

PROJECT REPORT

"Flight Price Prediction"

Submitted By:

ABHISHEK KUMAR BATCH -29

ACKNOWLEDGMENT

I would like to express my special gratitude to "Flip Robo Technologies" team, who has given me this opportunity to deal with a beautiful dataset related to flight dataset and it has helped me to improve my analyzation skills. And special thanks to our SME – **SWATANK MISHRA SIR** for his useful guidance.

A huge thanks to my academic team "<u>Data trained</u>" who are the reason behind what I am today. Last but not least my parents who have been my backbone in every step of my life. And also thank you for many other persons who has helped me directly or indirectly to complete the project.

ABSTRACT

Travelling through flights has become an integral part of today's lifestyle as more and more people are opting for faster travelling options. The flight ticket prices increase or decrease every now and then depending on various factors like timing of the flights, destination, duration of flights. various occasions such as vacations or festive season. Therefore, having some basic idea of the flight fares before planning the trip will surely help many people save money and time. The focus of this project is developing machine learning models that can accurately predict the flight price based on its features, in order to make informed purchases. We implement and evaluate various learning methods on a dataset consisting of the flight prices of different airline. We will compare the performance of various machine learning algorithms like XGB Regressor, Gradient Boosting Regressor, Bagging Regressor, Decision tree Regressor and choose the best out of it. Depending on various parameters we will determine the flight price. Regression Algorithms are used because they provide us with continuous value as an output and not a categorized value because of which it will be possible to predict the actual price.

Introduction:

Anyone who has booked a flight ticket knows how unexpectedly the prices vary. 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:

- 1. Time of purchase patterns (making sure last-minute purchases are expensive)
- 2. 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). So, we have to work on a project where we have to collect data of flight fares with other features and work to make a model to predict fares of flights.

Conceptual Background:

From the customer point of view, determining the minimum price or the best time to buy a ticket is the key issue. The concept of "tickets bought in advance are cheaper" is no longer working. It is possible that customers who bought a ticket earlier pay more than those who bought the same ticket later. Moreover, early purchasing implies a risk of commitment to a specific schedule that may need to be changed usually for a fee. The ticket price may be affected by several factors thus may change continuously. To address this, various studies were conducted to support the customer in determining an optimal ticket purchase time and ticket price prediction.

Review of Literature:

In this report, we discuss various applications and methods which inspired us to build our project. We did a background survey regarding the basic ideas of our project and used those ideas for the collection of information like the technological stack, algorithms, and shortcomings of our project which led us to build a better project. Paytm is a web platform where buyer can buy their tickets. It is a website with a simplified user interface which asks seller parameters like source, destination, date of journey and class (economy). These allow the web model to run certain algorithms on given parameters and predict the price. It is very difficult for the customer to purchase a flight ticket at the minimum price. For this several techniques are used to obtain the day at which the price of air ticket will be minimum. Most of these techniques are using sophisticated artificial intelligence (AI) research known as Machine Learning

Motivation:

Travelling through flights has become an integral part of today's lifestyle as more and more people are opting for faster travelling options. The flight ticket prices increase or decrease every now and then depending on various factors like timing of the flights, destination, duration of flights. various occasions such as vacations or festive season. Therefore, having some basic idea of the flight fares before planning the trip will surely help many people save money and time. In the proposed system a predictive model will be created by applying machine learning algorithms to the collected historical data of flights. This system will give people the idea about the trends that prices follow and also provide a predicted price value which they can refer to before booking their flight tickets to save money. This kind of system or service can be provided to the customers by flight booking companies which will help the customers to book their tickets accordingly.

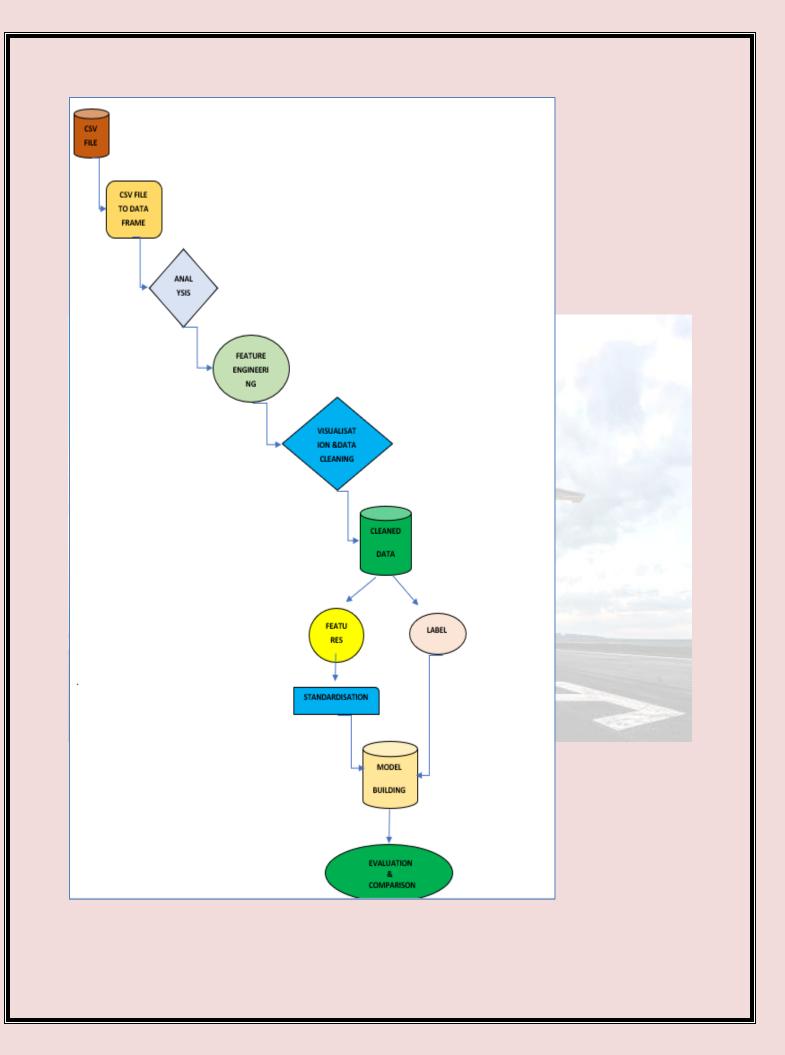
Description of dataset

- * Price: it is the label column.
- Flight name: it is the name of airline.
- ❖ Departure_time: The time at which flight leaves for the destination.
- Arrival time: The time at which flight arrives at destination.
- ❖ Source: It is starting point of journey.
- Destination: It is the ending point of the journey.
- ❖ Duration: it is the total time taken during the journey.
- * No of stop: it is the halt taken by flight during the journey.
- Date: It is date at which journey starts.

The task which is assigned is to fetch flight price data from a website and build a required model for that, I have used Paytm website for fetching the data as its URL is easy to modify and we can fetch data easily. The dataset consists of 8277rows and 10 attributes including target column price. The above mention features helped me in building machine learning model and predicting prices.

Architecture of dataset





Data Pre Processing

Data processing and feature engineering are crucial in machine learning to build a prediction model. Furthermore, a model cannot be made without some data processing. For instance, as shown in the experiment, the model could not be trained before handling the missing values and converting the text in the dataset into numerical values. Hence, from the experiment, we saw that preprocessing the data does improve the prediction accuracy.

- info: it is used to give info about not null value and data-type of features. here
 datatype is (object) which is been taken care and converted to float and int
 datatype
- Describe: The describe method is used for calculating some statistical data like percentile, mean and std of the numerical values of the Series or Data Frame. It analyses both numeric and object series and also the Data Frame column sets of mixed data types.it also give info about distribution of data.

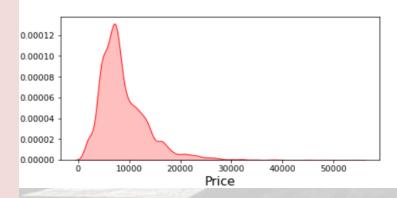
#checking basic info
df.describe()

		Unnamed: 0	Price
	count	8277.000000	8277.000000
	mean	4138.000000	7995.679957
	std	2389.508422	3999.594007
	min	0.000000	1694.000000
	25%	2069.000000	5177.000000
	50%	4138.000000	7161.000000
	75%	6207.000000	9744.000000
	max	8276.000000	33894.000000

Unique: This function is used to check the unique value in dataset

```
#checking unique in dataset
df.nunique().sort_values()
Source
Destination
                      4
No of stop
                      5
                      6
Flight name
                     9
Date
Departure time
                    197
Arrival time
                    242
                    360
Duration
Price
                   1761
Unnamed: 0
                   8277
dtype: int64
```

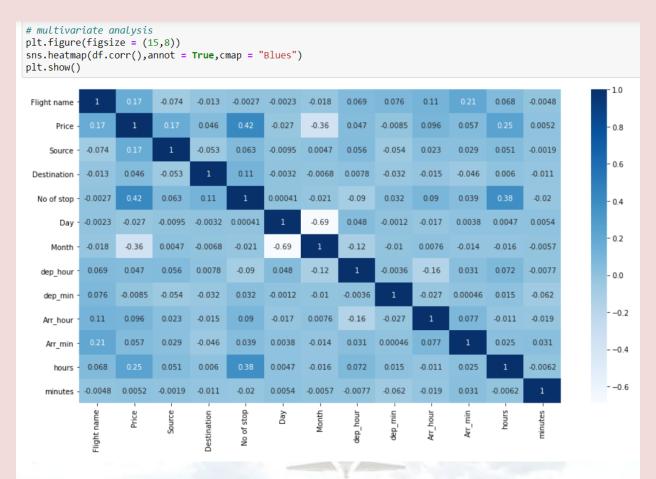
Univariate analysis: Univariate analysis is the simplest form of data analysis where the data being analysed contains only one variable. Since it's a single variable it doesn't deal with causes or relationships. The main purpose of univariate analysis is to describe the data and find patterns that exist within it. Here we can see from the graph, that price column is right skewed but it is our target column so we keep it as it.



Bivariate analysis: Bivariate analysis is used analyse relationship between feature and label and between two different variables. With the help of line plot, we can see the relationship between price and how it is varying with date. Last minute flight is costly as well as flight which is near some festive occasion is costly.



Multivariate analysis: Multivariate analysis is done on three or more than three variables. Through multivariate analysis we try to find out correlation of different feature with target variable as well as among themselves. I have used heatmap for multivariate analysis. We don't find any multicollinearity problem, only we can see strong correlation between day and month.



Feature Engineering & assumptions: Feature engineering involves extract features from raw dataset along with the use of domain knowledge. Feature engineering is useful to improve the performance of machine learning algorithms. Here we splitting date in day and month.

```
#splitting date into date and month
df["Date"]= pd.to_datetime(df["Date"])
df["Day"]=df["Date"].dt.day
df["Month"]=df["Date"].dt.month
```

Encoding: manually encoding certain column to make it meaningful and which is helping in building a better machine learning model with good accuracy.

```
# replacing non with 0 to make it a discrete column as mode
df["No of stop"] = df["No of stop"].replace("Non",0)

#chaning datatype of int
df["No of stop"]= df["No of stop"].astype("int64")

#replacing source manually with 1 and 2
df["Source"] = df["Source"].replace("Bengaluru",1)
df["Source"] = df["Source"].replace("Kolkata",2)
```

Time Delta: this function is used to split duration into hours and minute so that it is converted into numerical column as machine learning model only understands numerical data it does not understand text.

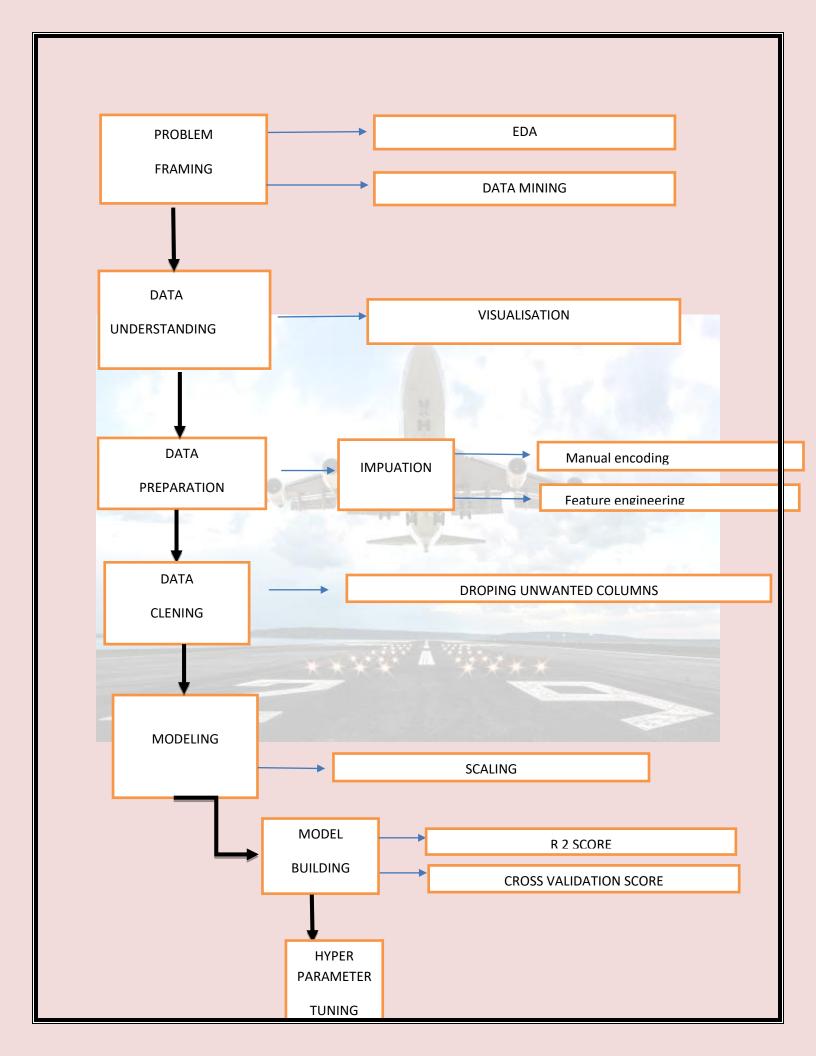
```
# splitting duration into and hour and min in seperate column for better understanding
s=pd.to_timedelta(df['Duration'])
df['hours']=s.dt.components['hours']
df['minutes']=s.dt.components['minutes']
```

Detection of Outliers: An outlier is an extremely high or extremely low-value value in the data set that tend to impact model performance. It is important to identify and take care of outliers to improve the model accuracy. The box plot is useful graphical display for describing the behavior of the data in the middle as well as the ends of distribution.it works on interquartile method to determine outlier.in this dataset only price column have outlier but is our target column so we don't perform any operation on target column so as keep it as.



Skewness: skewness is also an important factor which affects the model accuracy. In skewed data tail region act as an outlier for statistical model and may affect the model accuracy. In this dataset mostly column is categorical and skewness is only present in label column and we don't take care of skewness of label as well as categorical columns.

Model development and evaluation



Above is the architecture which shows the important steps involve in data preprocessing and model building. Firstly, I analysed the dataset then I found out that datatype of certain columns is not correct so I corrected that and with the help of feature engineering I have created numerical column out of object column as our machine learning model will not understand text, then for understanding I have used plots to visualise and understand data. After imputation comes data cleaning which is done by detecting outliers, checking skewness and removing unwanted columns which does not add value to model accuracy. After data cleaning splitting of data into test and train and then doing normalization of data so that it does not get biased toward a particular feature. Finally comes model building, I have used 4 classifier algorithm () which fits best in this dataset and predict good accuracy. After finding my best model with the help of accuracy and cross validation score comes hyper parameter tuning for best fit model.

Hardware requirement: -

- Minimum core i5 or higher
- Minimum8gb of ram

Software and

Tools required:

- Python is widely used in scientific and numeric computing:
- SciPy is a collection of packages for mathematics, science, and engineering.
- Pandas is a data analysis and modelling library.

Modules or library required for project data analysis and visualization:

- Pandas for data analysis and import
- NumPy to perform Mathematical operation
- Seaborn and matplotlib to data visualization
- ❖ Scikit-learn: All the models, metrics and feature selection etc are present inside of that module. We import from this library according to our need.

Model Building:

➤ I'm using the four different algorithms for model building:

Linear Regression:

```
from sklearn.linear_model import LinearRegression

LR=LinearRegression()
LR.fit(x_train,y_train)
print(LR.score(x_train,y_train))
LR_predict=LR.predict(x_test)

0.5008495384166481

print('MSE:',mean_squared_error(LR_predict,y_test))
print('MAE:',mean_absolute_error(LR_predict,y_test))
print('r2_score:',r2_score(LR_predict,y_test))

MSE: 8351570.491692461
MAE: 2076.99427338833
r2_score: -0.03586573173065255
```

From above I had predicted the linear regression method.

Random Forest Regressor:

```
# from sklearn.ensemble import RandomForestRegressor

RF=RandomForestRegressor()
RF.fit(x_train,y_train)
print(RF.score(x_train,y_train))
RF_PRED=RF.predict(x_test)

0.9775218438972029

print('MSE:',mean_squared_error(RF_PRED,y_test))
print('MAE:',mean_absolute_error(RF_PRED,y_test))
print('r2_score:',r2_score(RF_PRED,y_test))

MSE: 3274888.247600907
MAE: 934.8969252793194
r2_score: 0.7516358510544029
```

Above I have use the random forest regression model.

Decision Tree Regressor:

```
from sklearn.tree import DecisionTreeRegressor

DTR=DecisionTreeRegressor()
DTR.fit(x_train,y_train)
print(DTR.score(x_train,y_train))
DTR_PRED=DTR.predict(x_test)

0.9993451530366741

: print('MSE:',mean_squared_error(DTR_PRED,y_test))
print('MAE:',mean_absolute_error(DTR_PRED,y_test))
print('r2_score:',r2_score(DTR_PRED,y_test))

MSE: 4986415.538659208
MAE: 970.8929306487696
r2_score: 0.6907381694765916
```

➤ Above I have use the decision tree regressor model to predicted the values.

GradientBoosting Regressor:

```
from sklearn.ensemble import GradientBoostingRegressor

GBR=GradientBoostingRegressor()
GBR.fit(x_train,y_train)
print(GBR.score(x_train,y_train))
GBR_PRED=GBR.predict(x_test)

0.7212786031513447

print('MSE:',mean_squared_error(GBR_PRED,y_test))
print('MAE:',mean_absolute_error(GBR_PRED,y_test))
print('r2_score:',r2_score(GBR_PRED,y_test))

MSE: 5336026.923547953
MAE: 1557.2904131847945
r2_score: 0.45707116078525123
```

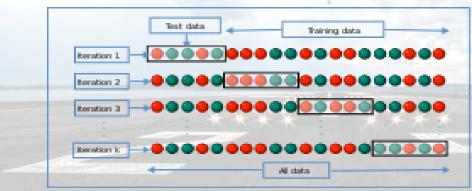
➤ Above I use the Gradient boosting Regressor to predict the value .

Standard Scaler: Scaling of Features is an essential step in modelling the algorithms with the datasets. So, the data obtained contains features of various dimensions and scales altogether. Different scales of the data features affect the modelling of a dataset adversely. It leads to a biased outcome of predictions in

terms of misclassification error and accuracy rates. Thus, it is necessary to Scale the data prior to modelling. StandardScaler removes the mean **and scales each feature/variable to unit variance**. This operation is performed feature-wise in an independent way.

Cross Validation Score:

Cross-validation is a statistical method used to estimate the skill of machine learning models. It is commonly used in applied machine learning to compare and select a model for a given predictive modelling problem because it is easy to understand, easy to implement, and results in skill estimates that generally have a lower bias than other methods. The procedure has a single parameter called k that refers to the number of groups that a given data sample is to be split into. As such, the procedure is often called k-fold cross-validation. When a specific value for k is chosen, it may be used in place of k in the reference to the model, such as k=10 becoming 10-fold cross-validation.



R-2 Score:

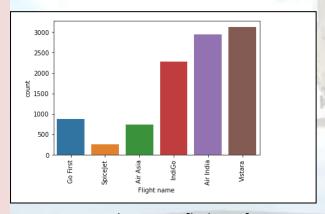
The most common interpretation of r-2 score is how well the regression model fits the observed data. for ex an r-2 squared of 70% revels that 70% of the data fit the regression model. Generally, a higher r-2 score indicates a better fit for the model.

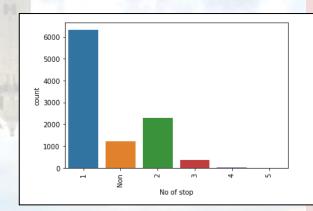
Visualisation:

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.

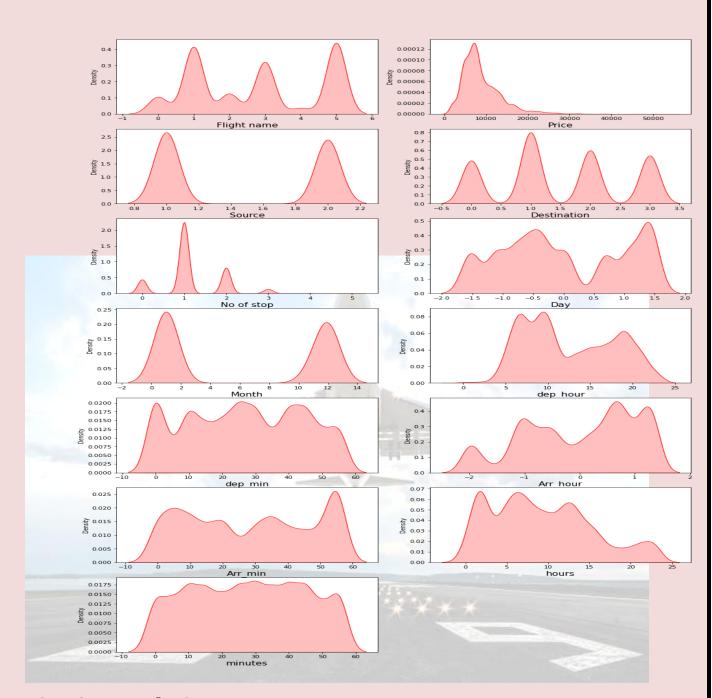
Univariate:

For univariate analysis I have used kde plot to analyse each column, it helps me in visualising skewness in the dataset as we can see price column right skewed.as we



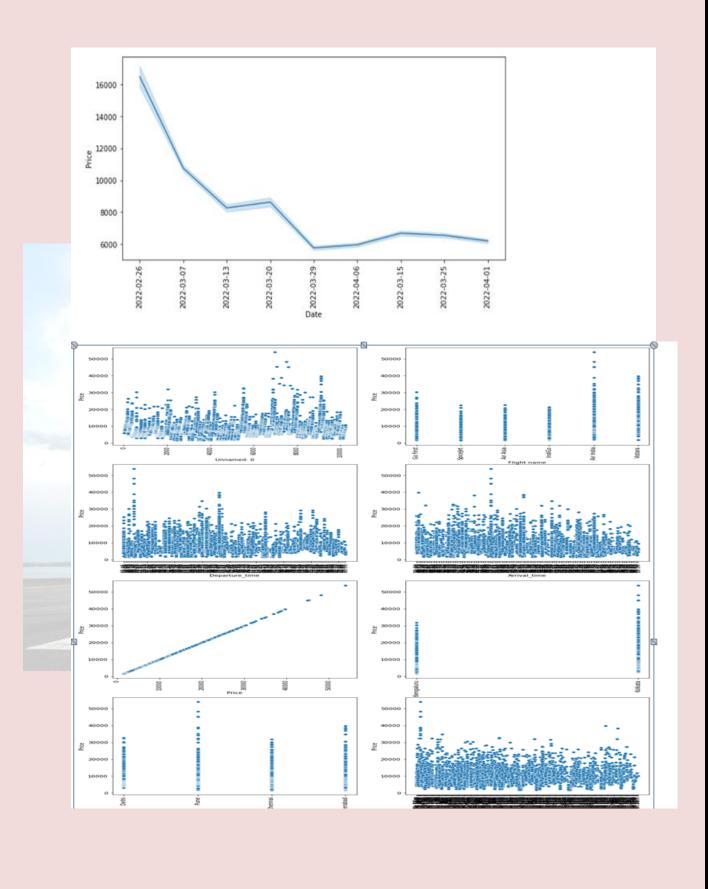


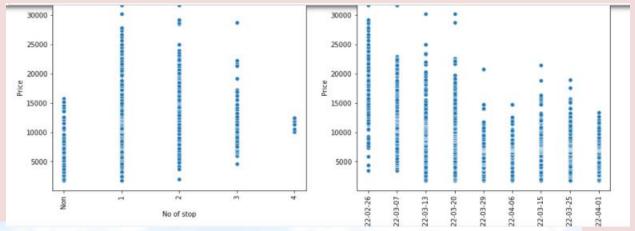
can in count plot most flight is from Vistara and mostly flight is one stop flight.



Bivariate analysis:

Bivariate analysis is done to analyse how each column is relating with the target(label) column. We can clearly see that last minute flight is expensive, also flight with maximum stop is expensive. Flight to pune is more expensive as compare to other cities. Air India flight have more price as compare to other flights.





Multivariate analysis:

Multivariate analysis is done to find out relationship between each column along with the label column. After going through heat map, I realise that there is no strong correlation of any column among themselves as well with the label which means that there is no problem of multicollinearity. As day is showing some correlation with month.



RESULT:

Many machine learning algorithms are used to predict. However, the prediction

accuracy of these algorithms depends heavily on the given data when training the model. If the data is in bad shape, the model will be overfitted and inefficient, which means that data pre-processing is an important part of this experiment and will affect the final results. Thus, multiple combinations of pre-processing methods need to be tested before getting the data ready to be used in training. After analyzing every model XGB Regressor shows good accuracy and cv with least difference and on doing hyper parameter tuning it accuracy reaches to 89%. If we compare the flight prices of the month of January, we can see that price of the flight changes frequently near the date of November. The flight prices increase in large increments near the departure dates. But increases and decreases in very small increments afterwards. Similarly, if we compare the flight prices in the month of November and December, we can see that flight prices increases. Therefore, we concluded that the flight prices tend to decrease over time, sometimes they show sudden increases but most of the time they tend to decrease.

CONCLUSION:

Flight price prediction can be a challenging task due to the high number of attributes that should be considered for the accurate prediction. The major step in the prediction process is collection and preprocessing of the data which we have successfully done using different machine learning algorithms like a Decision tree, Bagging, Gradient Boosting and XGB so it's clear that XGB have more accuracy in prediction when compared to the others and also my research

provides to find the attributes contribution in prediction. So, I would believe this research will be helpful for both the company and people, the future works are stated below Every system and new software technology can help in the future to predict the prices. Although, this model has achieved astonishing performance in flight price prediction problem our aim for the future research is to test this model to work successfully with various data sets.

FINDING:

The factors that have been studied in this study has a weak correlation with the label except some. Hence, by adding more features to the local dataset that affect the price prediction, such as meals, safety and many more.

LIMITATION:

As per my understanding and study of dataset I presented the best model with good accuracy, this is the best model until someone came with an extraordinary approach and technique. Moreover, this study will not cover all classification algorithms, instead, it is focused on the chosen algorithm, starting from the basic classification techniques to the advanced ones.

FUTURE WORK:

In future this machine learning model may bind with various website which can provide real time data for price prediction. Also, we may add large historical data of flight price which can help to improve accuracy of the

machine learning model. We can build an android app as user interface for interacting with user. For better performance, we plan to judiciously design deep learning network structures, use adaptive learning rates and train on clusters of data rather than the whole dataset.

REFERENCE:

