

**Flight Price Prediction**

Aaron Dsouza

Data Science

18-04-2021

# **Problem Definition:**

* Flight ticket prices can be something hard to guess, today we might see a price, and then, 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.
* As an aspiring Data Scientist, I have been allocated extensive data of various airlines. My job is to develop an accurate model that could predict the flight ticket prices.

**Data Analysis:**

* Data set link: **https://github.com/dsrscientist/Data-Science-ML-Capstone-Projects/blob/master/Flight\_Ticket\_Participant\_Datasets-20190305T100527Z-001.zip**
* The Flight price prediction dataset has 10,462 entries. Each entry contains the following information about the airline:
* **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.

### **Problem Type:**

### The target variable is Price.

### The target variable is continuous in nature.

### Solving it as a Regression Problem.

### On the circumstances given we need to predict the flight ticket prices.

**Data frame information:**

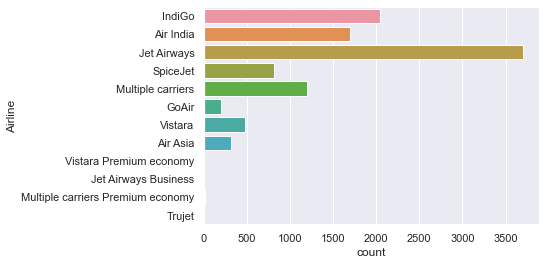
### The dataset has two datatypes Object and Int.

### There are 220 duplicated values in dataset.

### Some null values in the dataset.

### The dataset contains information about 12 different airlines.

**EDA (Exploratory Data Analysis):**

**Fig. 1 Airlines**

### The **Airline** feature describes the different working airlines in India. Fig. 1 Shows the distribution of Airlines in the data set. We can clearly observe that **“Jet Airways”** is dominating the Airline industry. **“Jet Airways”** has the highest number of flights running a total of 3700. From the above plot, we can also conclude that **“Jet Airways”**, **“Indigo” and “Air India”** are the leading airlines in the nation.

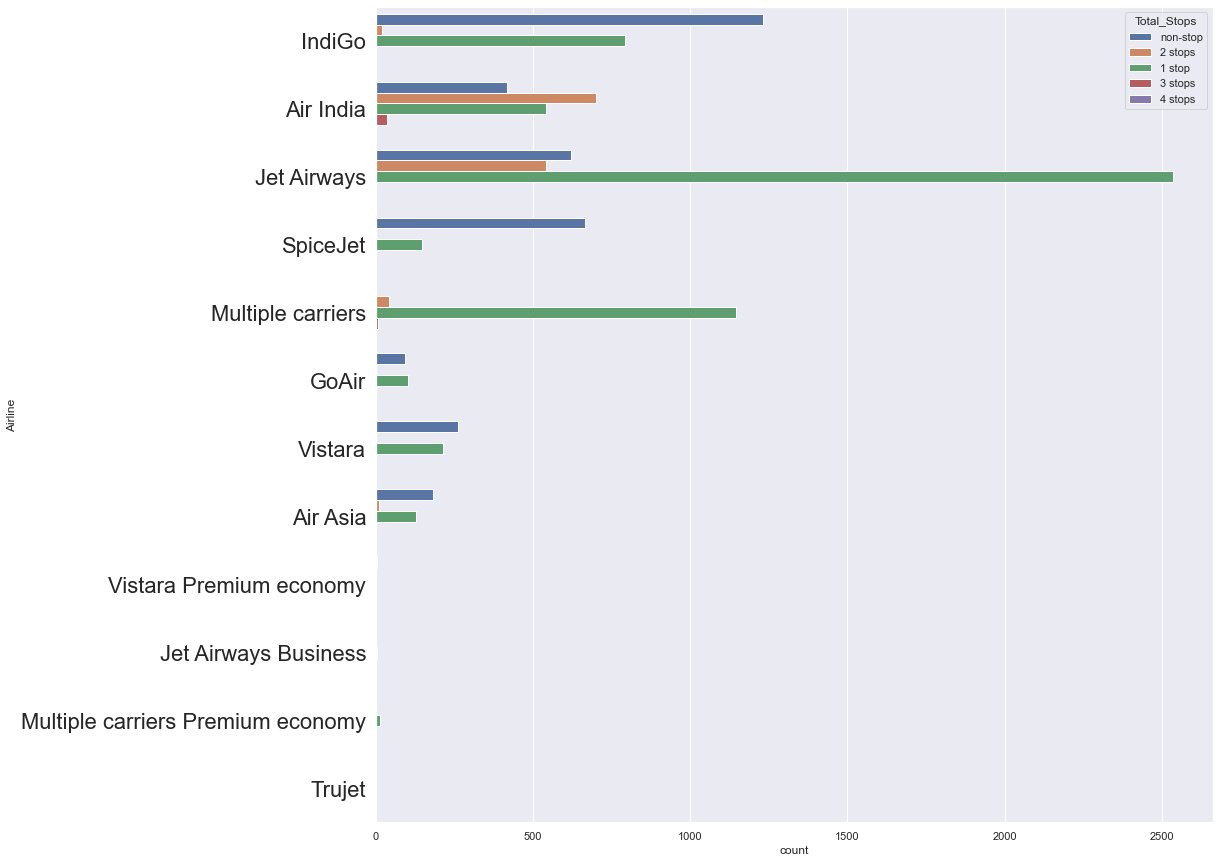
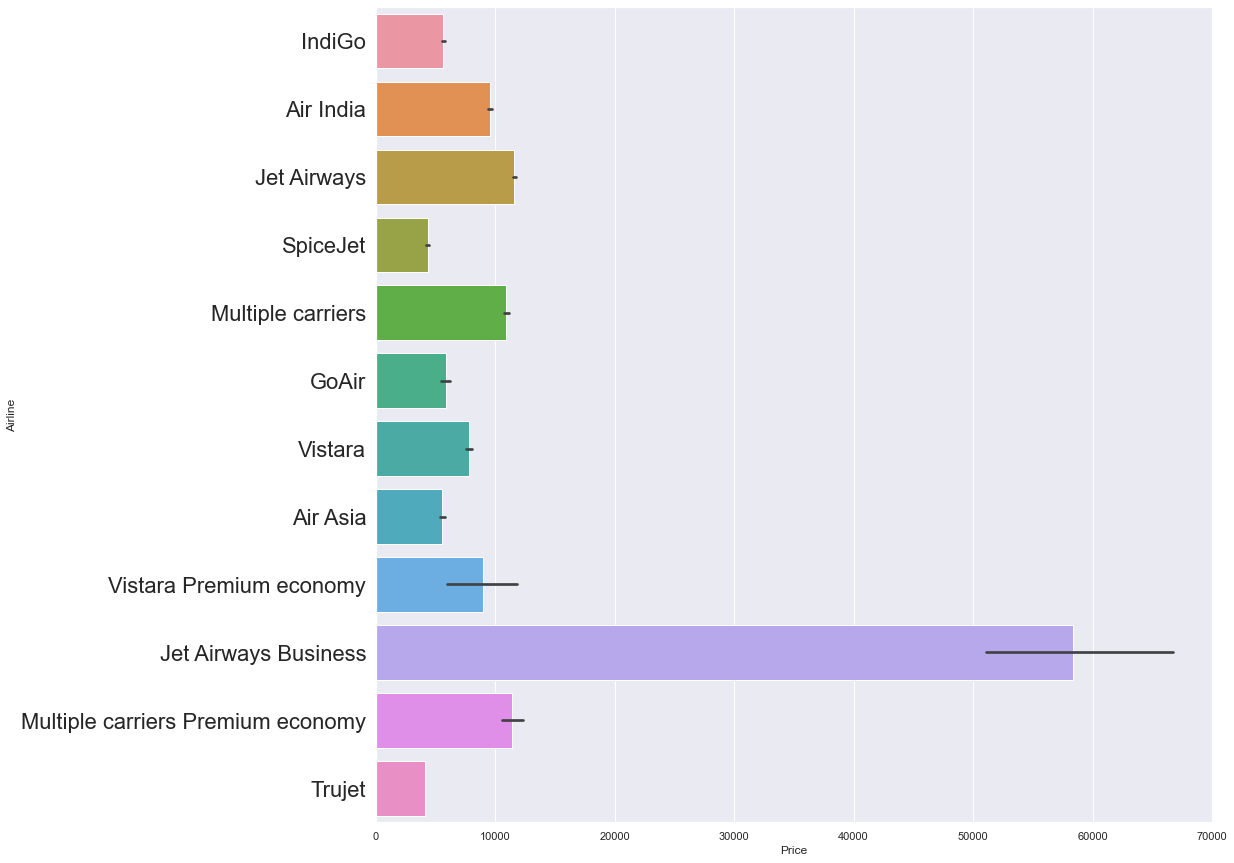
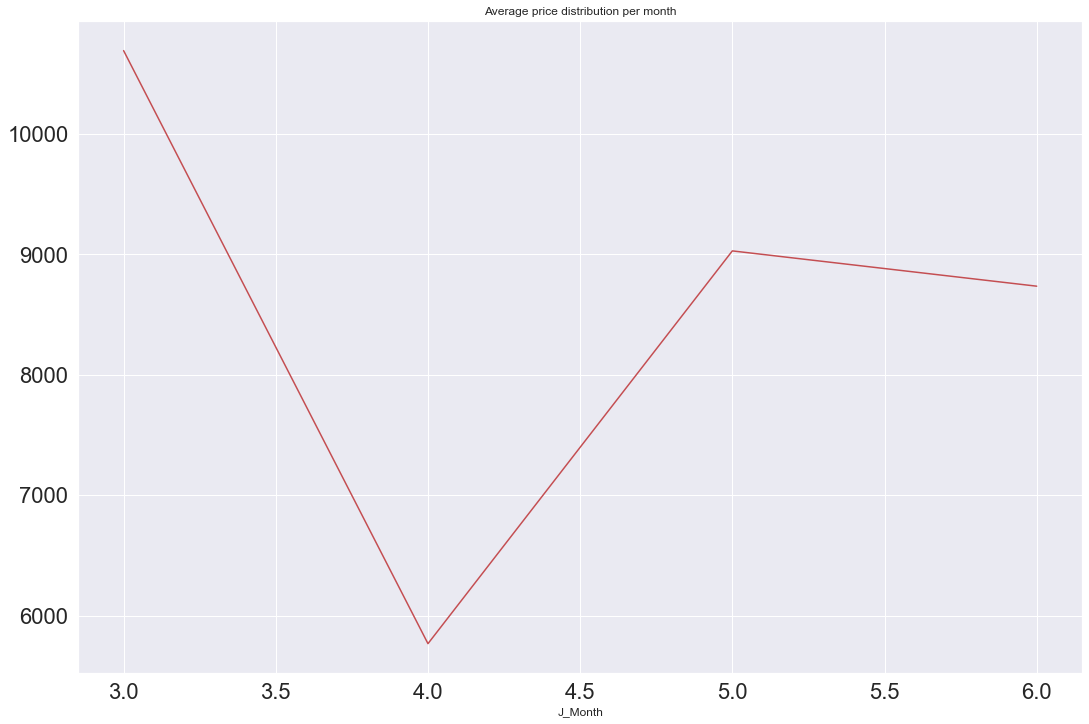
 **Fig. 2 Airlines with hue as total stops**

Fig. 2 shows the distribution of various airlines in India. With the help of count plot, and taking the hue attribute as Total stops, we conclude that Indigo Airline is leading when it comes to no stops (no halts). We can say that the longer the distance between source and destination location more will be the number of halts. Go Air, Vistara, and Air Asia are having a lower frequency of flights.



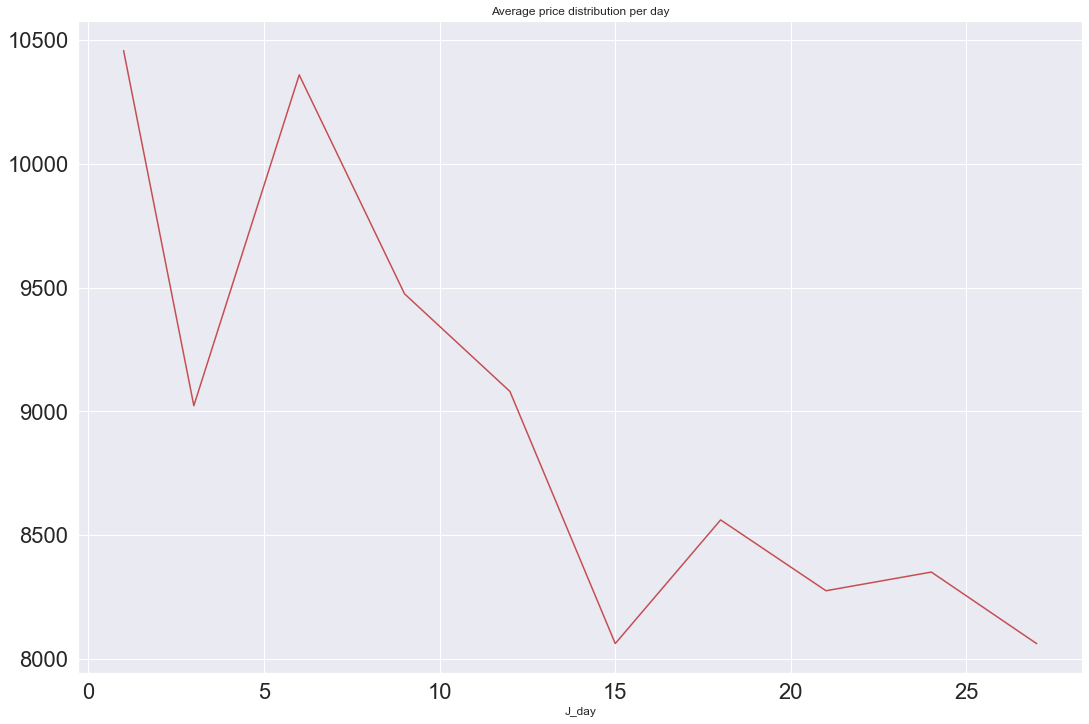
**Fig. 3 Airlines Vs Price**

Fig. 3 shows the relationships between various airline companies and their prices. From the above plot, we can clearly conclude that Jet Airways Business class is the most expensive airline company. Maybe most of the VIP clients travel in it. Indigo, Air India, and jet airways are somewhat economical but it also depends on the travel distance. And when it comes to business class only a few people can afford it.



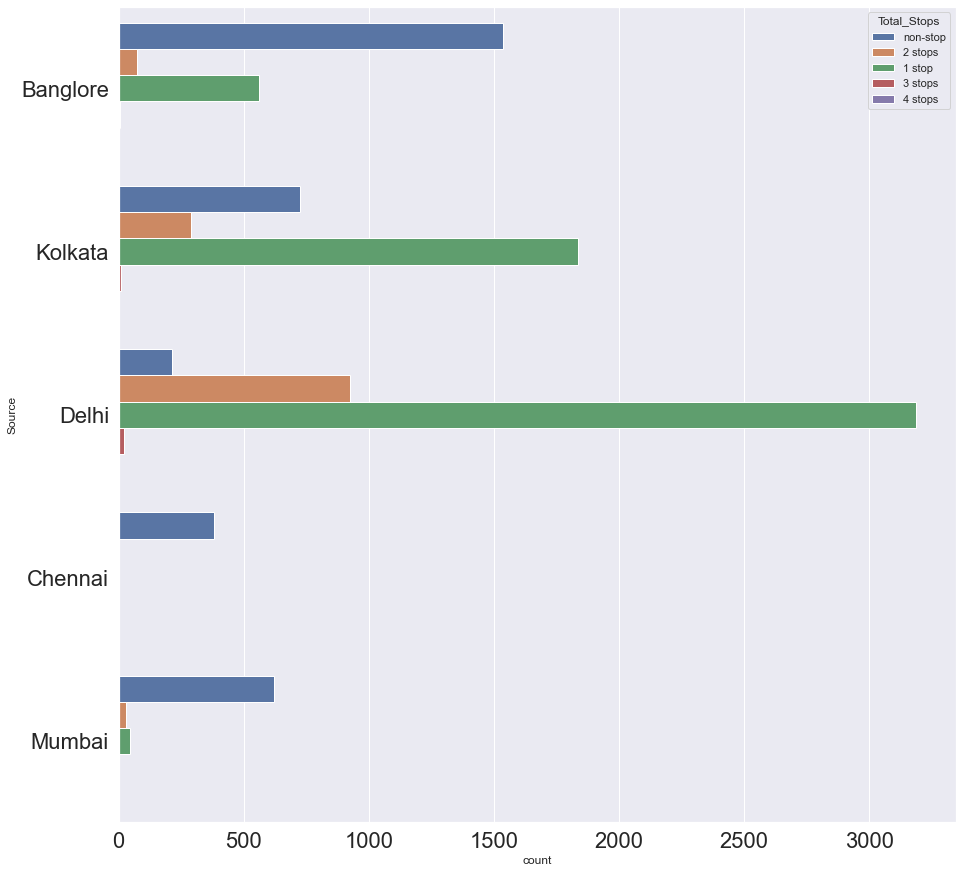
**Fig. 4 Price distribution per month**

Fig. 4 shows the average price distribution of Airline ticket prices per month. We have the data from the month of March to June. There is some pattern in the data the prices of tickets are very high in early March it can be due to some Festive season. The prices are somehow going down as the month-end in march approaches. The prices are economical till mid-April. Prices then again go high in the month of May and then the prices are somewhat stable.



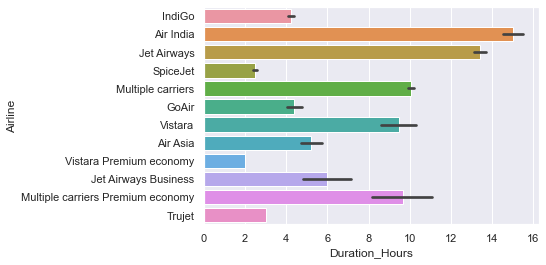
**Fig. 5 Price distribution per day**

Fig. 5 shows the average price distribution of Airline ticket prices per day. We have the data from the month of March to June. The prices are very high especially during the first two weeks of the month. The prices are economical and stable during the last two weeks of the month so depending on this data it is better to purchase tickets during month-end period for better savings.



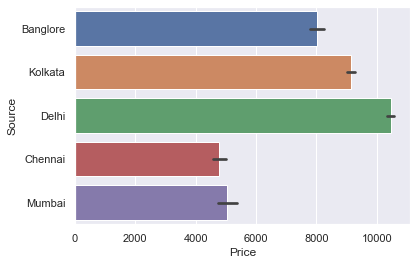
**Fig. 6 Source with hue as Total Halts**

Fig. 6 shows the distributions of all the Source destinations that the individuals travel to. Bangalore, Kolkata, Delhi, Chennai, Mumbai are the major source locations. For all the people traveling from Bangalore, in most cases, the flight will have no halts (non - stop). The people traveling from Delhi will have the greatest number of Halts. All the people traveling from Chennai will only have nonstop flights (no halts).



**Fig. 7 Airline vs Duration hours**

Fig. 7 shows the bivariate relationship between Airline companies and Duration hours. Air India is having the highest flight duration in terms of (Hours) i.e. when people are traveling to a foreign country. So, when traveling abroad most of the people are choosing either air India or jet airways. Spice jet and Vistara premium economy flights have the lowest duration hours.



**Fig. 8 Source vs Price**

Fig. 8 shows the bivariate relationship between Source locations and Ticket prices. All the people traveling from Delhi and Kolkata are having higher ticket prices. All the people traveling from Chennai are having lower ticket prices.

**Pre-processing Pipeline:**

* 220 duplicated values were dropped from the dataset
* The single null value in the “Total Stops” column was replaced by the mode of the “Total Stops” column.
* The “Route” column was dropped as it was proving to have less significance (in terms of correlation) in predicting the target variable.
* We need to extract "Day" and "Month" from the "Date of Journey" column and we need to convert the column from object datatype to pandas Date time
* Two new columns “J Day”, “J month” were created by extracting day and month information from the “Date of journey” column. Now we can drop the “Date of journey” column.
* Flight departure time and arrival time are important features we need to extract information like hours and minutes from it
* Two new columns “D Hour”, “D minute” were created by extracting hour and minute information from the “Departure Time” column. Now we can drop the “Departure Time” column.
* Flight arrival time is an important feature we need to extract information like hours and minutes from it.
* Two new columns “A Hour”, “A minute” created by extracting hour and minute information from the “Arrival Time” column. Now we can drop the “Arrival Time” column.
* Making separate columns for duration hours and duration minutes using the string split method. “D Hours” and “D minutes” were created from the “Duration” Column and now we can drop the original “Duration” column.
* There are some OUTLIERS in the columns “Price”, and “Duration Hours”.
* The OUTLIERS were removed with the help of the Z Score Technique and after removing the outliers the total data loss was 1 % so we can go ahead with it.
* The skewness range was considered between (-0.50 to +0.50).
* Skewness was only considered for continuous variables not for categorical variables.
* Skewness was reduced with the help of the **Log Transformation** method.
* Pandas get dummies method was used for one-hot encoding the categorical variables
* Train Test split from Sklearn was used in order to split the data for training and testing the model taking the test size as 0.25
* Min Max Scalar from Sklearn was used to transform the features in a given range of [0,1].

**Building Machine Learning Models:**

**A total of 6 Machine learning (Regression) models were used in order to predict the flight ticket prices.**

**Linear Regression:**

* **Linear Regression is a supervised machine learning algorithm where the predicted output is continuous and has a constant slope. It's used to predict values within a continuous range.**
* **For loop was used to decide the best random state for the algorithm**
* **K fold Cross validation technique was used.**
* **number of splits = 10 to decide the cross-validation score**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Training Score | R2 Score | K fold Cross validation Score |
| Linear Regression | 0.712 % | 0.714 % | 0.710 % |

**Random Forest Regression:**

* **Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean/average prediction (regression) of the individual trees.**
* **For loop was used to decide the best random state for the algorithm**
* **K fold Cross validation technique was used.**
* **number of splits = 10 to decide the cross-validation score**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Training Score | R2 Score | K fold Cross validation Score |
| Random Forest Regression | 0.98 % | 0.94 % | 0.92 % |

**Gradient Boosting Regression:**

* **Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees.**
* **For loop was used to decide the best random state for the algorithm**
* **K fold Cross validation technique was used.**
* **number of splits = 10 to decide the cross-validation score**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Training Score | R2 Score | K fold Cross validation Score |
| Gradient Boosting Regression | 0.84 % | 0.83 % | 0.83 % |

**ADA Boosting Regression:**

* **AdaBoost is one of the first boosting algorithms to be adapted in solving practices. Ada boost helps you combine multiple “weak classifiers” into a single “strong classifier”.**
* **For loop was used to decide the best random state for the algorithm**
* **K fold Cross validation technique was used.**
* **number of splits = 10 to decide the cross-validation score**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Training Score | R2 Score | K fold Cross validation Score |
| Ada Boosting Regression | 0.68 % | 0.67 % | 0.66 % |

**K-NN Regression:**

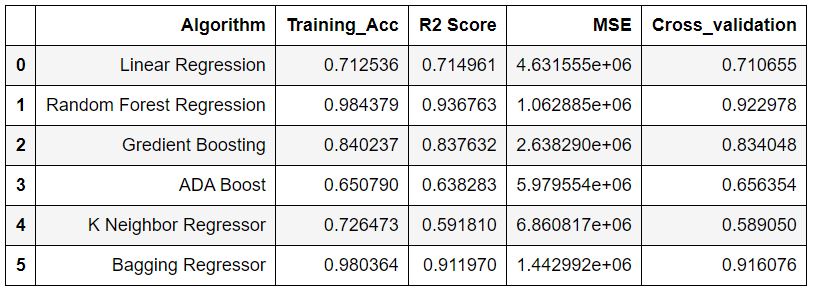
* **KNN regression is a non-parametric method that, in an intuitive manner, approximates the association between independent variables and the continuous outcome by averaging the observations in the same neighborhood.**
* **For loop was used to decide the best random state for the algorithm**
* **K fold Cross validation technique was used.**
* **number of splits = 10 to decide the cross-validation score**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Training Score | R2 Score | K fold Cross validation Score |
| K-NN Regression | 0.72 % | 0.59 % | 0.58 % |

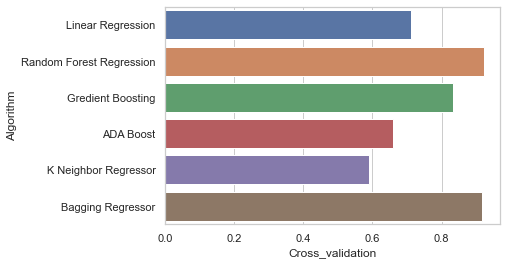
**Bagging Regression:**

* **A Bagging regressor is an ensemble meta-estimator that fits base regressors each on random subsets of the original dataset and then aggregate their individual predictions (either by voting or by averaging) to form a final prediction.**
* **For loop was used to decide the best random state for the algorithm**
* **K fold Cross validation technique was used.**
* **number of splits = 10 to decide the cross-validation score**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Training Score | R2 Score | K fold Cross validation Score |
| Bagging Regression | 0.98 % | 0.90 % | 0.91 % |

**Fig. 9 Algorithm performance table**

# **Algorithm performance:**



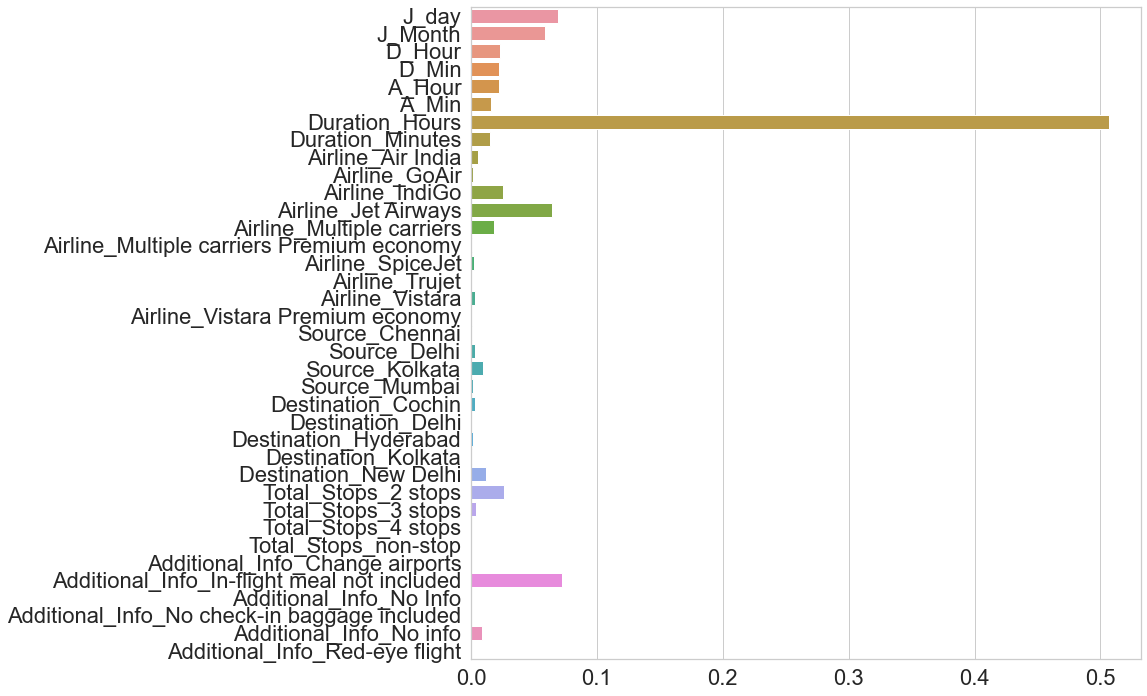
# **Fig. 10 Algorithm performance**

# **Based on the cross-validation score Random Forest Regression has the best performance.**

**Hyperparametric Tuning for Random Forest Regression Classifier:**

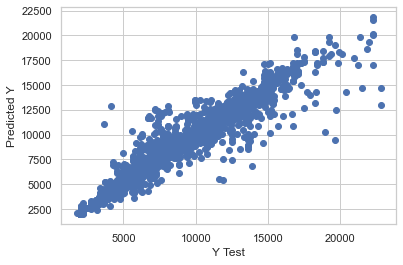
* Grid Search CV from Sklearn was used for hyperparametric tuning.
* Best Estimators (n\_estimators=400)
* Best Parameters ('criterion': 'mse', 'max\_features': 'auto', 'min\_samples\_leaf': 1, 'n\_estimators': 400)
* Best Score: 0.9096 %
* Training Score: 0.98 %
* R2 Score: 0.92 %
* Adjusted R2 Score: 91 %
* Explained Variance: 92 %

**Concluding Remarks:**



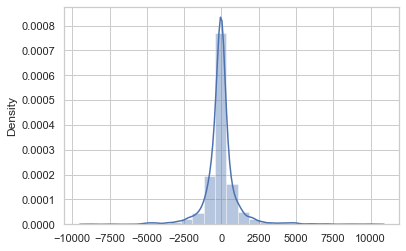
**Fig. 11 Feature Importance**

The first step was to Analyse the data, understanding the patterns with the help of visualizations, and also learning the relationships among different variables in order to predict the ticket prices. We also had to perform a lot of feature engineering in order to extract Date, time information which is very important in such use cases. From the analysis, I conclude that the most useful feature in predicting “Flight Ticket Prices” is **Flight Duration** (in hours). The Random Forest Regressor proved to be the best model for this problem based on the Cross-Validation Training and R2 scores. An accuracy of 0.92 % was achieved by hyperparametric tuning of the model.



**Fig. 12 Actual Vs Predicted**

From **Fig. 12 we can observe that when we plot the predicted values with the actual values, we get a graph that looks somewhat linear in nature. which shows that the model is performing good**

**Plotting a histogram of the residuals to make sure it looks normally distributed.**