**Flight Price Prediction using Machine Learning Techniques**



**1. Introduction**

Now a days, a lot of people are using flights to commute to various places. The flights may be booked for holidays or for a business trip but the demand for air travel is increasing day by day. Airline companies use complex algorithms to calculate flight prices based on the current situations at that time. They analyze various social, financial and market conditions to offer flight price and maintain flight price but the prices change dynamically due to various conditions. In this article we will talk about how to use machine learning to predict the flight prices based on previous booking preferences of the customers. Various data processing techniques and exploratory data analysis have been covered in this article. The performance of models were compared based of different parameters.

**2. 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

FEATURES:

**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.

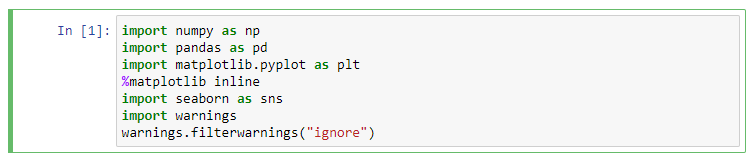
**Additional\_Info**: Additional information about the flight

**Price**: The price of the ticket

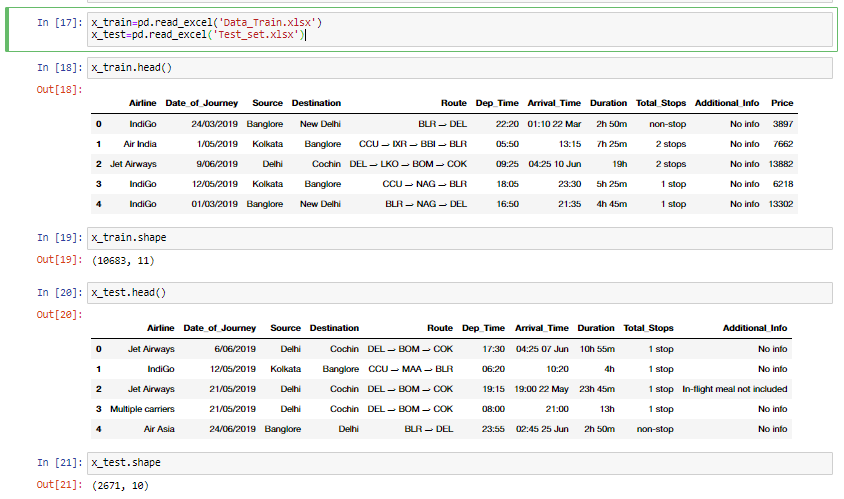
**3. Data Preprocessing and EDA**

The first step in building a model is to understand the dataset and prepare it for efficient prediction. In this step we use various cleaning techniques to produce a normally distributed dataset.

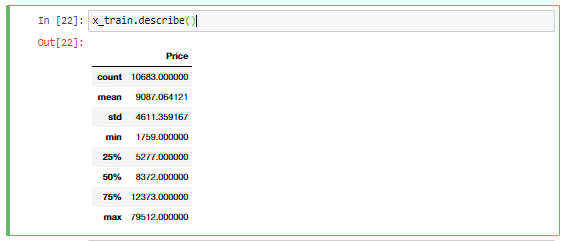
Initially we will import all the necessary libraries which we will use for our analysis



The next step is to load the csv file which contains our dataset. After loading the first thing which we do is check the dataset and understand the features available in the dataset. We will also check the size of the dataset, number of entries and number of columns. There are two sets of data, one is training set and other is test set. We will fit the training set into different models and check their performances. After the model is finalized we will predict the flight prices for the test set.



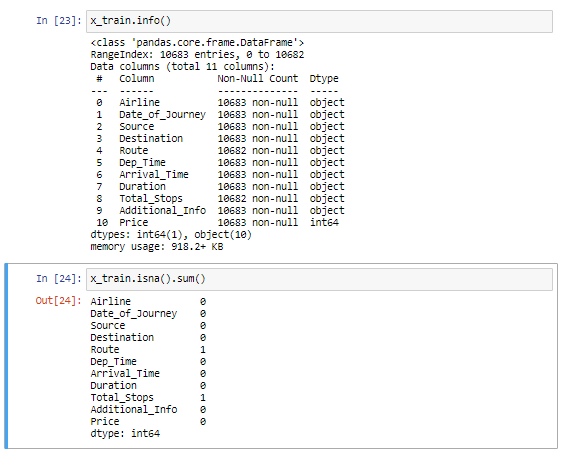
We see that there 10683 entries in training set and each of the entries have 10 features and 1 label. Our features are a mixture of categorical features and numerical features. We will identify the categorical features and numerical features and analyze them separately. There are 2671 entries in test set which have 10 features each. We will first process the training set and apply same pipeline for test set before prediction. Let us look at the data description to have an initial look at the variation of data.



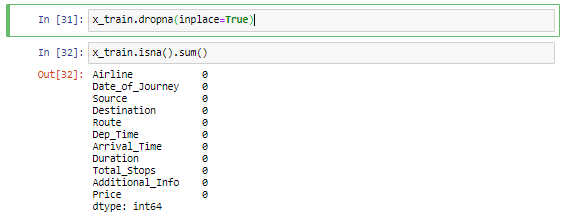
Only 1 column popped up using x\_train.describe() method which is the label while the rest 10 features must be categorical in nature. We are dealing with all the categorical features in this dataset. The count in all the features must be 614 but it is less than that for 4 features. Thus, we can notice that there are some missing values in the dataset. We will use appropriate imputation technique to fill the missing values.

From the first look analysis of all the columns it is clear that we need to do feature engineering to find out intermediate stops, flight time and hour of the day.

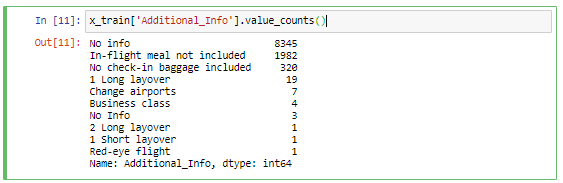
Lets find out how many missing values are there in the dataset.



The features which have missing values are Route and Total\_Stops. Just one value is missing from these two features so it is safe to drop the entries which have missing values without significant loss of information.

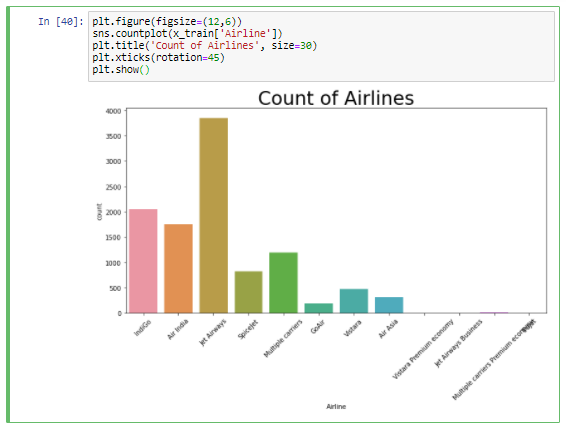


Lets us check the categories in the feature ‘Additional\_Info’.



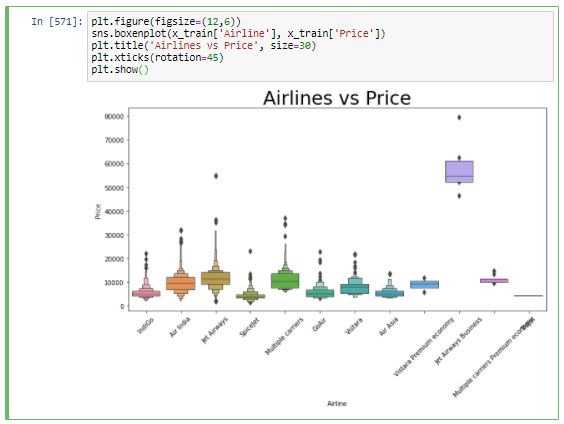
Majority of the categories in the feature 'Additional\_Info' has no info. This feature seems to be less informative and can be dropped from out data set.

Let us check the count of different airlines available for booking and plot count plot for the feature ‘Airline’.



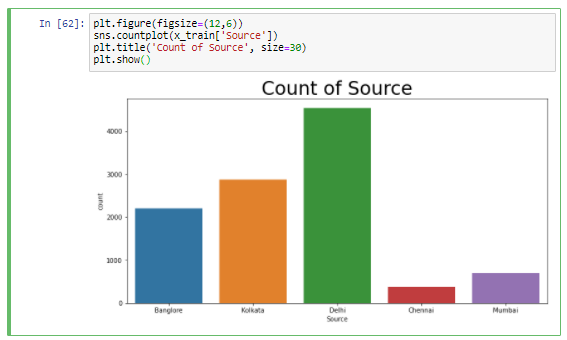
* Majority of the booking is done for Jet Airways followed by Indigo and then Air India.
* The number of bookings for Vistara Premium Economy, Jet Airways Business and Multiple Carrier Premium Economy is very less as compared to other economy class bookings.

We can check the boxen plot for Airline vs Price. This is going to give us an idea about the price range of flight tickets.



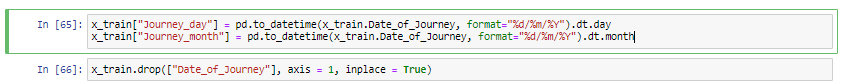
The flight price for Vistara Premium Economy is at an higher end, more than double the average price of other carriers.

Let see the source of the flight for out data set.

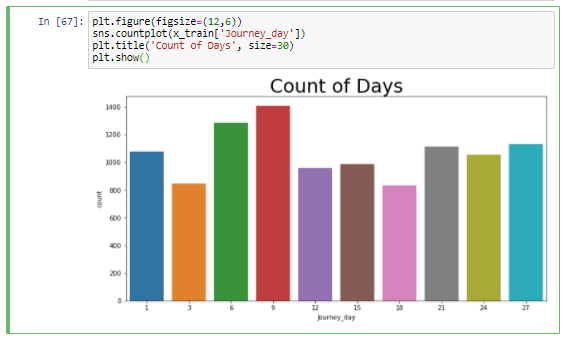


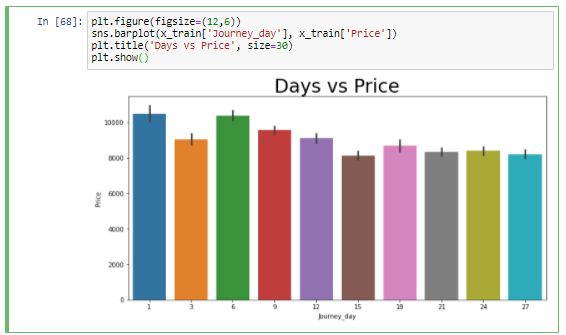
* Maximum number of flights are departing from Delhi while flight count is least for Chennai

The feature ‘Date of Journey’ contains date. For better analysis let us extract the day of the month for the journey and month of the journey.



We did feature engineering and created two new features ‘Journey\_day’ and ‘Journey\_month’.





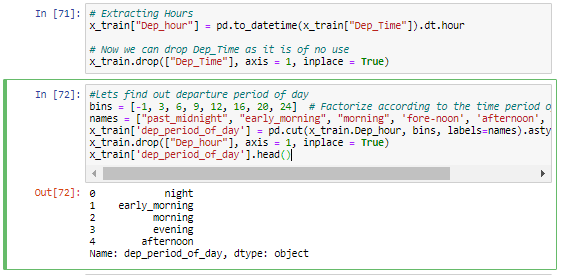
The plot of the day of the journey vs price shows that there is fluctuation in price throughout the month. You can identify difference in flight prices during weekends and weekdays. At this stage I have not done that analysis.

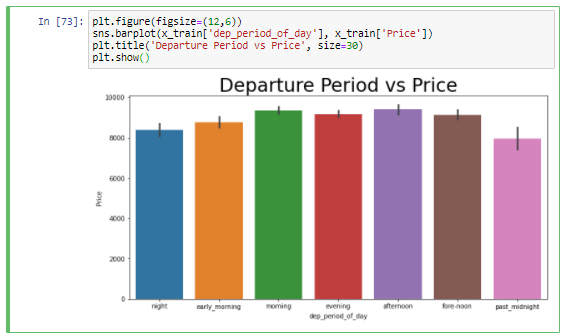
The new feature ‘Journey\_month’ which we created has numeric categories. Lets convert them to name of the month for better visualization and check the prices for different months.



The flight price were slashed down from March to April.

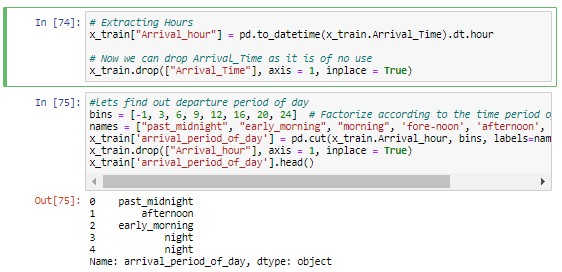
The feature ‘Dep\_Time’ contains the time of departure. We will do feature engineering by extracting the departure hour from that feature and convert it to different period during the day.





The plot of departure vs period of day shows that the customers have paid higher price for morning and afternoon while the flight price are comparatively less past midnight.

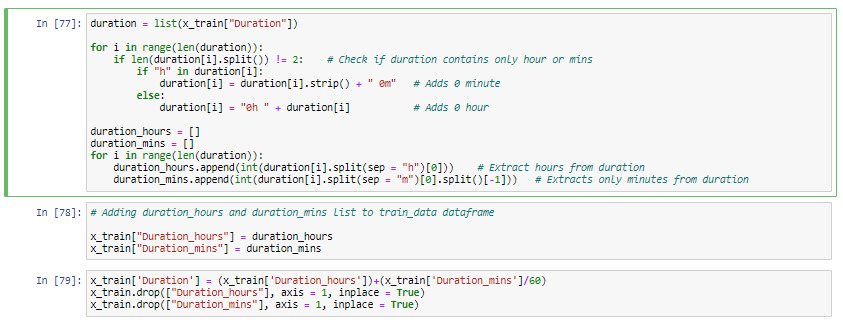
We can follow the same methodology for arrival time and visualize the price for different arrival period of the day.

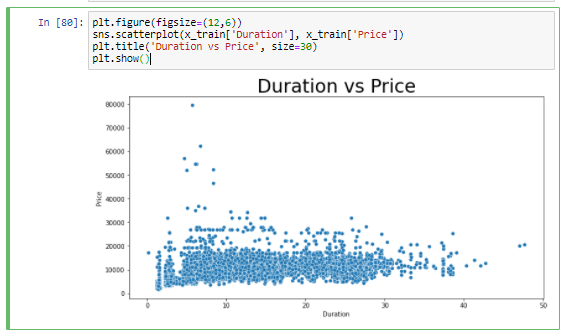




The plot for Arrival Period vs Price shows that the flight prices are highest when the arrival time is early morning followed by evening and it is least when the flight arrives past midnight.

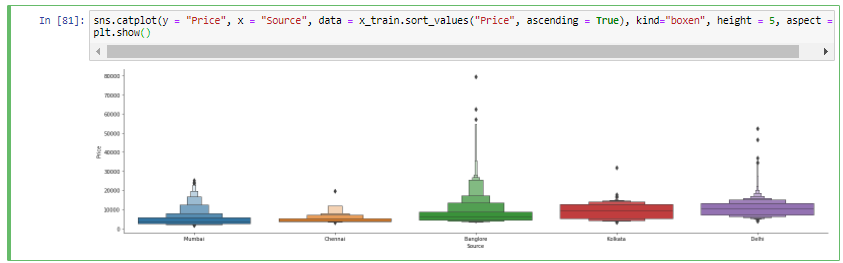
The feature ‘Duration’ is given in time format. So, from this feature we will calculate the flight duration in hours.



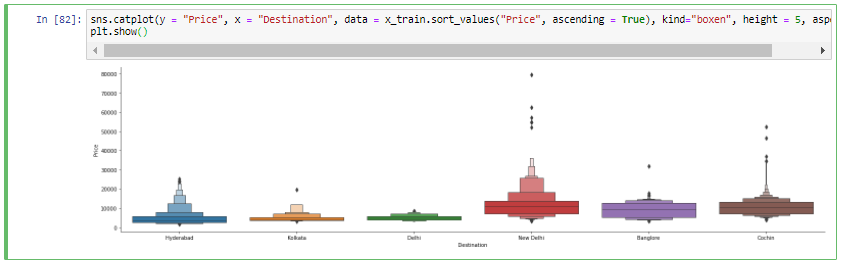


The flight prices are usually higher when the flight duration is long. One of the possible reasons may be the fact that duration increases for connecting flights.

The analysis of source of flight vs Price shows that the prices are maximum from Delhi and least for Mumbai.



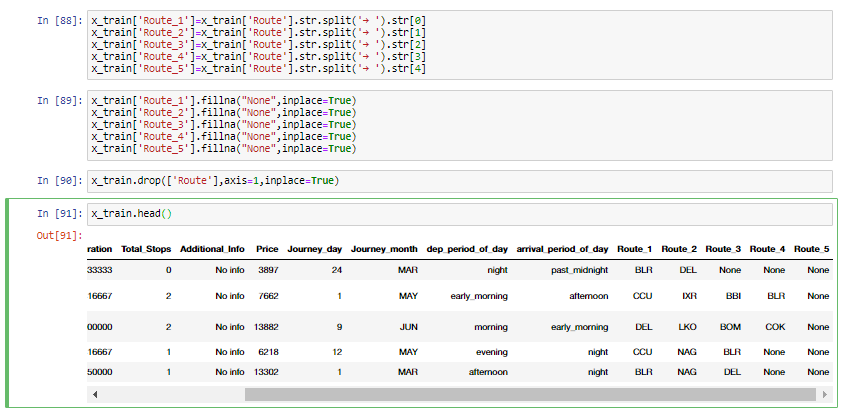
The flight prices are maximum for the Destination ‘Cochin’ and least for ‘Hyderabad’.



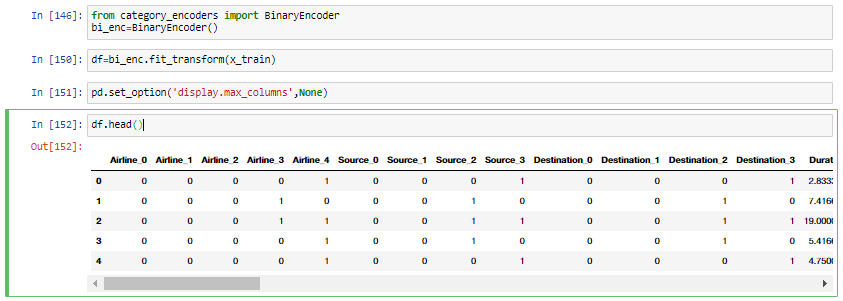
The feature ‘Total\_Stops’ has categorical values. We will modify this to numerical values which will be helpful in better analysis of the dataset.



For more number of stops the price is more. It is kind of obvious as the flight will be expensive for multiple stops. Now let us examine the feature ‘Route’. It has string values. We must extract the intermediate cities which the flight covers to understand the price flight path and help us in accurately predicting prices for different routes.



With this we are done with EDA and feature engineering. At this stage we need to encode all the columns which have categorical values. The model requires all input and output variables to be numeric, so we need to convert all categorical features into numerical values using proper encoding techniques. The categories can be encoded using binary encoder as this encodes the given information into a compact form. Binary encoder converts text attributes into numerical values for further processing.



After encoding we have 10682 entries and 54 features in our dataset. The next step in our analysis is to find out the features which are strongly correlated using correlation matrix and heatmap. This will help us to understand the relation between feature vs feature and in feature selection. We will drop the features which are highly correlated.



The heat map shows that the correlation between the features is not that significant. This was expected since all the features in our dataset have categorical values. So we are done feature selection. We will keep all the features for model prediction.

**EDA Concluding Remarks**

* The dataset has 10683 entries each entry has 10 features and 1 label.
* The features ‘Route’ and ‘Total\_Stops’ have only one value missing from the dataset. So, missing entries can be dropped without any loss of information.
* Majority of the categories in the feature 'Additional\_Info' has no info. This feature seems to be less informative and has been dropped from out data set.
* Majority of the booking is done for Jet Airways followed by Indigo and then Air India.
* The number of bookings for Vistara Premium Economy, Jet Airways Business and Multiple Carrier Premium Economy is very less as compared to other economy class bookings.
* The flight price for Vistara Premium Economy is at an higher end, more than double the average price of other carriers.
* Maximum number of flights are departing from Delhi while flight count is least for Chennai
* Two new features ‘Journey\_day’ and ‘Journey\_month’ from Date\_of\_Journey.
* The plot of the day of the journey vs price shows that there is fluctuation in price throughout the month.
* The new feature ‘Journey\_month’ which we created has numeric categories it is converted to name of the month for better visualization.
* The flight price were slashed down from March to April.
* The feature ‘Arrival\_Time’ contains the time of departure. The arrival hour was extracted from that feature and converted into different period during the day.
* The plot of departure vs period of day shows that the customers have paid higher price for morning and afternoon while the flight price are comparatively less past midnight.
* The feature ‘Dep\_Time’ contains the time of departure. The departure hour was extracted from that feature and converted into different period during the day.
* The plot for Arrival Period vs Price shows that the flight prices are highest when the arrival time is early morning followed by evening and it is least when the flight arrives past midnight.
* The feature ‘Duration’ is given in time format. Flight duration was calculated in hours from this feature.
* The flight prices are usually higher when the flight duration is long. One of the possible reasons may be the fact that duration increases for connecting flights.
* The analysis of source of flight vs Price shows that the prices are maximum from Delhi and least for Mumbai.
* The flight prices are maximum for the Destination ‘Cochin’ and least for ‘Hyderabad’.
* The feature ‘Total\_Stops’ has categorical values. It was converted to numerical values which was be helpful in better analysis of the dataset.
* For more number of stops the price is more. It is kind of obvious as the flight will be expensive for multiple stops.
* The feature ‘Route’ has string values. The intermediate cities were extracted from this info and added in the data set as different routes.
* The categories can be encoded using binary encoder.
* After encoding we have 10682 entries and 54 features in our dataset.
* The heat map showed that the correlation between the features is not that significant.

**4. Building Machine Learning Models**

The following models were tested to predict the approval of loan:

1. Linear Regression
2. Lasso
3. Ridge
4. KNN Regressor
5. Decision Tree Regressor
6. Random Forest Regressor
7. Ada Boost Regressor
8. Gradient Boost Regressor
9. XG Boost Regressor

The following table shows the MSE for all models fitted with our dataset without tuning hyper parameters.

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Model Name** | **Mean Squared Error** |
| 1. | Linear Regression | 2781.44 |
| 2. | Lasso | 2781.41 |
| 3. | Ridge | 2780.80 |
| 4. | KNN Regressor | 2583.39 |
| 5. | Decision Tree Regressor | 2355.88 |
| 6. | Random Forest Regressor | 1825.84 |
| 7. | Ada Boost Regressor | 3483.44 |
| 8. | Gradient Boost Regressor | 2101.14 |
| 9. | XG Boost Regressor | 1679.75 |

Decision Tree, Random Forest, Gradient Boost and XG Boost have performed better than other models. Lets tune the hyper paramwterers anch evaluate their performance.

The hyper parameters for the models were tuned using Randomized Search CV technique wherever applicable. The following table summarizes the performance of model:

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Model Name** | **Score** |
| 1. | Decision Tree Regressor | 0.72 |
| 2. | XG Boost Regressor | 0.83 |
| 3. | Random Forest Regressor | 0.80 |
| 4. | Gradient Boost Regressor | 0.81 |

The performance of XG Boost Regressor is better than any other model. Hence, XG Boost Regressor is selected as our final model

The hyper tuned parameters for the XG Boost Regressor are:

{'min\_child\_weight': 9,

'max\_depth': 7,

'learning\_rate': 0.25,

'gamma': 0.4,

'colsample\_bytree': 0.5}

The RMSE value of our final model is 1646.16.

There is a test set available with the data. All the data pre-processing techniques were applied for the test set as well so that we can accurately predict flight prices for test set as well.

**5. Concluding Remarks**

We cleaned the data, performed EDA and successfully trained a model and tuned the hyper parameters to predict the flight price for our test set. The XG Boost Regressor proved to be the best model in prediction as the model had the highest score and RMSE was minimum. I hope this has proven to be an informative topic.

You may refer to my repository for detailed code

https://github.com/dewangan-ashishk/Evaluation-Project/blob/d7f4a9741219cf4ffec99c16f6ca7e3ed5053799/Flight%20Price%20Prediction%20Final.ipynb