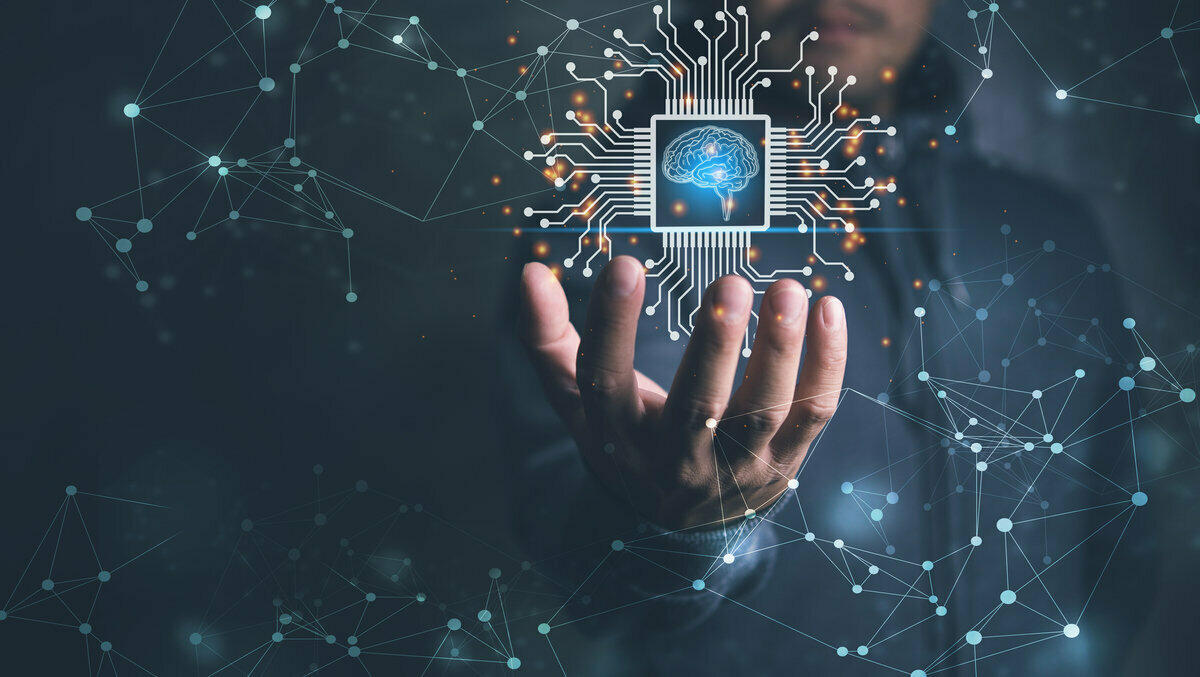
**DATATRAINED ACADEMY**

**BLOG ARTICLE – 1**

**(I’m not saying it is going to be very easy, I’m saying it is going to be worth it.)**

**FLIGHT PRICE PREDICTION PROJECT**

Submitted by: NITIN SHIMPI

Batch number: 1842



**INTRODUCTION:**

Flight prices are something unpredictable. It’s more than likely that we spent hours on the internet researching flight deals, trying to figure an airfare pricing system that seems completely random every day. Flight price appears to fluctuate without reason and longer flights aren’t always more expensive than shorter ones.

But now the question is how to know proper Flight price, for that I have built a Machine learning model which can predict the Flight price. Using various features like **Airline, Source, Destination, Arrival time, Departure time and the Price for the same travel**. So using all these previously known information and analysing the data I have achieved a good model that has **90% accuracy**. So let’s understand what all the steps we did to reach this good accuracy.



**Libraries used:**

* Python
* Numpy
* Matplotlib
* Seaborn
* Datetime

Now let’s get into the problem and build a best possible model to predict Flight price. **In this perticular problem we are going to deal with two sets of data one is train and other is test.** Let’s have a look.

**1.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 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.

Size of training set: **10683** records

Size of test set: **2671** records

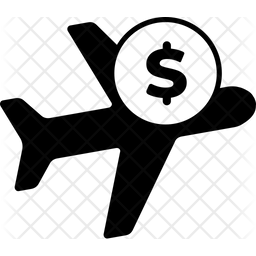
**Attribution Information:**

* **Airline:** The name of 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.

In this perticular problem we have two datasets so we have to build model using train dataset and save the best model, after that using the best model saved we have to predict the price for test dataset.

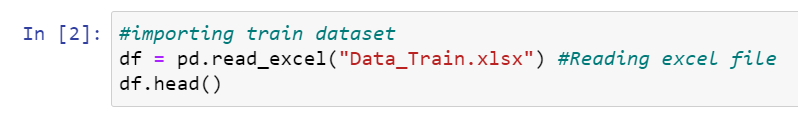
**Let’s do it step by step firstly analysing the dataset and doing exploratory data analysis, data visualization, data cleaning, pre-processing, model building, model saving and finally predictions.**

**2.Data Analysis:**

****

Let’s import the datasets first. I have two datasets one is train dataset and other is test dataset. I can merge both the sets, but I haven’t done that. Because this merging may cause data leakage so to avoid that we have to clean both the datasets separately with same steps did for first dataset. Now looking into the target ‘Price’ and I have to make sure the data type of target column to decide the type of problem.

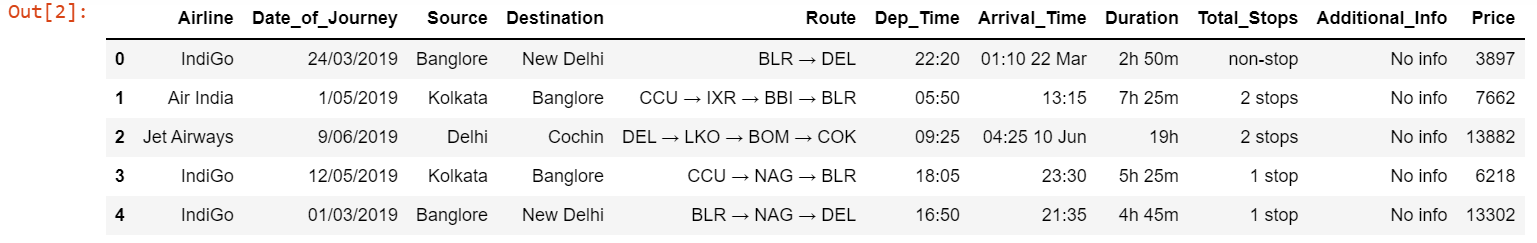
Since Price is my target and it is a numerical column with continuous entries. So it looks quite clear that this perticular problem is a **Regression problem** and I have to use all regression algorithms while building the model.



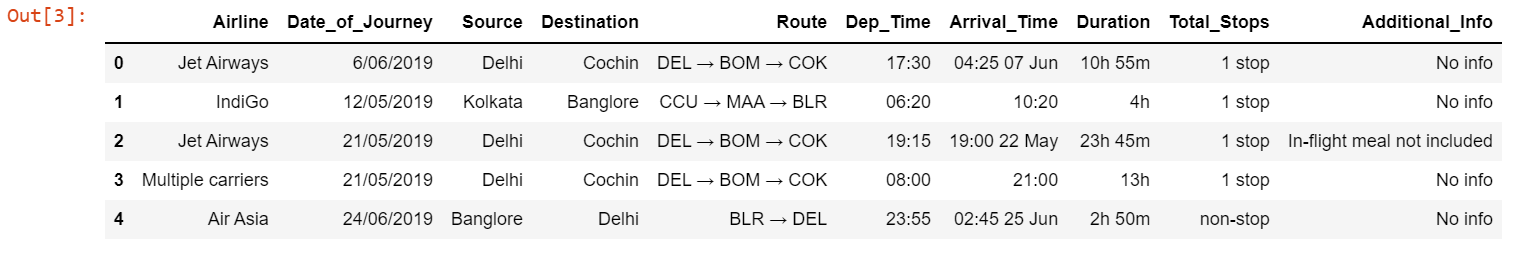


* Firstly I have imported both datasets which were in excel format as df for train dataset and dff for test dataset. Below are the datasets.

**Train dataset:**



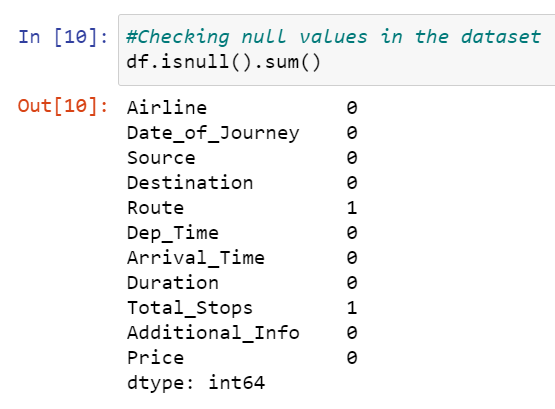
**Test dataset:**

****

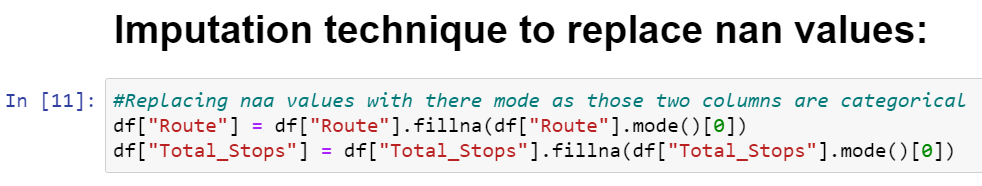
* Above are my datasets. By looking into the features I can say both the datasets has same features except target. So now we have to clean this data’s.

**Data Preparation and cleaning:**

* Firstly we have to do some statistical analysis like checking shape, nunique, value counts, info etc…..
* After reading the value counts if we find any unnecessary columns in the dataset we can drop those columns. But presently I don’t find any such kind of columns, so let’s keep everything as such.

**Checking for Null values and replacing them using imputation:**

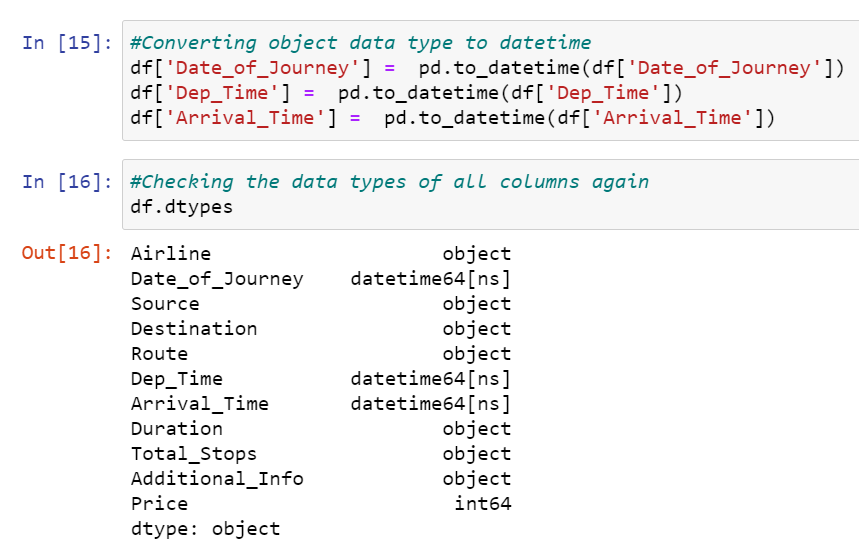
* I can observe only 1 null value in column Route and Total\_Stops.I have to replace these NaN values using Imputation technique.



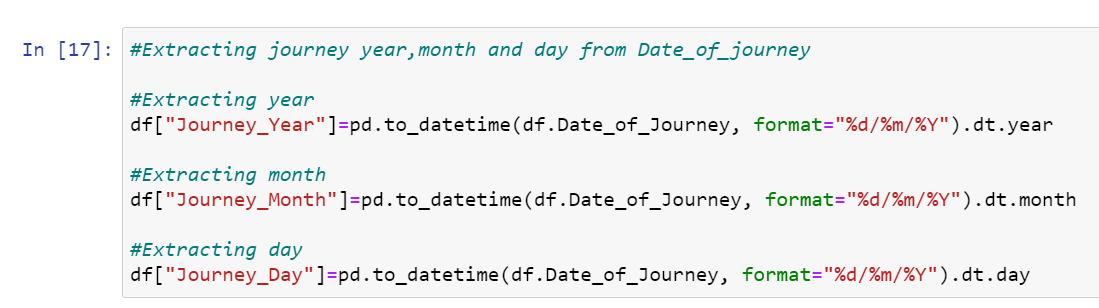
* I have replaced the NaN values with there mode since both Route and Total\_Stops are categorical columns.

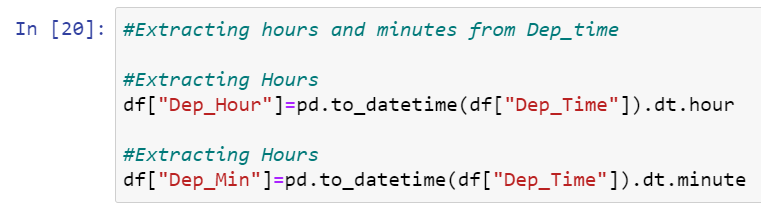
**Feature Extraction:**

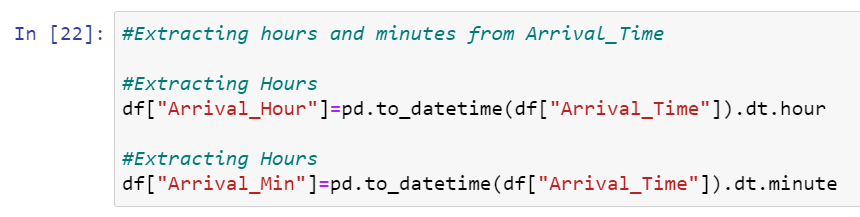
We have to do feature extraction from the date and time columns as the datatype of these columns are object we have to change there object data type to datetime data type.

****

* And I have extracted Journey\_day, Journey\_month and Journey\_year from Date\_of\_Journey column, Dep\_Hour and Dep\_min from Dep\_time and Arrival\_Hour, Arrival\_min from Arrival\_Time.



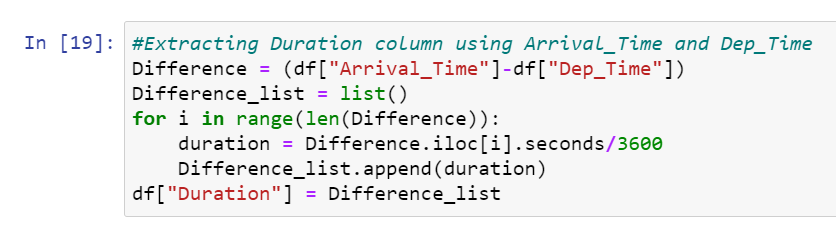




* After extracting all the necessary columns we can drop the old columns to avoid multicolinearity.



* With those old columns I have also droped Journey\_year extracted from Date\_of\_Journey. Because the value count of this column was having only single element 2019. Which means all the entries were 2019 in this column keeping all the entries same will not contribute anything for model building so I have droped this column.
* The present Duration column in the dataset is not in time format that is -h-m so I can extract Duration column from Dep\_Time and Arrival\_Time. Since Duration is the difference between these two features. And converted the Duration column as continuous data type.



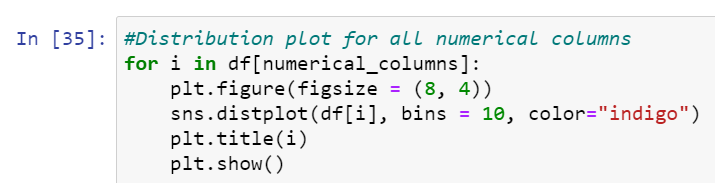
* Now the extraction has been done. Looking into all columns and there value count I noticed some duplicate entries that is in Airline, Destination, Additional\_Info there are 2 words with single meaning so I can group these entries with single entry.

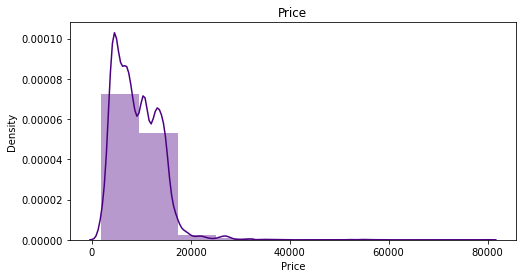


* Next looking at Total\_stops column there are specific entries like non-stop, 1-stop, 2-stop, 3-stop, 4-stop so I can use perticular labels for this column like 0,1,2,3,4 respectively to avoid misconception on the analysis.

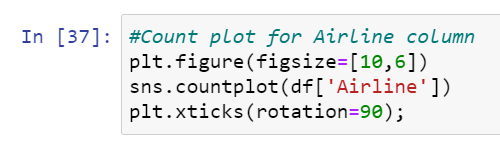


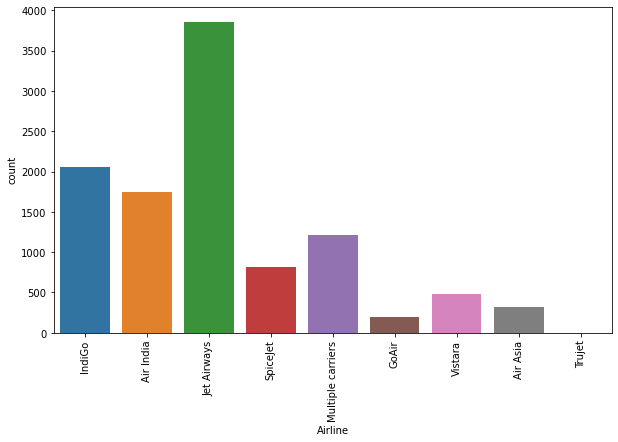
**Visualization:**

****

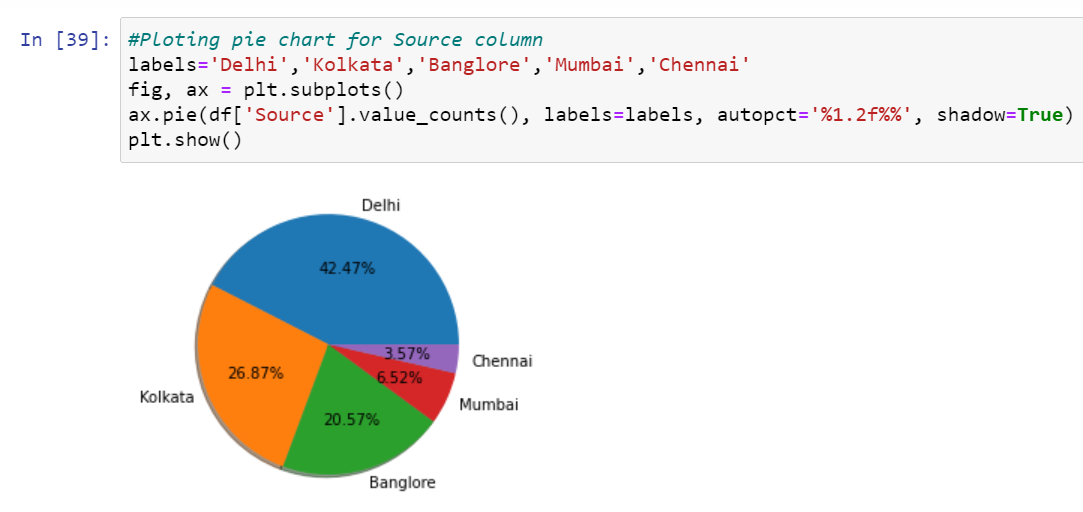
****

* I have plotted distplot for all numerical columns and since Price is my target column and I have observed skewness in target column. But still let me keep this skewness as it is, since I don’t want to manipulate my target.

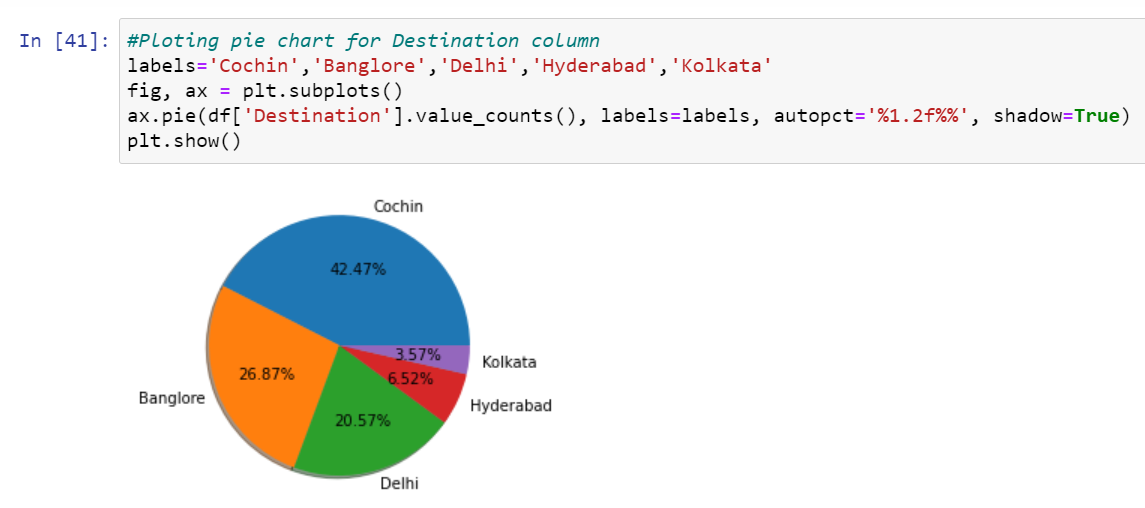


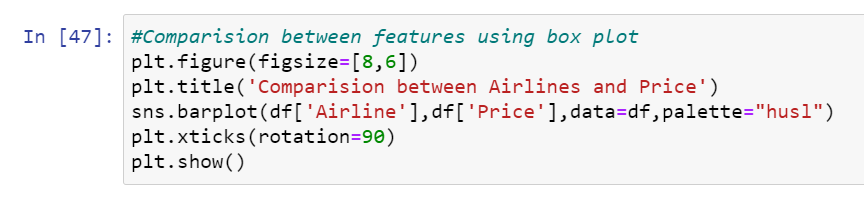
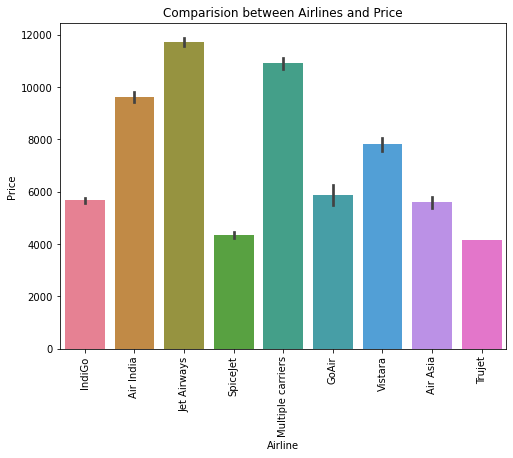


* Above is the count plot for Airlines I noticed that in all the cities between march and June Jet Airways is the famous Airlines.

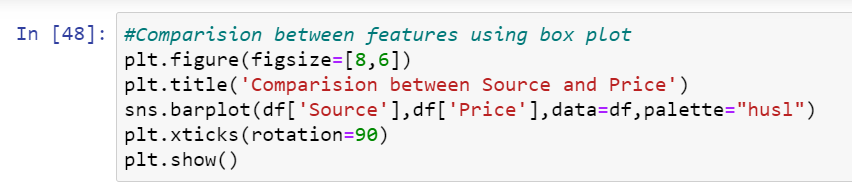
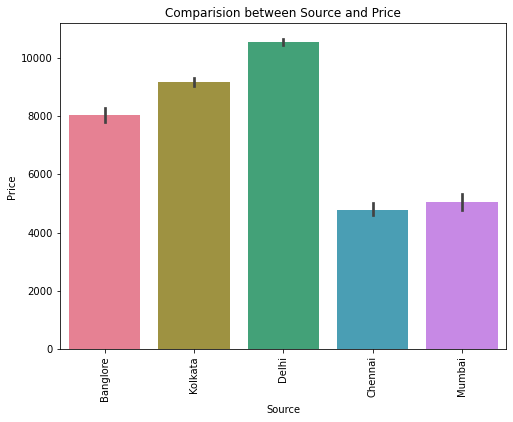


* Look at the pie chart for Source, most of the passengers chooses Delhi as there source. Since Delhi is the capital of India most of the people will be travelling in this state for there business or commercial purpose.
* Next plot is for Destination column. And the count is more for Cochin which means most of the passengers are having Cochin as there destination, since cochin is a busy city people will be going through this city for times.

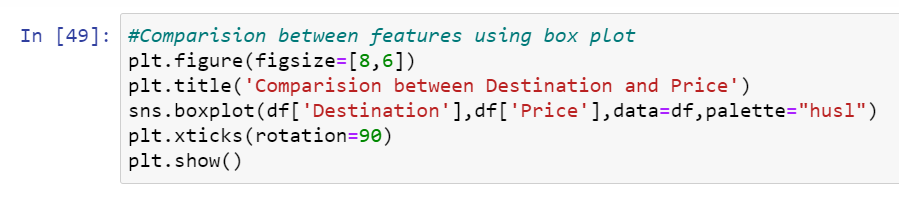


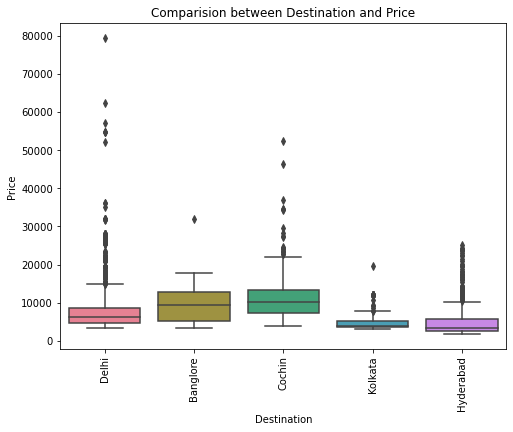


* Have a look into the plot, the price is high for Jet Airways, Multiple carriers and Air India. These three Airlines are expensive and also famous airlines. Most of the rich people from all cities prefers these airlines.

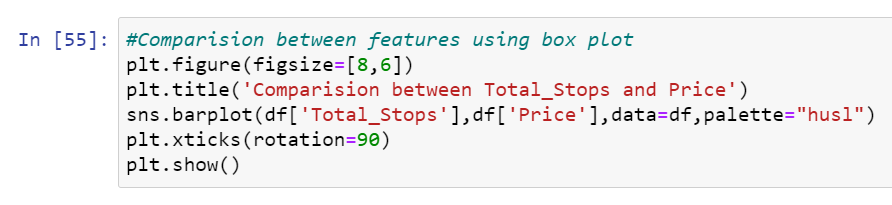


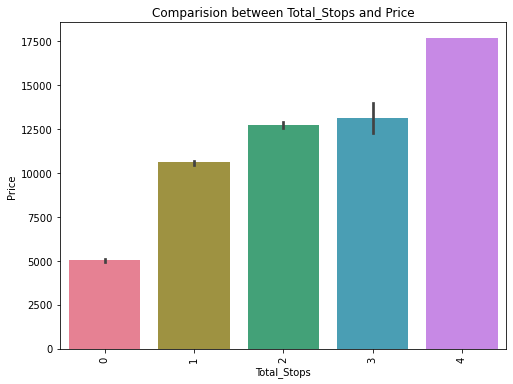
* Since Delhi is the capital and more and more passengers will take there source as Delhi so Price is also high in Delhi Airlines.





* This perticular plot is box plot for Destination and Price. Since Cochin was chosen as maximum Destination and Price for the same is also high.





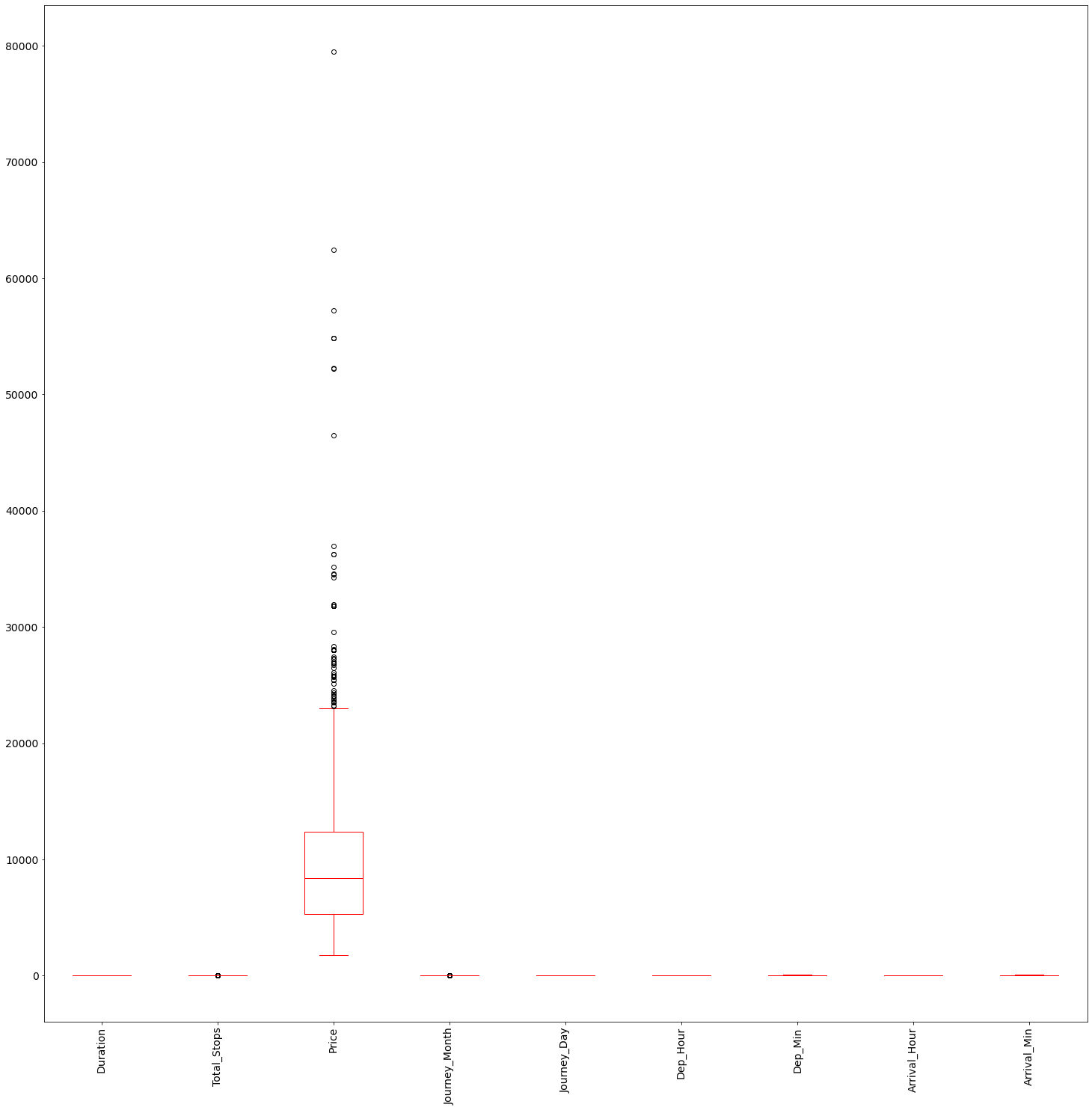
* It looks very evident that as the number of stops increses in the journey then the price is also going high.

**3.EDA Concluding Remark:**

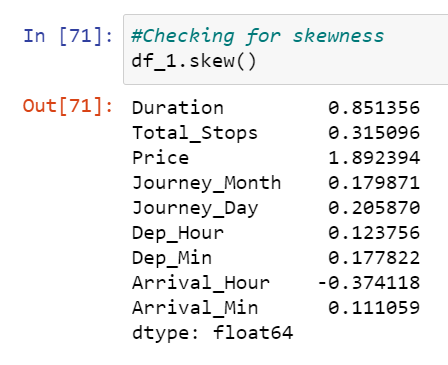
* I have used Imputation method to replace the NaN values.
* I have extracted the necessary features from existing features to get better accuracy and dropped the old columns to avoid multicolinearity. If I keep the old columns as it is then they will act as duplicates in the model.
* I have also grouped the duplicate entries in the features.
* I have used both matplotlib and seaborn to visualize the data.
* To get better insight on the features I have used distplot, barplot and boxplot since most of my columns were categorical I have used all categorical plots. For numerical columns I have used numerical ploting.

**Checking for Outliers and Skewness:**

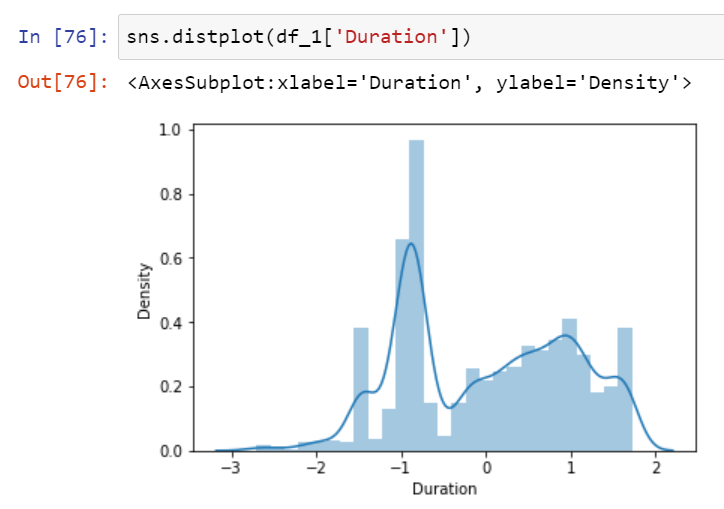
****

****

* I have used box plot to check outliers. And I found outliers in Total\_Stops, Price and Journey\_Month. Since Price is my target and Total\_Stops is a categorical column so I should not remove outliers from these two columns. Now I have to remove outliers only from Journey\_Month.
* To remove outliers I have chosen IQR method with dataloss of 8.9%, because in zscore the dataloss was 0% which means it is not removing even a single row. If it doesn’t remove any row then outliers will not be removed and using zscore has no meaning. After removing the outliers from IQR I have taken the new dataset as df\_1.

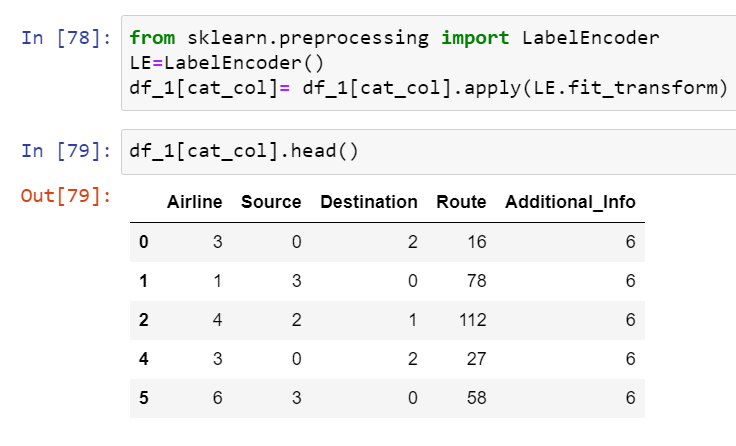


* I can notice skewness in only 2 columns Duration and Price. Since price is my target let me not remove the skewness and I have to remove skewness in Duration.
* To remove skewness I have used Yeo-johson method. I have also tried working on log, log1p, cbrt, sqrt but skewness was not at all reducing. After removing the skewness the distplot of Duration column is shown below.



* Now it looks good to proceed with the cleaned dataset df\_1.

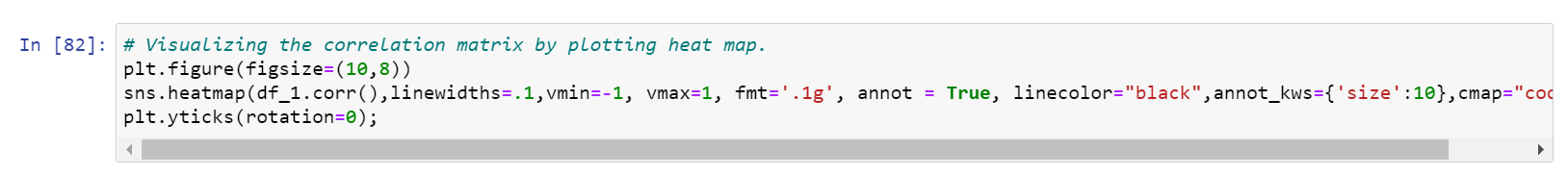
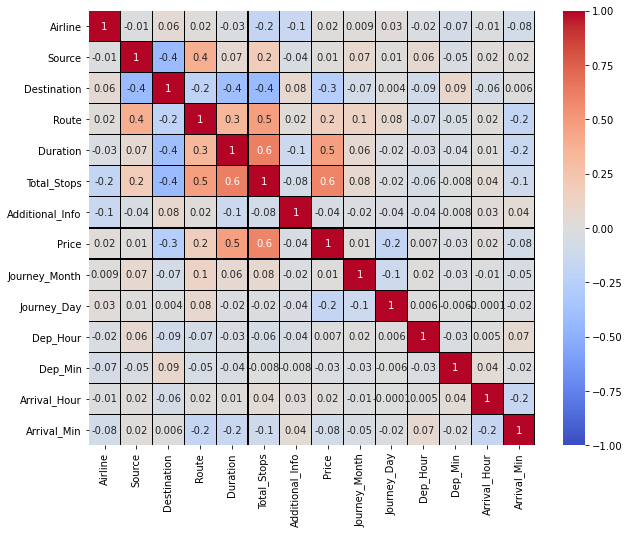
**Encoding:**



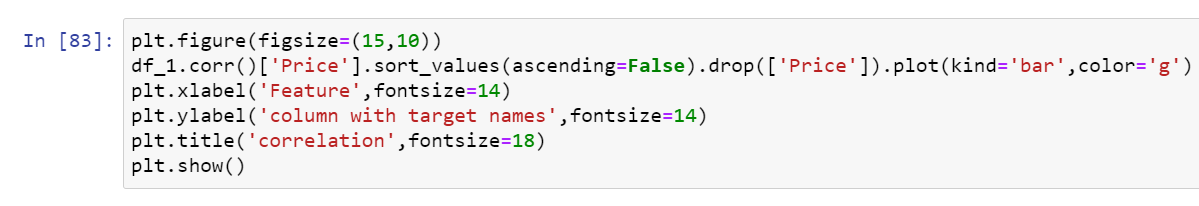
* For all the categorical columns in cleaned dataset df\_1 I have applied label encoding (Also tried with ordinal encoder but it decreased the model accuracy).

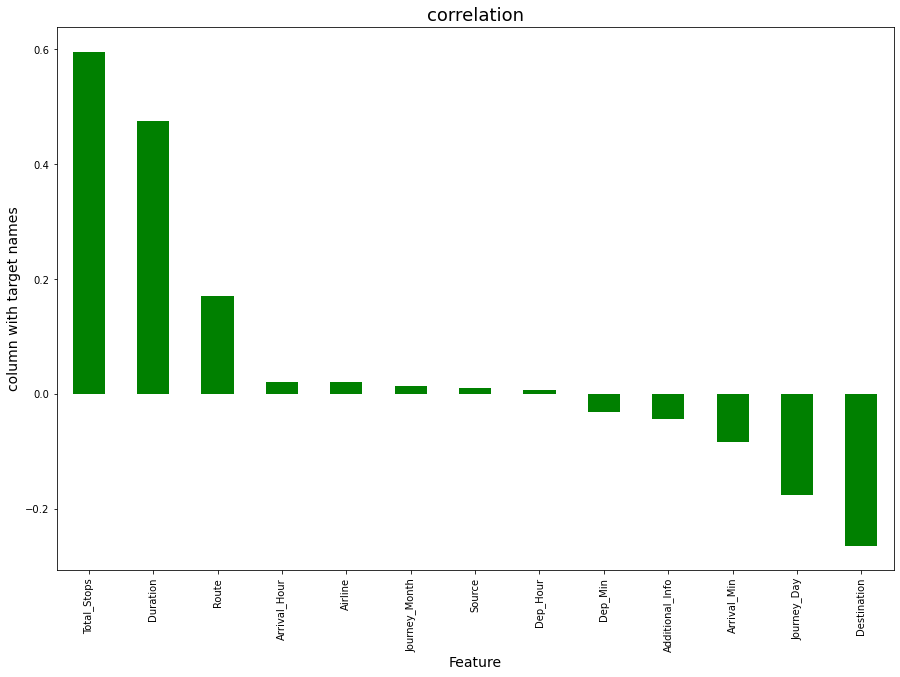
**Checking for correlation using heat map:**

After checking the correlation, to get better insight on the corr values I have plotted heat map. And this **correlation** has to be checked for **cleaned dataset**.

****

* Looking into the heat map I can say that there is no multicolinearity issue and to get better insight on targets correlation with other features I have ploted bar graph.

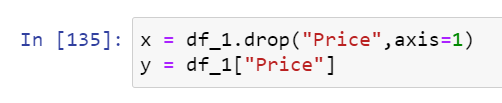
****

****

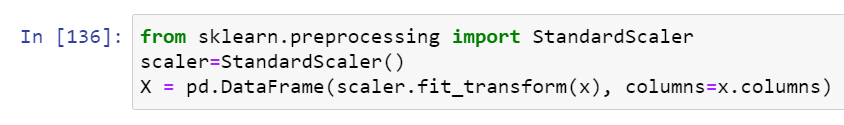
* Dep\_Hour and Source has very less correlation with target but let me keep the columns and build the model. Since I have very few columns dataloss may lead to less accuracy in models.
* Now the cleaning part for train dataset is complete. And I have to follow same steps and same methods used in train dataset for test dataset also. **To bring both the dataset’s features in same range, each and every step taken for train data has to be followed for test data also**.
* After doing all the EDA part for test dataset, we can proceed with preprocessing and model building using tarin dataset.

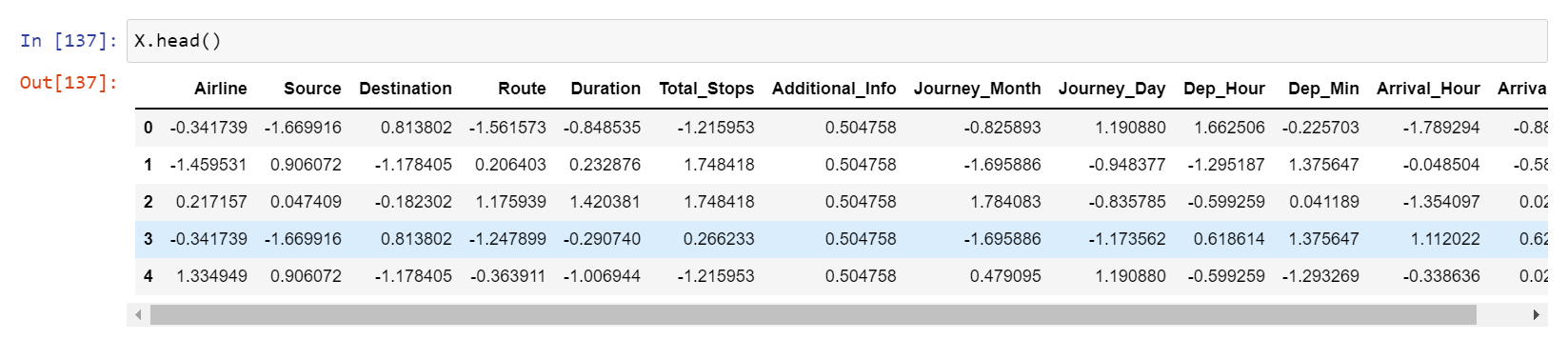
**4.Preprocessing Pipeline:**

* As a first step I have to separate the dependent and independent features.



* I have taken x as all independent features and y as dependent/target feature.
* Then I have to scale my independent features to get the same range of all columns. If I don’t scale my independent columns then there is a chance that my model may get baised. So In this perticular case I have used Standard scaling as I have removed all outliers and skewness from the dataset it is good to use standard scaling else we have MinMax scaler.



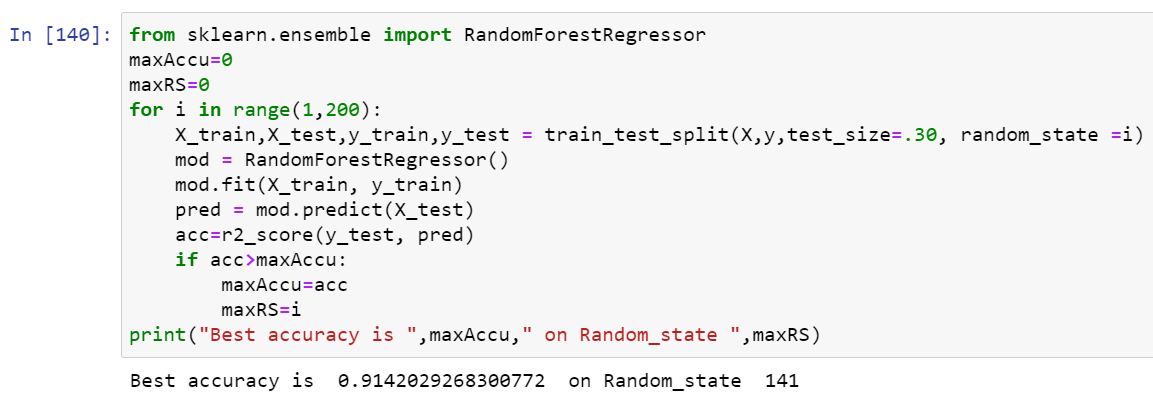


* Now my data is all set for model building. Let’s go ahead with regression algorithms since this is a Regression Problem.

**5.Building Machine Learning Models:**

1. **Finding best random state and accuracy:**

Let’s find the best Random state and accuracy first.



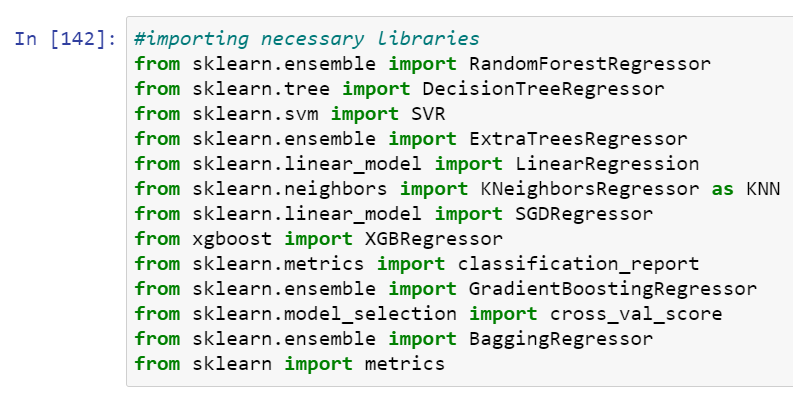
* I got best random state = 141 and accuracy = 91.42%. Now the task is to find the best fitting model.

****

* Created train and test data as X\_train, X\_test and y\_train, y\_test.

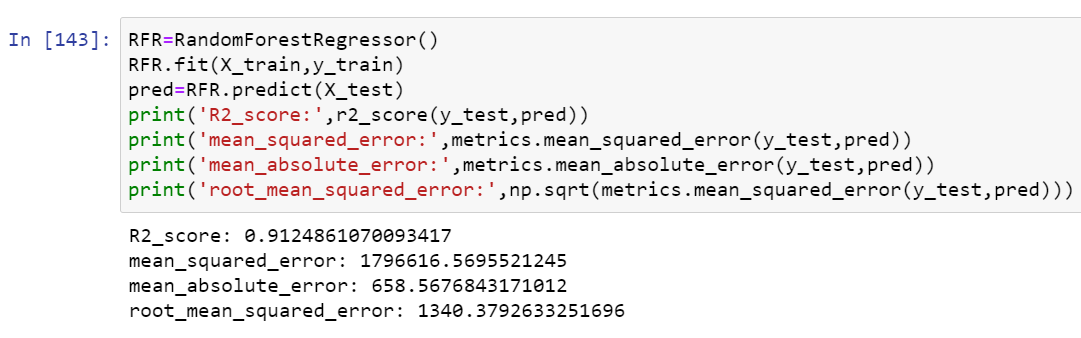
1. **Regression Algorithms:**

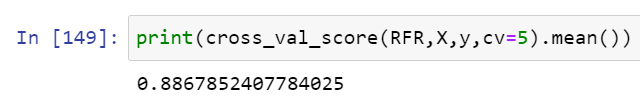
As a first step we have to import all the necessary libraries.



* I have used Cross validation as model evaluation metrics for all the algorithms. And I have used r2\_score, mse, mae, rmse metrics for model building.

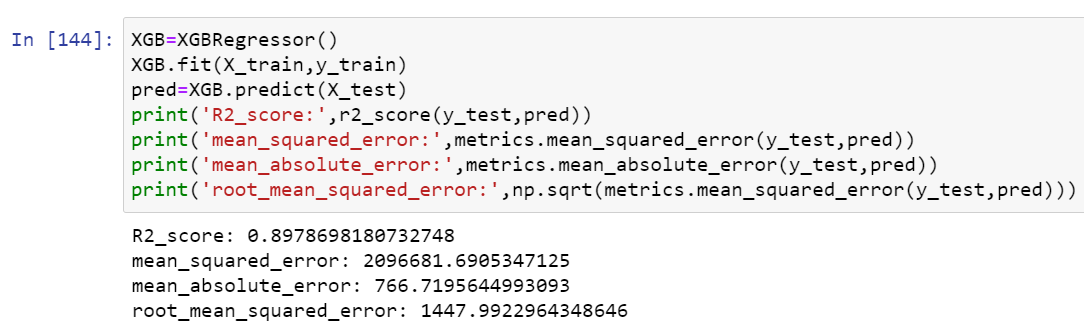
1. **Random Forest Regressor:**

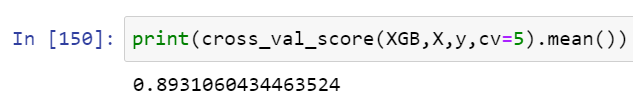


****

* Random Forest Regressor model is giving me 91.25% r2\_score and the cross validation is 88.68%. RFR is working good but I can not conclude it as good model before looking into multiple models.

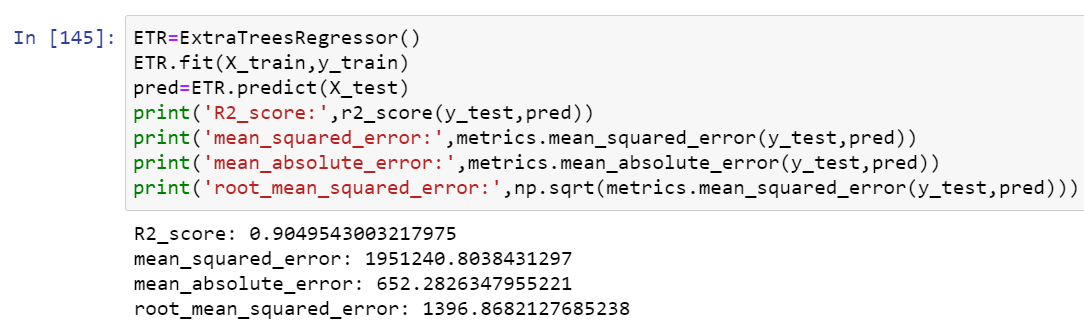
1. **XGB Regressor:**

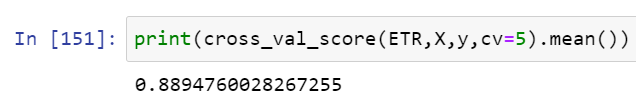
****

****

* XGB Regressor model is giving me 89.79% r2\_score and the cross validation is 89.31%.

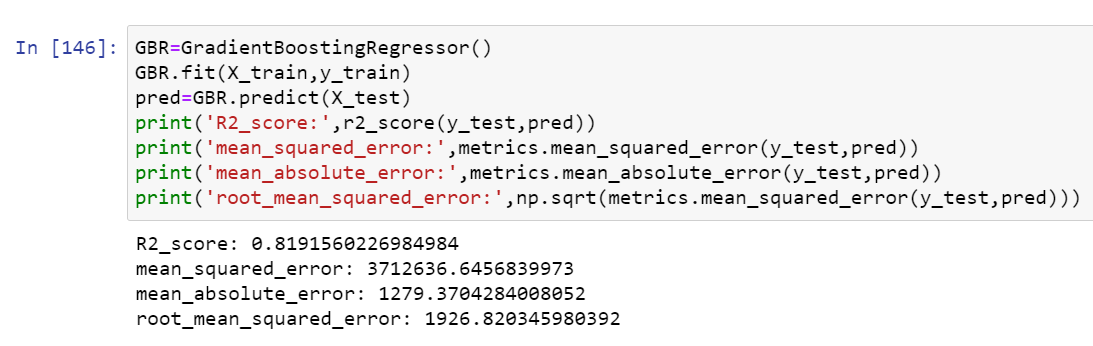
1. **Extra Trees Regressor:**

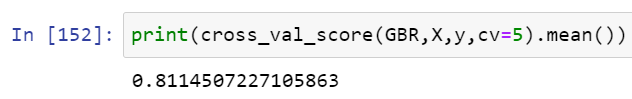
****

****

* Extra Trees Regressor model is giving me 90.50% r2\_score and the cross validation is 88.95%.

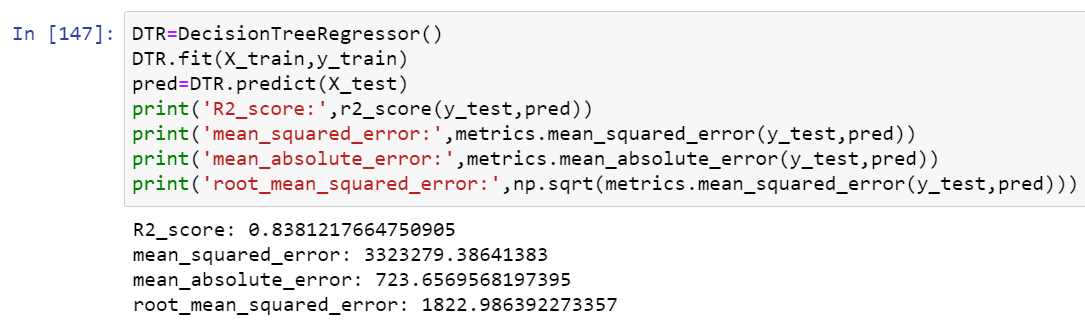
1. **Gradient Boosting Regressor:**

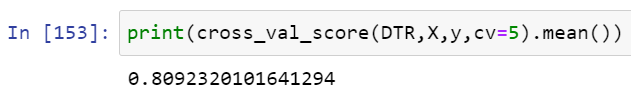
****

****

* Gradient Boosting Regressor model is giving me 81.92% r2\_score and the cross validation is 81.15%.

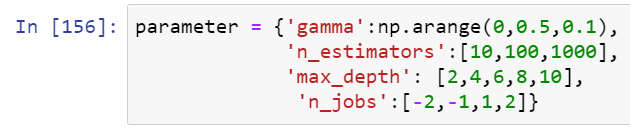
1. **Decision Tree Regressor:**

****

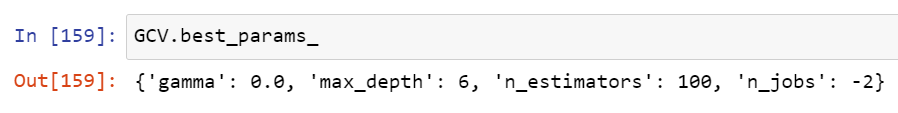
****

* Decision Tree Regressor model is giving me 83.81% r2\_score and the cross validation is 80.92%.
* **Now by looking into the difference of model accuracy and cross validation score I found XGB as best model with difference of 0.48%. And the model accuracy is 89.79% which is good but I can improve the model accuracy by tuning it. Let’s try to improve the model accuracy now.**

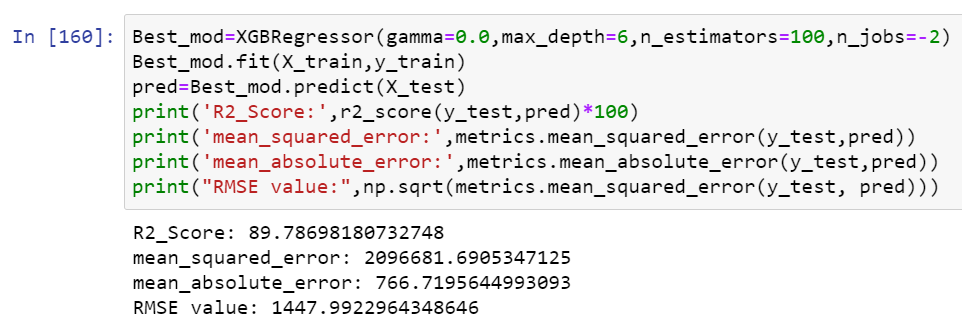
**Hyper Parameter Tuning:**



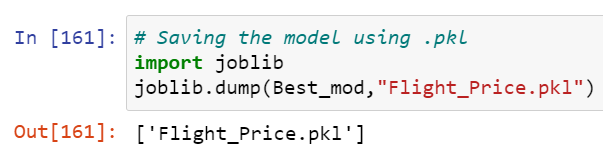
* Using the above parameters list I’m tuning my best model i.e., XGB. And I have to choose the best parameters in above parameter list, with those parameters I have to build my model.



* After knowing the above best parameters I have to run for improving model accuracy.

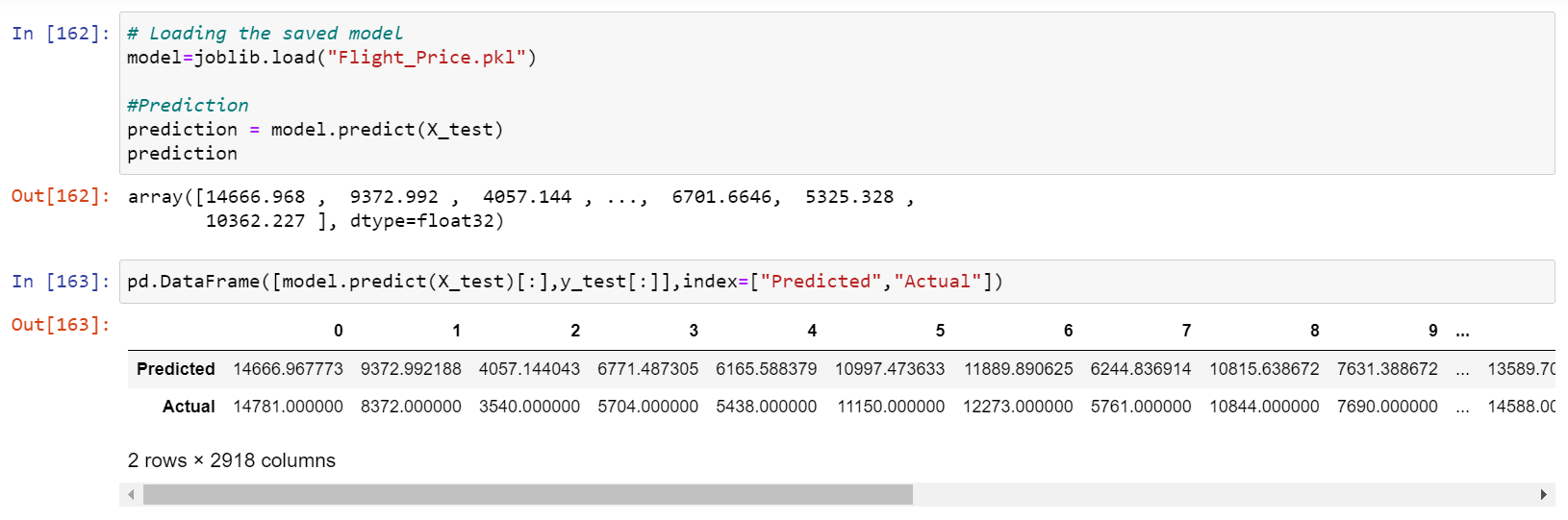


* Even after tunning the model accuracy is same which means the default parameters used by the model were giving the best accuracy. And the model is now ready with 89.79% accuracy which looks quite good.
* After getting this best model I have saved it using .pkl. As Flight\_Price.

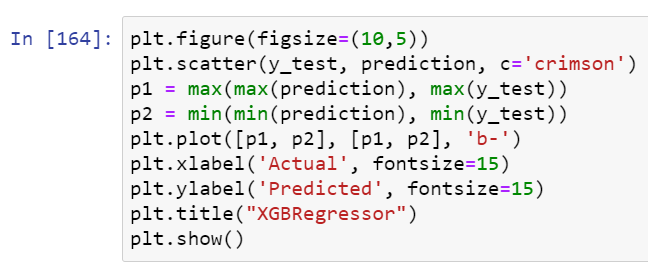


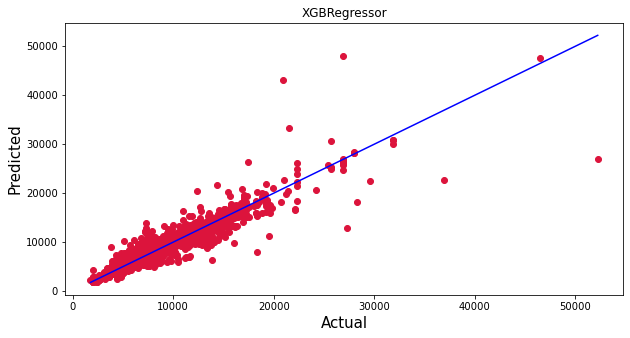
**Predictions:**

* Now using the saved model and cleaned test dataset I have to predict price for test dataset using best model of train dataset.



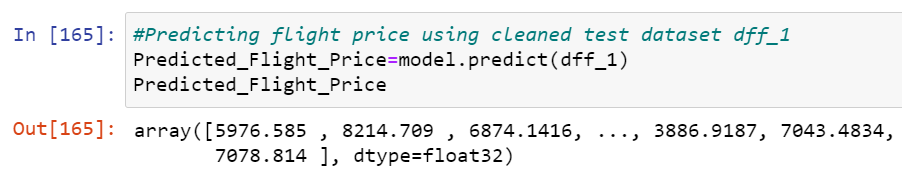
* After saving the best model we have to load the saved model and checking the actual verses predicted values.

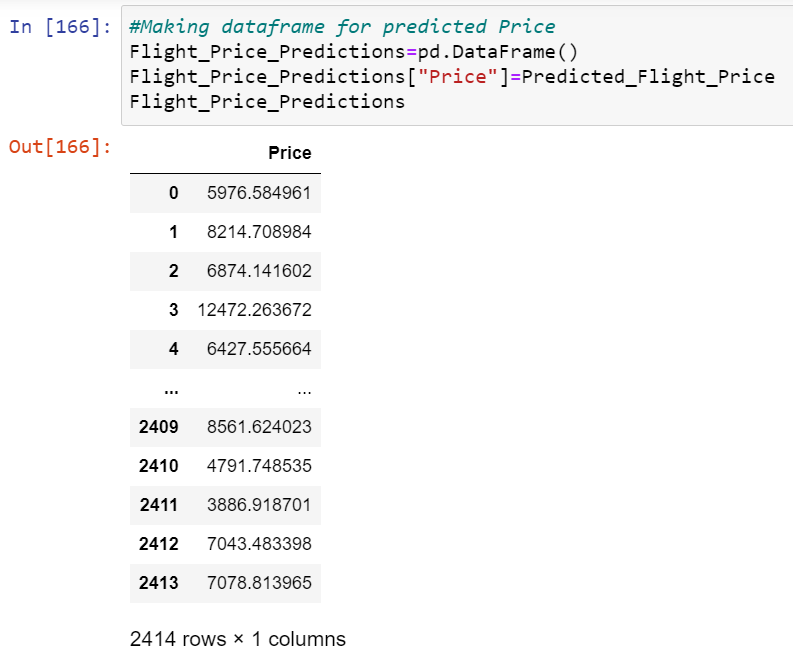




* Blue line is the actual values and red dots are predicted values and it’s pleasure to see my model is working good!!!😊.

**Predicting Price for test dataset using Best model:**





* I have predicted the price for cleaned test dataset. I have also framed the predited price as above.



* I have saved the prediction as csv file for further assistance or further analysis.

**6.Concluding Remarks:**

* ****This perticular problem needs a good vision on data, and in this problem Feature Engineering is the most crucial thing.
* You can see how we have handled numerical and categorical data and also how we build different machine learning models on the same dataset.
* Using hyper parameter tunning we can improve our model accuracy, for instance in this model the accuracy remained same.
* Using this machine Learning Model we passengers can easily predict the flight price and we could also save some amount while booking the Flight smartly.

****