Flight Price Prediction

Problem Definition

Any individual who has booked a flight ticket previously knows how dynamically costs 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, or night. Cost may likewise change with the seasons like winter, summer, and celebration seasons. The extreme goal of the carrier is to build its income yet on the opposite side purchaser is searching at the least expensive cost. Purchasers generally endeavour to purchase the ticket in advance to the take-off day. Since they trust that airfare will be most likely high when the date of buying a ticket is closer to the take-off date, yet it is not generally true. Purchaser may finish up with the paying more than they ought to for a similar seat. The Traveler often find the flight prices unpredictable as the flight prices tomorrow will not be the same as the same flight today. The system is complicated because each flight has limited number seats to be sold .In case the demand of air tickets is high ,Than the price will increase and on the other hand if the seats are left unsold than the cost of the air tickets might decrease as it represents a loss of revenue .To solve this problem of predicting flight prices, Machine Learning is great idea to learn from historical data of the past flight prices and build logic on given data.

Data information

The prices of flight tickets for various airlines are between the months of march and June of 2019 and between various cities. We have two date sets i.e. Train data and Test data. The size of **Training Set** is 10,683 records which consists of both categorical and numeric data. Some special Characters are also seen with in data to which we will apply data transformation before using it on the Model.

The **Features** considered initially for each flight are: -

Air lines: 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.

Deep 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 Stop: Total stop between the source of destination.

Additional Info.: Information about the flight.

Price: The price of the Ticket.

The Size of **Testing Set** is 2671 records. The testing data is like the training data, except for the "price" column which will be predicted using the model.

Data Analysis

From numerous sources the data was collected. The flight ticket detailed information is retrieved from an online data source (Github.com). We took out this data from the website which is in the form of csv record .The file consists of the information with input features and its target variable required for analyzing data. Days to departure can be obtained by calculating the difference between the departure date and the date on which data is taken. This parameter is within 45 days. Also, the day of departure plays an important role in whether it is holiday or weekday We

have retrieved additional features from the existing variables to get more accuracy in the results Features such as “Arrival Time”, “Arrival Date”, “Arrival Month”, “Day”, “Month” & “Year” are generated to make analysis of data

Cleaning And Data Preparation

All the collected data needed a lot of work so after the collection of data, it is needed to be clean and prepare according to the model requirements. All the unnecessary data is removed like duplicates and null values. In all machine learning this technology, this is the most important and time-consuming step. Various statistical techniques and logic built-in python are used to clean and prepare the data. For example, the price was character type, not an integer.

Data analysis using visualizations

Data preparation is done by breaking down the information, understanding patterns and then applying different ML algorithm in this case we divide our data into three different data frames based on its type such as nominal, ordinal, or continuous types of data.

Visualizations

We use cat plot, count plot for nominal/categorical data as it gives the frequency of the columns. The following observations were made while analyzing data.

1)The maximum the flights flying is from jet airways, indigo And Air India and lest are from True jet and Vistara Premium economy.

2)Most of the flights are flying from Delhi and Lest from Chennai.

3)Majority of the flights are landing at the Cochin Airport and lest are landing at the Kolkata Airport.

4)The majority flights have only one stop in between the journey and the most of them are also Non-stop but very few flights have 3 and 4 stops.

5)The flights have journey mostly in the month of June, May, and March.

Visualization of Continuous Type of Values

The observation says:

- The data is broadly scattered in all columns excepts for the price columns.
- The data is the price columns is right skewed, but it is a target variable.
- In the day column, the maximum number of flights are flying between the date 3 and 7

EDA Concluding Remarks

After performing all the transformation integration and cleaning of data, we get all the relevant variables and significant information required for building an ML model. We end Up having 11 variables and 10,683 records in data set. The final data det consists of important features used for analysis are:

- Airlines
- Source
- Destination
- Route
- Deep Time
- Arrival Time
- Duration
- Total Stops
- Price
- Months
- Day

Pre-processing pipeline

Divide the data set into test and train

1)Now that we have converted all the categorical columns into numerical using the encoding technique. The next step is to split the data into test and train and drop the price column from the data set as we need to predict the price.

2) Finding the best Random_ state: We have used Random Forest Regressor to obtain the best random state.

3) Final dimension of data: 10682 rows & 29 columns.

4) Created train test split: We have split the train & test data in 0.2 test size with the best random state.

Building Machine Learning Models

1)Linear Regression: It is a linear model, e.g., a model that assumes a linear relationship between the input variables (x) and the single output variable (y). More specifically, that y can be calculated from a linear combination of the input variables (x).

Mean Squared Error: 0.7944639940035147
Root Mean Squared Error: 0.2818623767024458
R squared value: 0.6200430559667731

2) Decision Tree Regression: Decision Regression trees are needed when the response variable is numeric or continuous. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The result is a tree with decision nodes and leaf nodes.

Mean Squared Error: 0.7570431513651493
Root Mean Squared Error: 0.2751441715474179
R squared value: 0.5796665216771614

3) Random Forest Regressor: It uses the ensemble learning met hod for regression. The ensemble learning method is a technique that co mbines predictions from multiple machine learning algorithms to make a more accurate prediction based on majority voting/averaging.

Mean Squared Error: 0.7570431513651493

Root Mean Squared Error: 0.2751441715474179
R squared value: 0.637939787749886

Concluding Remark

The r2 score achieved linear Regression is 62%.

The r2 score achieved for Decision Regression is 57%.

The r2 score achieved for Random Forest Regression is 63%.

Following the same procedures for the testing file as done for the training file (the complete EDA process), we have used the best saved model of the training file to predict the analysis of the testing file.