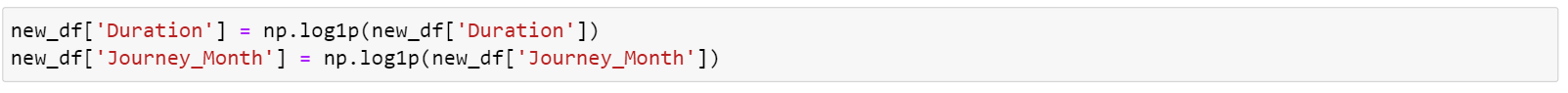
Skewness is a measure of the symmetry of a distribution.

If the values of a certain independent variable are skewed, depending on the model, skewness may violate model assumptions or may impair the interpretation of feature importance.



Presence of skewness more than +0.5 and -0.5 is not acceptable as it will impact on model accuracy.

We can find the skewness present in Price, Duration and Journey\_Month columns. As “Price” is the target variable no need to remove skewness from it and removing from “Duration” and “Journey\_Month”.



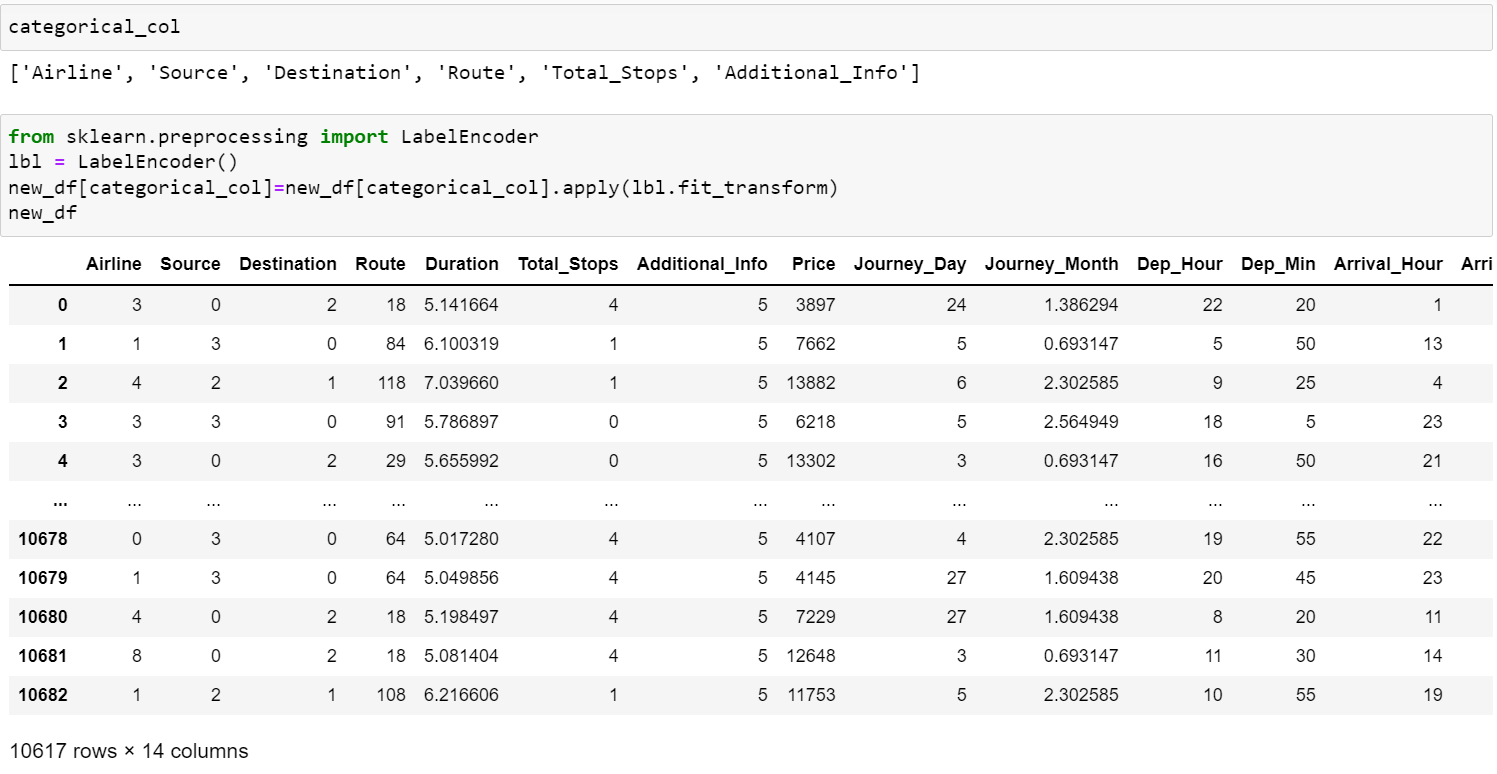
Skewness can be removing using various techniques. Here I used “log transformation” top remove skewness from “Duration” and “Journey\_Month” columns.

* ***EDA Concluding Remark*:**
* The dataset entailed **missing values** and **duplicate** **values** in the features. As nan values were present in categorical variables, handled them using “mode” method. Handled the duplicate categories using value\_counts method and assigned the unique values.
* Engineered datetime features by converting strings into datetimes, which exposes all the **pandas dt** properties.
* The number of flights is highest in the month of **January** and the ticket price is also expensive in this month compared others.
* The number of flights varied from city to city and **Delhi** being the source of most flights and **Cochin** seems to be the destination of most flights.
* Removed outliers and skewness present in the dataset using zscore and log transformation method respectively.
* The feature “Duration” is **highly positively correlated** with the target variable "Price" whereas the feature “Total\_Stops” is **highly Negatively correlated** with the label “Price”.

# ***Pre-processing Pipeline:***

In this section, we undertake data pre-processing steps to prepare the datasets for Machine Learning algorithm implementation.

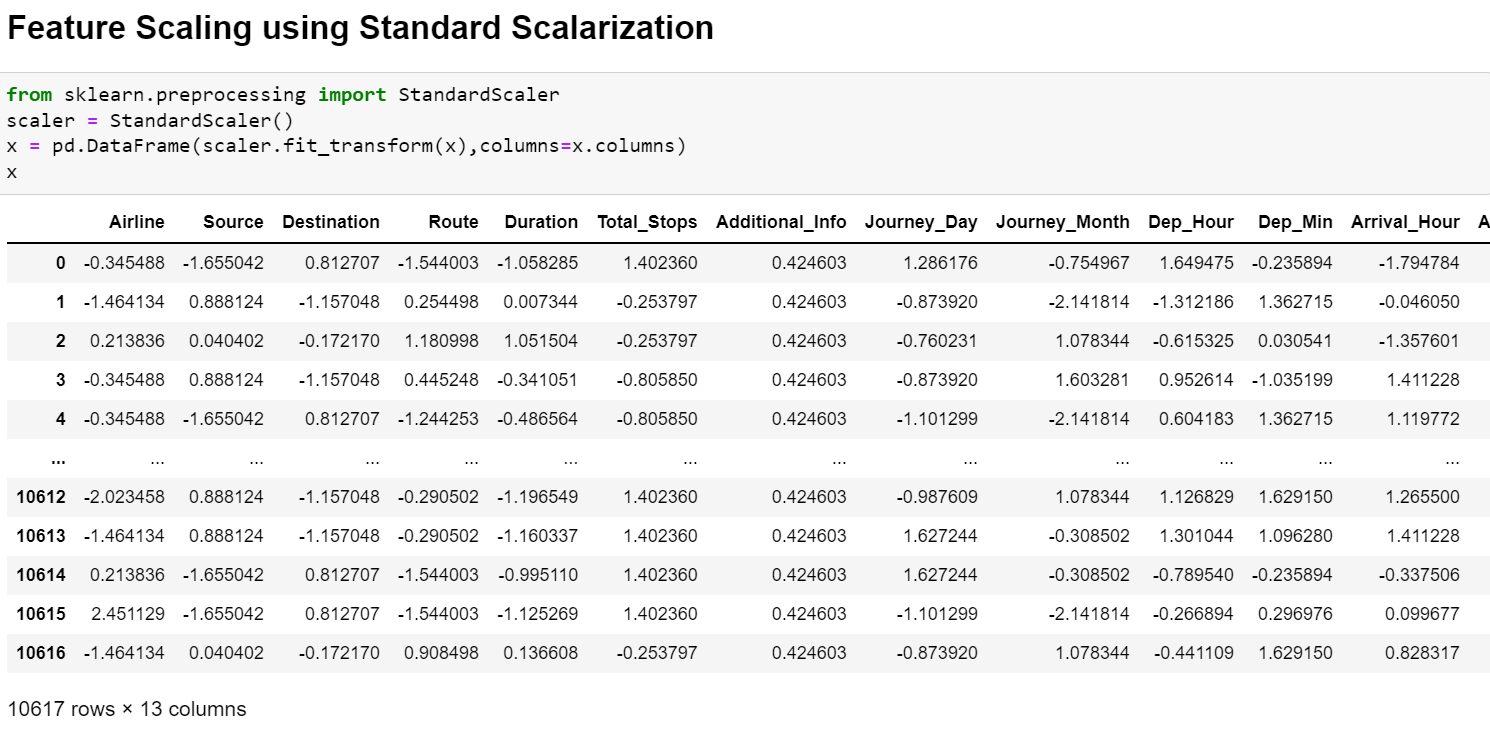
## **Label Encoding Categorical data:**



Encoded the categorical data using LabelEncoder method ().

**Feature Scaling:**

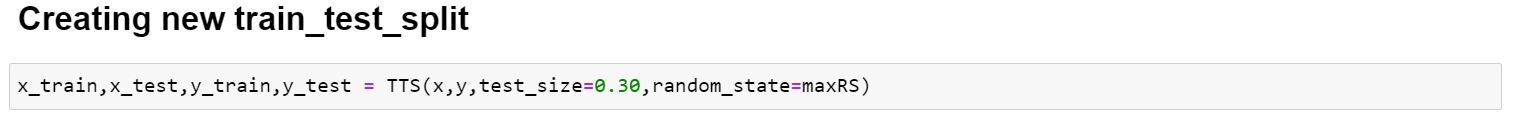
Feature scaling is a method used to normalize the range of independent variables or features of data.



## **Splitting data into training and testing sets**

The train-test split is a technique for evaluating the performance of a machine learning algorithm.





* ***Building Machine Learning Model :***

Building machine learning models that can 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

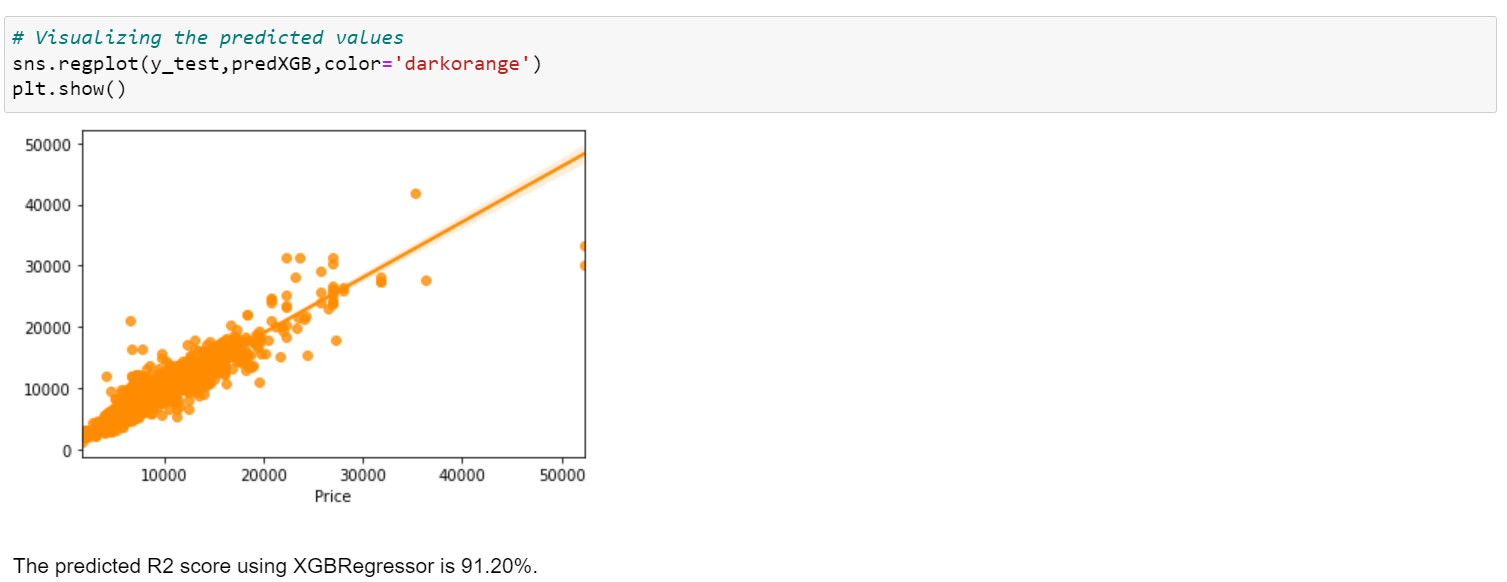
# Gradient Boosting Regressor

# Bagging Regressor

# Extra Trees Regressor

# XGB Regressor





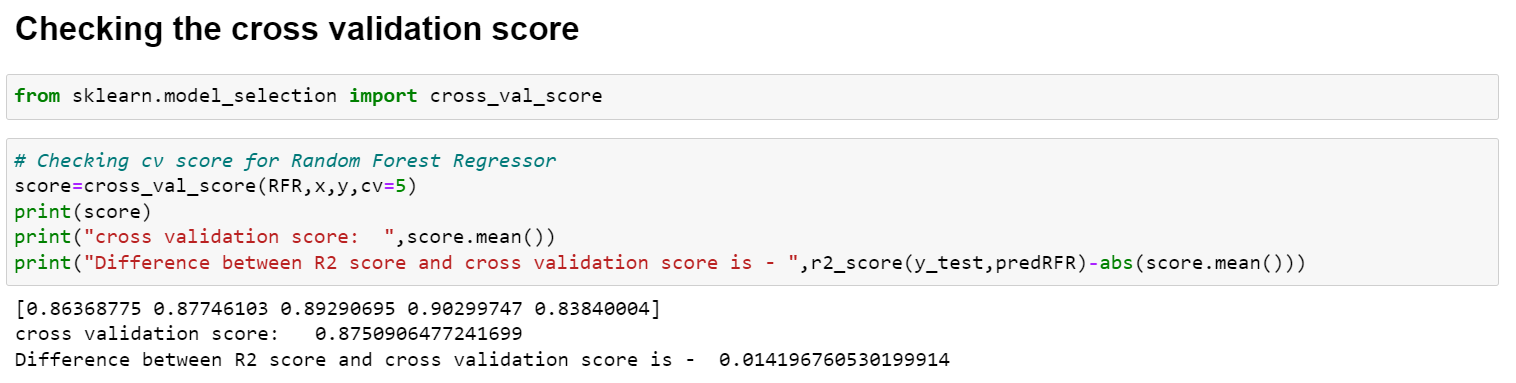
There are three error metrics that are commonly used for evaluating and reporting the performance of a regression model; they are:

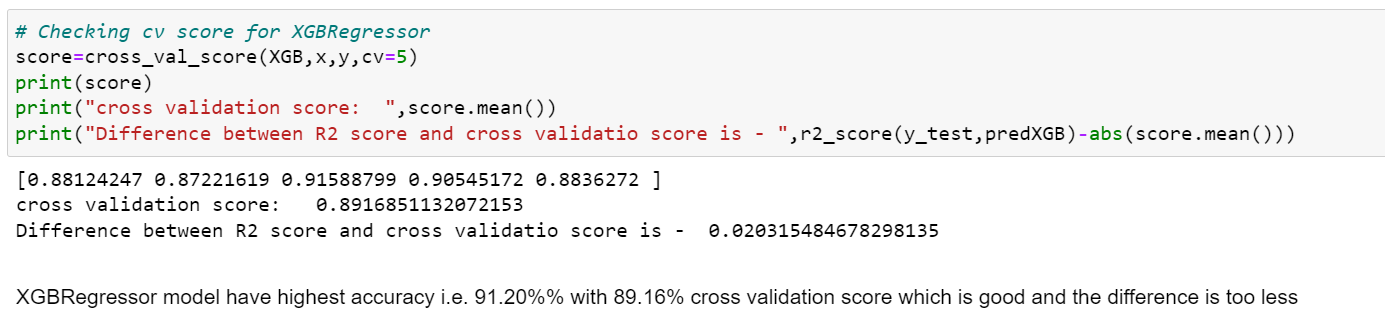
* Mean Squared Error (MSE).
* Root Mean Squared Error (RMSE).
* Mean Absolute Error (MAE)

After successfully building the model by training the data using training set and with the help of testing set getting the prediction for every model, checked the error metrics for less generalization.

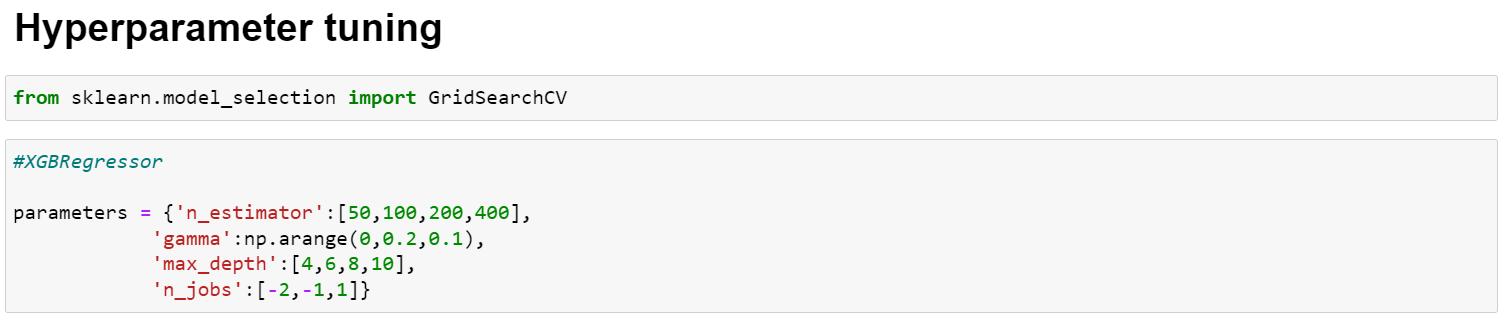
Here we got best R2 score 91.20% using **XGB Regressor**.

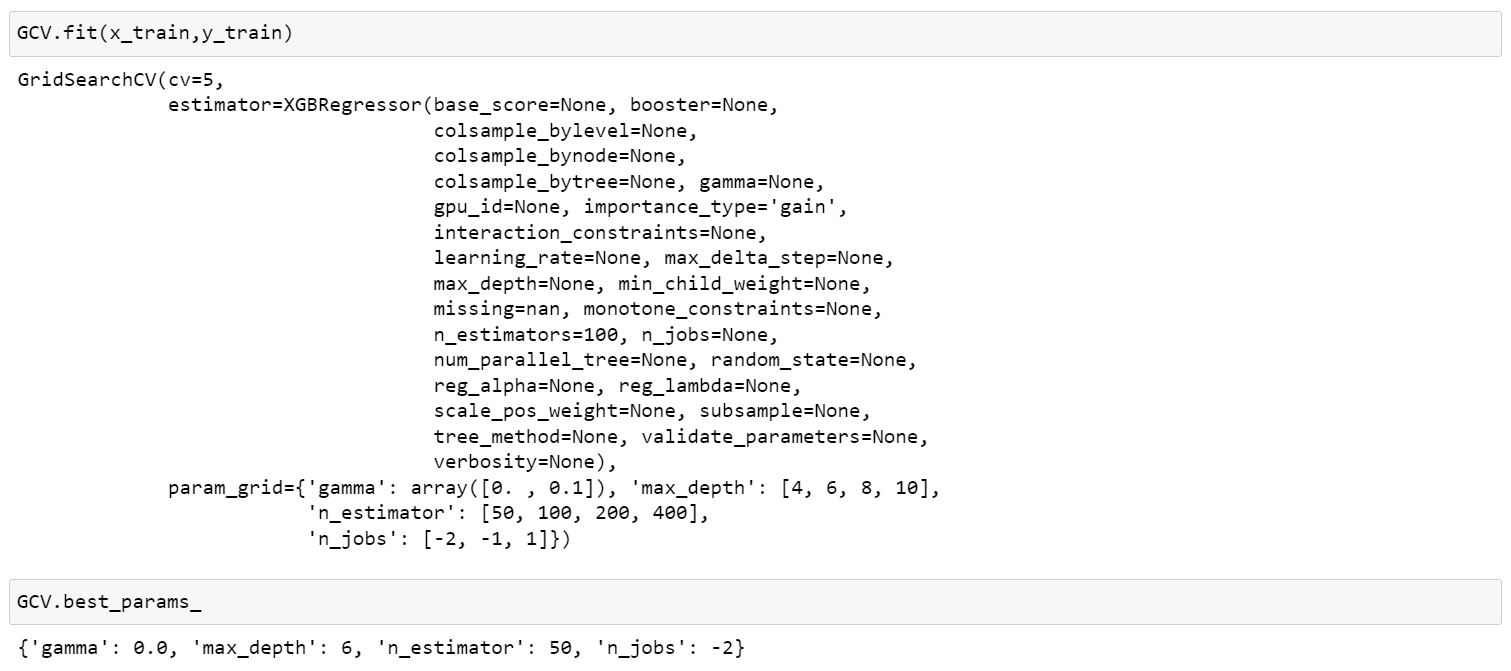
In order to check if the model is overfitted or not, let’s perform cross validation.



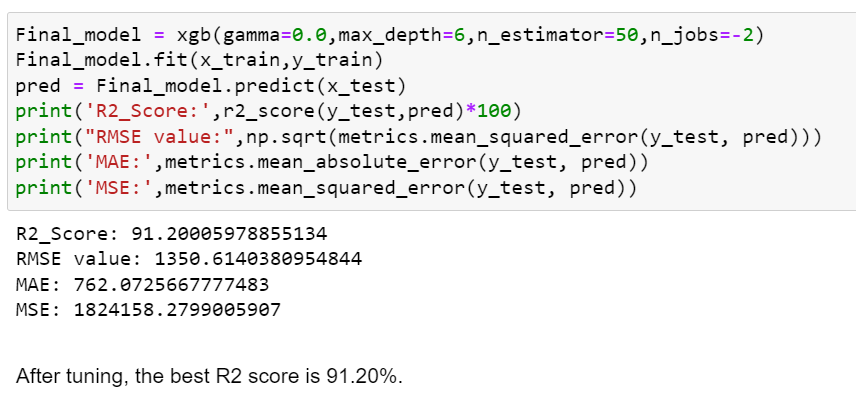


Model selection refers to the process of selecting the right model that fits the data. This is done using test evaluation matrices. The results from the test data are passed back to the hyper-parameter tuner to get the **most optimal hyperparameters**.

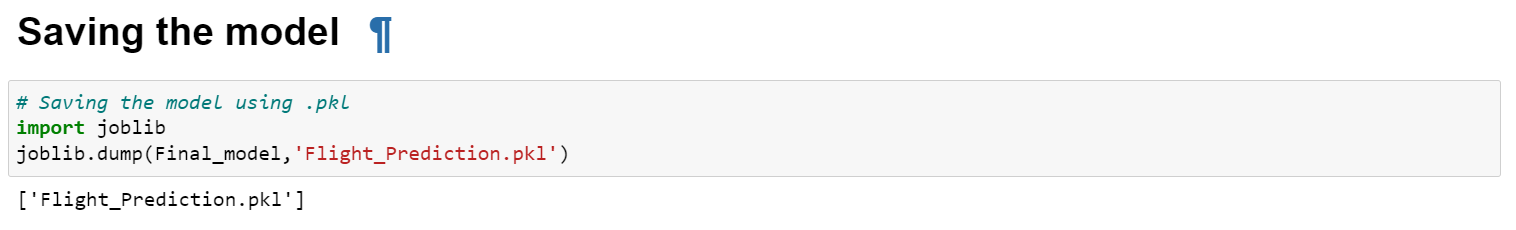


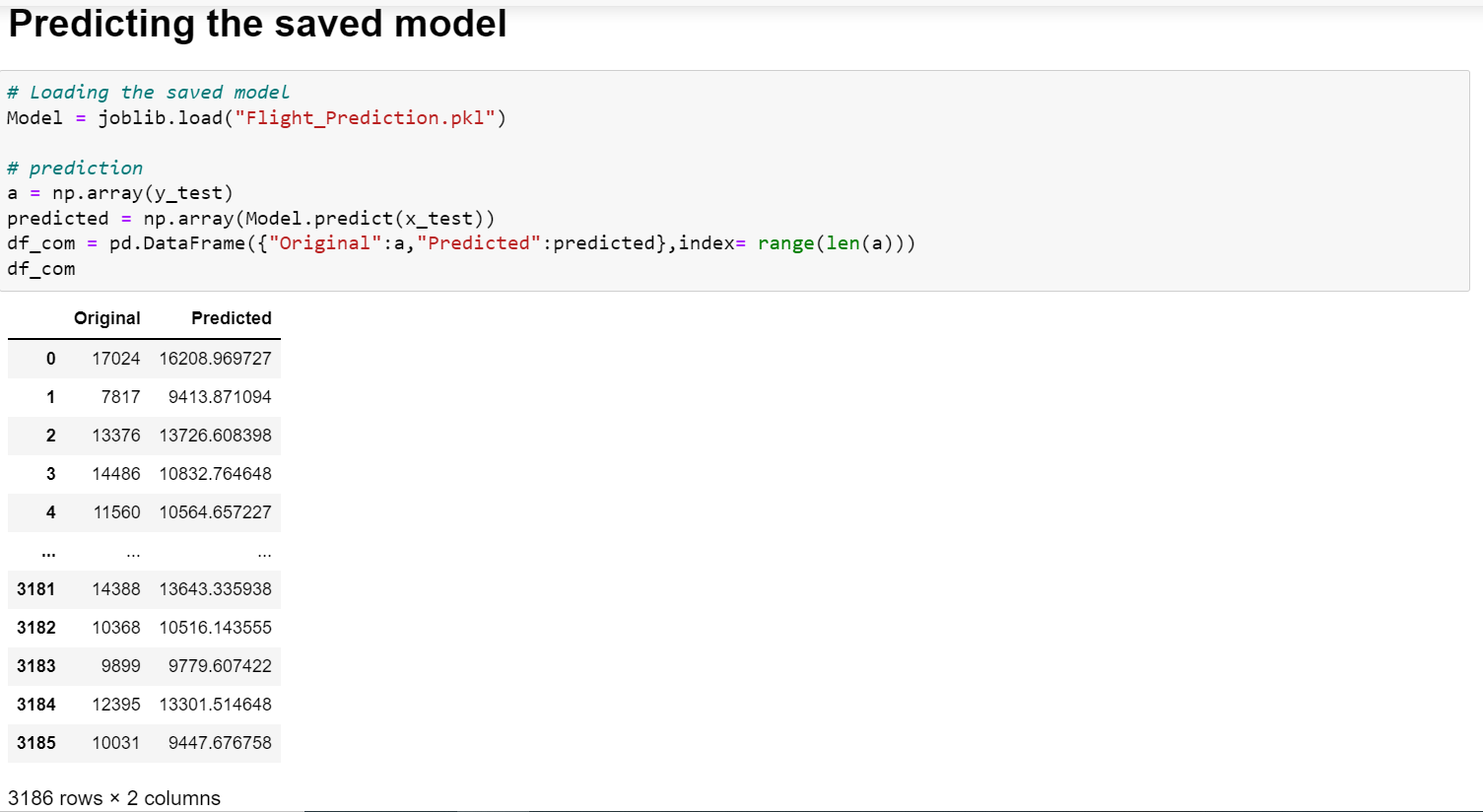


Let’s build the model using these hyperparameters.

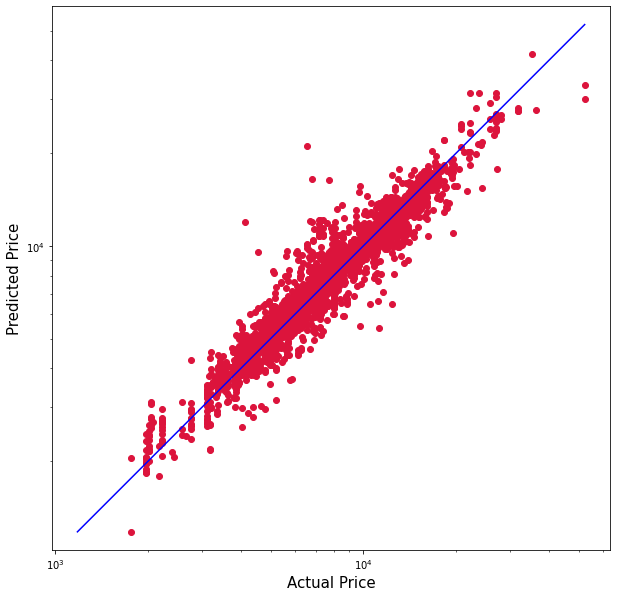


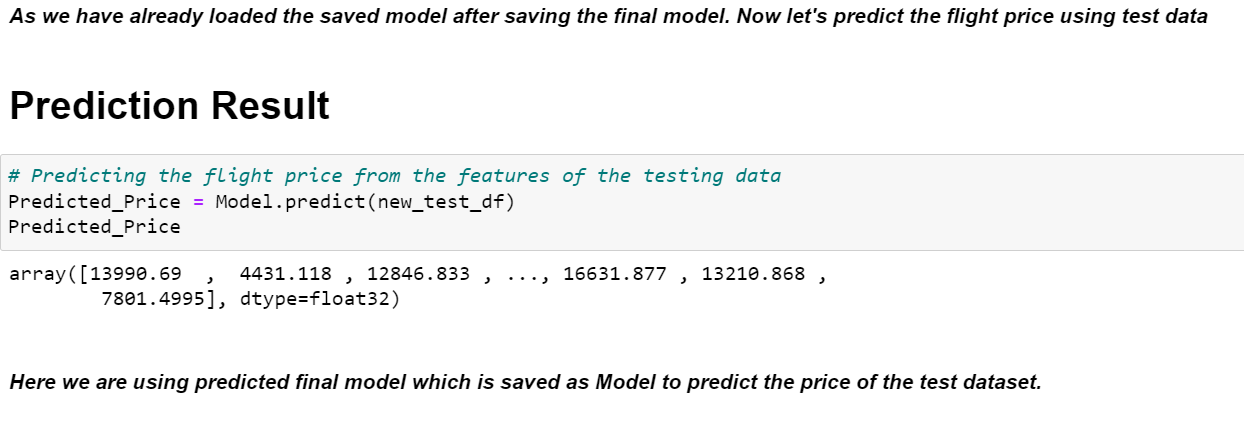
We have built the model and performed the hyper parameter tuning, now we will save the model to reuse it again while processing test data.

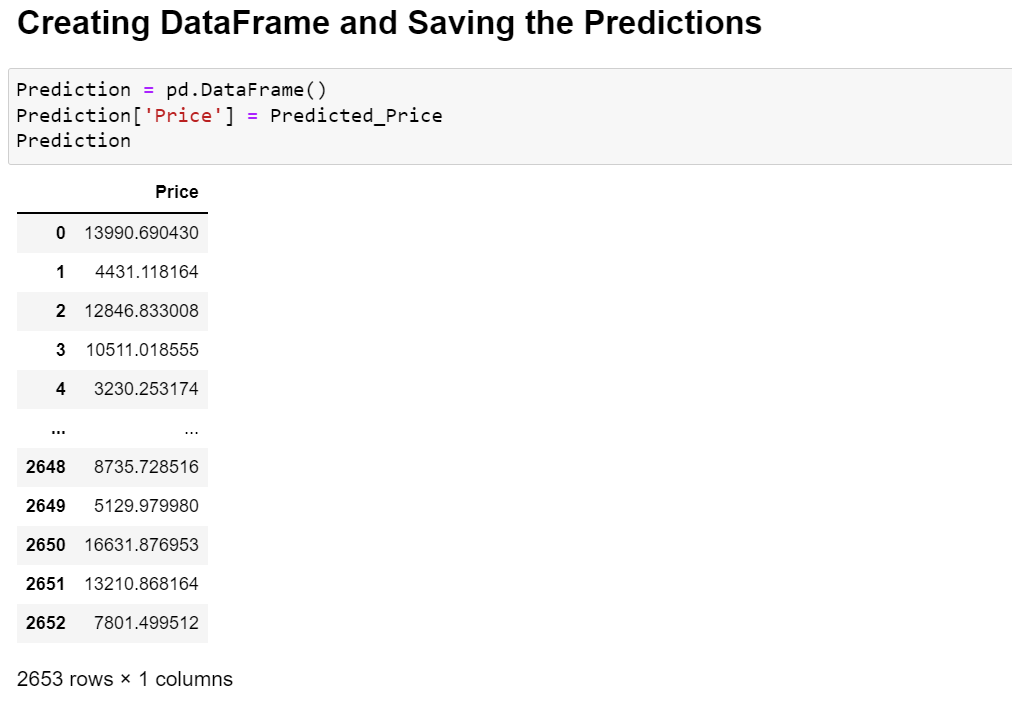


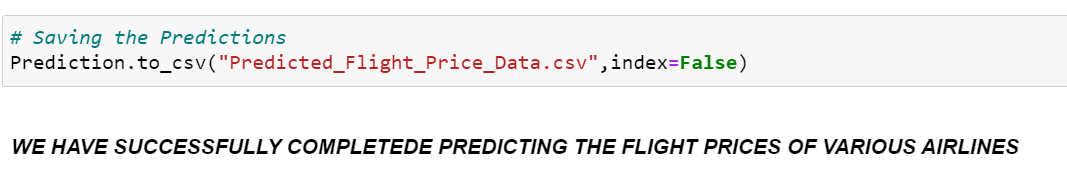




****







# ***Concluding Remarks:***

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.

Github-link of the project:

<https://github.com/GuggillaMamatha/DataScience_Projects/blob/main/Flight_Price_Prediction.zip>

