**Flight Ticket Price Prediction**



Rasha Jamsheer

Data Trained Education, Noida

**Introduction**

Any individual who has booked a flight ticket previously know how dynamically cost change. Aircraft uses advanced strategies called revenue management to execute a distinctive valuing strategy. The least expensive accessible ticket changes over a period the cost of a ticket might be high or low. This valuing method naturally modifies the toll as per the time like morning, afternoon, evening, night or late night. cost may likewise change with the seasons like winter, summer, and celebration seasons. Therefor , it is challenging for consumers to predict the price change in the future.

Our project is aimed at building up models to predict the airline ticket price. The input of our models are the factors that many influence the price, such as the source, destination and number of stops etc. I applied linear regression, K Neighbours Regressor and Decision Tree Regressor to predict the corresponding price.

**Data set and 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.

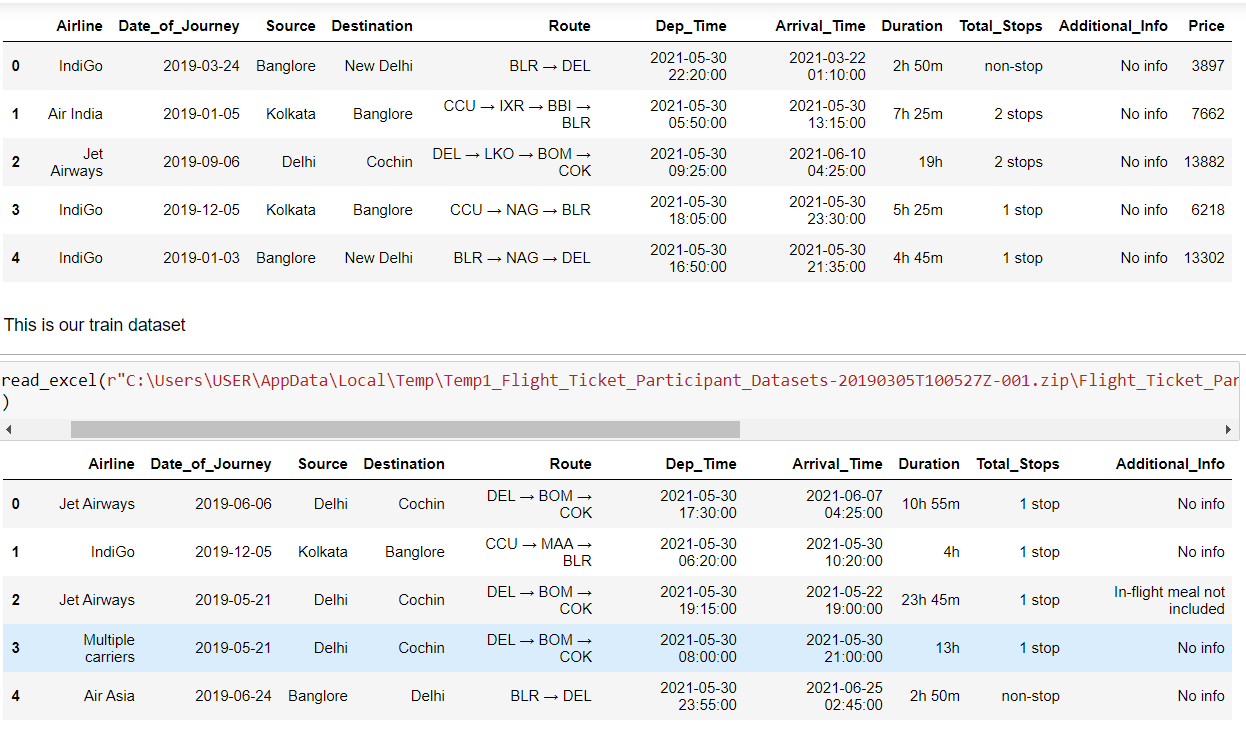
**Additional Info**: Additional information about the flight

**Price**: The price of the ticket

For this problem we have two data set which is train data and test data.

First and foremost I started with the usuals by importing necessary libraries like numpy, pandas, matplotlib and seaborn, then I moved on to reading the file (Train and Test) to enable work.

Train & Test Dataset:



I started by checking the information of the data and I discovered there were few missing values in the test dataset. so I was begin my work with filling the missing value after some analysis.

For better understanding of the model I created a new column, which represent convert the column Dep. Time (The time when the journey starts from the source) according to the period of the day like morning, evening, afternoon, night and late night using dt. hour. I applied it in both test and train data. After converted column named as ‘day time’. Now it is a categorical type column.

Extracting of columns I used value count of each columns. After extract its I notice that some categorical column has repeated attribute with same meaning. I replace it has a single attribute. When we check the Airlines and Destination column, we can see that name of the airline and name of destination were repeated (e g: New Delhi replace with Delhi and Jet Airways Business replace with Jet Airways)

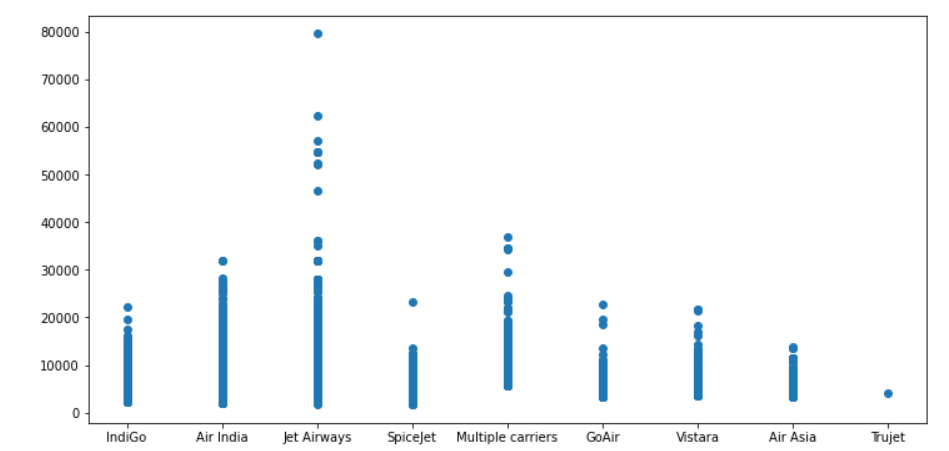
After I moved to prepare proper duration column. For that I extract arrival time and starting time (Dep Time). And create a list and append the value in that list. after finding the proper value of Duration, it change to original column. Then I moved to drop unnecessary columns from the dataset. we already have the information of Dep. Time, and we got duration time and day time so not necessary for Arrival Time. So removed Dep. Time, Arrival Time and Rout column also.

Then I moved to fill the nan value. For that I again checking the presents of nan value and find out that only one column have missing value. So I tried to fill nan with mean.

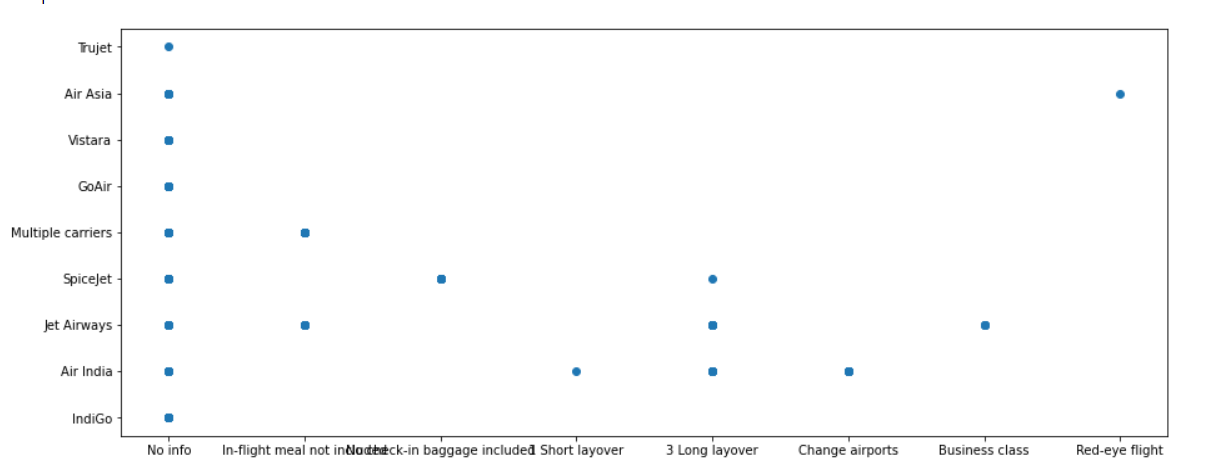
**Visualization:**

Now my data are ready for the visualization. Data visualization is the discipline of trying to understand data by placing it in a visual context so that patterns, trends and correlations that might not otherwise be detected can be exposed. Python offers multiple great graphing libraries that come packed with lots of different features.

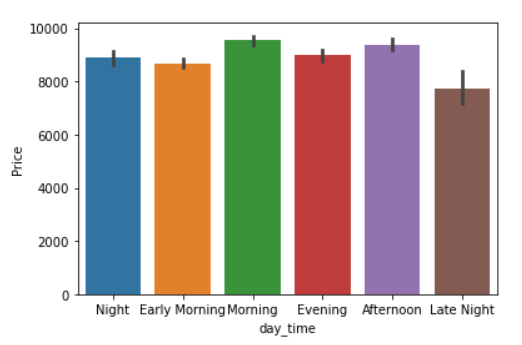
When draw a scatter plot I can understand that Jet Airways is the airline charged highest price and least price charged by TruJet.



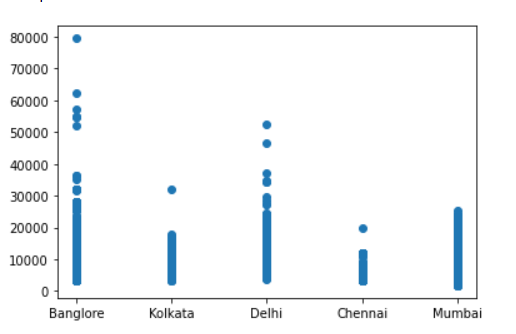
When I check the relation between additional information and Airline , we can see that Jet Airways provide high number of additional facilities like meals, Business class and 3Long layover. Go Air, Vistara and TruJet has not providing anything additionally. Their price might be only for the traveling expense.



After I plotting a bar plot I can understand more about level of price. Here I tried with price and day time feature. Highest price were charged at morning and least are charged at late night.



Price also mainly depend with Source (The source from which the service begins). If the flight take off from Bangalore, has charged highest price. Least price charged source is Chennai.



you already know that if you have a data set with many columns, a good way to quickly check correlation among columns is by visualizing the correlation matrix as a heatmap.

But we have only two numerical columns which is price and Duration. When I plot a heatmap for getting correlation between these two features, I can find out that both are equally correlated with each other. If the two independent features are equally correlated with each other, we can drop any one of that colum. but in my case price is my target column and Duration is independent column so not need to do anything more.

Now I reached the final preprocessing stage that is Encoding. We have mainly two types of encoding Label encoding and one hot encoding. In label encoding ,we replace the categorical value with a numeric value between 0 and the number of classes minus 1. If the categorical variable value contains 5 distinct classes, we use (0,1,2,3 and 4). One hot encoding is the method convert the data as 0 and 1as per the number of value similar as label encoding but instead of create the value in single column created number of column according to the number of values.

In this problem I was used label encoding because when we use one hot encoding it will increase the number of columns. It might be difficult to learn model.

This is the final step for train the model. For that I did split as input(x) and output(y) .

Standard scaler:

Standardizes a feature by subtracting the mean and then scaling to unit variance. Unit variance means dividing all the values by the standard deviation. Standard scale makes the mean of the distribution 0. About 68% of the values will lie be between -1 and 1. So I also used standard scaling for make the data in a single scale.

After that I used train test split for training test process.

Models: Linear regression, K Neighbour Regressor, Decision Tree Regressor.

**Linear Regression**

Linear Regression fits a linear model with coefficients w = (w1, …, wp) to minimize the residual sum of squares between the observed targets in the dataset, and the targets predicted by the linear approximation.

After training the linear model I got r2 score is 77 and I also try to tuning the performance of the model using Grid search CV I got batter r2 score.

**K Neighbour Regressor**

K nearest neighbours is a simple algorithm that stores all available cases and predict the numerical target based on a similarity measure.

My K Neighbour accuracy is 66 and after tuning process I got better performance.



**Decision Tree**

It is the a non-parametric supervised learning method used for [classification](https://scikit-learn.org/stable/modules/tree.html#tree-classification) and [regression](https://scikit-learn.org/stable/modules/tree.html#tree-regression). The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. A tree can be seen as a piecewise constant approximation.

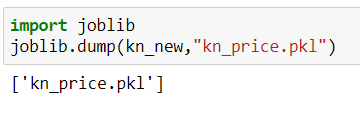
My decision Tree value is 64 and after tuning the model I got 66.

**Conclusion:**

Comparing the above three model I conclude that K Neighbour is the best model.

This study shows that it is feasible to predict the airline ticket price based on historical data. Adding more features will increase the accuracy of the models. In the future, more features, like available seats, departure day is a holiday or not can be included. It will help the model for better prediction. In this data set we have a column named ‘additional info’ , not provide much information, most of the values are similar like blank(no info).

**Save model**:

****