A Case study of Flight Price Prediction

Submitted by:

Sharad Yadav

ACKNOWLEDGMENT

Data fetch from makemytrip web by selenium, analysis done by Sharad Yadav under guidance of Mr. Keshav Bansal, articles content written by myself regarding Flight price prediction.

INTRODUCTION

- **Business Problem Framing**
- 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, I have extracted data with help of selenium prices of flight tickets for various airlines the months of October 2021 New Delhi to Bengaluru c, using which we aim to build a model which predicts the prices of the flights using various input features

- **Conceptual Background of the Domain Problem**

  Building Flight price prediction model that can help to understand how prices vary with the variables. Then we can accordingly offer better plan to customer as well as client.

- **Review of Literature**

  In this dataset, there are 225 observations with 10explanatory variables describing (almost) every aspect of Flight name , distance ,source destination fair etc. Descriptive analysis and quantitative analysis will use subsets of it depending on models.

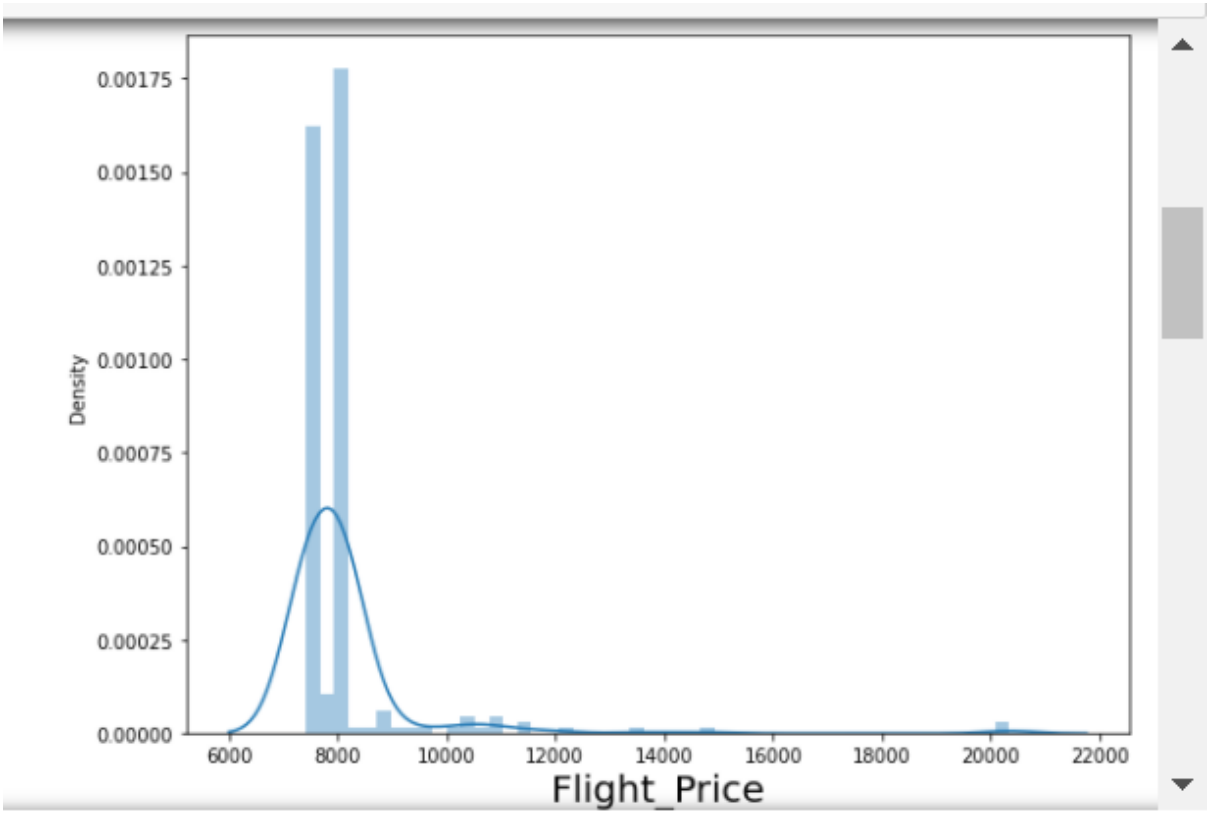
- **Motivation for the Problem Undertaken**

  The tourism industry is changing fast and this is attracting a lot more travelers each year. The airline industry is considered as one of the most sophisticated industry in using complex pricing strategies. Now-a-days flight prices are quite unpredictable. The ticket prices change frequently. Customers are seeking to get the lowest price for their ticket, while airline companies are trying to keep their overall revenue as high as possible.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

  In this project we have to predict flight price after plotting distribution plot of sales price data is right skewed which sold at higher price than aver



  average price.

- **Data Sources and their formats**

- Dataset is in csv file, to read that data I have to used pandas library to read file further describe method to analysis get overview of data distribution,datainfo to identify object integer float data types

```ipynb
#import imp library
import pandas as pd
import numpy as np
import seaborn as sns
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split ,GridSearchCV
from sklearn.metrics import r2_score,mean_absolute_error,mean_squared_error
import statsmodels as sm
import matplotlib.pyplot as plt
from string import digits
import warnings
from pandas.plotting import scatter_matrix
import math
warnings.filterwarnings('ignore')
```

```ipynb
flight=pd.read_csv(r"C:\Users\INPshy\Desktop\DATA Science\data.csv")
flight.head()
```

']:

	Unnamed: 0	Flight_Name	Dep_Time	Arrival_Time	Source	Destination	Travel_Hour	Flight_Price
0	0	AirAsia	21:20	11:30	New Delhi	Bengaluru	14 h 10 m	₹ 7,423
1	1	AirAsia	19:25	11:30	New Delhi	Bengaluru	16 h 05 m	₹ 7,423
2	2	Go First	19:45	22:20	New Delhi	Bengaluru	02 h 35 m	₹ 7,424
3	3	Go First	18:50	00:30	New Delhi	Bengaluru	05 h 40 m	₹ 7,424
4	4	NaN	NaN	NaN	NaN	NaN	NaN	NaN

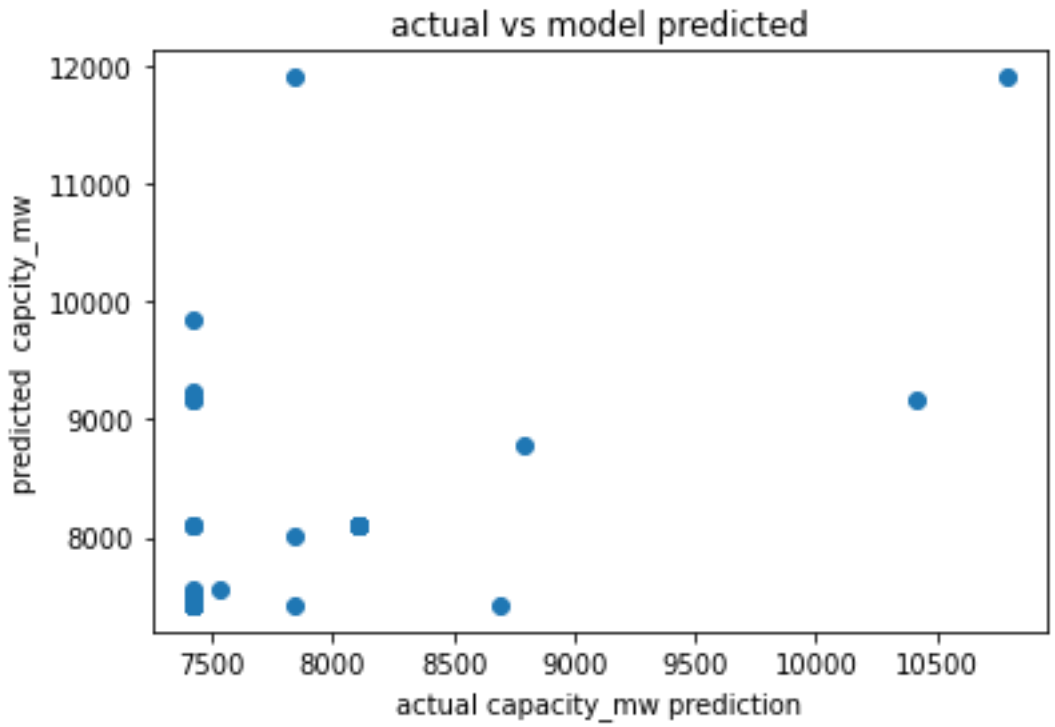
- **Data Preprocessing Done**
  First import important library then describe method to see data distribution ,check null values ,data types,datainfo, data.shape & conversion of all object data into numeric values which factors have more effects on flight fare i.e. noted down on notebook itself. Fill all missing values,encode object data with Ordinal encode after that distribution plot to check how data distributed ,box plot to check outliers in dataset, heatmap to check multicollnearity, scatter plot to check correlation between features ,correlation plot to check

how features have correlation with Price ,features selection ,train test split after that model building and tune the model and cross validation ,visualization of actual sale price vs predicted sale price with scatter plot on different model and GradientBoost model found best performer among all to save for future analysis of these dataset.

- **Data Inputs- Logic- Output Relationships**

  In this data set 10 columns including target variable i.e. flight Price ,after cleaning and pre-processing ,visualization of which features have strong positive and negative relationship with target variable

  Predicting Sale price is regression type problem ,building model I have predicted sale price and compare to actual sale price i.e. look like as shown in image below:-



- Hardware and Software Requirements and Tools Used

```
#import imp library
import pandas as pd
import numpy as np
import seaborn as sns
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split ,GridSearchCV
from sklearn.metrics import r2_score,mean_absolute_error,mean_squared_error
import statsmodels as sm
import matplotlib.pyplot as plt
from string import digits
import warnings
from pandas.plotting import scatter_matrix
import math
warnings.filterwarnings('ignore')
```

Screenshot of imported library used to build predictive model

Pandas library used to read csv file,pie chart analysis, distribution plot to check how data distributed, standardscaler to scale features traintestsplit dataset,sklearn.metrics used to check model accuracy and other parameters evaluation ,matplotlib used for visualization. Different model imported from sklearn libray to build model like Linear ,GradientBoost,decision tree,RandomForest regression.

Model/s Development and Evaluation

- **Identification of possible problem-solving approaches (methods)**

  First step to read csv file, check data shape, missing values,describe method to view data mean, median, mode, std, max data, data types, few columns are object type then it converted to numerical value with help of encoding techniques. After encoding plot distplot of each column to see how data distributed

	Flight_Name	Dep_Time	Arrival_Time	Source	Destination	Travel_Hour	Flight_Price
0	AirAsia	21:20	11:30	New Delhi	Bengaluru	14 h 10 m	7423.0
1	AirAsia	19:25	11:30	New Delhi	Bengaluru	16 h 05 m	7423.0
2	Go First	19:45	22:20	New Delhi	Bengaluru	02 h 35 m	7424.0
3	Go First	18:50	00:30	New Delhi	Bengaluru	05 h 40 m	7424.0
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Reading of csv file with help of pandas library

- **Testing of Identified Approaches (Algorithms)**
  1. from sklearn.linear_model import LinearRegression
  2. from sklearn.ensemble import Randomforest Regressor
  3. from sklearn.tree import DecisionTreeRegressor
  4. from sklearn.ensemble import GradientBoostingRegressor
- **Run and Evaluate selected models**

First model is Linear regression to predict Sale price of flight,linear regression model based on to find best fit line works on linear equation Y=mX+C

Linear Regression

```
:  1 from sklearn.linear_model import LinearRegression
```

```
:  1 Lr=LinearRegression()
   2 Lr.fit(x_train,y_train)
```

```
171]: LinearRegression()
```

```
:  1 y_pred=Lr.predict(x_test)
```

```
:  1 #train model score
   2 Lr.score(x_train,y_train)
```

```
173]: 0.0452938860174279
```

Mean absolute and square error is high so we need to tune the model with Ridge and Lasso after tuning the model

```
178]:
1 lasso_reg=Lasso(alpha)
2 lasso_reg.fit(x_train,y_train)

Out[178]: Lasso(alpha=18.026395497992908)

179]:
1 lasso_reg.score(x_train,y_train)

Out[179]: 0.04454281019540085
```

No improvement after parameter tuning

```
180]:
1 ridgecv=RidgeCV(alphas=np.arange(0.001,0.1,0.01),normalize=True)
2 ridgecv.fit(x_train,y_train)

Out[180]: RidgeCV(alphas=array([0.001, 0.011, 0.021, 0.031, 0.041, 0.051, 0.061, 0.071, 0.081,
       0.091]),
        normalize=True)

181]:
1 ridgecv.alpha_

Out[181]: 0.09099999999999998

182]:
1 ridge_model=Ridge(alpha=ridgecv.alpha_)
2 ridge_model.fit(x_train,y_train)

Out[182]: Ridge(alpha=0.09099999999999998)

183]:
1 ridge_model.score(x_train,y_train)

Out[183]: 0.045293875085881674
```

Linear model score after tuning and cross validation of model

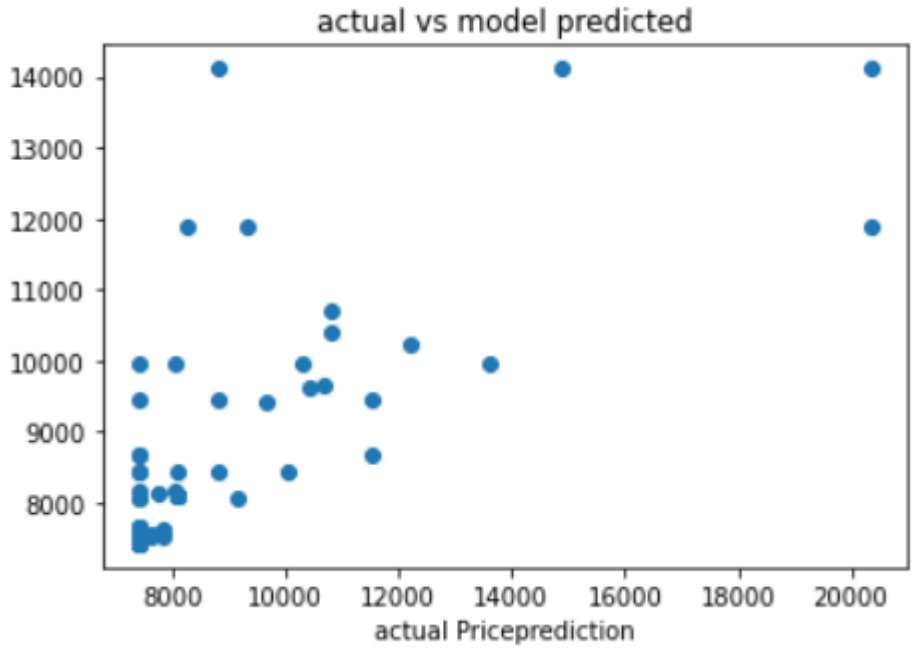
Next model is RandomForest Regressor which use import from sklearn library it actually work on principal ensemble technique boost the performance of decision tree.

RandomForestRegression Model

```
]:  1 from sklearn.ensemble import RandomForestRegressor

]:  1 rfr=RandomForestRegressor()
    2 rfr.fit(x_train,y_train)

[185]: RandomForestRegressor()

]:  1 y_pred=rfr.predict(x_train)

]:  1 #train data model score
    2 rfr.score(x_train,y_train)

[187]: 0.5590696012711919

]:  1 #test data model score
    2 rfr.score(x_test,y_test)

[188]: -0.2943313693174352
```

RandomForest regression Model

RandomForest R2 score is not impresive and when I plot scatter plot actual vs predicted model is also bad



Scatter plot of RandomForest actual vs predicted model

RandomForest score is 55% so I have to tune the model to achieve more accuracy of model with randomizedsearchCV

Hyperparameter Tuning

```
from sklearn.model_selection import RandomizedSearchCV
```

```
param_grid={'n_estimators':[100,130,150,170,180,200],
            'max_features':['auto','sqrt'],
            'max_depth':[5,8,10,12,14,16,18],
            'min_samples_split':[2,4,5,7,8,10,12],
            'min_samples_leaf':[1,2,4,6,10]}
```

```
grid_search=RandomizedSearchCV(rfr,param_distributions=param_grid,cv=5)
```

```
grid_search.fit(x_train,y_train)
```

```
96]: RandomizedSearchCV(cv=5, estimator=RandomForestRegressor(),
                        param_distributions={'max_depth': [5, 8, 10, 12, 14, 16, 18],
                                             'max_features': ['auto', 'sqrt'],
                                             'min_samples_leaf': [1, 2, 4, 6, 10],
                                             'min_samples_split': [2, 4, 5, 7, 8, 10,
                                                                   12],
                                             'n_estimators': [100, 130, 150, 170,
                                                              180, 200]})
```

```
                                                              180, 200]})
```

```
#find best parameters
grid_search.best_params_
```

```
197]: {'n_estimators': 180,
       'min_samples_split': 4,
       'min_samples_leaf': 10,
       'max_features': 'sqrt',
       'max_depth': 18}
```

```
#tune model with best parameters
rfr=RandomForestRegressor(n_estimators=100,min_samples_split=7,min_samples_leaf=10,max_features='sqrt',max_depth=10)
rfr.fit(x_train,y_train)
```

```
198]: RandomForestRegressor(max_depth=10, max_features='sqrt', min_samples_leaf=10,
                            min_samples_split=7)
```

```
#train data score post tuning
rfr.score(x_train,y_train)
```

```
199]: 0.18619795360978586
```

```
#test data score post tuning
rfr.score(x_test,y_test)
```

```
200]: -0.005238830208567968
```

RandomForest model score and cross validation score

Similarly I have also build KNN, decision tree & GradientBoosting regression model

DecisionTree regression model

Model performece is not so good like adaboost here is scatter plot

Post tuning Decision Tree regression Model scatter plot actual vs predicted

GradientBoost Regression model

```
from sklearn.ensemble import GradientBoostingRegressor
gbr=GradientBoostingRegressor()
```

```
gbr.fit(x_train,y_train)
```

```
GradientBoostingRegressor()
```

```
#check model score
gbr.score(x_train,y_train)
```

```
0.5780131013869461
```

```
param={'loss':['ls'],
    'learning_rate':[0.1,0.2,0.3,0.4],
    'n_estimators':[100,150,200,250],
    'subsample':[1.0,2,3,4,7],
    'criterion':['friedman_mse']}
```

```
grid_search=GridSearchCV(estimator=gbr,param_grid=param,cv=5)
```

```
grid_search.fit(x_train,y_train)
```

```
GridSearchCV(cv=5, estimator=GradientBoostingRegressor(),
             param_grid={'criterion': ['friedman_mse'],
                         'learning_rate': [0.1, 0.2, 0.3, 0.4], 'loss': ['ls'],
                         'n_estimators': [100, 150, 200, 250],
                         'subsample': [1.0, 2, 3, 4, 7]})
```

Gradientboost regression model score and cross validation score

Model score is 57% but cross val score tell that model have overfitting issue which we have to resolve by hyperparameter tuning

```
grid_search.best_params_
```

```
{'criterion': 'friedman_mse',
 'learning_rate': 0.1,
 'loss': 'ls',
 'n_estimators': 100,
 'subsample': 1.0}
```

```
gbr=GradientBoostingRegressor(criterion='friedman_mse',learning_rate=0.1, loss='ls',n_estimators=100,subsample=1)
```

```
gbr.fit(x_train,y_train)
```

```
GradientBoostingRegressor(subsample=1)
```

```
gbr.score(x_train,y_train)
```

```
0.5780131013869461
```

```
y_pred=gbr.predict(x_train)
```

```
cross_val_score(gbr,X_scaler,Y,cv=5).mean()
```

```
0.0034004897382907016
```

w score but among all modle cross val score is positive so i will save this model

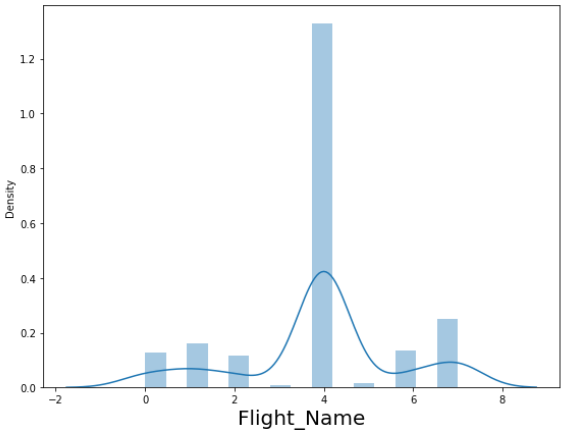
Model score is improve i.e. 57% now

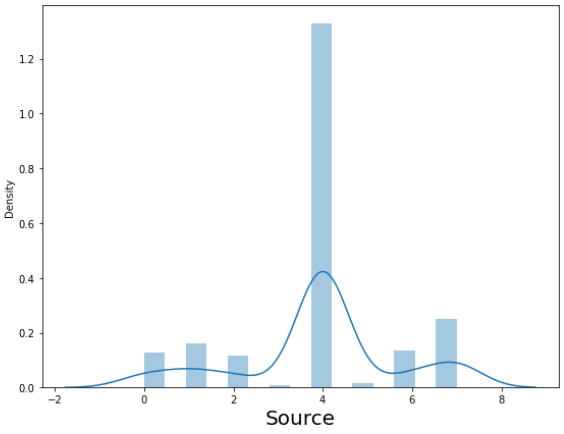
Cross validation of gradientboosting regression model post tuning is 56% it means model is not improved and scatter plot of actual vs predicted is just ok.

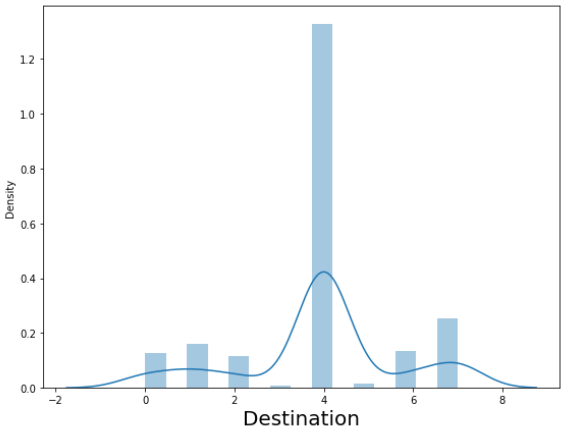
- **Key Metrics for success in solving problem under consideration**

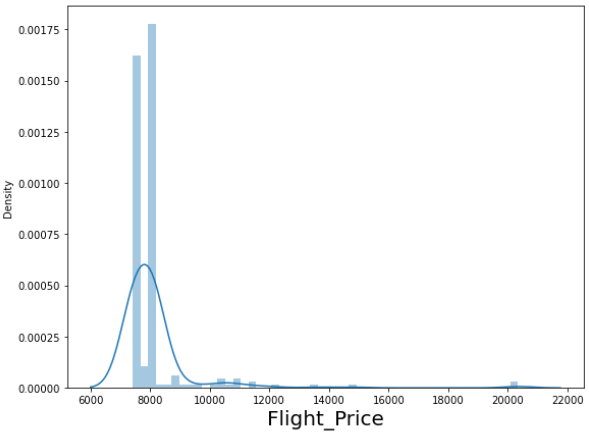
  R2 score ,mean squared error score , give idea about which model score performance is accurate ,cross validation score of each model test overfitting issue ,hyperparametr tuning boost model score ,scatter plot between actual vs predicted model.

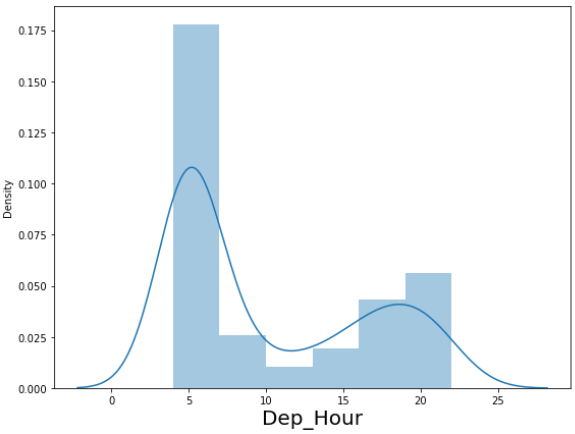
- **Visualizations**

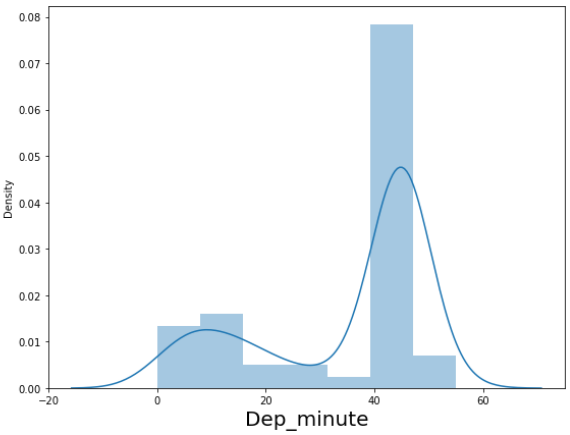

Density
Flight_Name

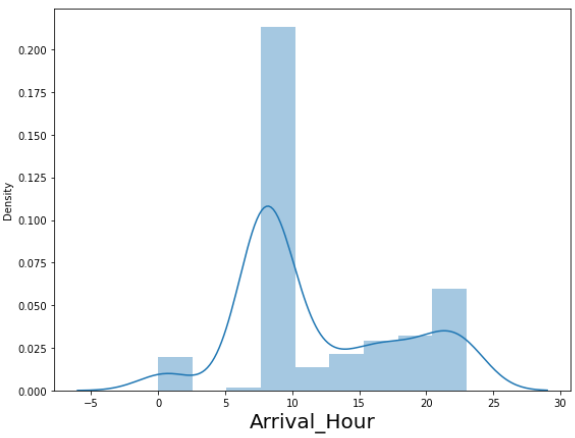

Density
Source

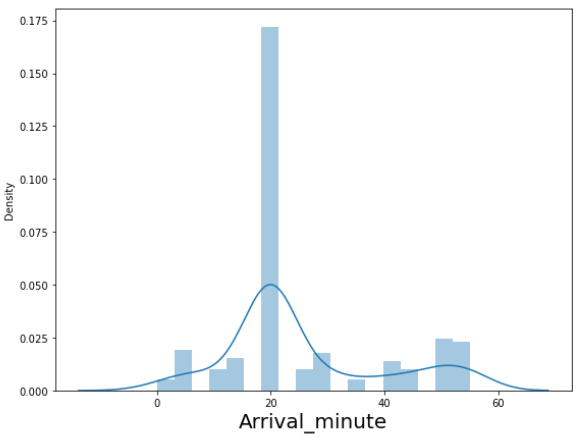

Density
Destination

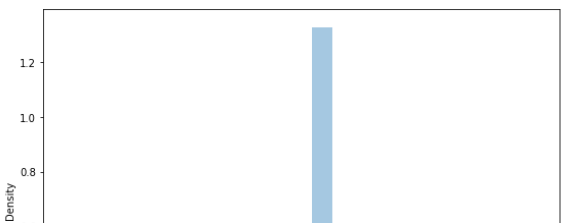

Density
Flight_Price


Density
Dep_Hour


Density
Dep_minute

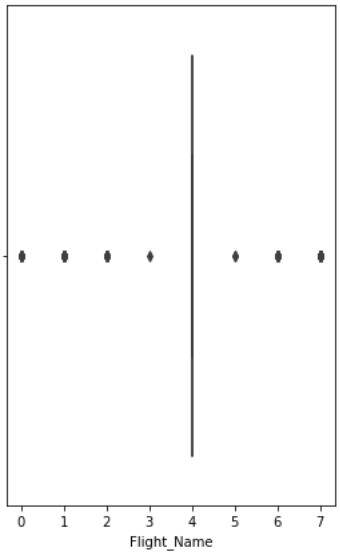

Density
Arrival_Hour

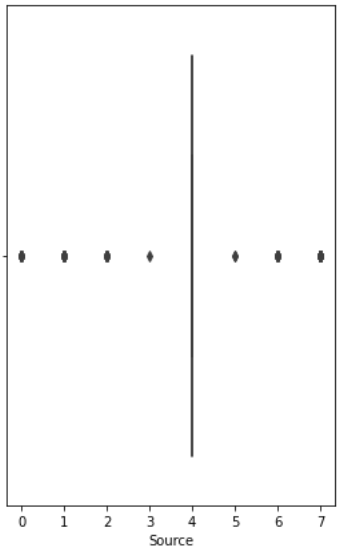

Density
Arrival_minute

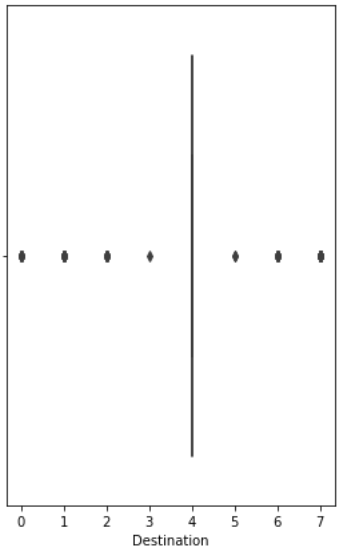

Density

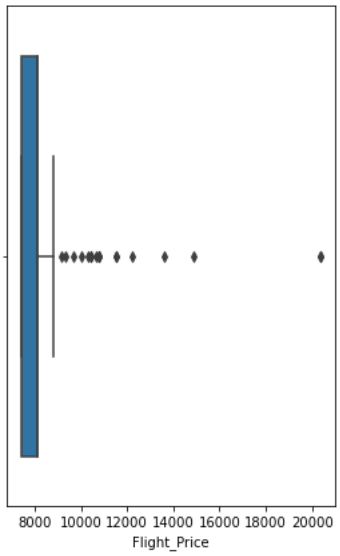

Density

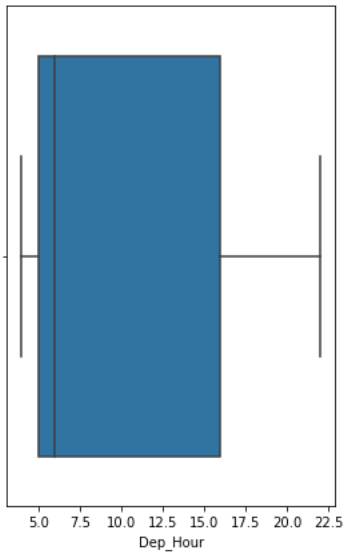
Distplot to visualization of data distribution

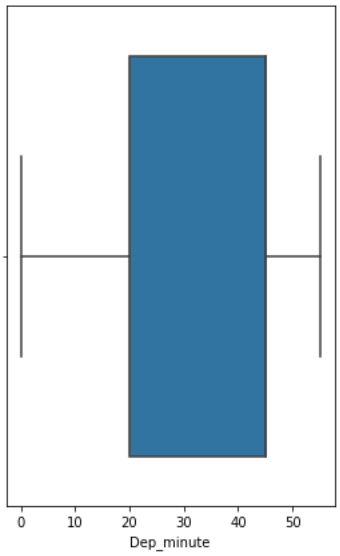

0 1 2 3 4 5 6 7
Flight_Name

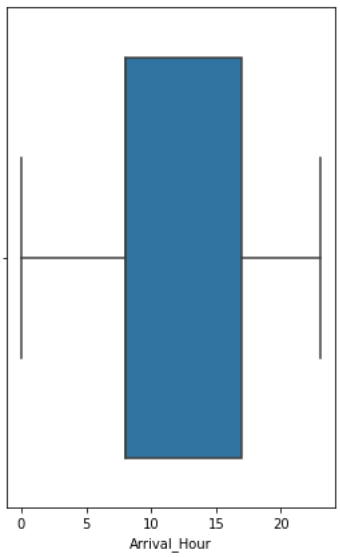

0 1 2 3 4 5 6 7
Source

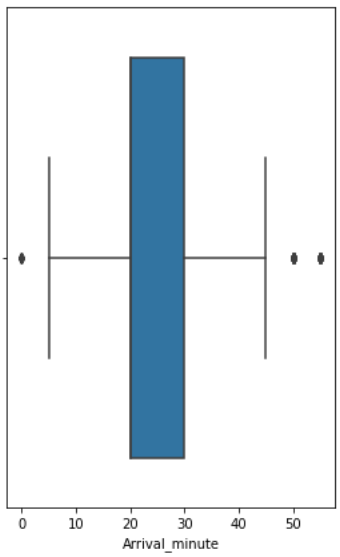

0 1 2 3 4 5 6 7
Destination

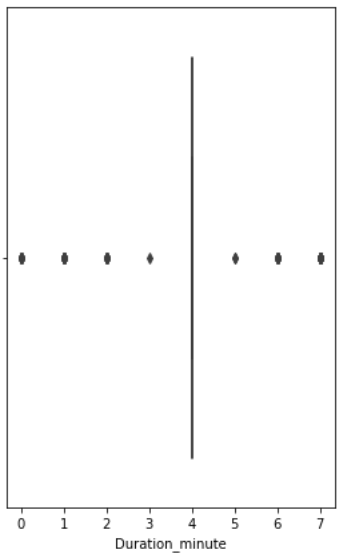

8000 10000 12000 14000 16000 18000 20000
Flight_Price

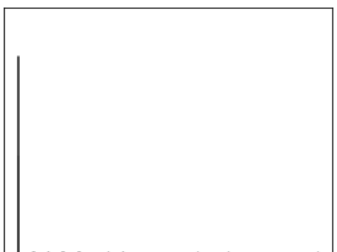

5.0 7.5 10.0 12.5 15.0 17.5 20.0 22.5
Dep_Hour


0 10 20 30 40 50
Dep_minute

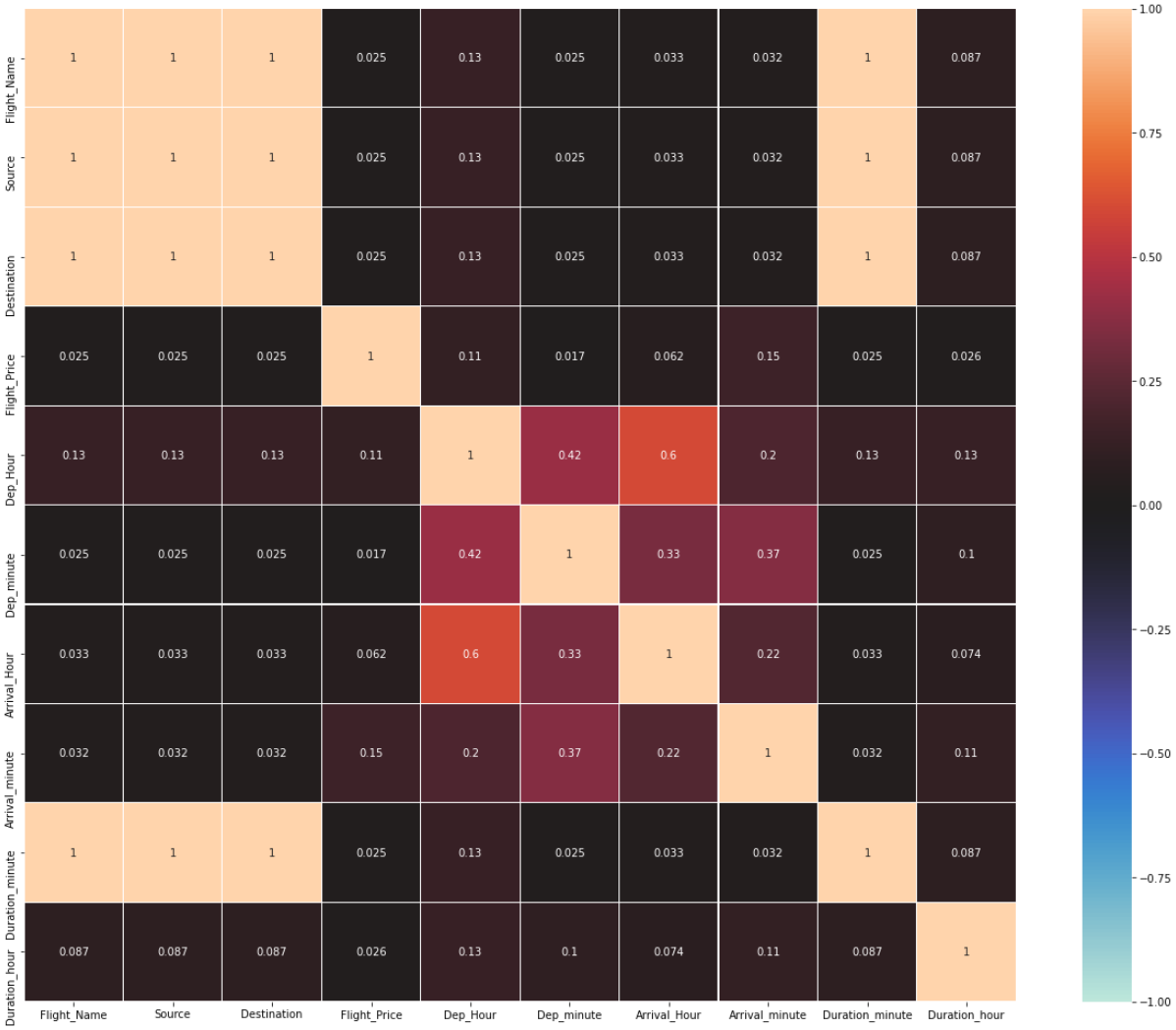

0 5 10 15 20
Arrival_Hour


0 10 20 30 40 50
Arrival_minute

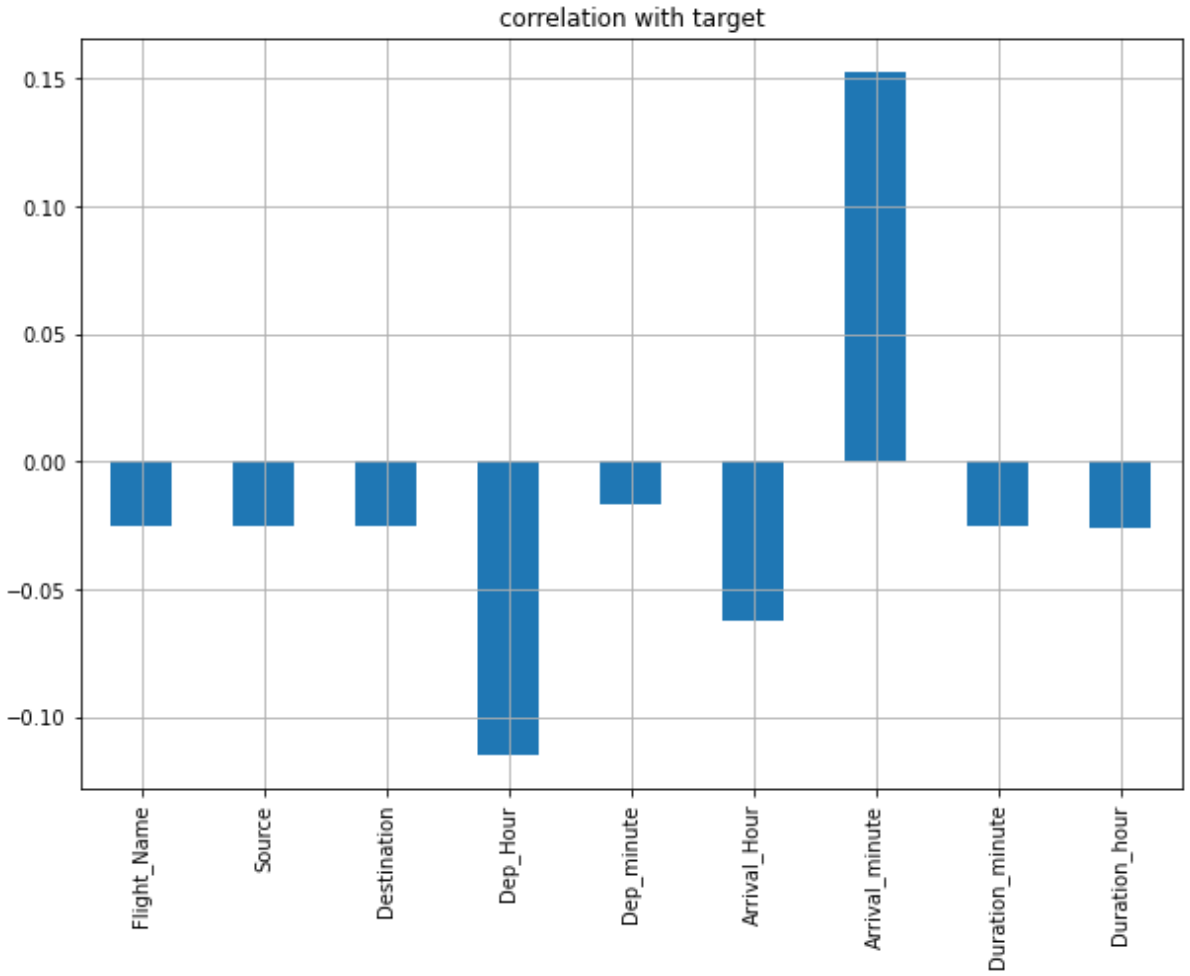

0 1 2 3 4 5 6 7
Duration_minute



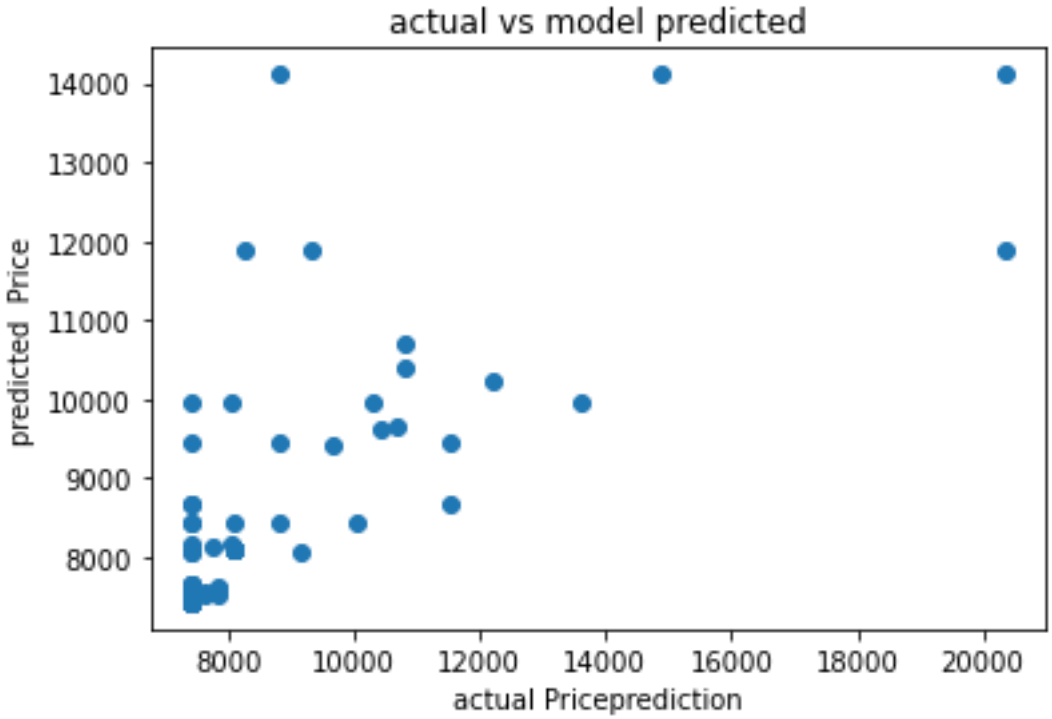
Boxplot to detect outliers in dataset



Heatmap to check multicollinearity present in dataset



Visualization the correlation with class



Scatter plot check actual price vs predicted price

- **Interpretation of the Results**

  Model score is low ,R2 score is also low

CONCLUSION

- **Key Findings and Conclusions of the Study**

  Flight sale price data is right skewed as per distplot
  Add more route, and dates different time zone is some where lacking.
  R2 score is not good ,model is not trained well

- **Learning Outcomes of the Study in respect of Data Science**

  First model Linear reg what I attempted ad its not good result but disappointed with R2 score after tuning model score reduce and cross validation score is high ,moving forward when I build RandomForest regression model score ,cross validation and scatter plot R2 score all reflect model performing better compare to decision tree,but gradientboost regression is quite good among all

- **Limitations of this work and Scope for Future Work**

  I will add more route dates try tobuild model with different approach more EDA techniques to identify insights, due to time limitation may extend different approach to build more model and find best accuracy and R2 score.

  .