1. Problem Definition

Flight prices change a lot, and it may seem that the prices are unpredictable at times – however airlines use complex algorithms to maximize their profit. A week before the flight, during the day the lowest fare may be Rs.2500 and some hours later it may rise to Rs.4000. Our task here is to train a machine learning algorithm which helps us better understand/predict how the prices work.

Airlines have collected huge amounts of data for the past few decades. This huge amount of data helps them better understand the market i.e. people looking for flights and predict how much they are willing to pay to get on the flight. The airlines viewpoint is simple enough – maximize the profit. While the people buying the ticket may be looking for alternative sources of travel if the flight fare is too high. It is a game of balance between the two.

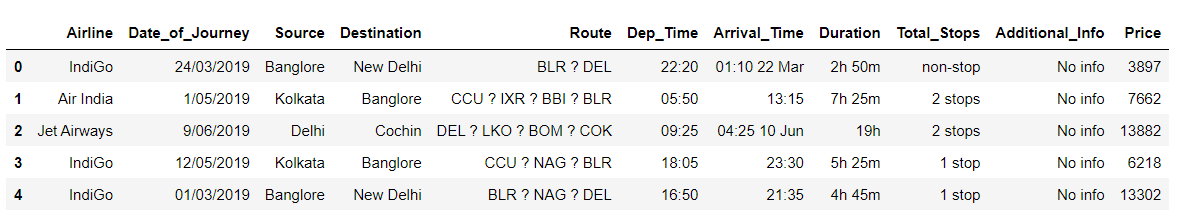


Regression is one of the most important and broadly used machine learning and statistics tools out there. It allows you to make predictions from data by learning the relationship between features of your data and some observed, continuous-valued response. Regression is used in a massive number of applications ranging from predicting stock prices to understanding gene regulatory networks.

2. Data Analysis

Let’s understand the data by looking at few of its elements.

First 5 columns of the data. Using df.head()



These are the columns which we have to work with.

[ 'Airline', 'Date\_of\_Journey', 'Source', 'Destination', 'Route', 'Dep\_Time', 'Arrival\_Time', 'Duration', 'Total\_Stops', 'Additional\_Info', 'Price' ]

Defining the various columns-

Airline: Contains all the various different airlines

Date\_of\_Journey: Date of the flight

Source: Where the flight originated

Destination: Final destination where flight landed

Route: Path of the flight through various cities and where it lands

Dep\_Time: Departure time of the flight

Arrival\_Time: Arrival time of the flight

Duration: Total length of the flight

Total\_Stops: Total number of stops the flight took

Additional\_Info: Additional information provided by the airlines

Price: The price at which the ticket is available

Price is our target variable which we have to predict. We have over ten thousand observations in this data set. The price of the flights depends on many more variables, but we are going to try to make a predicative model based on these specific variables.

Now we will probe some of the data to gain more insight into our data.

Checking how many variables each column has:

Airline 11

Source 5

Destination 6

Route 123

Additional\_Info 3

Price 1796

Day 14

Month 4

Departure 4

Arrival 4

Duration\_minutes 283

Total stops 5

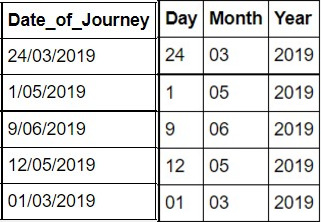
We have data from over 11 airlines, 5 source cities, 6 destination cities, 123 routes, a simplified additional information column with 3 variables. Over 4 months and 14 unique days of the month.

We can observe that the average domestic flight price is Rs. 9000, keeping in mind that this also includes premium tickets. The average flight lasts well over 7 hours and on average each flight takes one stop.



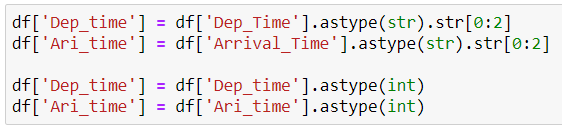
**Data pre-processing:**

Before we further analyze our data, there are some considerations to be made. Conversion of date of Journey into Day, month and year as separate variables.



In the data, we only have the data for year 2019 so it won’t be helpful in our analysis, so I will be dropping the year column. It is essential to break the date of journey into more variables to have more control over our data and get a better understanding of it.

Below, slicing the hour element to only consider the first two elements of departure and arrival time, to get the hour and converting it into an integer for further processing.



Now I have created two new columns, which use the same model which various airlines use to categorize the flights based on the time of the flight into 4 categories.

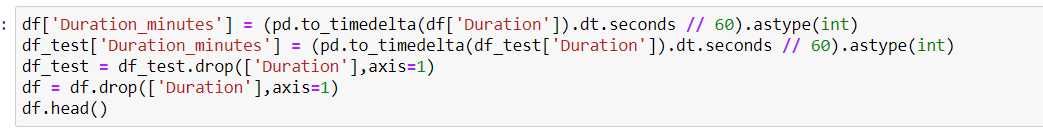
* Early Morning (00:00 – 06:00 hours)
* Late Morning (06:01 – 12:00 hours)
* Early Evening (12:01 – 18:00 hours)
* Late Evening (18:01 – 00:00 hours)

df['Departure'] = df.Dep\_time.apply(lambda date: "Early Morning" if 0 <= date <= 6 else "Late Morning" if 7 <= date <= 12 else "Early Evening" if 13 <= date <= 18 else "Late Evening")

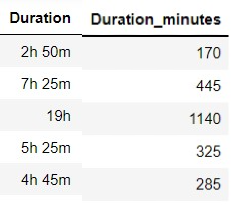
df['Arrival'] = df.Ari\_time.apply(lambda date: "Early Morning" if 0 <= date <= 6 else "Late Morning" if 7 <= date <= 12 else "Early Evening" if 13 <= date <= 18 else "Late Evening")

This will give us better insight into the data.

Now, converting the duration from hours and minutes into purely minutes and converting it into an integer value so that we can use it for our EDA.



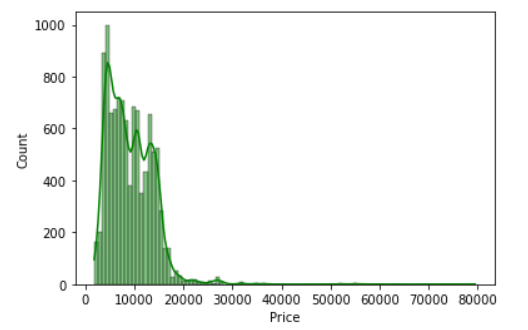
Below, is the converted flight duration. I have converted it to minutes so that we are able to use it as a integer value instead of a categorical value.

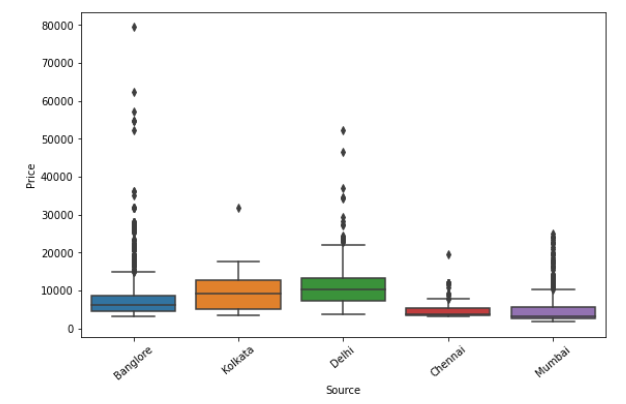


Looking at the summary of the numerical variables which we have.

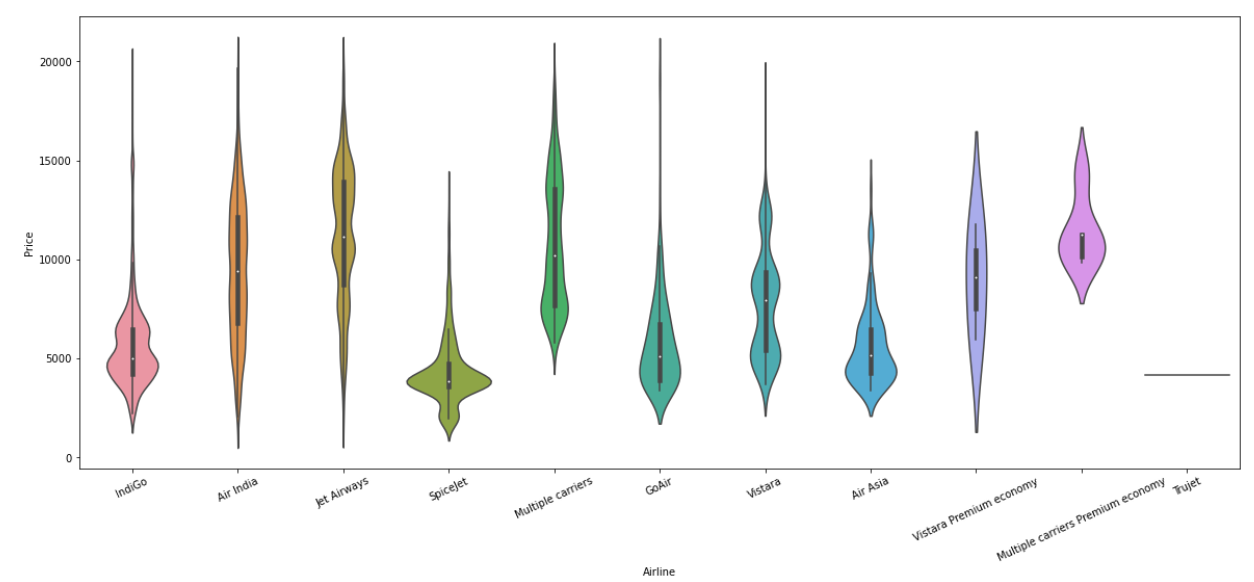


This is the price distribution for all the flights, we can observe here that most of the flights range from Rs. 2000 to Rs. 20000. Which agrees with our previous findings. Flight prices above Rs.20000 exists but are extremely few in number compared to the bulk of our data.



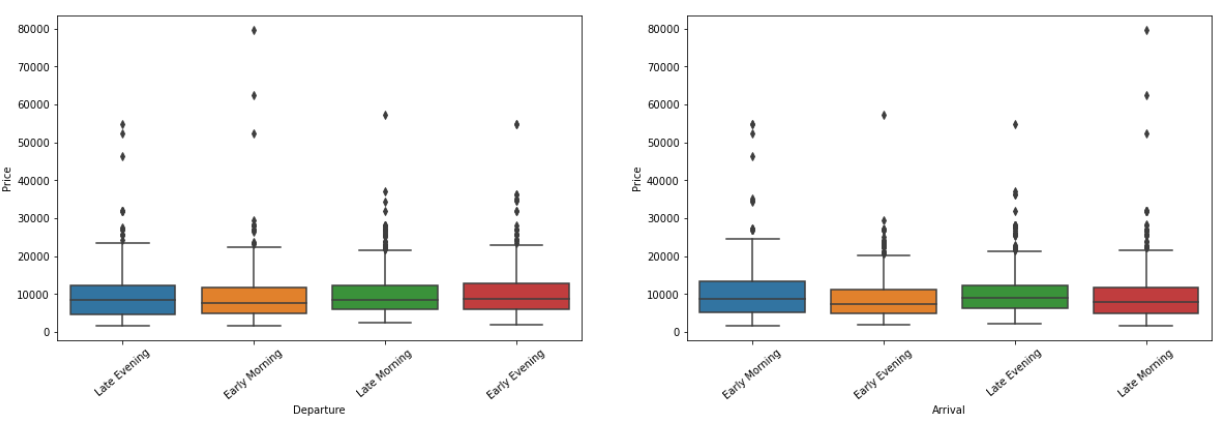
Looking at the source vs flight price, we can observe that flights originating in Delhi are on average the most expensive and that’s flights from Chennai are the cheapest.

Comparing various airline prices using a violin plot, we can observer that the price scheme between various airline varies as they all use their own proprietary algorithms which differ from each other.

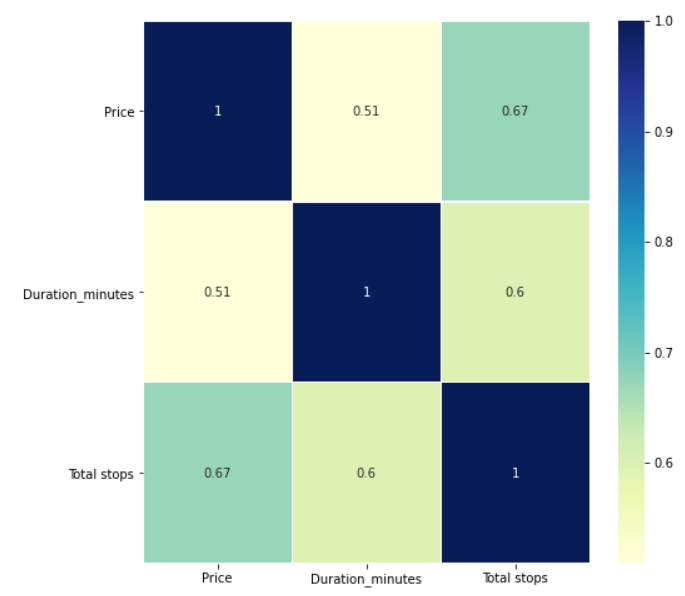


Looking at the co-relation between Price, duration of flight and total stops we can observer that both duration of flight and stops are in a positive co-relation with price.

We can observe that, in the grand scheme of thing the time of the flight (Arrival / Departure time) doesn’t affect the flight prices very much. It may do so in extreme cases, such as a really late flight is usually less expensive than the same flight in the evening. Here still we can see that early morning flights are the cheapest and early evening flights (Departure time) are the most expensive.

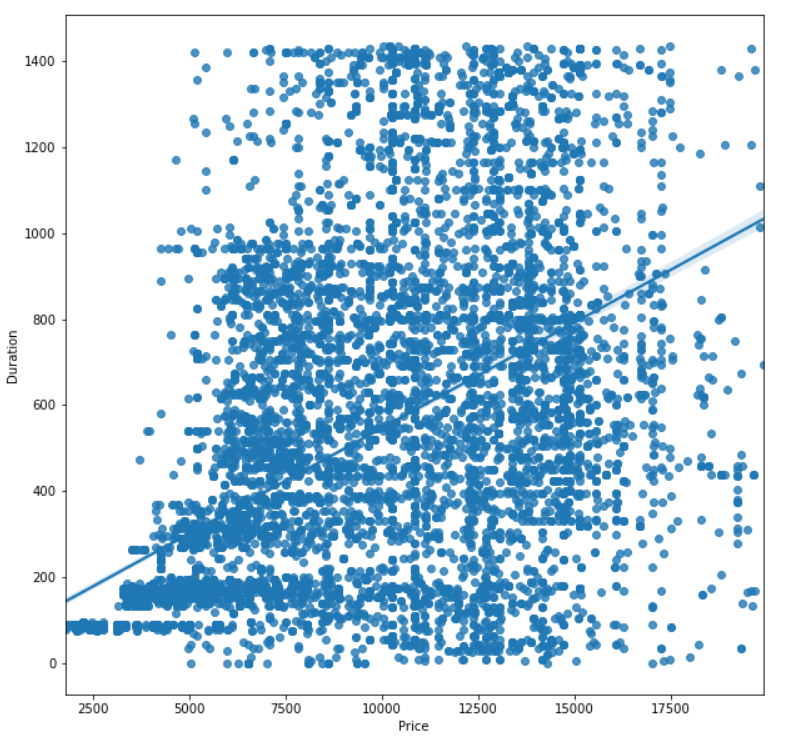


Making a co-relation heat map:

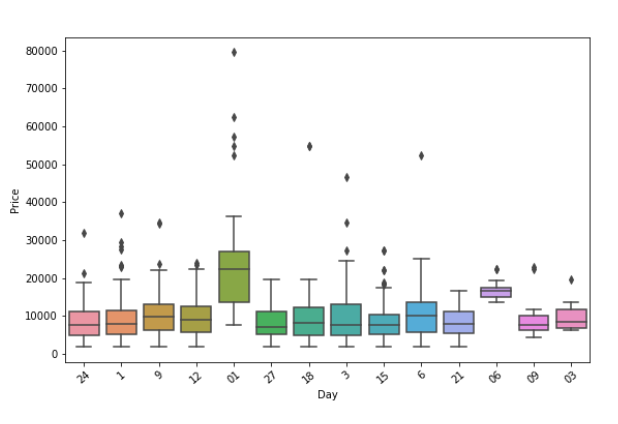


Here we can see the regression plot between Price and duration of the flight, and we can observe a positive co-relation between the two.

In the regression plot below, between flight price and flight duration in minutes, we can observe a linear trend – as the flight duration increases, the flight price increases as well. This makes sense as flights with a longer duration usually cover longer distances, which is one of the main variables in determining the flight price.



Based on the day of flight, we can observe that the flight on 1st day of the month is usually more expensive than any other day of the month.



3. EDA Concluding Remarks

Based on our analysis we can conclude that flight prices can be extremely unpredictable. Based on certain variables. With more data available we can make more accurate predictions.

Price is directly related to the duration in minutes, and the number of stops of the flight.

We can also observe that, various airlines have different price philosophies which correspond to the various different configurations of planes. i.e. the same model of a plane say the airbus A320 can have 150 or 180 seats depending on the different classes which the airline provides.

Different airlines also provide different perks, such as in-flight meals, leg room etc etc. all of this (and many more variables) have to be taken into consideration when calculating of prices.

4. Pre-processing pipeline

Checking how many flights have prices more than Rs.20000



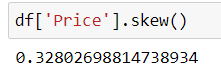


We can observer that only 148 flight prices are available with price more than Rs.20000

Based on the dataset we have available, out of more than 10000 data points, only 148 flight prices are available with price more than Rs.20000, these 148 said flights highly skew our data so in the spirit of removing outliers I will remove flights with prices more than Rs.20000.



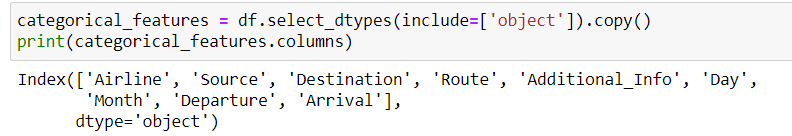
Removing the rows with prices more than Rs.20000



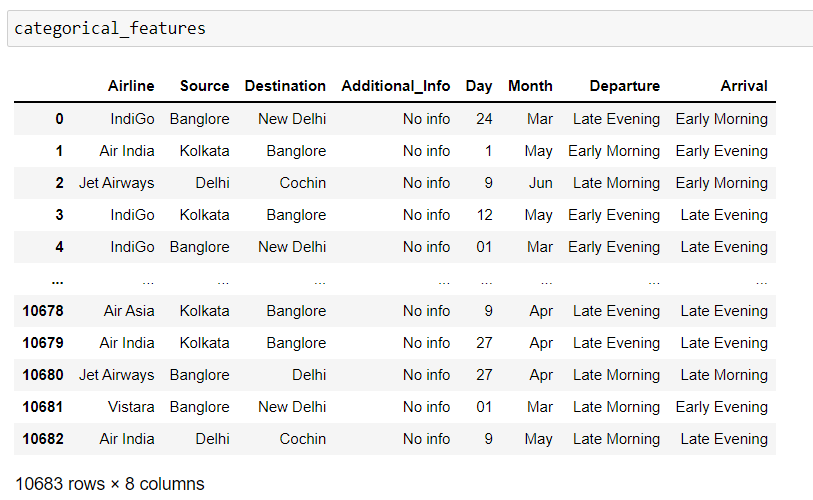
The skew of our target variable, i.e. price is very important as we want the lowest skew possible. Here we can observer that we have a skew of 0.32 which is fine for our analysis.

On side note: I have removed various columns for eg. Route, Duration, Dat\_of\_journey and year to simplify the analysis and remove depreciated columns which have been created in other coulmns.

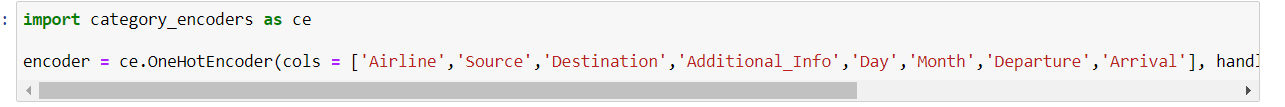
Taking a look at the categorical features which we have.



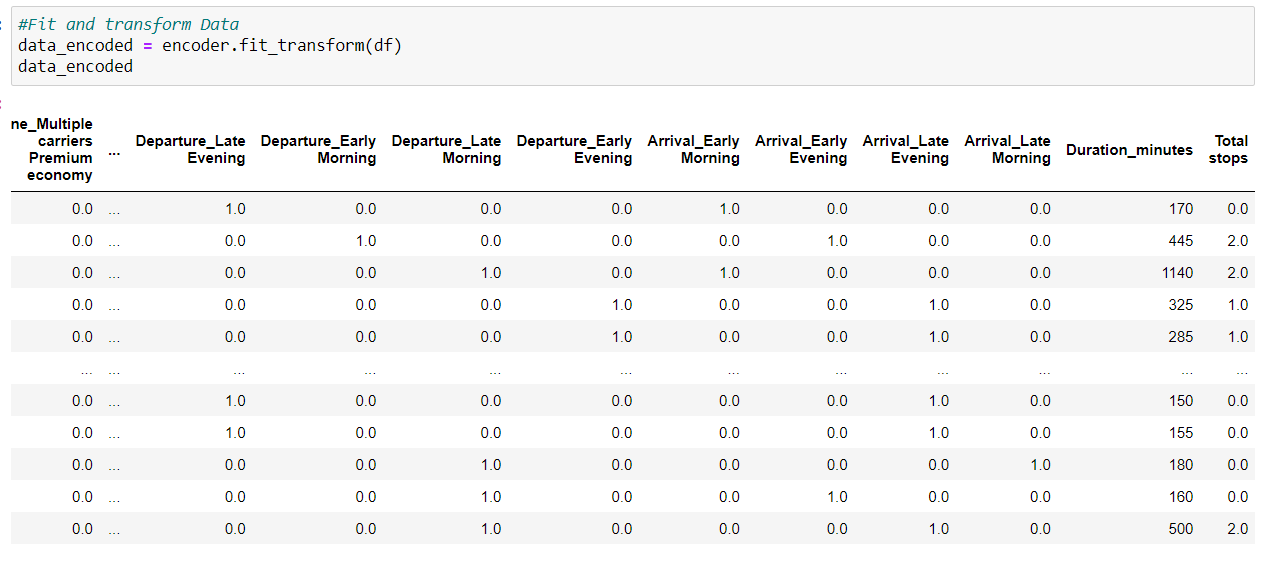
We will be encoding these categorical features:



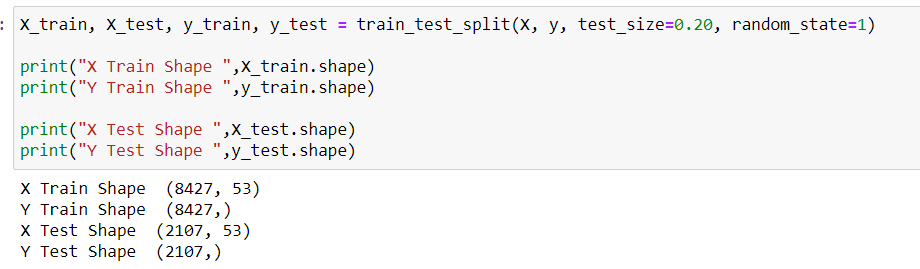
I will be using the OneHotEncoder to convert these data points. It is necessary to convert categorical variables to numerical variables as the machine learning algorithms we apply require all of the attributes to be numerical.



One Hot Encoding will convert each categorical value into a new categorical column and assign a binary value of 1 or 0 to those columns. As we can observe below:



Now, I will be splitting the data into two parts. Training data includes 80% of the total data and after training I will be testing the accuracy of the various models on the remaining 20% of the data.



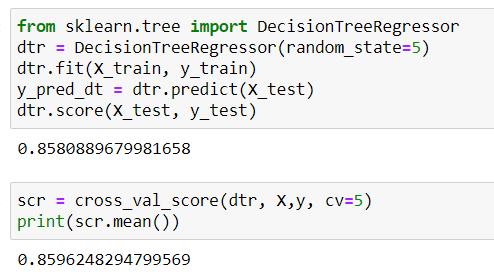
5. Building Machine Learning Models

In this analysis, I will be using 5 different machine learning techniques. Let’s look at the scores one by one. I will also be running a 5 fold cross validation to remove any bias from the training data and taking mean of all the five additional runs.

Regression is a supervised machine learning technique which **is used to predict continuous values**. The ultimate goal of the regression algorithm is to plot a best-fit line or a curve between the data.

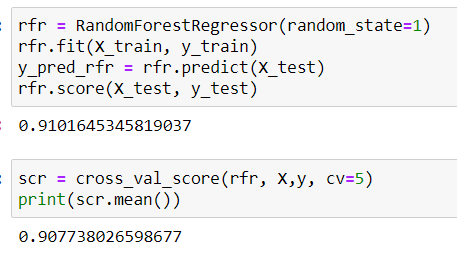
* DecisionTreeRegressor

Using the Decision tree regression algorithm a R-squared value of 0.859 was achieved.



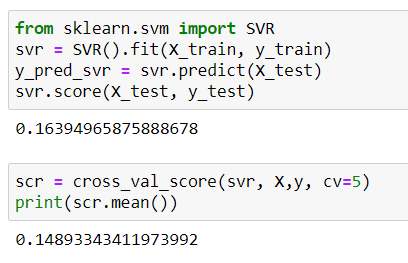
* RandomForestRegressor

Using the random forest regression algorithm a R-squared value of 0.907 was achieved.



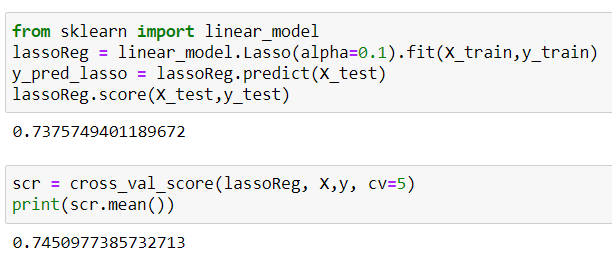
* SVR (SupportVectorRegressor)

Using the support vector regression algorithm a R-squared value of 0.148 was achieved.



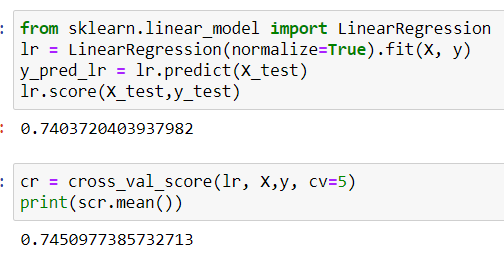
* LassoRegression

Using the lasso regression algorithm a R-squared value of 0.745 was achieved.



* LinearRegression

Using the linear regression algorithm a R-squared value of 0.745 was achieved.



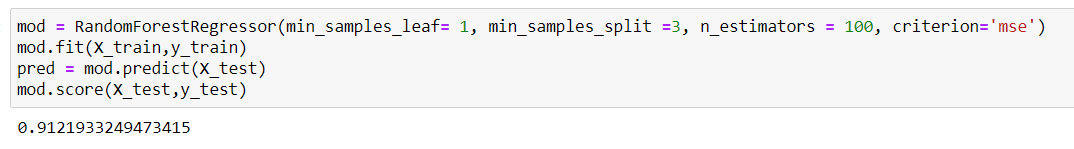
Random forest regression is the best algorithm in this case.

Based on the testing, we can conclude that Random forest regression is the best algorithm in our case. Let’s dive deeper into the algorithm and tune the hyperparameters.

After tuning the hyper-paramters, the best parameters were found to be:

min\_samples\_leaf= 1, min\_samples\_split =3, n\_estimators = 100, criterion='mse'

New training based on these hyper paramters:



Mean absolute error = 668.01

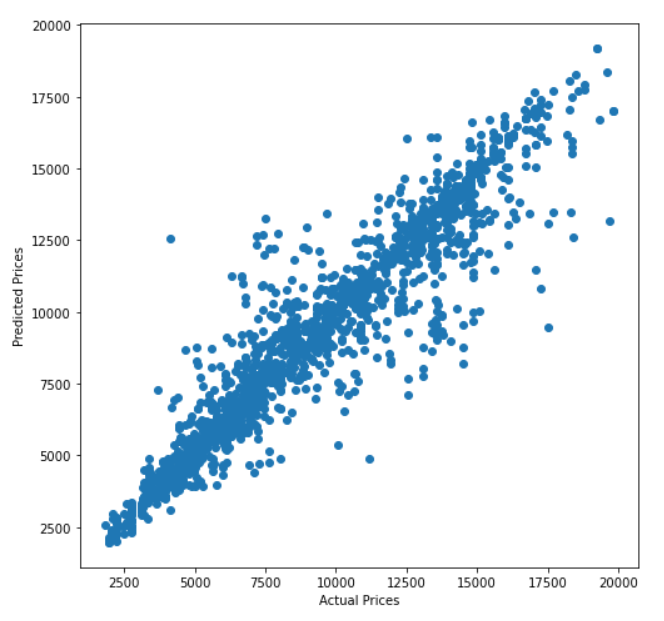
Mean squared error = 1432748.94

Median absolute error = 293.91

Explain variance score = 0.91

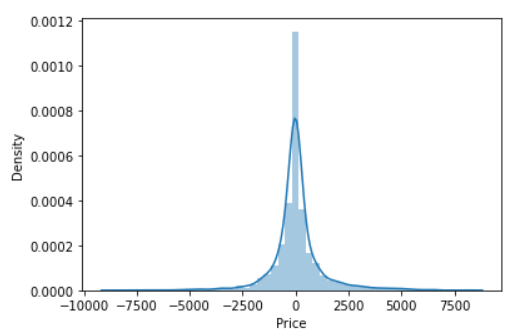
R2 score = 0.9121933249473415

We can observer that we have achieved a R-squared value of 0.912 which is great considering the un predictability of flight prices.



Here we can observer the accuracy of our final model in this scatter plot.

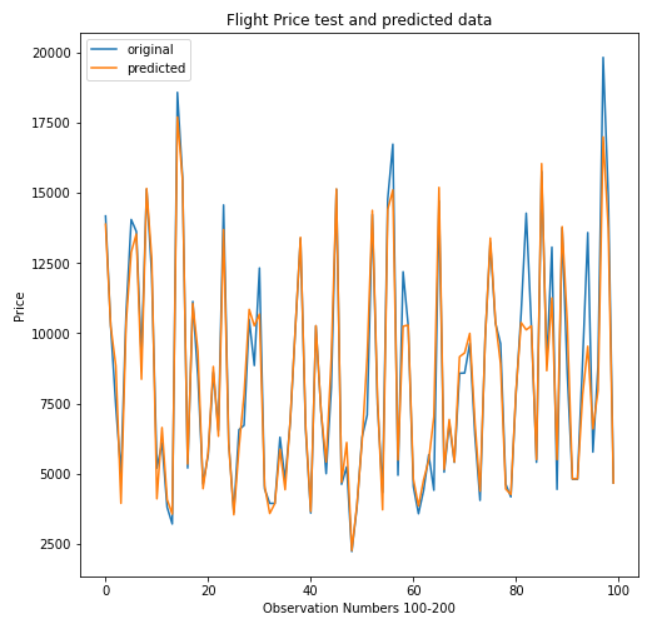
On the x-axis we have the actual flight prices from our data set, and on the y-axis, we have the predicted values of the same. We can observer that our algorithm is determining the prices very accurately.

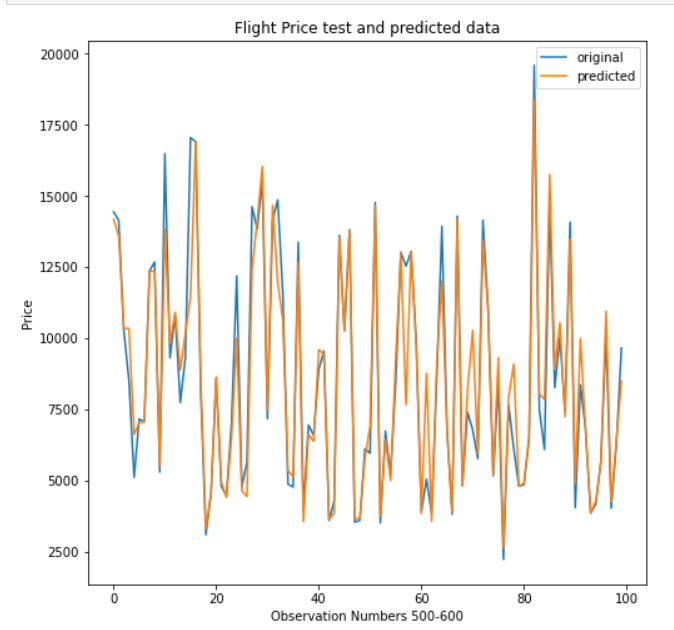


Distribution plot of the difference between actual price and predicted price. We can observe that most of the prediction are extremely accurate with almost a 100% match to the actual price.

Visualizing the results further:

Using a simple code for plotting I will be showcasing two sets of prediction which this algorithm has made. In these graphs, the blue line indicated the actual price and the orange line represents the predicted price. We can see they overlap for most of the values.





Our algorithm is predicting the flight prices very accurately.

6. Concluding Remarks

Using the Random Forest algorithm, I have achieved a R-squared value of 0.91

Flight prices are unpredictable at first glance, but if we have enough data to train our machine learning models we can determine the flight prices very accurately.

This dataset only includes flights from a handful of cities in India, for a handful of days and only 3 months of one specific year. When airlines are building a machine learning model they do the same but, they have access to much more data and many more variables points and more information on the people. A big portion of an airline’s budget is spent on determining flight prices so that they flights are running at 100% capacity.

With the limited data I have created a fairly accurate flight prediction model. With better hardware and more data this model can be improved a lot.

Furthermore, a 100% fitting model cannot be made because there are too many variables which change every day, every hour, every minute and sometimes within seconds.