

**Flight Price Prediction**

Submitted by:

Charitha Lanka

# Introduction

## Business Problem Framing

Machine Learning is a field of technology developing with immense abilities and applications in automating tasks, where neither human intervention is needed nor explicit programming.

The power of ML is such great that we can see its applications trending almost everywhere in our day-to-day lives. ML has solved many problems that existed earlier and have made businesses in the world progress to a great extent.

Today, we’ll go through one such practical problem and build a solution(model) on our own using ML.

We are about to deploy an ML model for Flight price prediction and analysis. This kind of system becomes handy for many people.

So, to be clear, this model will provide you will the approximate fare for your flight based on the flight name, timing of the flight, detinaiton and soruce, the number of stops,

**Abstract:**

Due to Economic pricing & people wanting to travel in short period of time, Airline industry has seen a rise in ticket booking during this decade. Proper scheduling & good knowledge of airfares can help economic passengers have good value for money on their travel. The cheapest available ticket on a given flight gets more and less expensive over time. The airfares dramatically vary depending on time & demand of purchase of tickets. We have to work on a project where you collect data of flight fares with other features and work to make a model to predict fares of flights

## Conceptual Background of the Domain Problem

The goal of this statistical analysis is to help us understand the relationship between flight features and how these variables are used to predict flight fare.

## Review of Literature

From the dataset I get to know that it is a Regression problem. And there are many features which help to find it.

## Motivation for the Problem Undertaken

I am doing this for practice, to get more hands-on data exploration, Feature extraction and Model building.

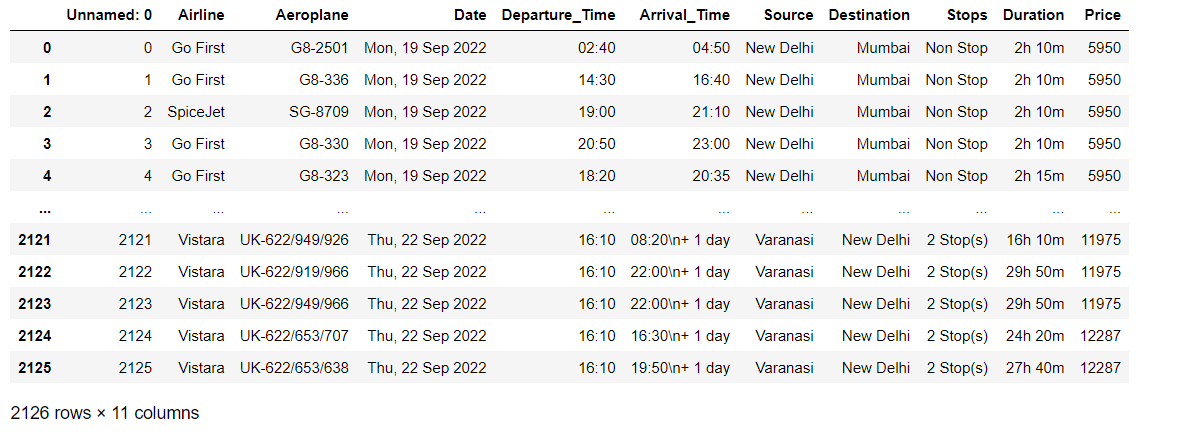
# Analytical Problem Framing

## Mathematical/ Analytical Modelling of the Problem

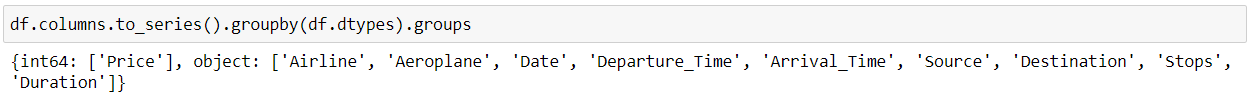
First phase of problem modelling involves data scraping of flights from internet. For that purpose, flight data is scrap from www.yatra.com for timeframe of 19 sep 2022 to 30 sep 2022. Data is scrape for flights on route of New Delhi to Mumbai and Varanasi to New Delhi. Data is scrap for Economy class, Premium Economy class & Business class flights. Next phase is data cleaning & pre-processing for building ML Model. Our objective is to predict flight prices which can be resolve by use of regression-based algorithm. Further Hyperparameter tuning performed to build more accurate model out of best model.

## Data Sources and their formats

Data is collected from www.yatra.com for timeframe of 19 Sep 2022 to 30 Sep 2022 using selenium and saved in excel file. Data is scrape for flights on route of New Delhi to Mumbai. Data is scrap for around 2126 flights details is collected for this project.



Unnecessary column of index name as ‘Unnamed: 0’ is drop out. There are 10 features in dataset including target feature ‘Price’. The data types of different features are as shown below:



## Data Pre-processing Done

The dataset is large and it may contain some data error. In order to reach clean, error free data some data cleaning & data pre-processing performed data.

• Data Integrity check –

No missing values or duplicate entries present in dataset.

• Conversion of Duration column from hr & Minutes format into Minutes –

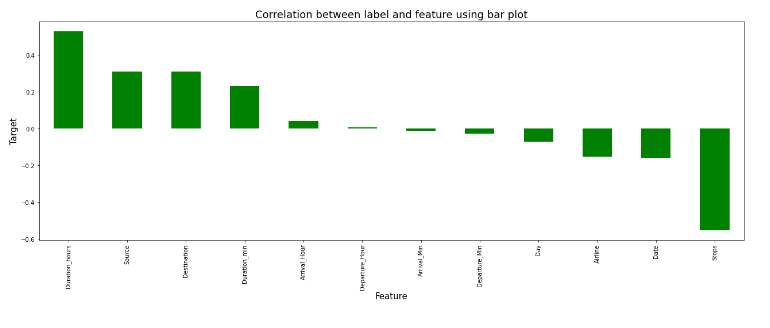
By default, Duration of flights are given in format of [(hh) hours: (mm)minute] which need to convert into uniform unit of time. Here we have written code to convert duration in terms of minute. For example,





## Data Inputs- Logic- Output Relationships

I have found out that with continuous numerical variable there is a Relationship with the flight fare. And for categorical variable, I have used Boxplot for each categorical feature that shows the relation with the median sale price for all the sub categories in each categorical variable. For continuous numerical variables I have used scatter plot to show the relationship between continuous numerical variable and target variable.



## Hardware and Software Requirements and Tools Used

The system requirements are description of features and functionalities of the target system. Requirements convey the expectations of users from the software product. The requirements can be obvious or hidden, known or unknown, expected or unexpected from client’s point of view. System requirements are all of the requirements at the system level that describe the functions which the system as a whole should fulfil to satisfy the stakeholder needs and requirements, and is expressed in an appropriate combination of textual statements, views, and non-functional requirements; the latter expressing the levels of safety, security, reliability, etc., that will be necessary.

**Hardware requirements**: -

1. Processor — core i5 and above

2. RAM — 8 GB or above

3. SSD — 250GB or above

**Software requirements**: -

Anaconda

**Libraries**: -

**From sklearn.preprocessing import StandardScaler**

As these columns are different in **scale**, they are **standardized** to have common **scale** while building machine learning model. This is useful when you want to compare data that correspond to different units.

**from sklearn.preprocessing import Label Encoder**

 Label Encoder and One Hot Encoder. These two encoders are parts of the SciKit Learn library in Python, and they are used to convert categorical data, or text data, into numbers, which our predictive models can better understand.

**from sklearn.model\_selection import train\_test\_split,cross\_val\_score**

Train\_test\_split is a function in Sklearn model selection for splitting data arrays into two subsets: for training data and for testing data. With this function, you don't need to divide the dataset manually. By default, Sklearn train\_test\_split will make random partitions for the two subsets.

The algorithm is trained and tested K times, each time a new set is used as testing set while remaining sets are used for training. Finally, the result of the K-Fold Cross-Validation is the average of the results obtained on each set.

**from sklearn.neighbors import KNeighborsRegressor**

K Nearest Regressor (KNN) is a very simple, easy to understand, versatile and one of the topmost machine learning algorithms. KNN used in the variety of applications such as finance, healthcare, political science, handwriting detection, image recognition and video recognition

**from sklearn.linear\_model import LinearRegression**

The library sklearn can be used to perform linear regression in a few lines as shown using the LinearRegression class. It also supports multiple features. It requires the input values to be in a specific format hence they have been reshaped before training using the fit method.

**from sklearn.tree import DecisionTreeRegressor**

Decision Tree is a white box type of ML algorithm. It shares internal decision-making logic, which is not available in the black box type of algorithms such as Neural Network. Its training time is faster compared to the neural network algorithm. The time complexity of decision trees is a function of the number of records and number of attributes in the given data. The decision tree is a distribution-free or non-parametric method, which does not depend upon probability distribution assumptions. Decision trees can handle high dimensional data with good accuracy

# Model/s Development and Evaluation

## Identification of possible problem-solving approaches (methods)

For feature transformation I have used Log normal transformation to make the continuous non zero variables close to normal distributed. Use of Annona test to check the importance of categorical features. Use of Pearson’s correlation coefficient to check the correlation between dependent and independent features. Use of Min Max scaler to scale down the features and one label encoding to encode categorical features in numeric.

## Testing of Identified Approaches (Algorithms)

Web Scraping Strategy employed in this project as follow:

1. Selenium will be used for web scraping data from www.yatra.com

2. Flights on route of New Delhi to Mumbai and Varanasi to New Delhi in duration of 19 Sep 2022 to 30 Sep 2022.

3. Selecting features to be scrap from website.

4. In next part web scraping code executed for above mention details.

5. Exporting final data in Excel file.

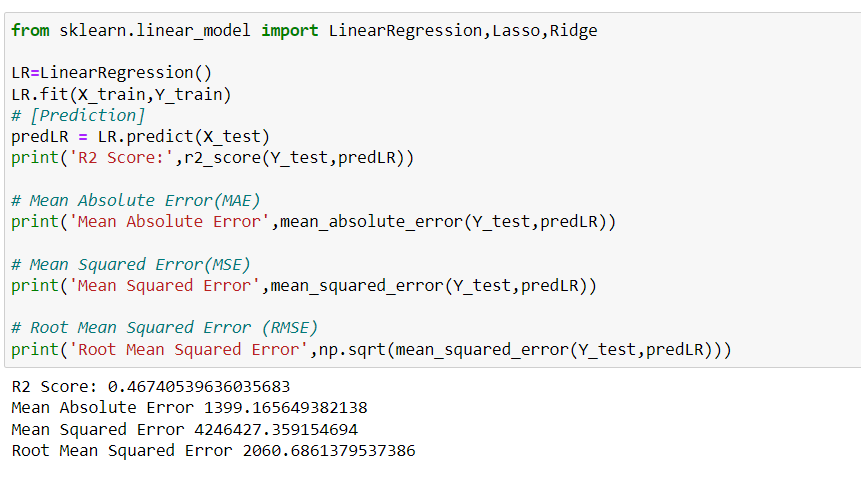
Listing down all the algorithms used for the training and testing.

* KNeighbors Regressor
* Linear Regression
* Linear Regression with Ridge
* Linear Regression with Lasso
* Gradient Boosting Regressor
* Bagging Regressor
* Extra Trees Regressor
* XGB Regressor
* Decision Tree Regressor
* Random Forest Regressor

I applied all these algorithms in the dataset.

## Run and evaluate selected models

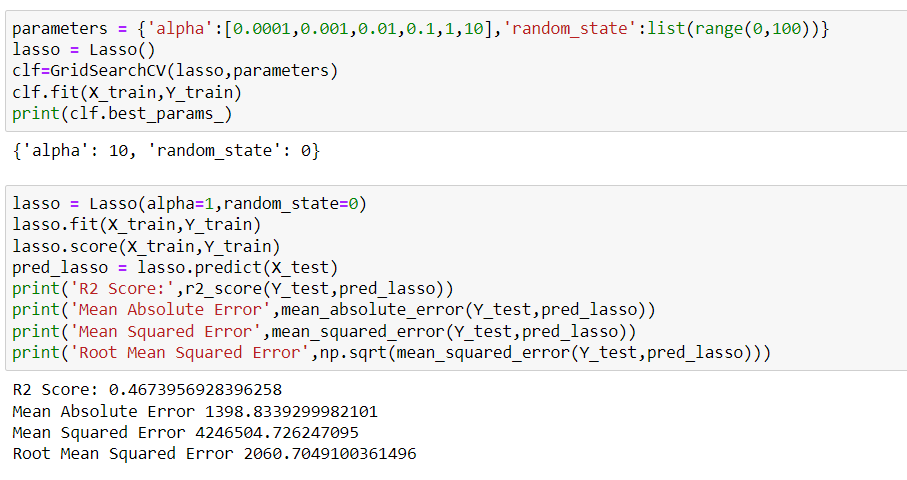
Linear Regression



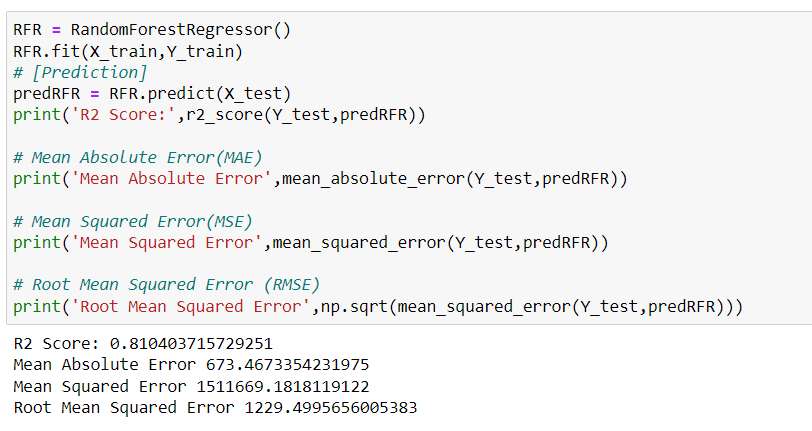
Linear Regression with Ridge



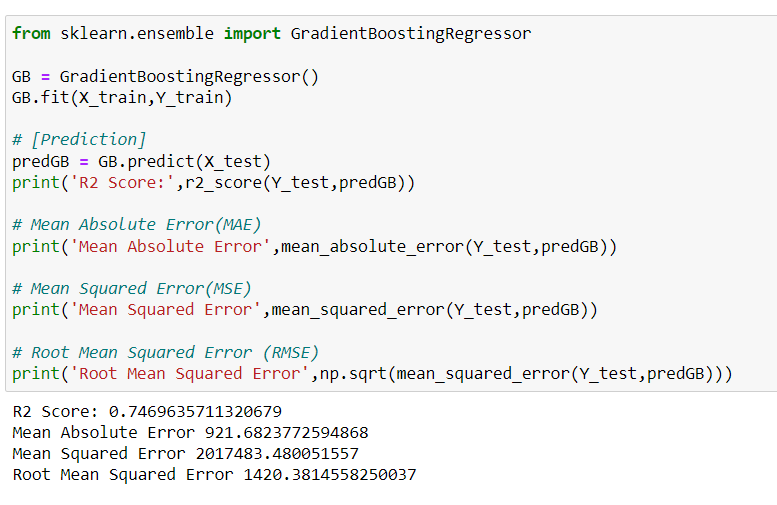
Linear Regression with Lasso



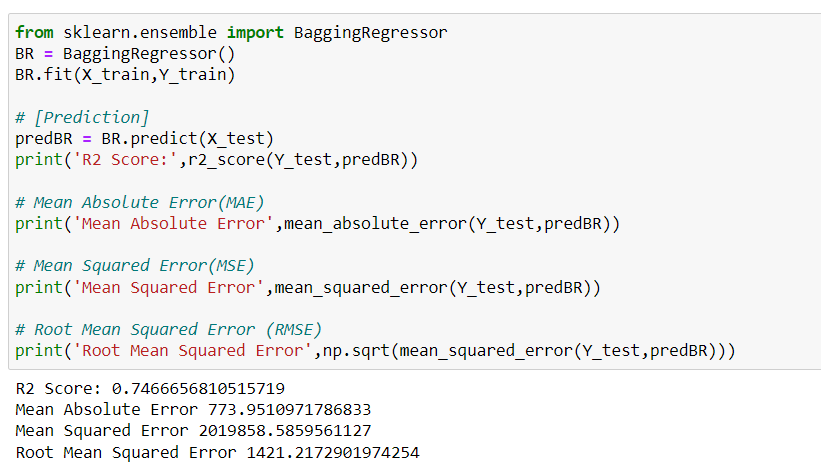
Random Forest Regressor



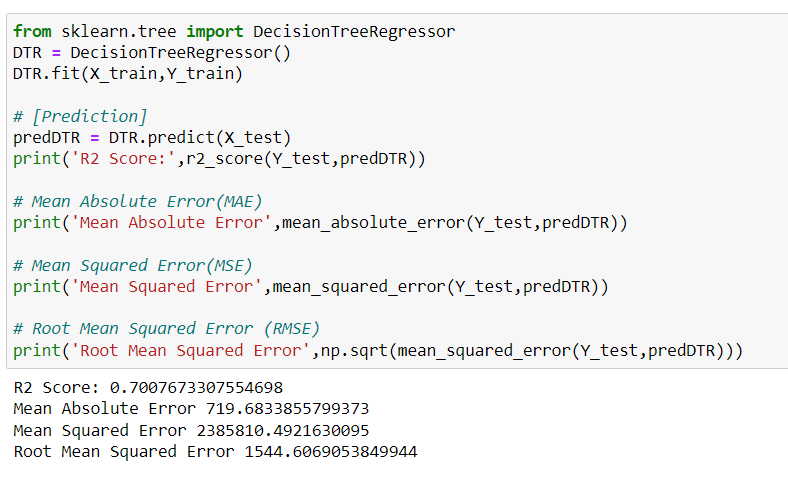
Gradient Boosting Regressor



Bagging Regressor



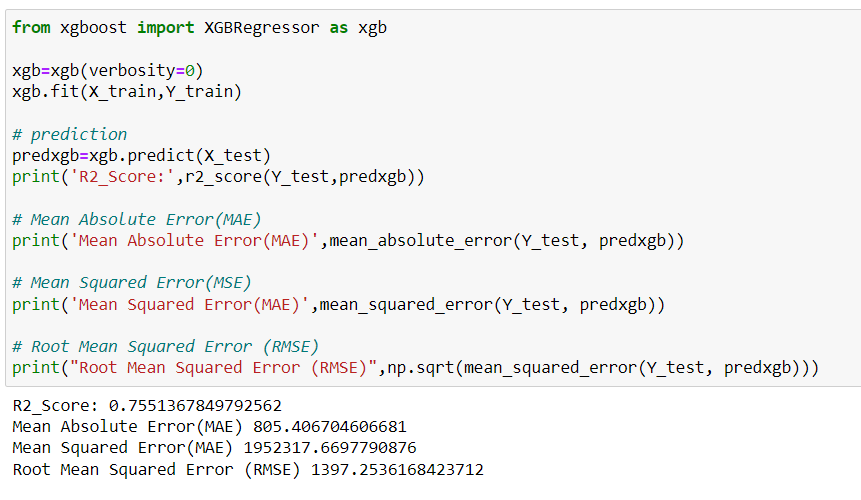
Decision Tree Regressor



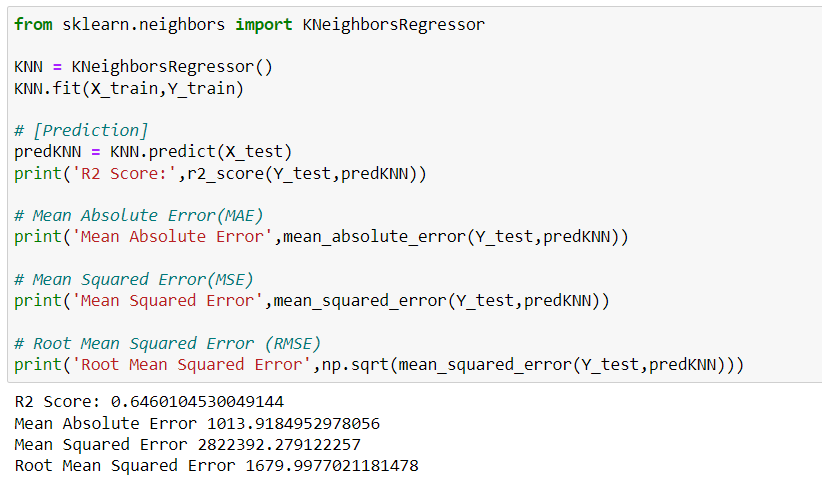
Extra Trees Regressor



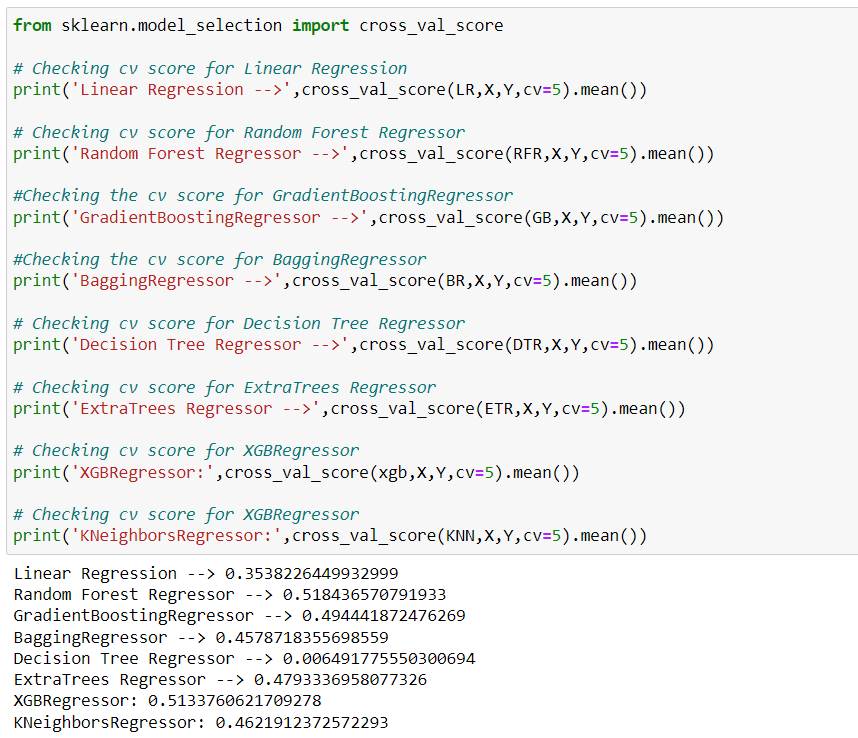
XGB Regressor

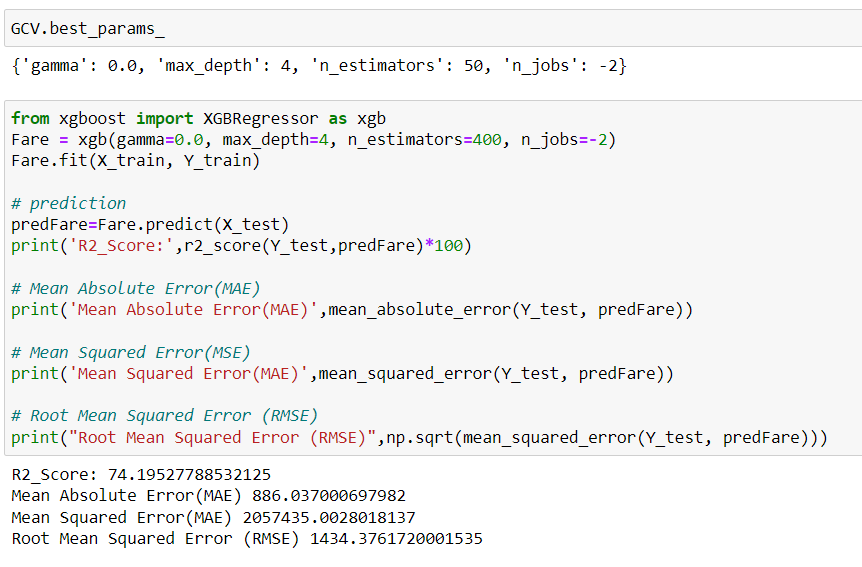
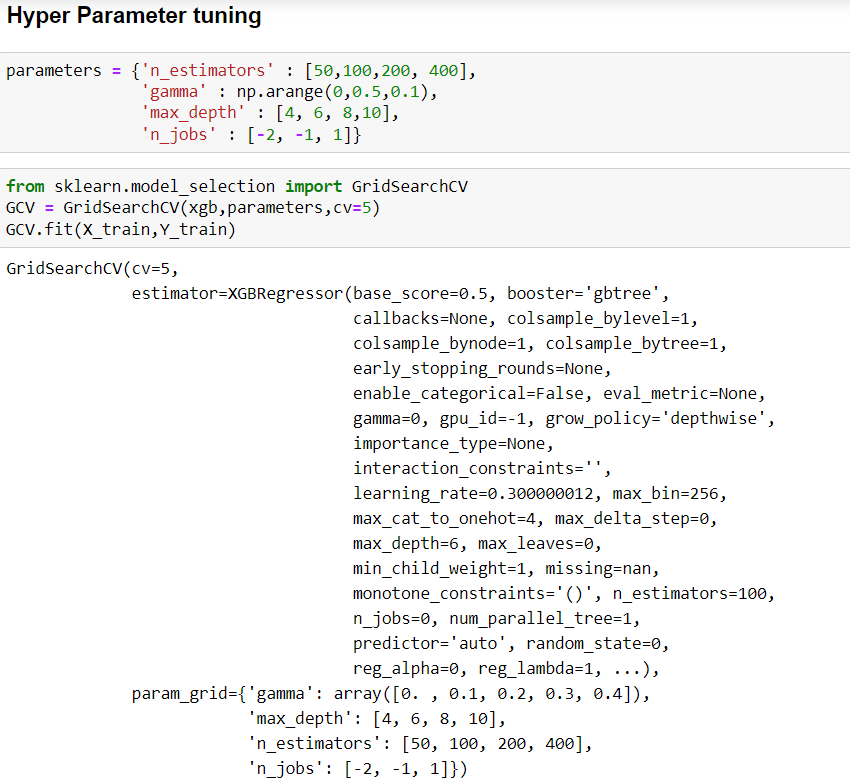


KNeighbors Regressor



Cross Validation





## Key Metrics for success in solving problem under consideration

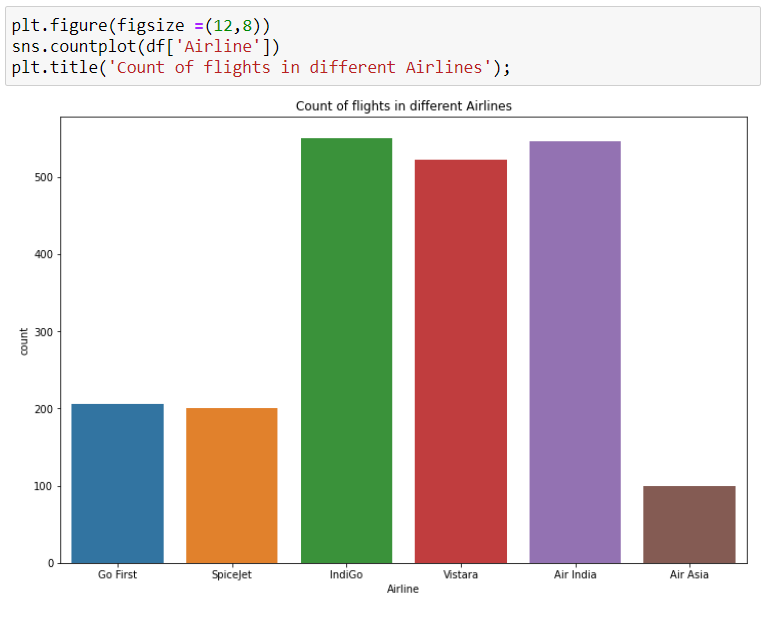
Following metrics used for evaluation:

1. Mean absolute error which gives magnitude of difference between the prediction of an observation and the true value of that observation.

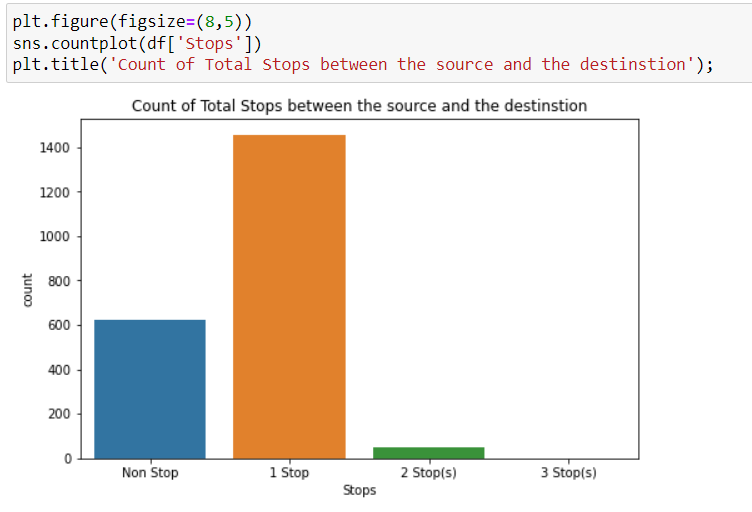
2. Root mean square error is one of the most commonly used measures for evaluating the quality of predictions.

3. R2 score which tells us how accurate our model predict result, is going to important evaluation criteria along with Cross validation score.

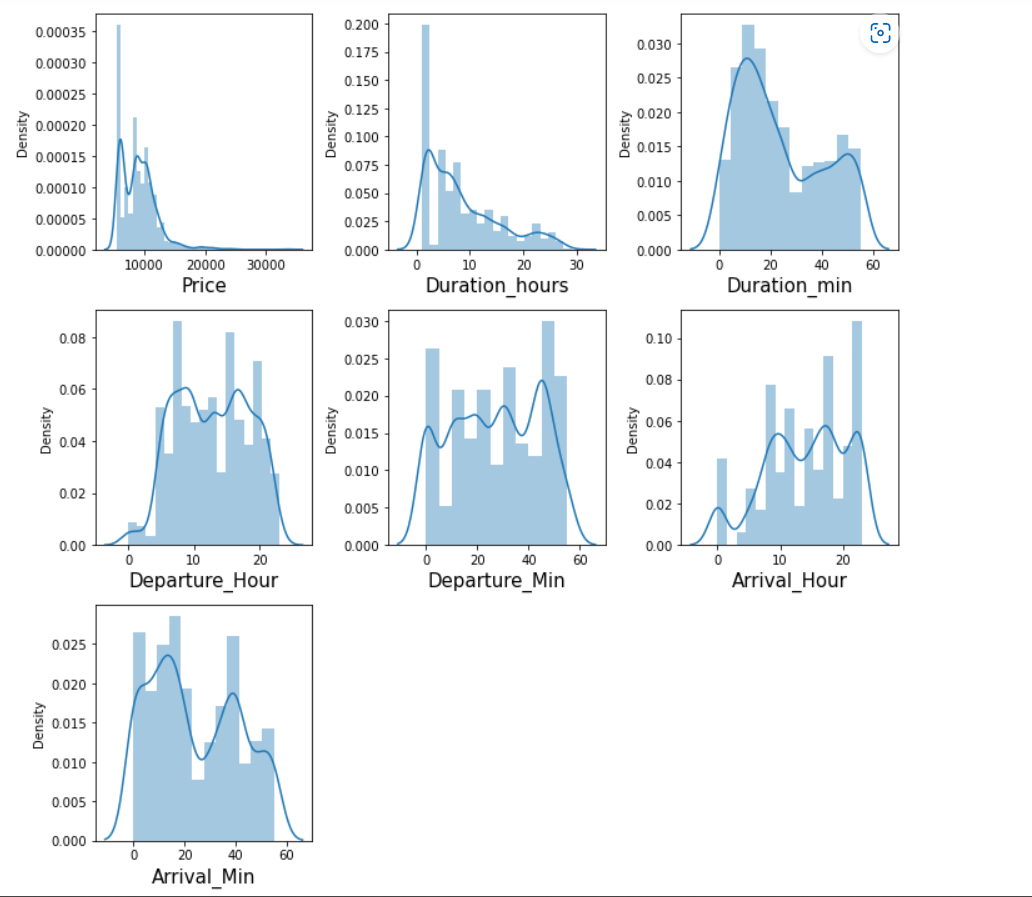
## Visualizations

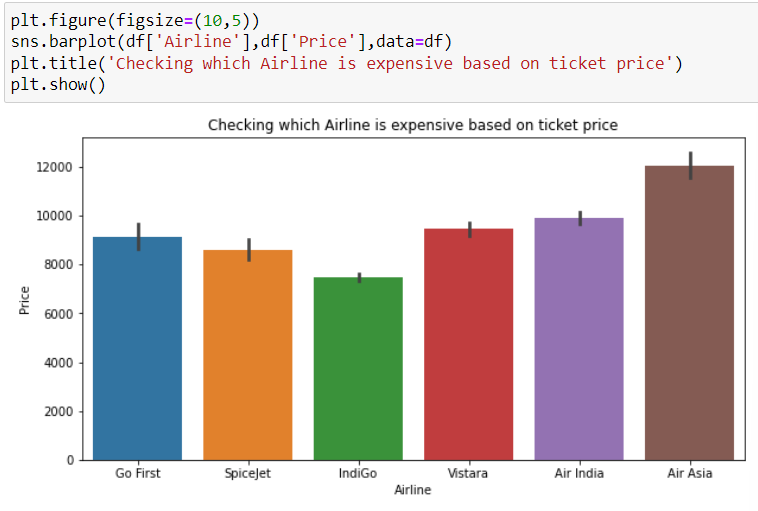


We can see from the graph that indiGo and Air India has the highest counts.



We can see from the graph that 1 Stop has the highest counts.

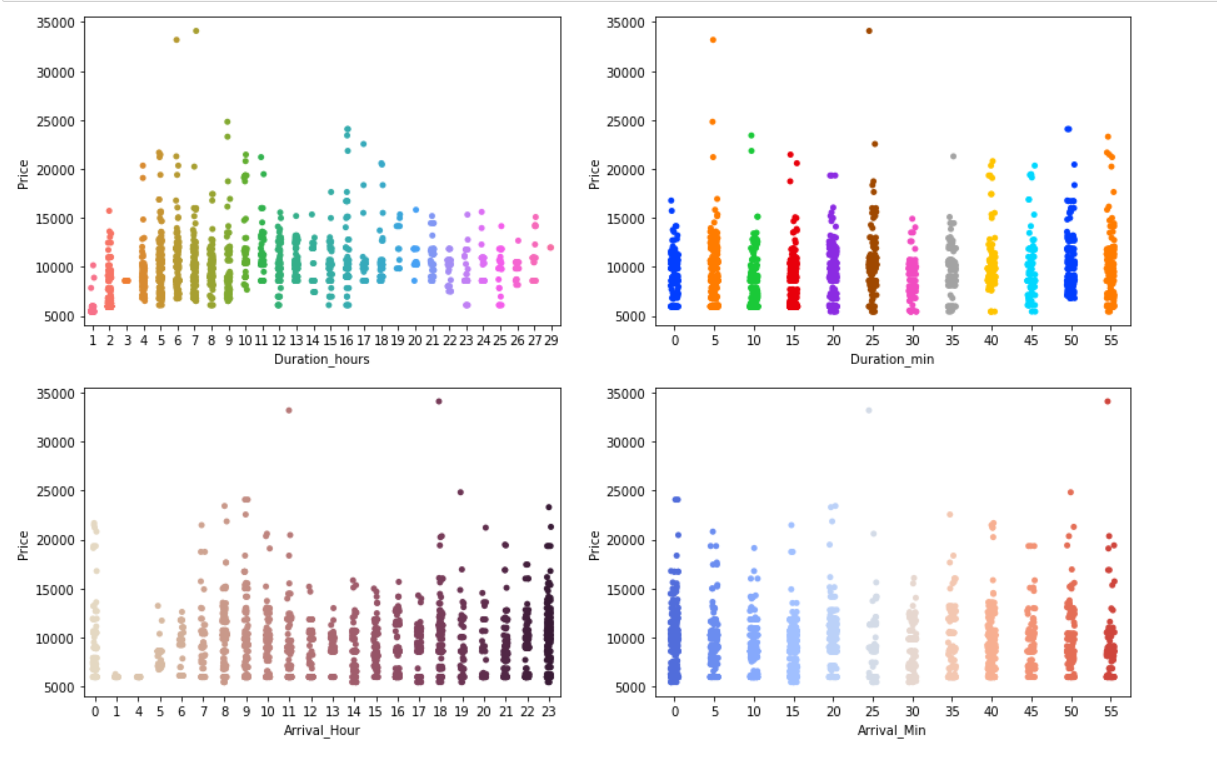


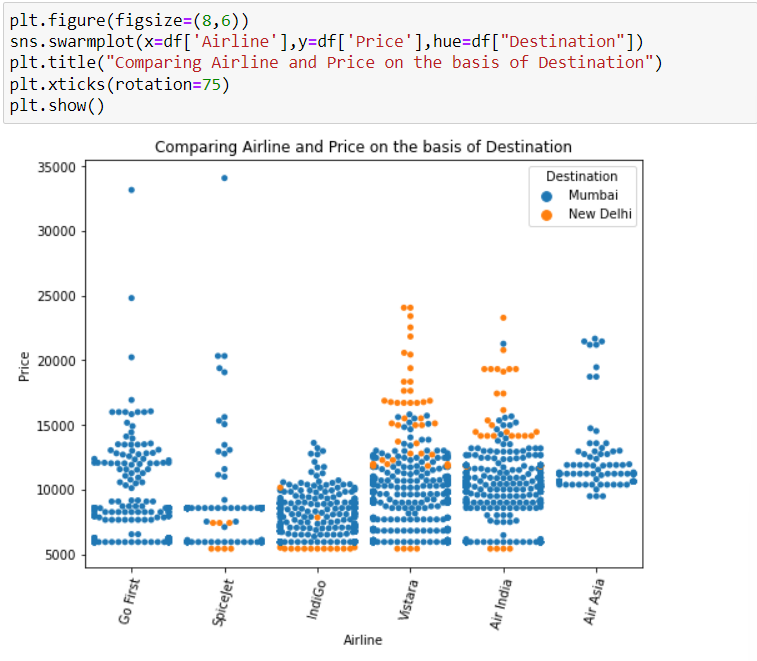


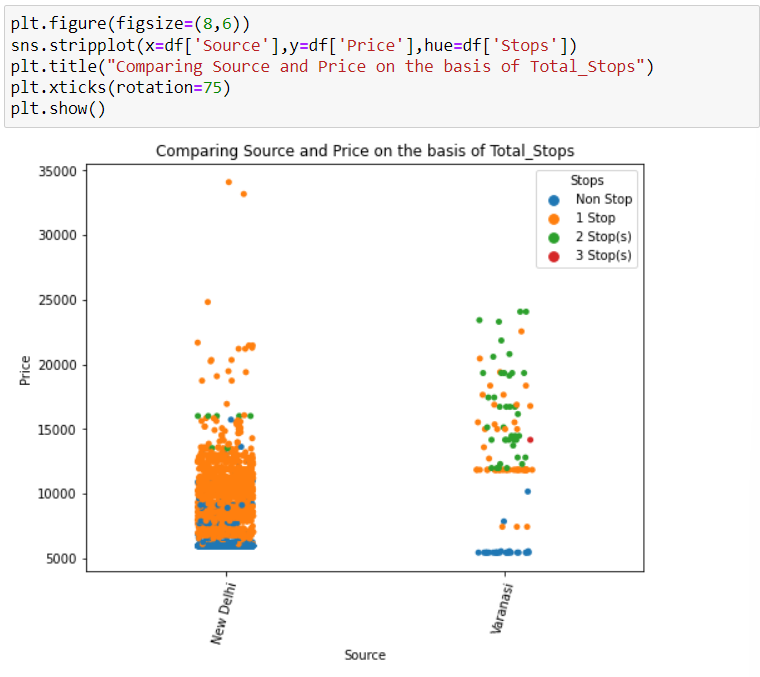
From the plot we can visualize that Air Asia is the most expensive and SpiceJet and IndiGo are the cheapest.

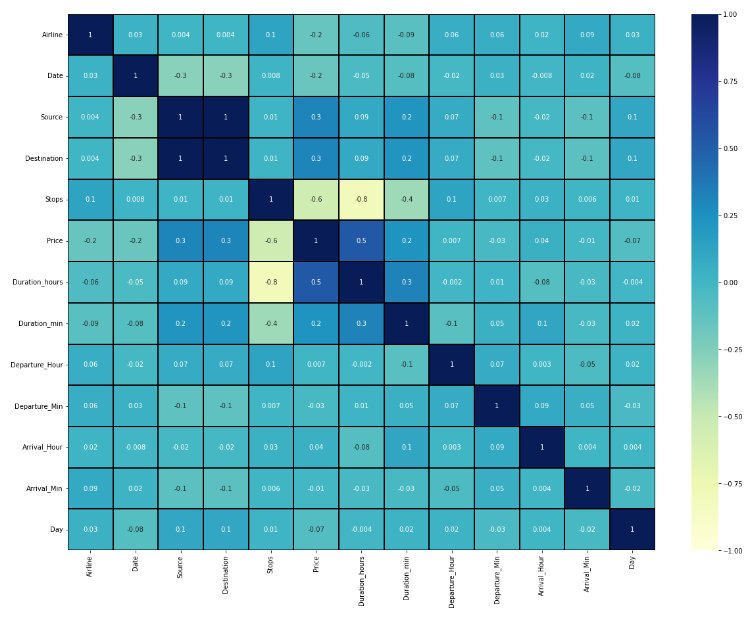


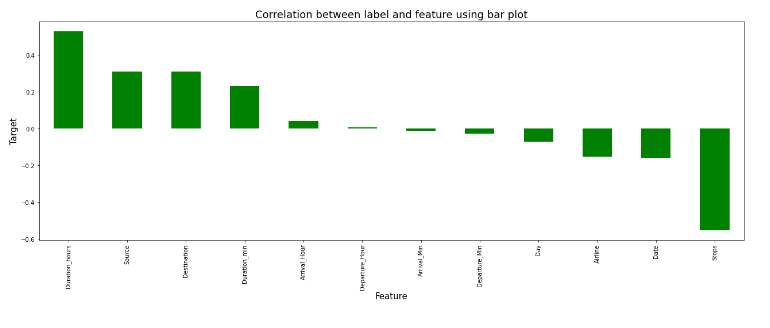
From the plot we can visualized that the flights with 2 Stops have highest price followed by flights having 2 stops and the flights which have no stops is having very less ticket price compared to others.

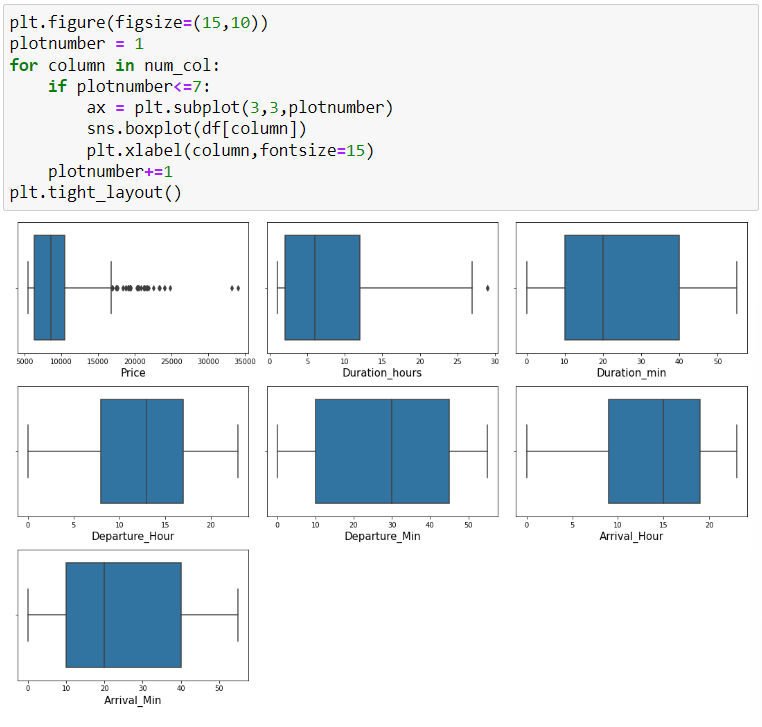




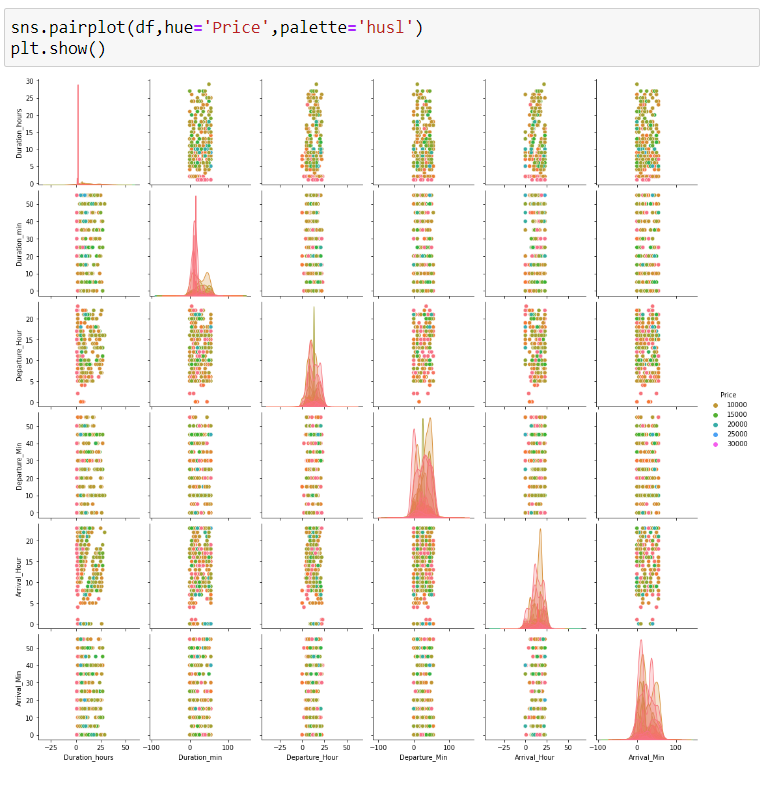








The outliers present in Price, Duration\_hours. Since Price is our target column so no need to remove outliers in these two columns.



## Interpretation of the Results

XGB Regressor R2 score was 75.51%, means 88% of the variance of the dependent variable being studied is explained by the variance of the independent variable.

Higher the R2 score means the model is well fit for the data. However, if R2 score is very high, it might be a case of overfitting. Other metrics Mean Absolute Error, Mean Squared Error and Root Mean Squared Error, with gradient boosting these scores are less then compared to other models. If these errors are less that means the model shows less errors.

# Conclusion

## Key Findings and Conclusions of the Study

From this dataset I get to know that each feature plays a very import role to understand the data. Data format plays a very important role in the visualization and Appling the models and algorithms.

## Limitation of this work and scope for Future Work

* In this study we focus on flights on route of New Delhi to Mumbai and Varanasi to New Delhi, more route can incorporate in this project to extend it beyond present investigation.
* This investigation focuses on short timeframe (11 days prior flights take off) which can be extended variation over larger period.
* Time series analysis can be performed over this model