***FLIGHT PRICE PREDICTION***

***DETAILED PROJECT REPORT***

**SOHAM PAWAR**

**MUAAZ PATEL**

**DIPALI BHANUSHALI**

**KHUSHABU PATIL**

# Introduction

Flight ticket prices can be something hard to guess, today we might see a price, check out the price of the same flight tomorrow, and it will be a different story. To solve this problem, we have been provided with prices of flight tickets for various airlines between the months of March and June of 2019 and between various cities, using which we aim to build a model which predicts the prices of the flights using various input features

1. **Problem Statement**

## Anyone who has booked a flight ticket knows how unexpectedly the prices vary. Airlines use using sophisticated quasi-academic tactics which they call "revenue management" or "yield management". The cheapest available ticket on a given flight gets more and less expensive over time. This usually happens as an attempt to maximize revenue based on –

## Time of purchase patterns (making sure last-minute purchases are expensive)

1. Keeping the flight as full as they want it (raising prices on a flight which is filling up in order to reduce sales and hold back inventory for those expensive last-minute expensive purchases)

Airline industry is one of the most sophisticated in its use of dynamic pricing strategies to maximize revenue, based on proprietary algorithms and hidden variables. Therefore, it is challenging for consumers to predict the price change in the future i . With the information of the airfare available online, buyers are trying to track the prices of the flight over a certain period of time, and anticipate the price change in the future. However, it turns out to be rather difficult to predict the price of the flight precisely only by observation.

1. **Objective**

The goals of this project is to predict price of the flight as per user’s needs from the data using machine learning techniques.

The table below describes the features of the data.

Feature D

Following is the description of features available in the dataset –

1. **Airline**: The name of the airline.

2. **Date\_of\_Journey**: The date of the journey

3. **Source**: The source from which the service begins.

4. **Destination**: The destination where the service ends.

5. **Route**: The route taken by the flight to reach the destination.

6. **Dep\_Time**: The time when the journey starts from the source.

7. **Arrival\_Time**: Time of arrival at the destination.

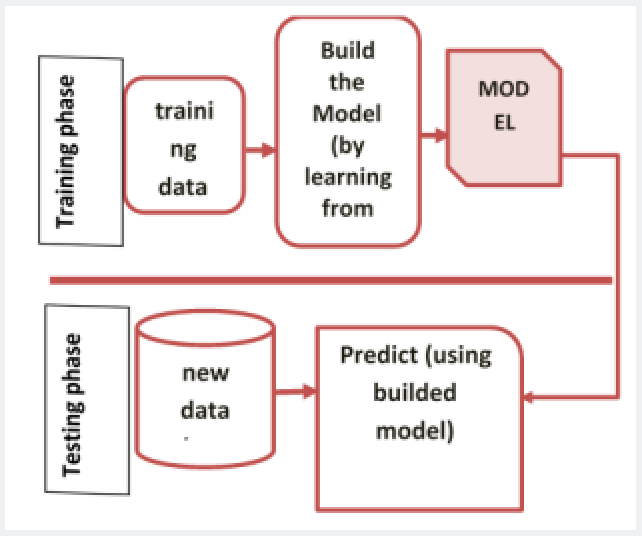
8. **Duration**: Total duration of the flight.

9. **Total\_Stops**: Total stops between the source and destination.

10. **Additional\_Info**: Additional information about the flight

11. **Price**: The price of the ticket.

1. **Architecture**

1. **Overview :**

**The whole project is divided into 7 steps:**

1. Importing dependencies and loading Data set

2. Data Preprocessing

3. Exploratory Analysis

4. Statistical Analysis

5. Train Test Split

6. Training the Model

7. Testing the model accuracy

**Step 1: Importing Dependencies and Loading Dataset**

In order to do the predictive analysis we need to import some python libraries which will help in data visualization, dealing with data set and will also provide pre-implemented Machine Learning models.

**Step 2: Data Preprocessing**

This is the most important step of all Machine Learning and Data Science projects. It is about **80%** of the overall work. For this project I have done data cleaning manually by identifying the relation between multiple columns, although there are some tools and standard procedures available but I found it more suitable as per the accuracy.

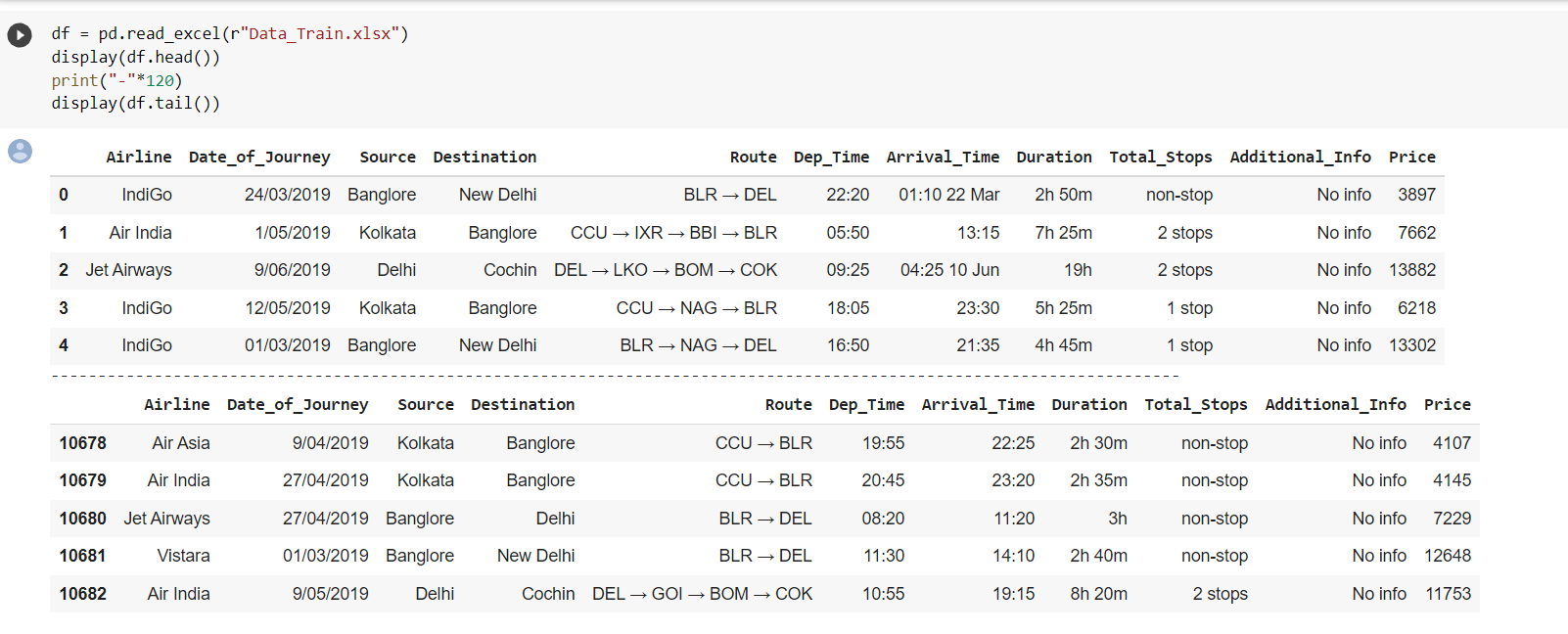
We identify the below mentioned points in the first look –

1. The Route column contains a list of cities which we will need to separate, since we would have multiple combinations in our dataset.

2. The Arrival time column has dates attached along with, which we will need to separate. These are the cases when the flight takes off from the source on a date and reaches its destination on the next day.

3. The Duration is in a string format, which we will need to convert to integer type.

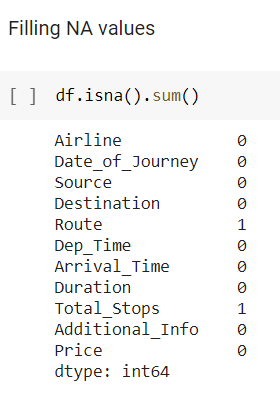
4. The total stops also has text ‘stops’ added along with the number of stops, and certain columns as ‘non-stop’, which we will need to convert to integer types.



We run the data.info() command, which gives us the information about number of values present in each column, and data types of each column.

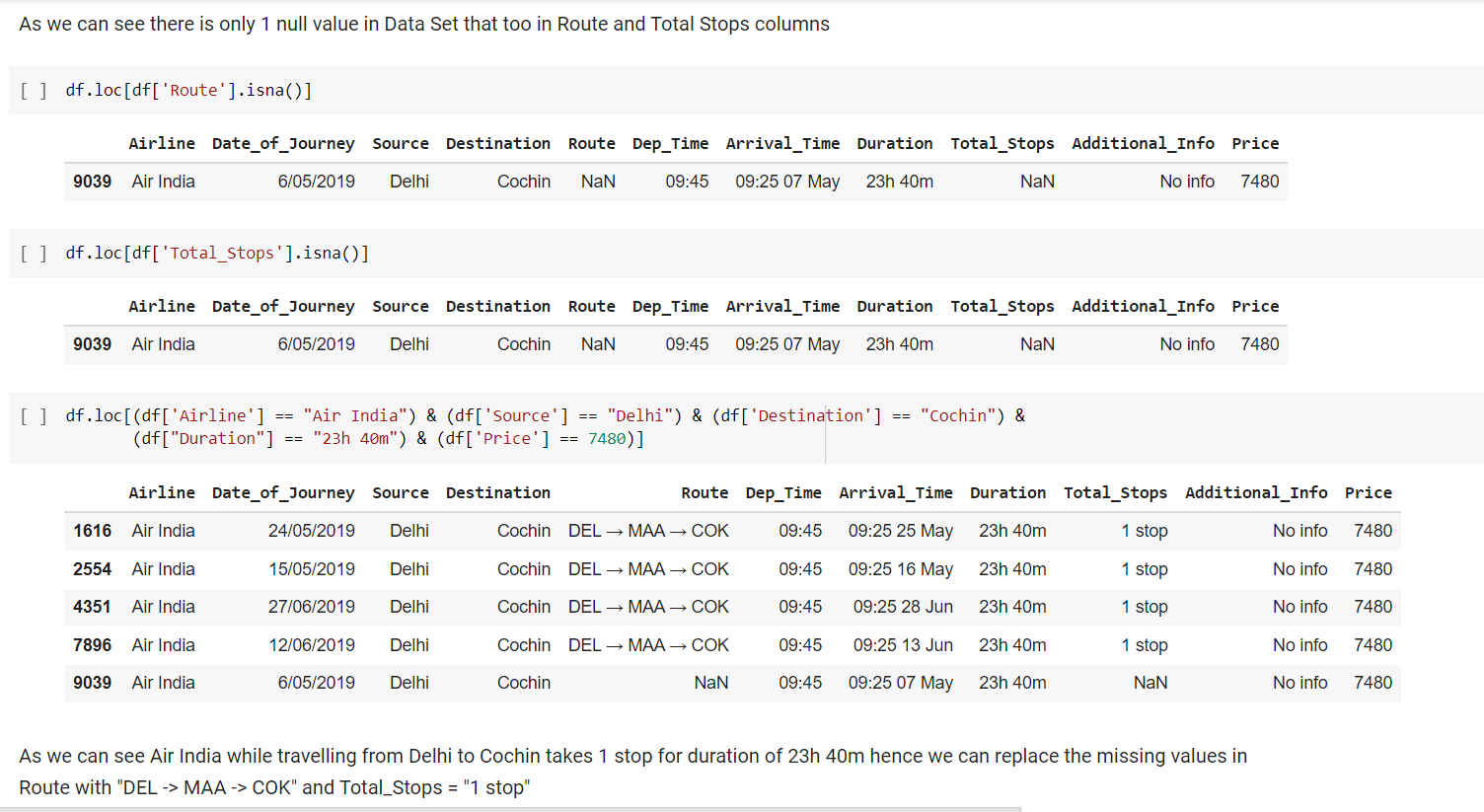
We observe that we have all the columns as ‘object’ data types, and only ‘Price’ column (the output) is of integer type. Since we know what our columns signify, we know which columns we need to treat!

We now check the count of NaN (null) values in our dataset, which turns out to give the following result –



We have 1 missing value in Route column, and 1 missing value in Total stops column. We will meaningfully replace the missing values going further.

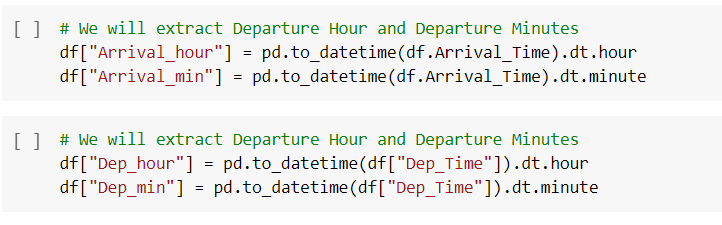
We now start exploring the columns available in our dataset. The first thing we do is to create a list of categorical columns, and check the unique values present in these columns –



We observe that –

1. Airline column has 12 unique values - 'IndiGo' , 'Air India', 'Jet Airways' , 'SpiceJet' , 'Multiple carriers' , 'GoAir', 'Vistara', 'Air Asia', 'Vistara Premium economy' , 'Jet Airways Business', 'Multiple carriers Premium economy', 'Trujet'.
2. 2. Source column has 5 unique values – ‘Bangalore’, ‘Kolkata’, ‘Chennai’, ‘Delhi’ and ‘Mumbai’.
3. 3. Destination column has 6 unique values - 'New Delhi', 'Banglore', 'Cochin', 'Kolkata', 'Delhi' , 'Hyderabad'.
4. 4. Additional info column has 10 unique values - 'No info', 'In-flight meal not included', 'No check-in baggage included', '1 Short layover' , 'No Info', '1 Long layover', 'Change airports' , 'Business class', 'Red-eye flight' , '2 Long layover'.

We now split the Date column to extract the ‘Date’, ‘Month’ and ‘Year’ values, and store them in new columns in our dataframe

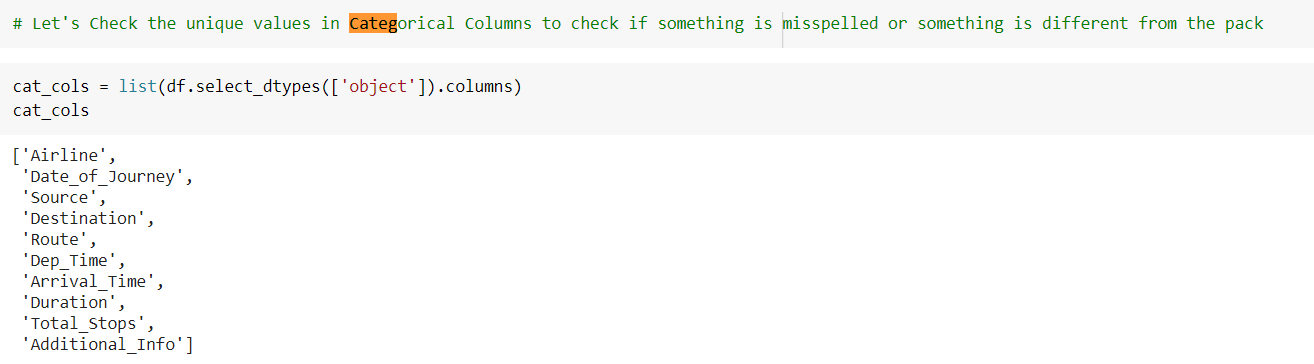


Further, we split the Route column to create multiple columns with cities that the flight travels through. We check the maximum number of stops that a flight has, to confirm what should be the maximum number of cities in the longest route –

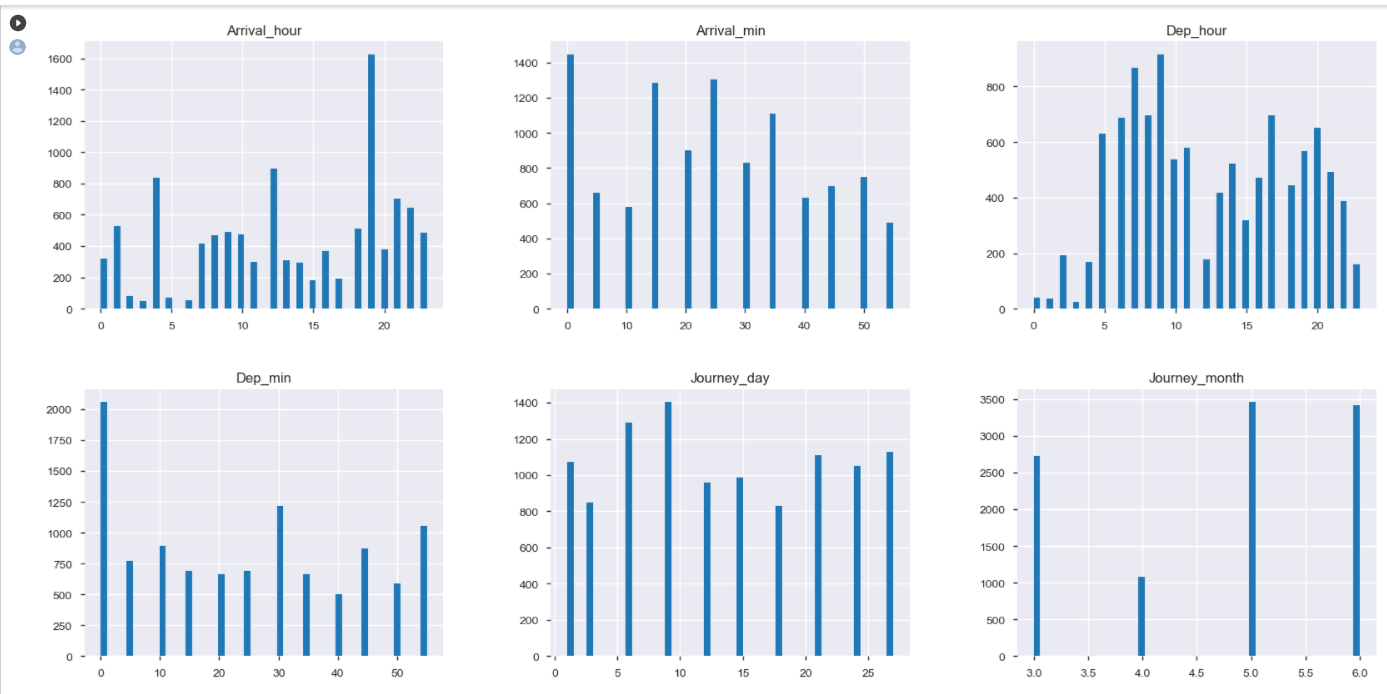
# Replacing the missing values

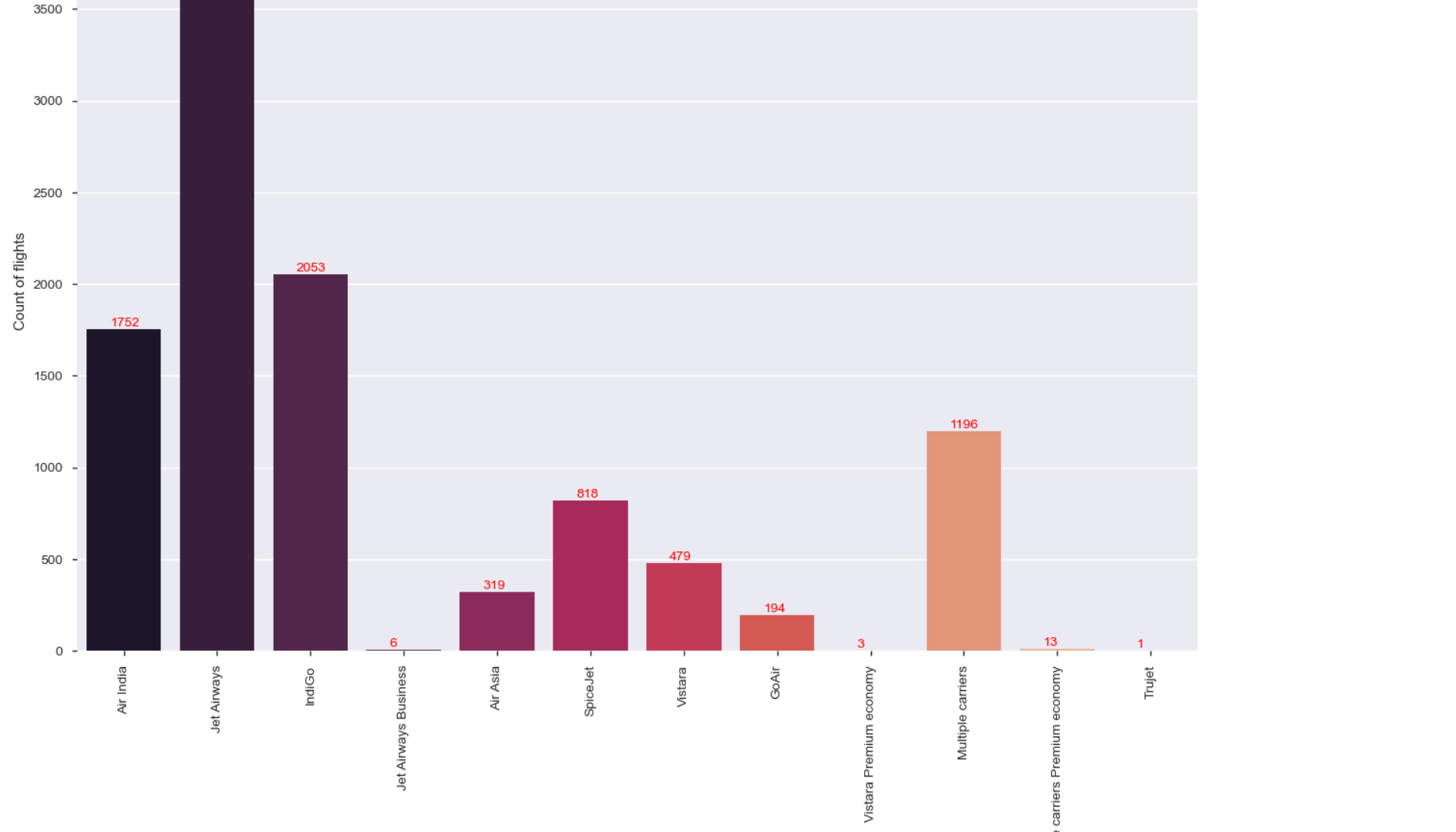
Due to introduction of new columns, we introduced many NaN values in our data set, which we now need to sort out.

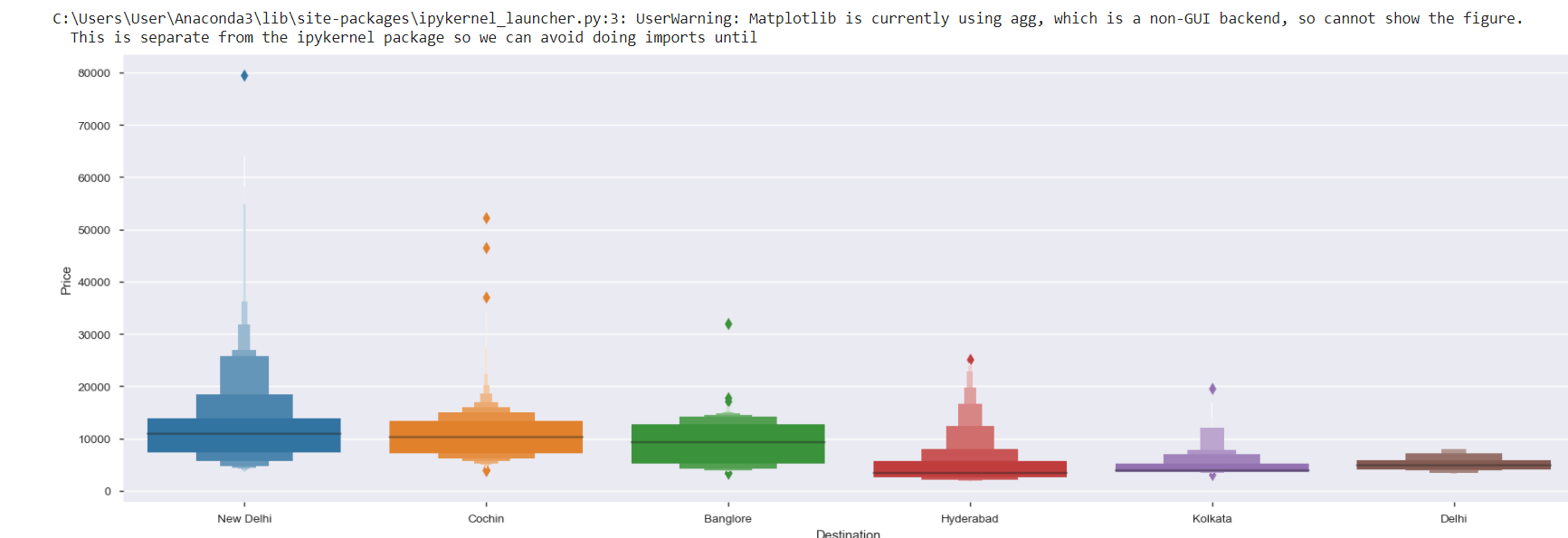
List of categorical columns



A few visualizations







We make the below mentioned observations using the plots above –

**Airlines**

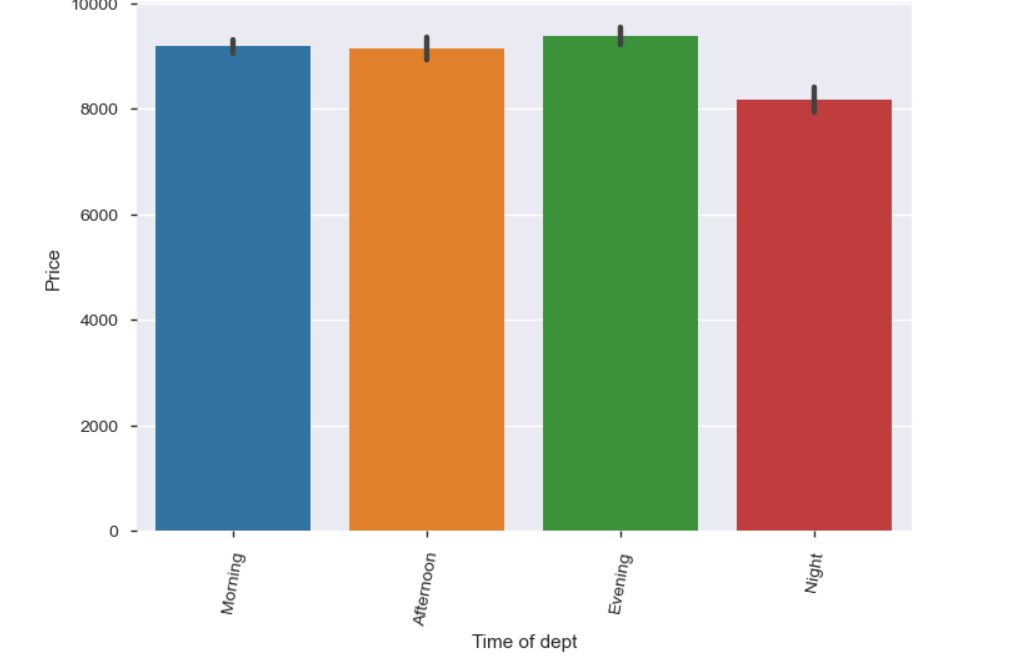
* Jet Airways is the most preferred airline with the highest row count, followed by Indigo and AirIndia.
* Count for Vistara Premium economy, Trujet, Multiple carries premium economy and Jet airways business is quite low.

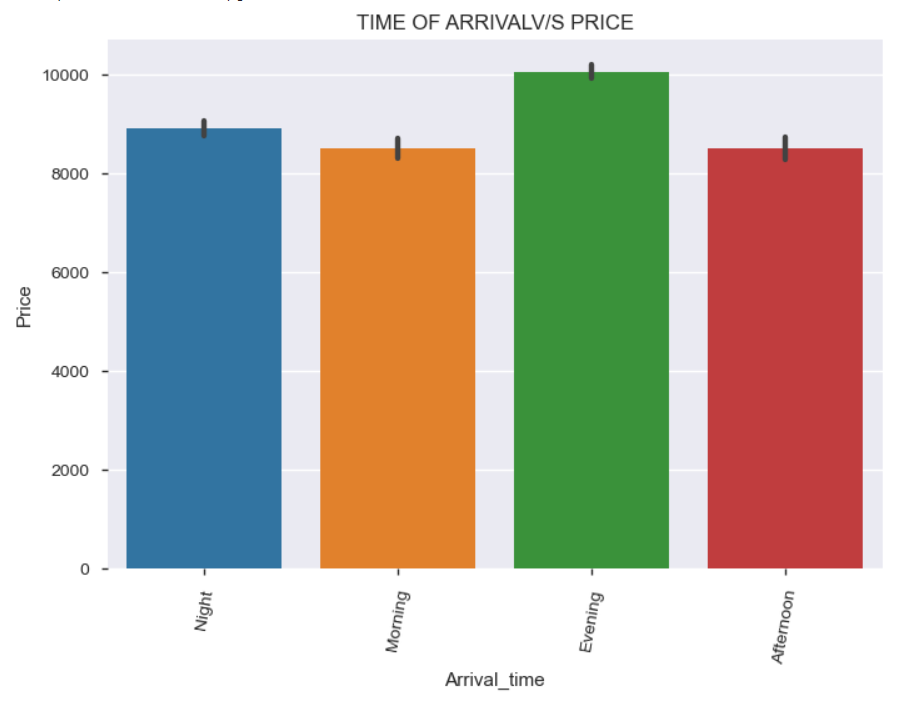
**Source**

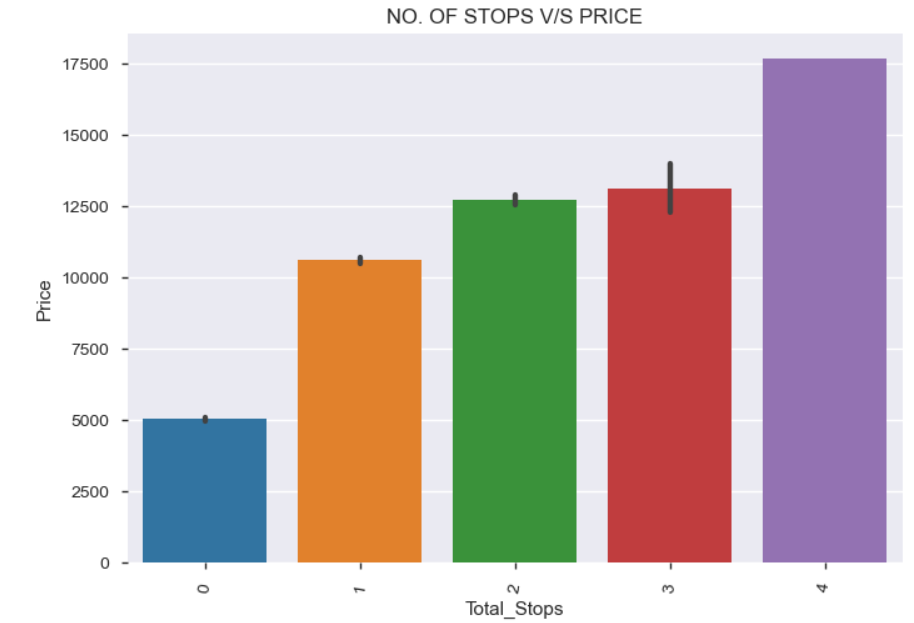
* Majority of the flights take off from Delhi
* Chennai has the minimum count of flight take-offs

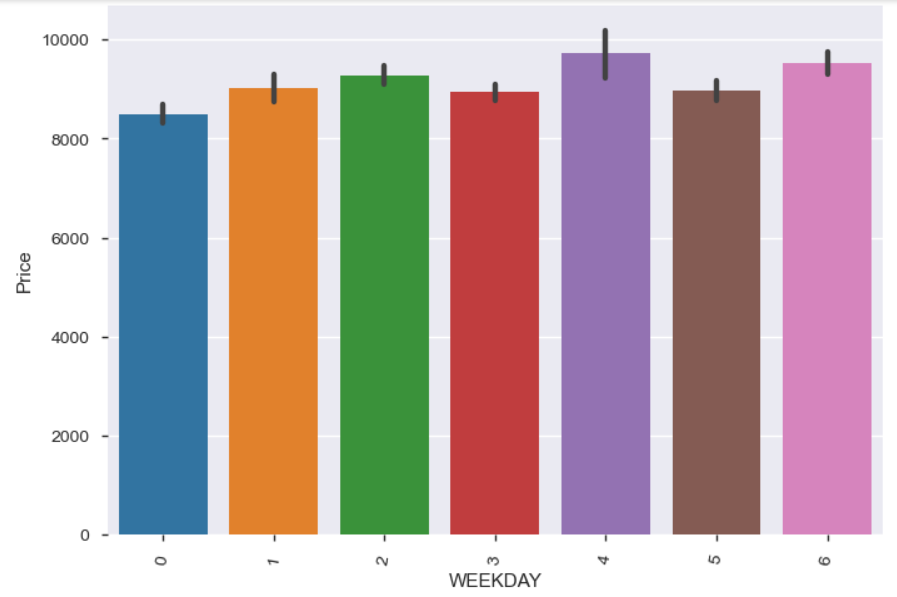
**Destination**

* Maximum flights land in Cochin
* Kolkata has the lowest count of receiving the flights









**Date**

* There are no specific dates when the flights travel; the distribution is almost similar for all dates

**Month**

* People tend to travel less in April
* - Flights in May and June have a higher count, seems like people travel during holiday months

**Year**

* This column has only 2019 as a value and can be dropped

​**Dep\_Time\_Hour**

* Majority of the flights tend to fly in the early morning time
* Count of flights taking off during 16:00 - 23:00 is also high, Afternoon flights are less in number.

​**Dep\_Time\_Min**

* Most flights take off at whole hours (Mins as 00)

​**Arrival date**

* In majority of the cases, flights take off and land on the same day

​**Arrival time hour**

* Majority of the flights reach its destination in the evening time around 18:00-19:00
* This seems to be because majority of the flights have take-off times in the morning and hence land after in the evening

**Arrival time min**

* This distribution is similar and does not give out any dedicated information

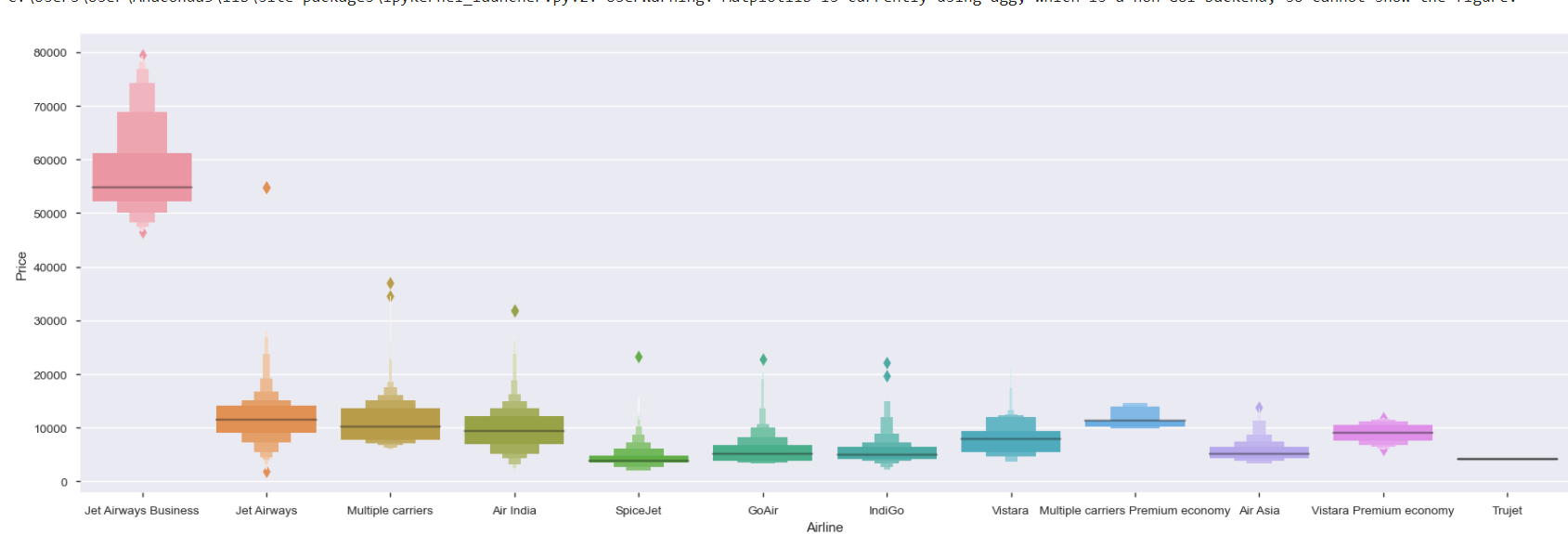
**Travel hours**

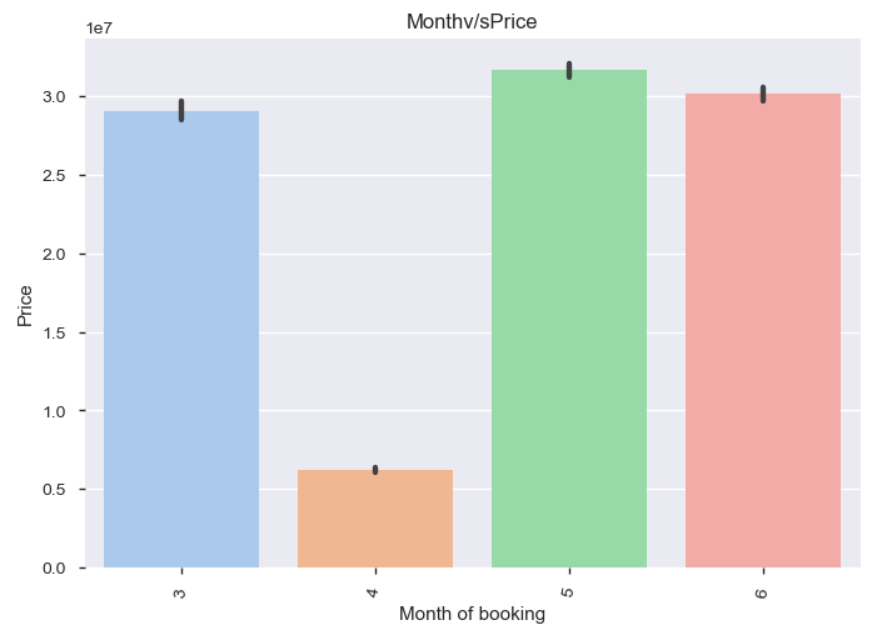
* Majority of the flights have travel time for around 2-3 hours, which seems ok since these are domestic flights
* Some flights have time around 30 hours too, this could be because of the number of stops in between

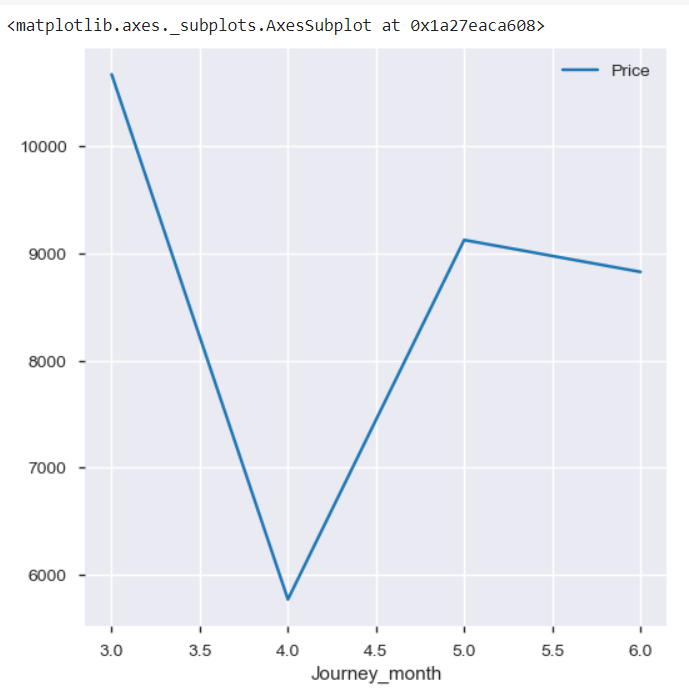
**Travel mins**

- The data is divided and is not pointing towards any specific points

**Distibution of ‘Price’ column**









**Step 4: Statistical Analysis**

**We have performed following analysis.**

1. Examine the data

2. Data type and length of the variables

3. Check for Missing Values

4. Numerical and Categorical Variables Identification

5. Summarizing Numerical Variables

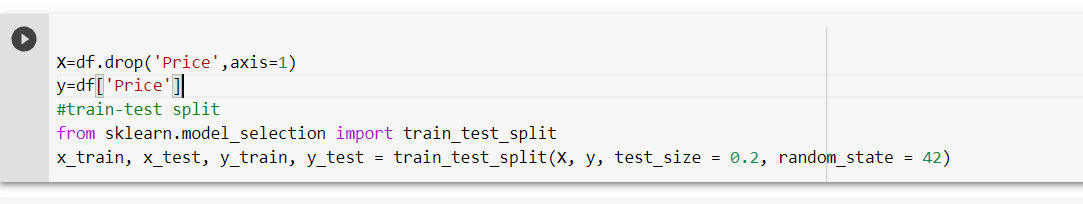
6. Summarizing Categorical Variables

7. Categorizing Quantitative and Qualitative Variables

8. Outliers

**Step 5: Train and Test Data Sets**

Once the dataset is processed, we need to divide it into two parts: training and test set. We will import and use the train\_test\_split function for that.



**Step 6 : Training the Machine Learning Model**

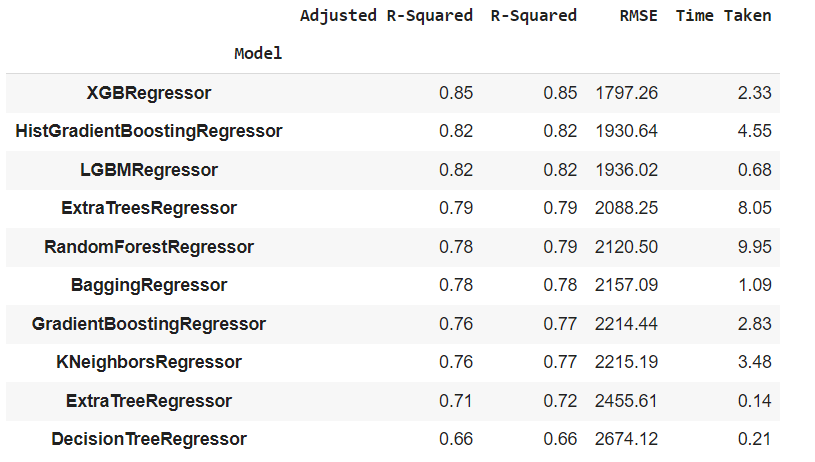
Since there are multiple conditions it depends on what parameters the user choosed accordingly the price will be predicted

Also visualization of data also gave an intuition that there are decision boundaries which can be used as the basis of selecting the Machine Learning model.

**Checking Model Accuracy**

Final step is to check the accuracy of the Machine Learning model which we have created for ad click prediction:

**Experimental Results:**



**Conclusion**

We further proceed to test the object that we saved using pickle and the model can be reused. We used XGBRegressor as it can approximately accurate best predicted price from the model.