1. INTRODUCTION

In the wake of global economic challenges and widespread technological layoffs, individuals across the world, including India, are grappling with financial uncertainties. Every rupee counts more than ever, prompting consumers to seek ways to save money wherever possible. One significant area where this quest for savings intersects with unpredictability is the airline industry. Fluctuating prices in this domain often leave consumers bewildered and frustrated, especially amidst financial strain.

Recognizing this pressing need for predictability and affordability in air travel, I embarked on a project leveraging the power of machine learning. The objective was clear: to assist airline customers in India in predicting the price of air tickets accurately. By harnessing various features such as the origin and destination of travel, departure and arrival dates, type of airline, and number of stoppages, I aimed to develop a robust model capable of forecasting flight prices with precision.

This project serves a dual purpose. Firstly, it addresses the immediate concern of individuals facing financial difficulties by empowering them with tools to make informed decisions and potentially save on their flight bookings. Secondly, it underscores the broader potential of machine learning in alleviating real-world challenges, demonstrating its applicability in enhancing consumer experiences and mitigating economic burdens.

In this endeavor, I employed five different supervised machine learning models to predict flight prices. Through rigorous evaluation and comparison, one model emerged as the frontrunner: XGBoost. Leveraging its superior performance, I deployed the model on the Heroku cloud platform, ensuring accessibility and ease of use for consumers across India.

At its core, this project embodies a commitment to leveraging technology for the betterment of society, particularly in times of economic hardship. By providing individuals with the means to navigate the complexities of flight pricing, it strives to make air travel more accessible and affordable for all.

2. Literature Review

Airline ticketing congestion and tight delivery time windows force passengers to exorbitant prices which they wouldn't have paid under normal circumstances. This poses a lot of strains on consumers and in the past few years, machine learning has been applied into different time series problems. Airline companies use many different variables to determine the flight ticket prices: indicator whether the travel i during the holidays, the number of free seats in the plane etc. Such variables have be used to optimize the pricing and routing of airlines. Stefan Klein and co. found that during the early years of e-Commerce, the tourism sector had high expectations for online booking. All the major airlines invested huge sums of money, not only to make booking features available, but also to integrate them into attractive, easy-to-use Web offerings. Nevertheless, even after years of investment and improvements, the booking ratio for all but the no-frills airlines is still disappointing. In spite of the rapid growth in Internet purchasing of products and services in the world, the buying rate of online flight ticketing remains low. Prior research investigates the factors that influence online ticket purchasing through a survey of Internet consumers to determine the relationships between convenience, willingness to purchase, price and trust. They found that ticket pricing still remains the major factor influencing consumer purchasing of airline tickets.

3. Objectives

Predictive Accuracy: Develop a machine learning model capable of accurately predicting flight prices within India based on various features such as source and destination cities, departure and arrival dates, airline types, and stoppages.

Affordability: Assist airline customers in India in making informed decisions about flight bookings by providing them with a reliable tool to predict prices. This aims to help individuals save money during times of economic hardship and financial uncertainty.

Model Selection and Optimization: Evaluate and compare multiple supervised machine learning algorithms to identify the most accurate predictor of flight prices. Optimize the selected model through hyperparameter tuning to enhance its predictive performance.

Accessibility: Deploy the optimized machine learning model on a cloud platform like Heroku to ensure easy accessibility for consumers across India. Develop a user-friendly interface to facilitate seamless utilization of the prediction tool.

Continuous Improvement: Implement mechanisms for monitoring the model's performance in real-time and gather user feedback to refine the model iteratively. Stay updated on industry changes and adapt the model accordingly to ensure its continued relevance and effectiveness over time.

Empowerment: Empower consumers with the means to navigate the complexities of flight pricing, enabling them to make informed decisions and potentially save on their air travel expenses amidst economic hardships.

Demonstration of Machine Learning Applications: Showcase the applicability of machine learning techniques in addressing real-world challenges, particularly in enhancing consumer experiences and mitigating economic burdens in the airline industry.

4. Technical Implementation

4.1 Methodology Diagram:

The project commenced with the acquisition of a comprehensive dataset encompassing flight prices within India, capturing crucial variables such as the origin and destination cities, departure and arrival dates, airline types, and stoppages. Subsequently, a meticulous data preprocessing phase ensued, involving thorough cleansing to rectify missing values, discrepancies, and outliers. Feature engineering techniques were employed to extract pertinent information and normalize numerical attributes, preparing the data for modeling.

Following data preparation, five distinct supervised machine learning algorithms—ExtraTreesRegressor, RandomForestRegressor, CatBoostRegressor, LGBMRegressor, and XGBRegressor—were implemented. The dataset was partitioned into training and testing subsets to facilitate robust evaluation. Each model underwent rigorous training on the training data, with performance assessment conducted using metrics including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2).

The ensuing phase involved meticulous model evaluation and comparison to ascertain the most accurate predictor of flight prices. Subsequently, the selected model underwent further refinement through hyperparameter tuning to optimize its accuracy and generalization capabilities. Upon achieving the desired level of performance, the optimized XGBoost model was deployed on the Heroku cloud platform. A user-friendly interface was developed to enable seamless access and utilization of the prediction tool by consumers.

To ensure ongoing effectiveness, mechanisms for real-time performance monitoring were instituted. User feedback was actively solicited to facilitate iterative model refinement. Additionally, the project remained attuned to industry dynamics, ensuring the model's continued relevance amidst evolving market conditions. This methodology was designed to leverage machine learning effectively, empowering consumers with accurate flight price predictions to navigate the complexities of air travel amidst economic uncertainties.

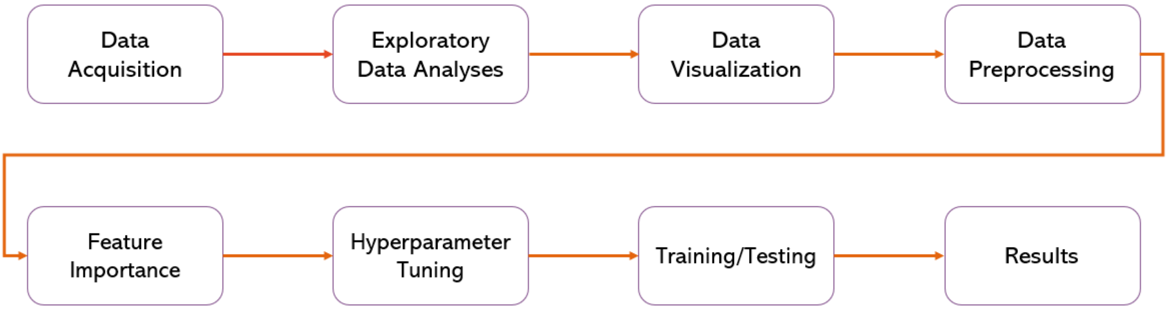


Figure: Methodology Diagram

5. Data Collection and Preprocessing:

5.1 Data Collection:

Octo parse scraping tools will be used to extract data from the website. An estimated total of 300261 distinct flight booking options will be extracted from the site. Data will be collected for 50 days, from February 11th to March 31st, 2022. Data source will be secondary data and collected from Ease my trip website. The various features of the cleaned dataset are explained below:

● Airline: The name of the airline company is stored in the airline column. It is a categorical feature having 6 different airlines.

● Flight: Flight stores information regarding the plane's flight code. It is a categorical feature.

● Source City: City from which the flight takes off. It is a categorical feature having 6 unique cities.

● Departure Time: This is a derived categorical feature obtained by grouping time periods into bins. It stores information about the departure time and have 6 unique time labels.

● Stops: A categorical feature with 3 distinct values that stores the number of stops between the source and destination cities.

● Arrival Time: This is a derived categorical feature created by grouping time intervals into bins. It has six distinct time labels and keeps information about the arrival time.

● Destination City: City where the flight will land. It is a categorical feature having 6 unique cities.

● Class: A categorical feature that contains information on seat class; it has two distinct values: Business and Economy.

● Duration: A continuous feature that displays the overall amount of time it takes to travel between cities in hours.

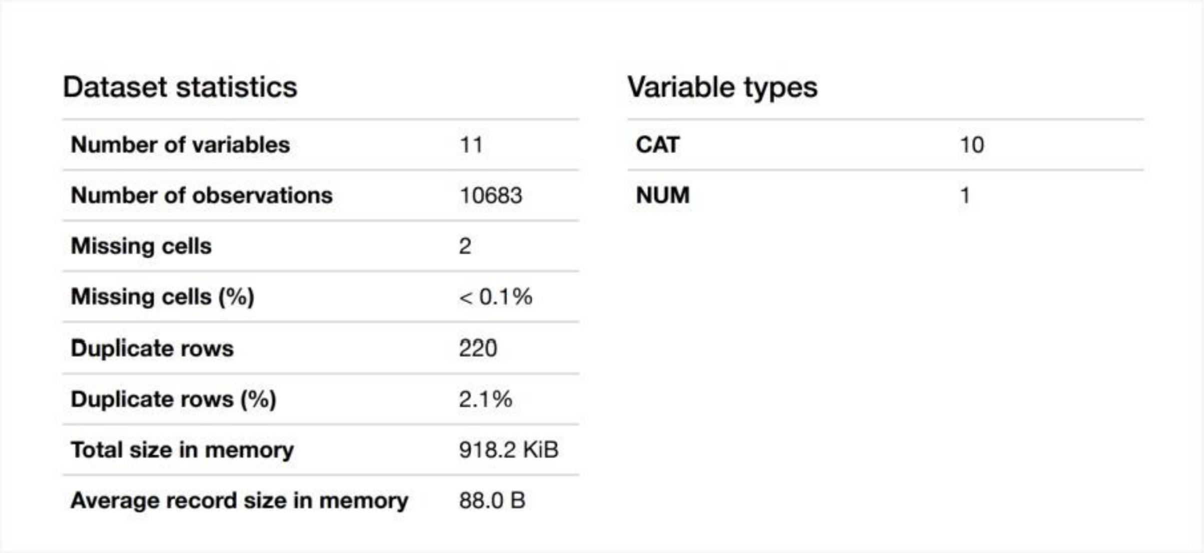
● Days Left: This is a derived characteristic that is calculated by subtracting the trip date by the booking date.

● Price: Target variable stores information of the ticket price.

5.2 Data Preprocessing

Exploratory data analysis(EDA) is one of the most important steps in machine learning. In order to get the data right and build robust models, I did a lot of EDA on the collected data. The following steps are what I used in arriving at the final dataset for building the machine learning models:

1. Automated Exploratory Data Analysis Using Pandas Profile Report



Dataset statistics

Number of variables	11
Number of observations	10683
Missing cells	2
Missing cells (%)	< 0.1%
Duplicate rows	220
Duplicate rows (%)	2.1%
Total size in memory	918.2 KiB
Average record size in memory	88.0 B

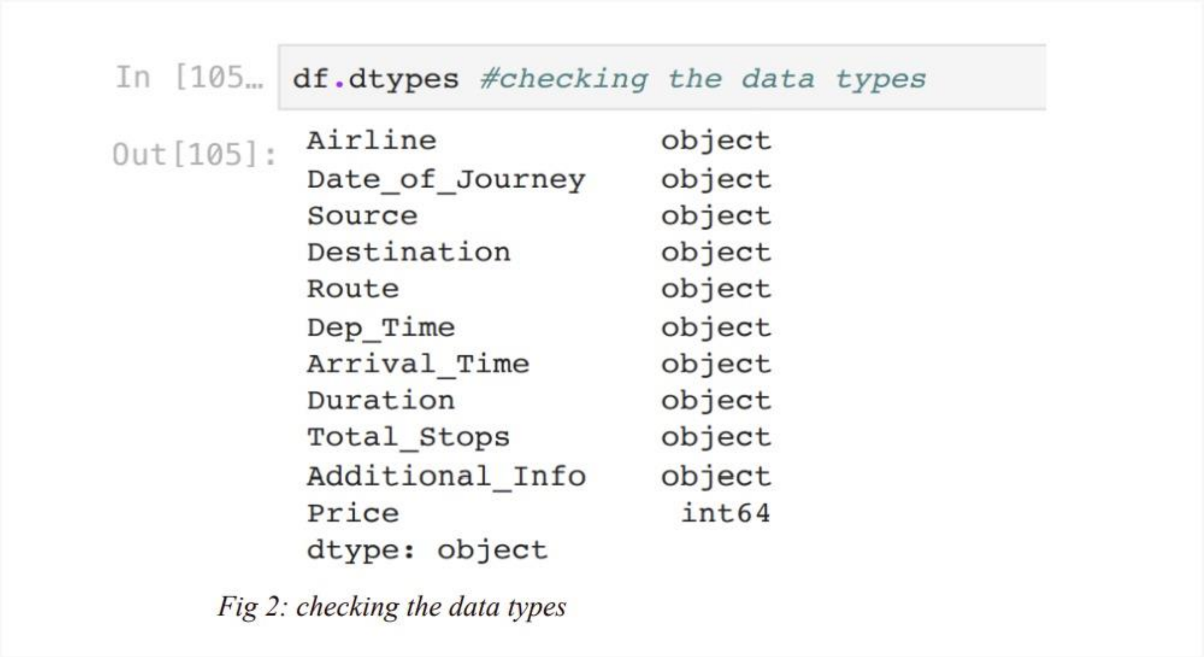
Variable types

CAT	10
NUM	1

Fig. 1: Pandas profiling report showing the EDA of the extract data. The above summary output shows the statistical EDA output of the variables in the dataset.

2. Manual Exploratory Data Analysis

A) First I check to see the data types of the features in the dataset, to make sure they are in the right type.

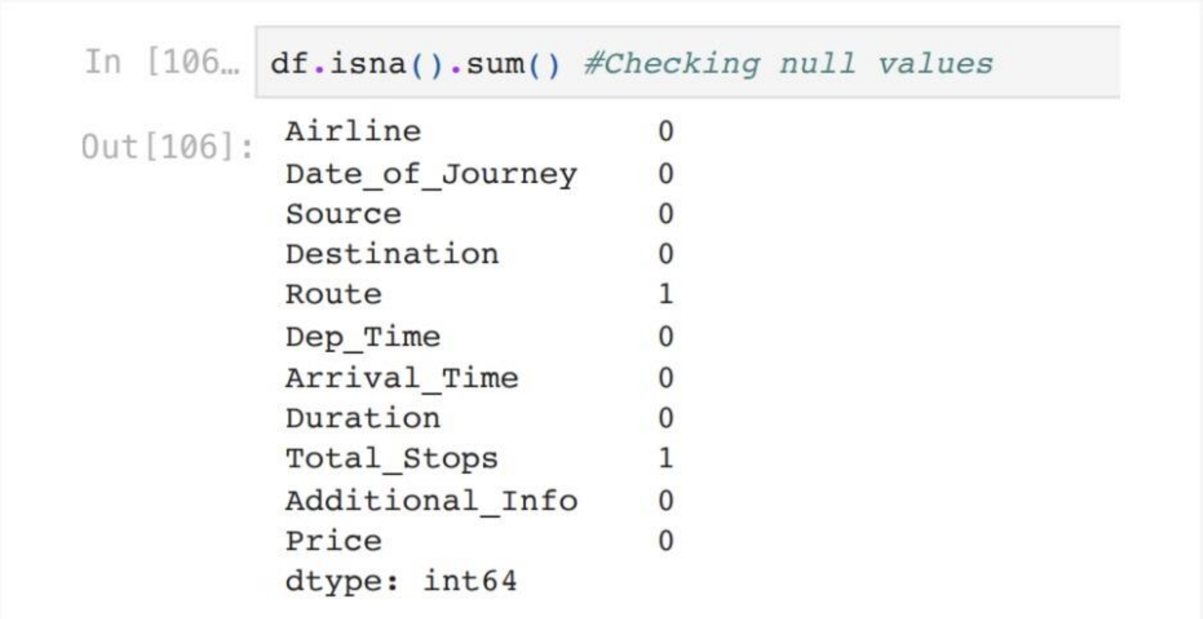


```
In [105… df.dtypes #checking the data types

Out[105]: Airline            object
          Date_of_Journey    object
          Source             object
          Destination        object
          Route              object
          Dep_Time           object
          Arrival_Time       object
          Duration           object
          Total_Stops        object
          Additional_Info    object
          Price               int64
          dtype: object
```

Fig 2: checking the data types

B) Second I checked to see if there missing values present in my dataset:

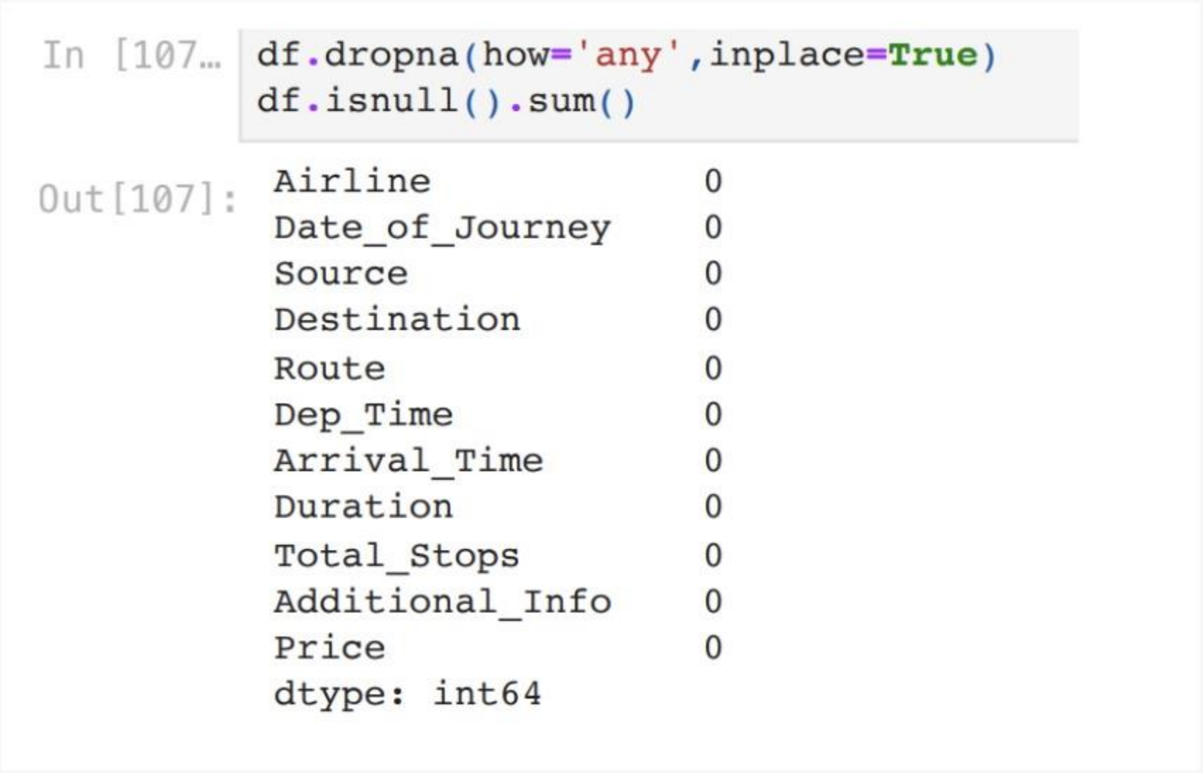


```
In [106…  df.isna().sum() #Checking null values

Out[106]: Airline            0
          Date_of_Journey    0
          Source             0
          Destination        0
          Route              1
          Dep_Time           0
          Arrival_Time       0
          Duration           0
          Total_Stops        1
          Additional_Info    0
          Price              0
          dtype: int64
```

Fig 3: checking for missing values

C) Next I removed the missing values from the dataset



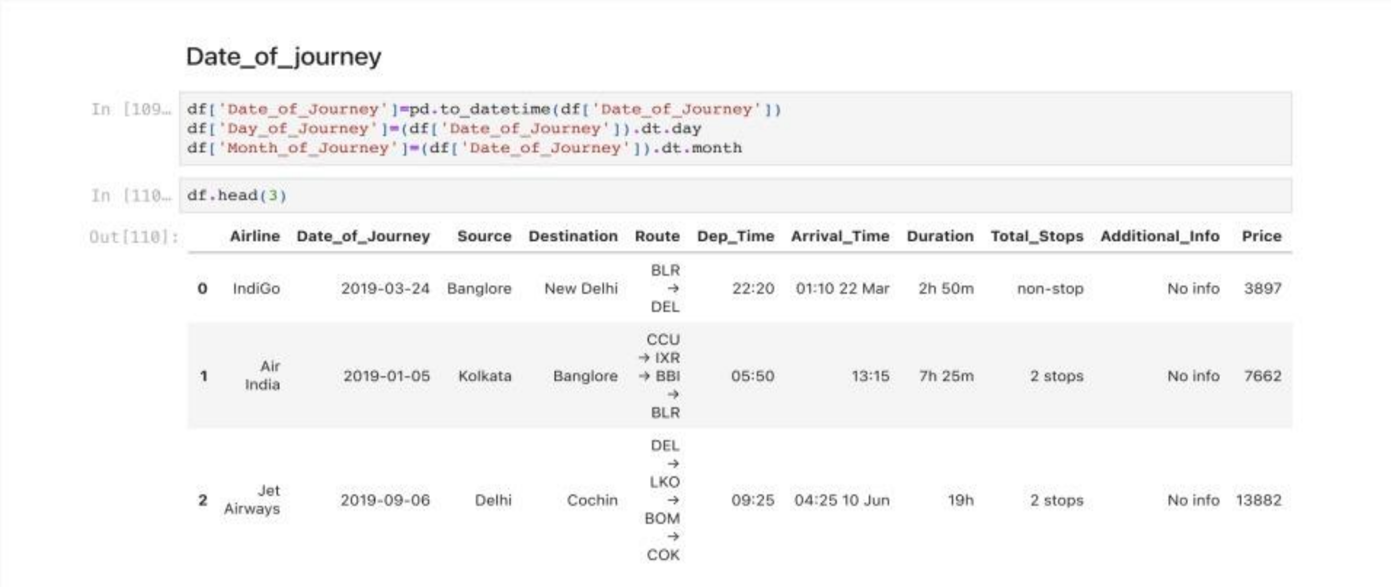
```
In [107…  df.dropna(how='any',inplace=True)
          df.isnull().sum()

Out[107]: Airline            0
          Date_of_Journey    0
          Source             0
          Destination        0
          Route              0
          Dep_Time           0
          Arrival_Time       0
          Duration           0
          Total_Stops        0
          Additional_Info    0
          Price              0
          dtype: int64
```

Fig 4: removing missing values

5.3 Feature Engineering

A) Again there was some feature engineering that needed to be done on the dataset in order to get the right data for the analysis. The first feature engineering to be done was to convert Date_of_Journey feature in the dataset to its appropriate format as datetime with regards to day and month.



Date_of_journey

```
In [109…  df['Date_of_Journey']=pd.to_datetime(df['Date_of_Journey'])
          df['Day_of_Journey']=(df['Date_of_Journey']).dt.day
          df['Month_of_Journey']=(df['Date_of_Journey']).dt.month
```

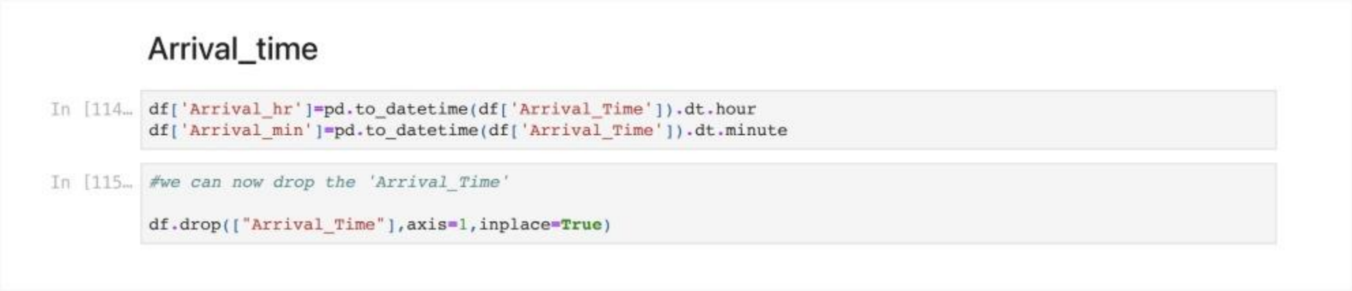
```
In [110…  df.head(3)
```

Out[110]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
0	IndiGo	2019-03-24	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897
1	Air India	2019-01-05	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	No info	7662
2	Jet Airways	2019-09-06	Delhi	Cochin	DEL → LKO → BOM → COK	09:25	04:25 10 Jun	19h	2 stops	No info	13882

Fig 5: convert ***Date_of_Journey*** feature to datetime format

I repeated the above strategy for couple of features that needed to be engineered as shown below:

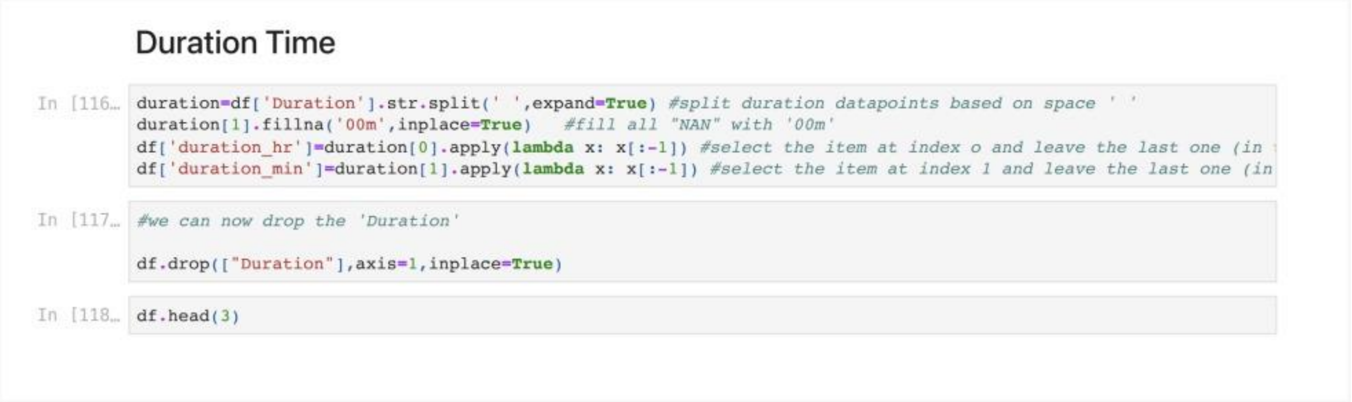


Arrival_time

```
In [114…  df['Arrival_hr']=pd.to_datetime(df['Arrival_Time']).dt.hour
          df['Arrival_min']=pd.to_datetime(df['Arrival_Time']).dt.minute
```

```
In [115…  #we can now drop the 'Arrival_Time'

          df.drop(["Arrival_Time"],axis=1,inplace=True)
```



Duration Time

```
In [116…  duration=df['Duration'].str.split(' ',expand=True) #split duration datapoints based on space ' '
          duration[1].fillna('00m',inplace=True)   #fill all "NAN" with '00m'
          df['duration_hr']=duration[0].apply(lambda x: x[:-1]) #select the item at index o and leave the last one (in
          df['duration_min']=duration[1].apply(lambda x: x[:-1]) #select the item at index 1 and leave the last one (in
```

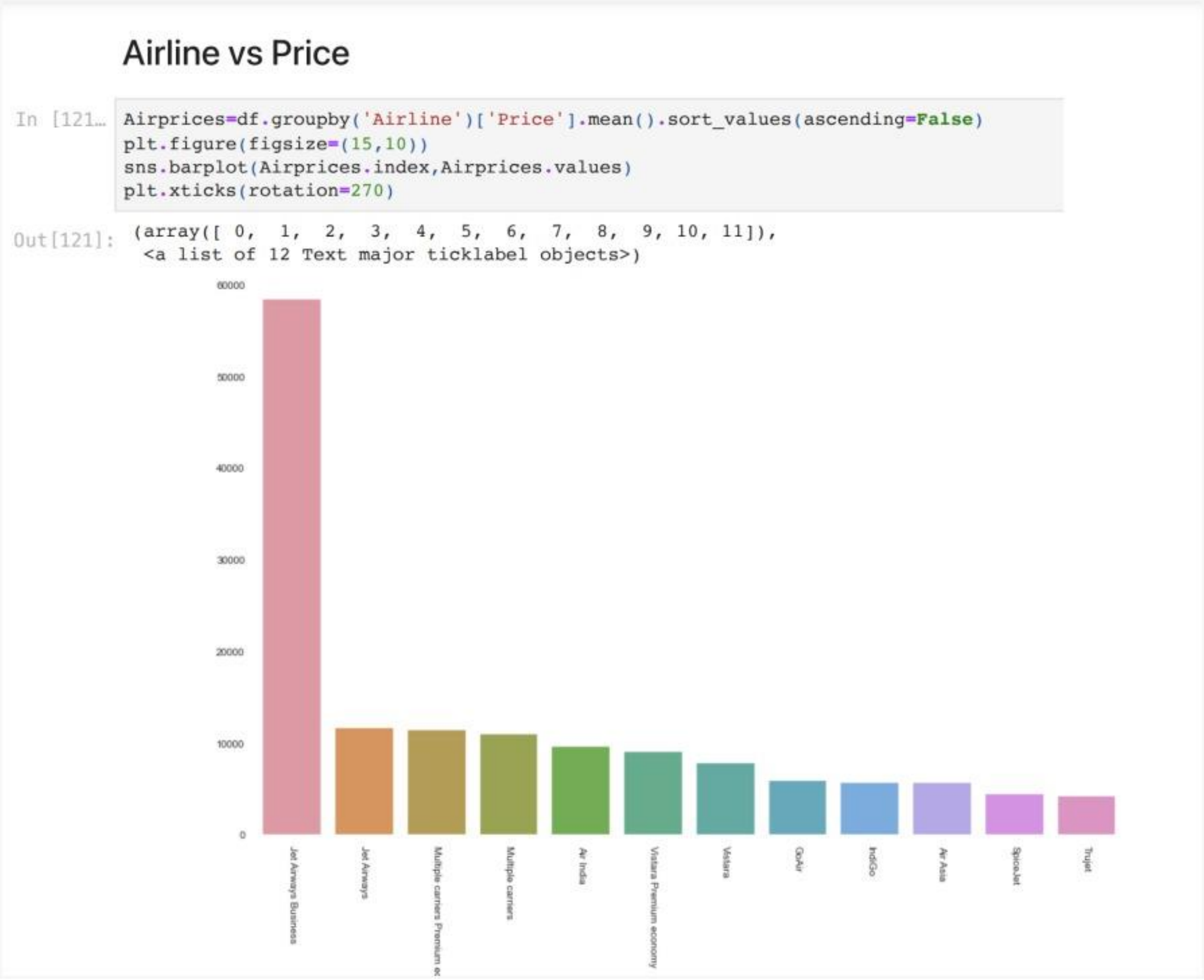
```
In [117…  #we can now drop the 'Duration'

          df.drop(["Duration"],axis=1,inplace=True)
```

```
In [118…  df.head(3)
```

5.4 Data Visualization

The next activity was to perform data visualization in order to have a better visual understanding of my dataset. I then started with visualizing the various airlines and their pricing components and realised that Jet Airways Business has the highest price with Trujet having the lowest



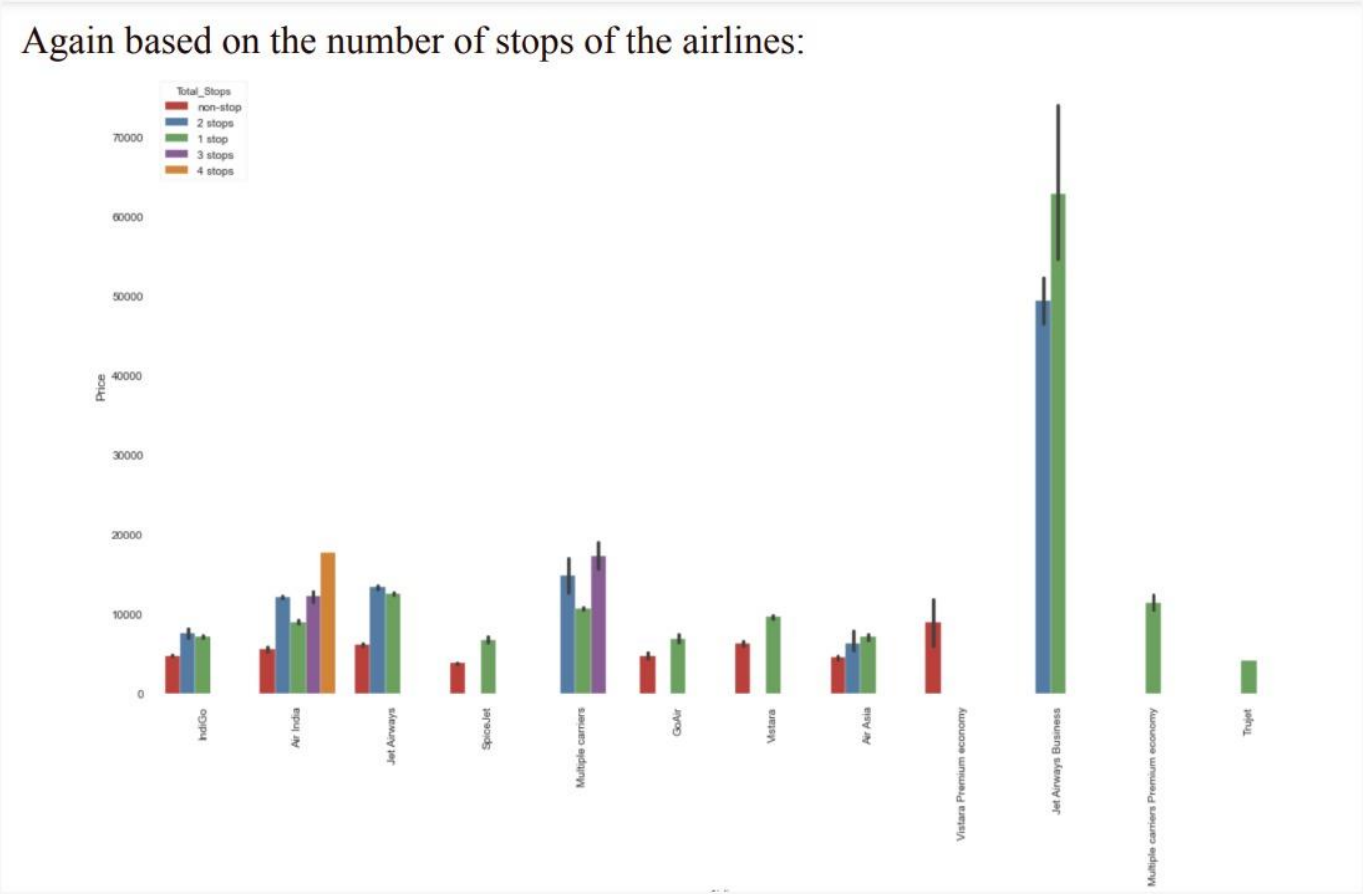
Airline vs Price

```
In [121…  Airprices=df.groupby('Airline')['Price'].mean().sort_values(ascending=False)
          plt.figure(figsize=(15,10))
          sns.barplot(Airprices.index,Airprices.values)
          plt.xticks(rotation=270)
```

```
Out[121]:  (array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11]),
            <a list of 12 Text major ticklabel objects>)
```

Fig 7: visualization of Airling Vs Pricing

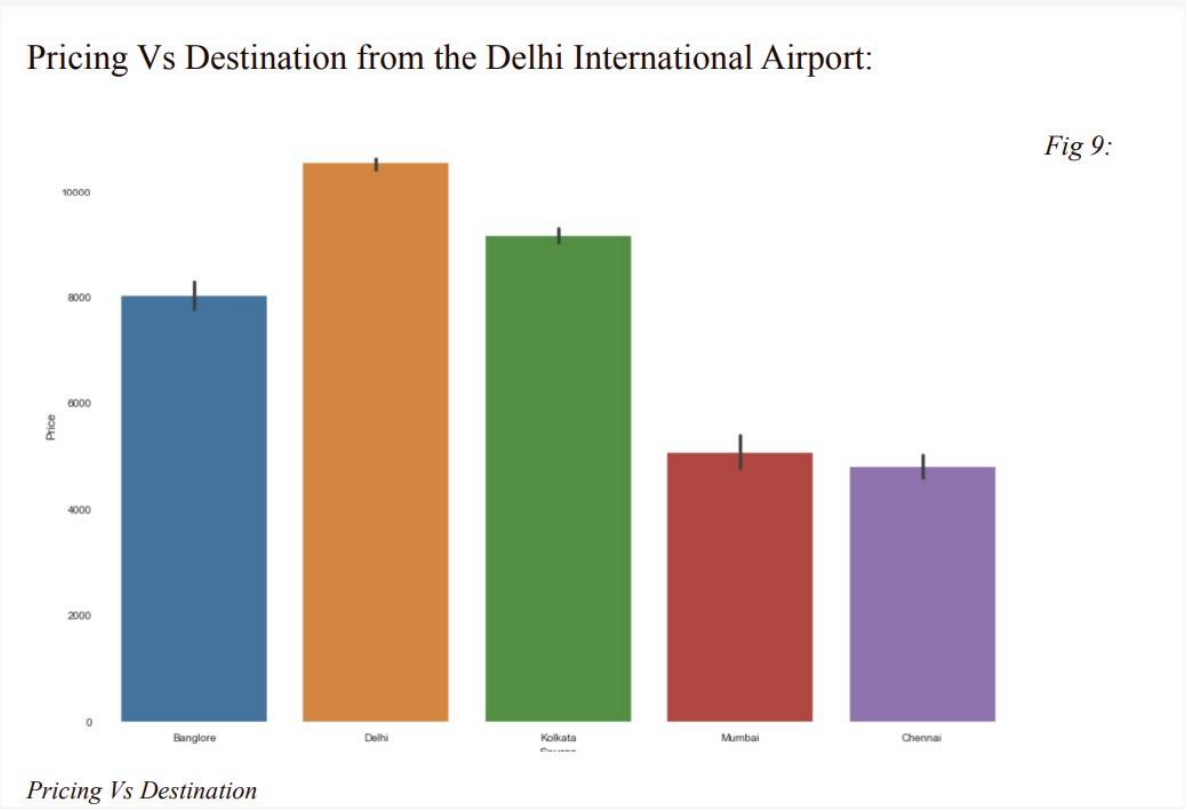
Again based on the number of stops of the airlines:



Again based on the number of stops of the airlines:

Fig 8: number of stops

One stop and two stops Jet Airways Business is having the highest price.



Pricing Vs Destination from the Delhi International Airport:

Fig 9:

Pricing Vs Destination

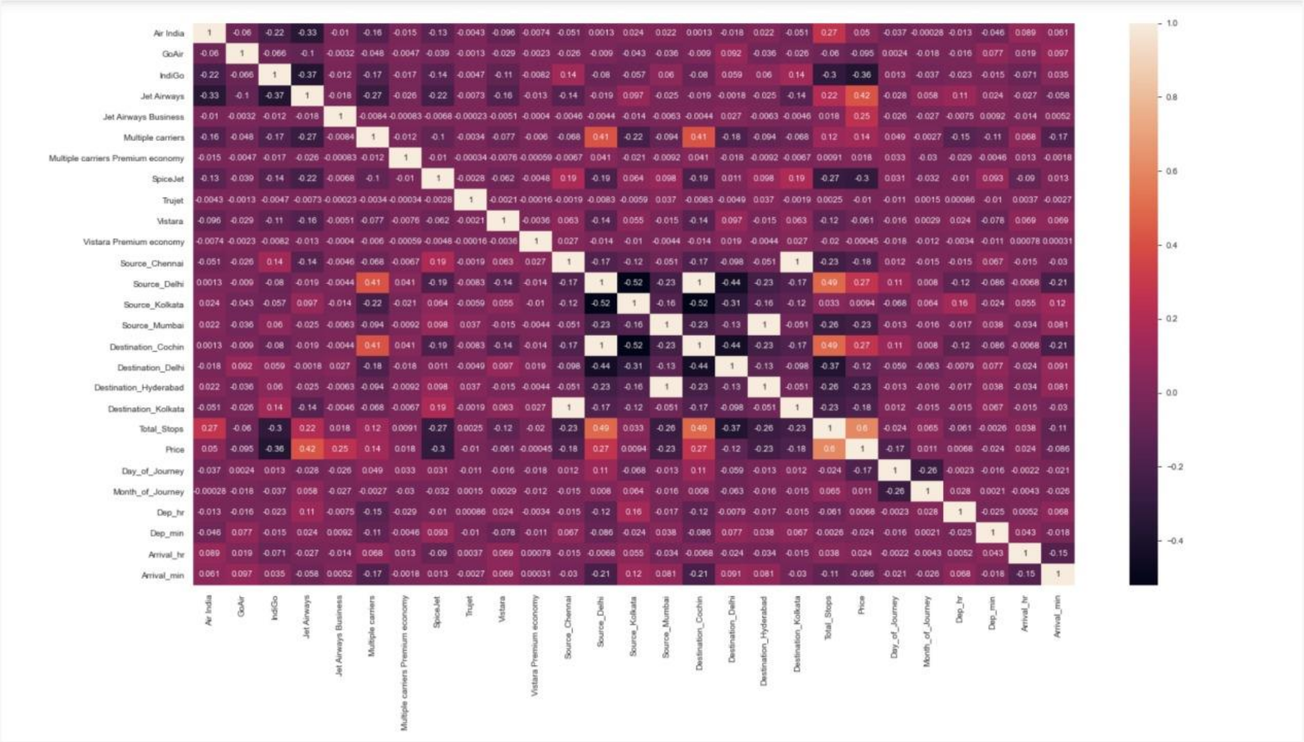


Fig 9: Pricing Vs Destination

6 Model Selection And Training:

ExtraTreesRegressor: This model achieved an R-squared (R^2) score of 0.76. R-squared is a statistical measure that represents the proportion of the variance in the dependent variable (in this case, flight prices) that is explained by the independent variables (features) in the model. An R-squared value closer to 1 indicates a better fit of the model to the data.

RandomForestRegressor: Initially, this model achieved an R-squared score of 0.79. However, the performance improved slightly to 0.80 after hyperparameter tuning. Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the average prediction of the individual trees.

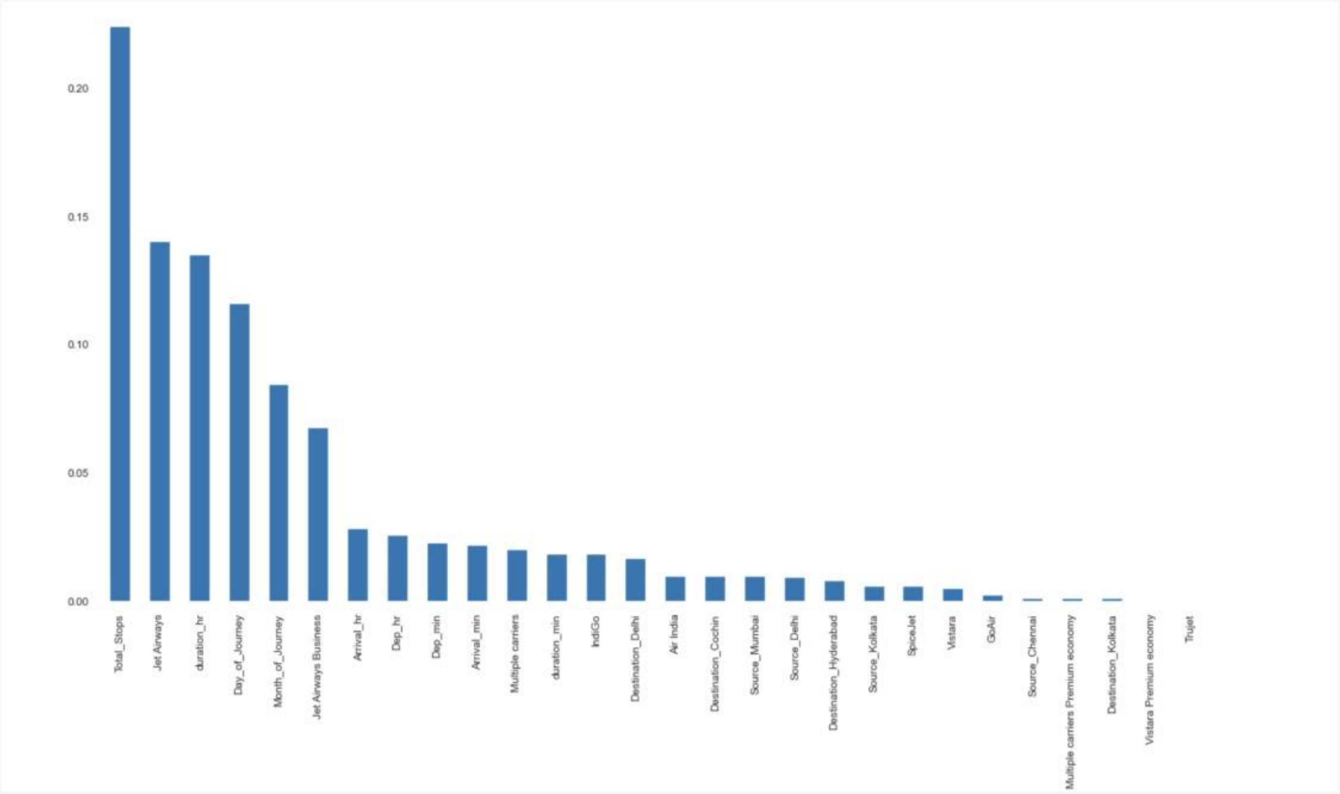
CatBoostRegressor: This model outperformed the others with an R-squared score of 0.83. CatBoost is a gradient boosting algorithm that is known for its ability to handle categorical features effectively without the need for extensive preprocessing.

LGBMRegressor: This model achieved an R-squared score of 0.80. LightGBM (LGBM) is another gradient boosting framework known for its high efficiency and fast training speed.

XGBoost: This model achieved an R-squared score of 0.82. XGBoost is an optimized gradient boosting algorithm that is widely used for regression and classification tasks due to its high performance and scalability.

6.1 Feature Importance:

To know which of the 28 features are contributing the most in making a good prediction of the flight prices, I checked the feature importance of all the features and the results is shown in fig 10 below:



6.2 Hyperparameter Tuning:

Hyperparameter tuning is a crucial step in optimizing the performance of machine learning models. In this project, we employed the RandomizedSearchCV method for hyperparameter tuning to enhance the predictive accuracy and generalization capabilities of our flight price prediction model.

RandomizedSearchCV is a technique that randomly selects a set of hyperparameters from a predefined search space and evaluates the model's performance using cross-validation. This approach is particularly effective when the hyperparameter space is large, as it allows for a more efficient exploration of the parameter space compared to traditional grid search methods.

The hyperparameters tuned in our model included parameters specific to the XGBoost algorithm, such as learning rate, maximum depth of trees, minimum child weight, subsample ratio, and regularization parameters. By varying these hyperparameters within specified ranges, RandomizedSearchCV systematically explores different combinations to identify the optimal set that maximizes the model's performance metrics.

During the hyperparameter tuning process, RandomizedSearchCV conducts multiple iterations of model training and evaluation, adjusting the hyperparameters based on the observed performance results. This iterative approach helps to refine the model and find the combination of hyperparameters that yields the best performance on the validation data.

By leveraging RandomizedSearchCV for hyperparameter tuning, we were able to fine-tune our XGBoost model efficiently and effectively. This optimization process ultimately resulted in a model with improved predictive accuracy, enabling more accurate and reliable predictions of flight prices for consumers across India.

7 Model Evaluation:

After the data preprocessing stage and doing all the necessary exploratory data analysis(EDA), I moved to the model building part. At this stage, since my problem is a regression problem, I choose 5 regression algorithms in building and optimizing my machine learning model. Among the 5 models built, XGBoost quite outperformed the other algorithms with an accuracy of 82% hence I choose that as my base model.

Model Name	**R2 Score**
ExtraTreesRegressor	0.76
RandomForestRegressor	0.79
RandomForestRegressor (with Hyperparameter Tuning)	0.80
CatBoost	0.81
LGBMRegressor	0.80
XGBoost	0.82

8 Conclusion:

This project is an eye opener and helping Indian citizens to better make decisions was a great achievement. Overall the data analysis and machine learning model building was a great lesson to go through. The data extraction part was the most part that I learnt the most and hope to learn more in further projects.