Fliprobo

Flight Price Prediction Model

Report

Submitted by:

Arjun Verma,

Intern Data Scientist

ACKNOWLEDGEMENT

I would like to express my greatest appreciation to the all individuals who have helped and supported me throughout the project. I am thankful to Fliprobo team for their ongoing support during the project, from initial advice, and encouragement, which led to the final report of this project.

A special acknowledgement goes to my institute Datatrained who helped me in completing the project and learning concepts.

I wish to thank my parents as well for their undivided support and interest who inspired me and encouraged me to go my own way, without whom I would be unable to complete my project.

Below following are the other references:

www.towardsdatascience.com

www.medium.com

www.stackoverflow.com

Datatrained lectures

INTRODUCTION

➢ Business Problem Framing

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 collect data of flight fares with other features and work to make a model to predict fares of flights.

➢ Conceptual Background of the Domain Problem

Companies such as sastasafar.com, yatra.com, skyscanner.com, makemytrip etc which are booking site of flights. But before booking a flight we used to predict good and relevant value of the flight price. Similarly in the given task we have to build a model that can predict flight price from the extracted dataset from the older booking price of the flight.

➢ Review of Literature

Data has been collected from website sastasafar.com. We collected most of the important flight price dataset that can impact the booking price of the flight. Model is created using the data by splitting the data as dependent and independent variable. These dataset are further split into test and train. The train data is trained through various regression algorithms. The algorithm having the lease difference between r2 score and cross val score will be used for hyperparameter tuning. The best parameters are used to tune the model. This model is given to the client in further using to visualise data for future flight price prediction.

➢ Motivation for the Problem Undertaken

Genuinely it's a need of the any seller to complete their goal with higher revenue and low expenditure. Hence this model can brings higher revenue because we can predict upcoming booking flight booking prices and make booking accordingly for our clients.

➢ Mathematical/ Analytical Modeling of the Problem

Data is statistically analysed through variance inflation factor. Analysed through correlation and multicollinearity. Graphical modelling done through seaborn and matplotlib to understanding how different features impact dataset.

Statistical models used

- ➢ Linear Regression
- ➢ DecisionTreeClassifier
- ➢ Random forest regressor
- ➢ Gradient Boosting Regressor
- ➢ Ada Boost Regressor
- ➢ PCA
- ➢ Standard Scalar

➢ Data Sources and their formats

Datasets are extracted by site sastasafar.com for building machine learning model to predict booking price of the flight based on given parameter.

Dataset is having 1949 rows and 9 columns including target.

The information about features are as follows

```
'Price', 'Flight_Name', 'Arrival_Location',
'Destination_Location', 'Arrival_time', 'Destination_time', 'No
of Stops', 'Duration', 'DateDay'
```

Dataframe Description:

- **Price : Actual traveling cost of the flight**
- **Flight_Name : Flight names**
- **Arrival_Location : Arrival Location**
- **Destination_Location : Destination Location**
- **Arrival_time : Takeoff timing of the flight.**
- **Destination_time : Reahout timing of the flight to their destination.**
- **No of Stops : No of stops is the flight interval stops.**
- **Duration : Total time taken to reach out destination place.**
- **DateDay : Represent date and day of the flight.**

```
df.head() # checking first 5 rows
```

	Price	Flight_Name	Arrival_Location	Destination_Location	Arrival_time	Destination_time	No of Stops	Duration	DateDay
0	₹9419	Spice Jet	DEL	BLR	20:00	22:30	Non Stop	NaN	20 Jul, Wed
1	₹9419	Go Air	DEL	BLR	20:30	23:25	Non Stop	02h 55m	20 Jul, Wed
2	₹9419	Air Asia	DEL	BLR	08:10	10:40	Non Stop	02h 30m	20 Jul, Wed
3	₹9419	Spice Jet	DEL	BLR	06:05	08:55	Non Stop	02h 50m	20 Jul, Wed
4	₹9945	Air Asia	DEL	BLR	09:35	12:25	Non Stop	02h 50m	20 Jul, Wed

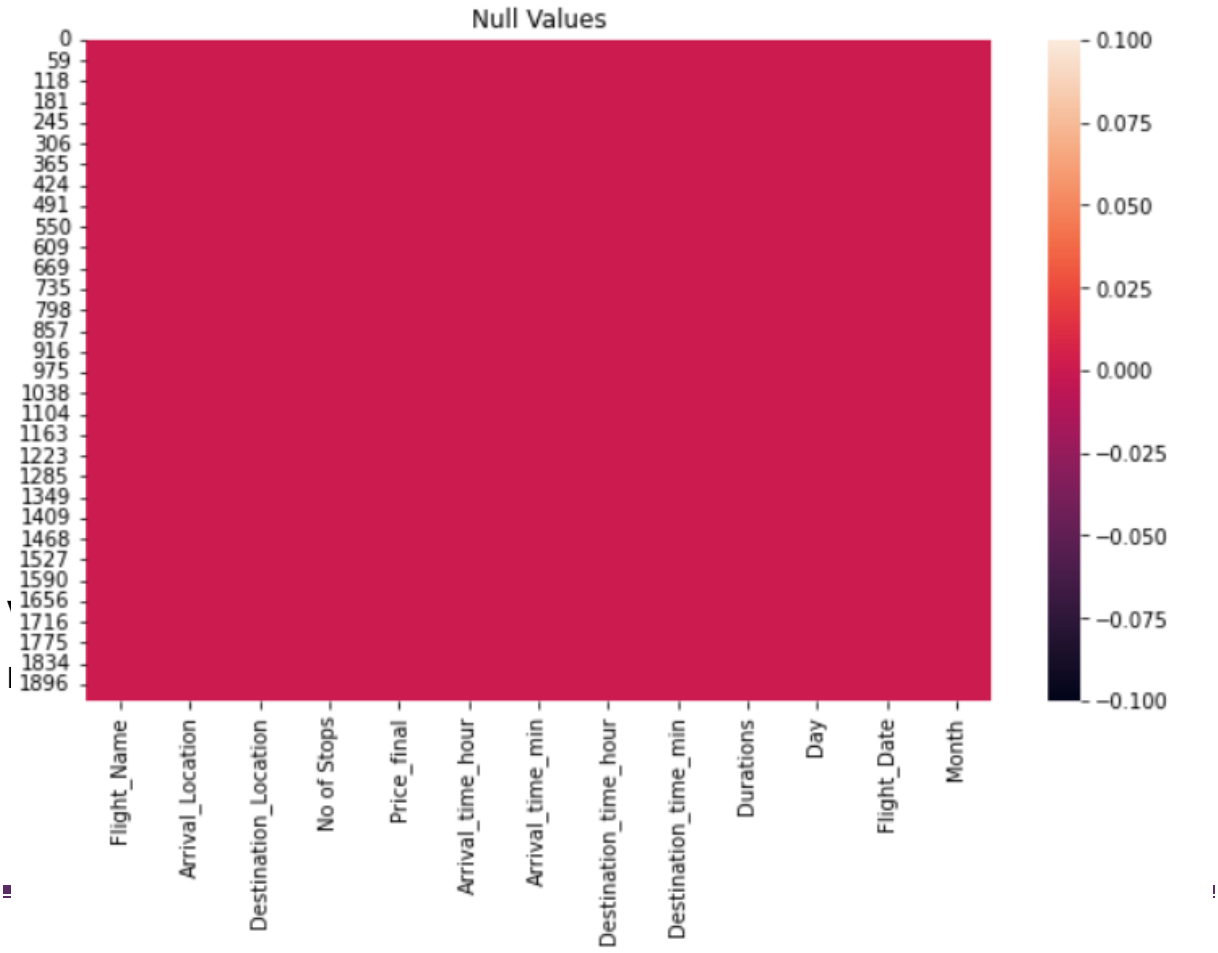
We have done feature engineering for preprocessing dataset and get below information

Dataset Information

`'Flight_Name', 'Arrival_Location', 'Destination_Location', 'No of Stops', 'Day', 'Flight_Date', 'Month'` **are of object type data columns.**

`'Price_final', 'Arrival_time_hour', 'Arrival_time_min', 'Destination_time_hour', 'Destination_time_min', 'Durations'` **are of numerical type data columns.**

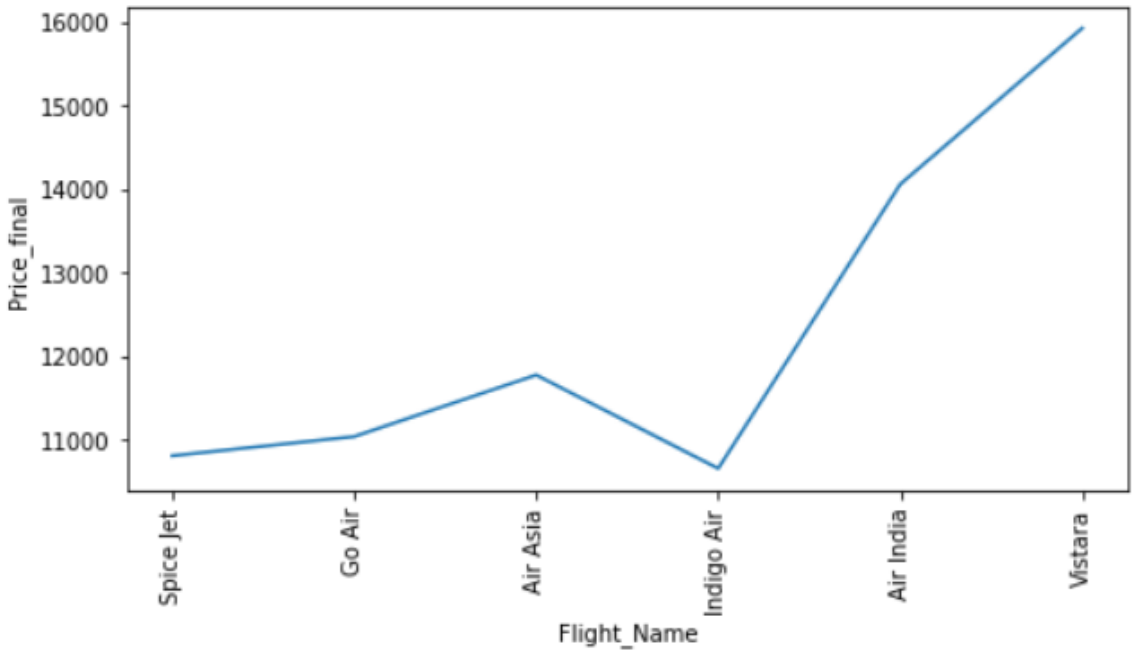
Checking Null Values of the dataset



There are no null values present in the dataset.

Visualization of important features for understanding

Flight name according to prices



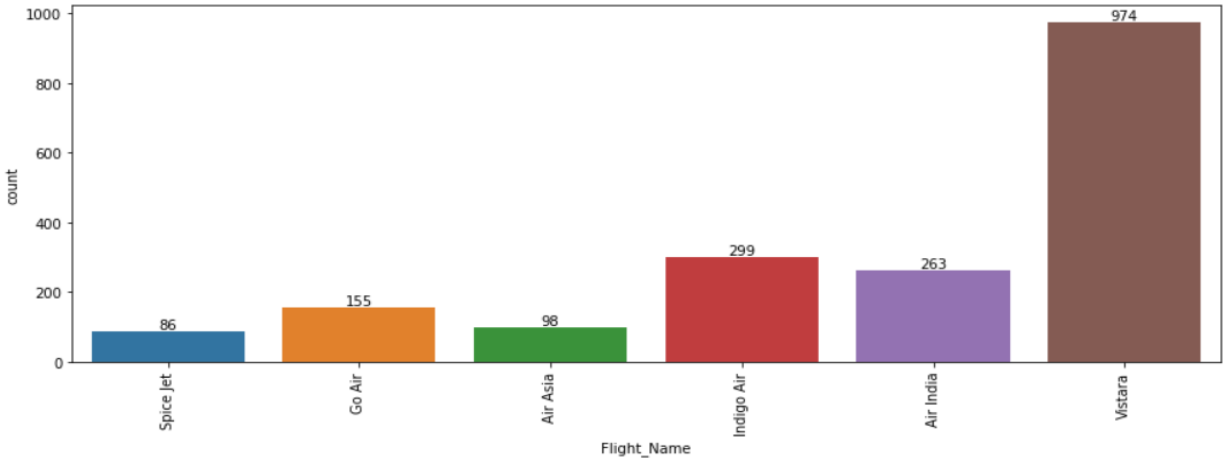
Observations:

- **From above graph we can find that Air Asia, Air India and Vistara having high price value flight while rest flights having lower cost.**

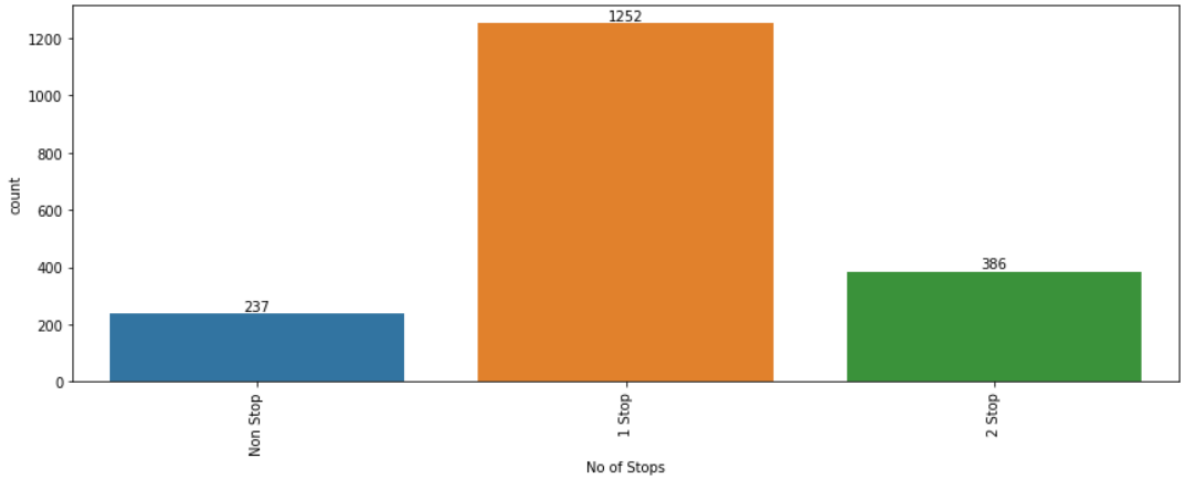
.

Count Plot

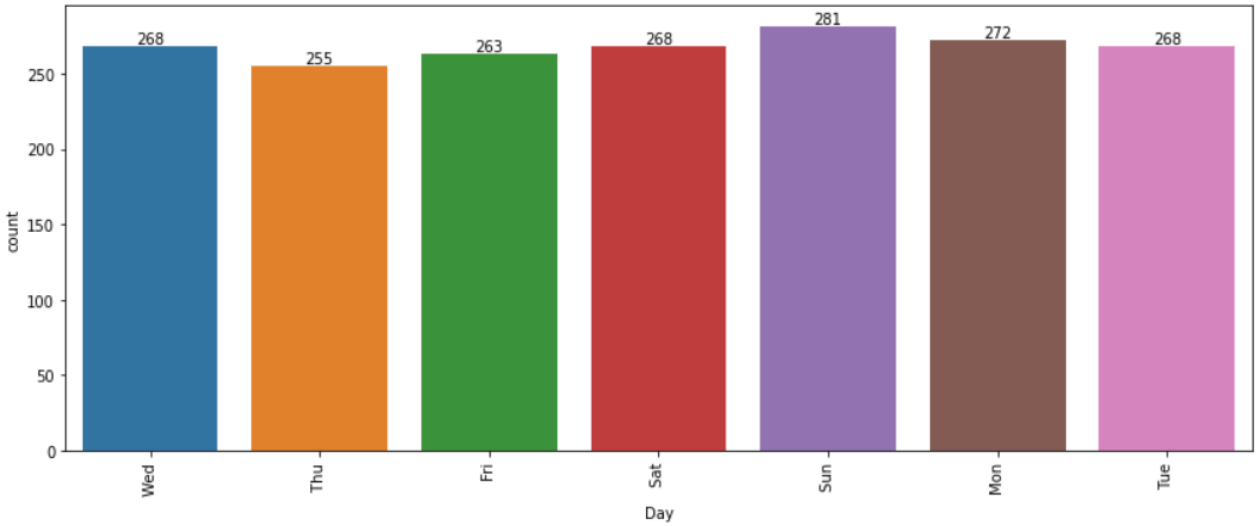
Flight Name



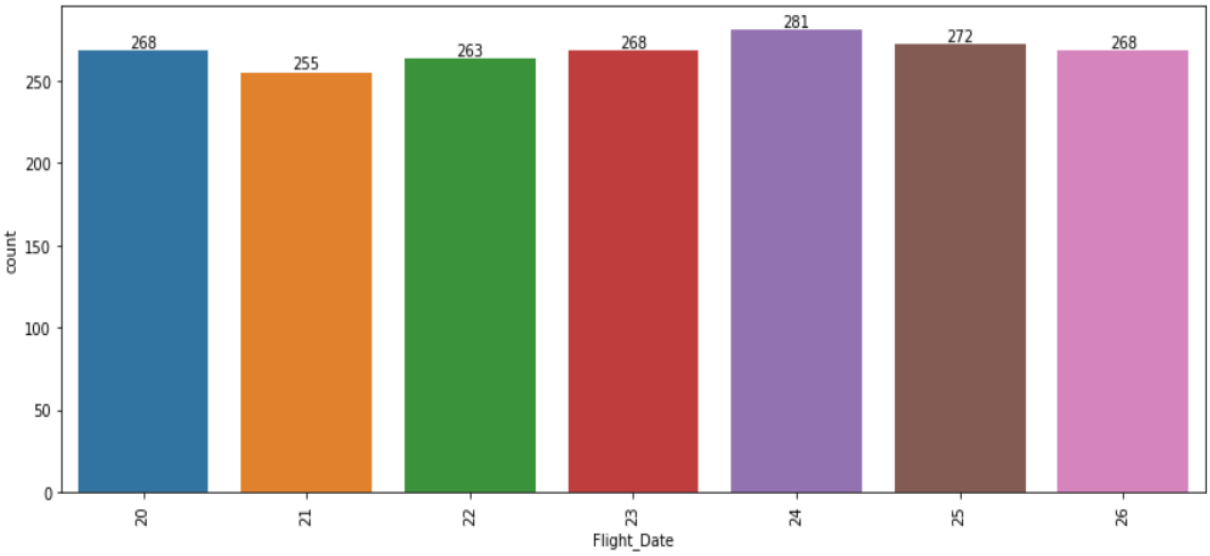
No of stops



Flight Counts on different week days

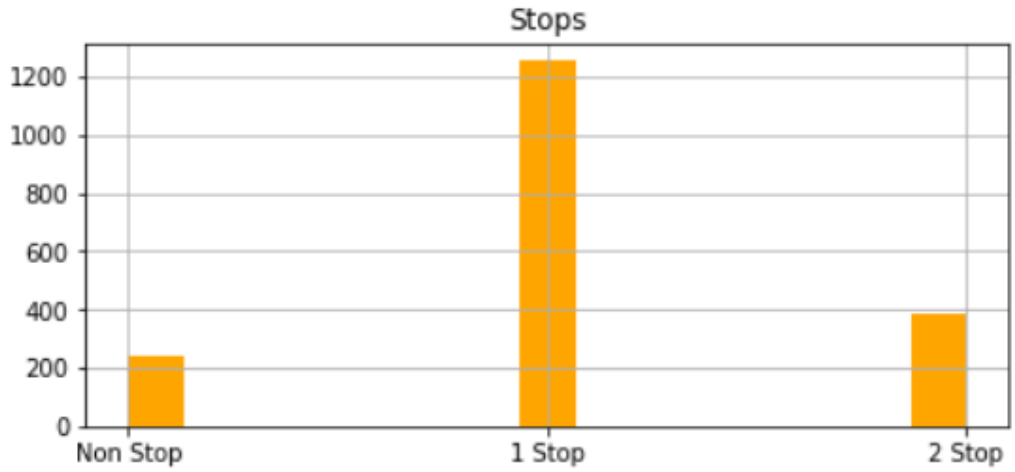


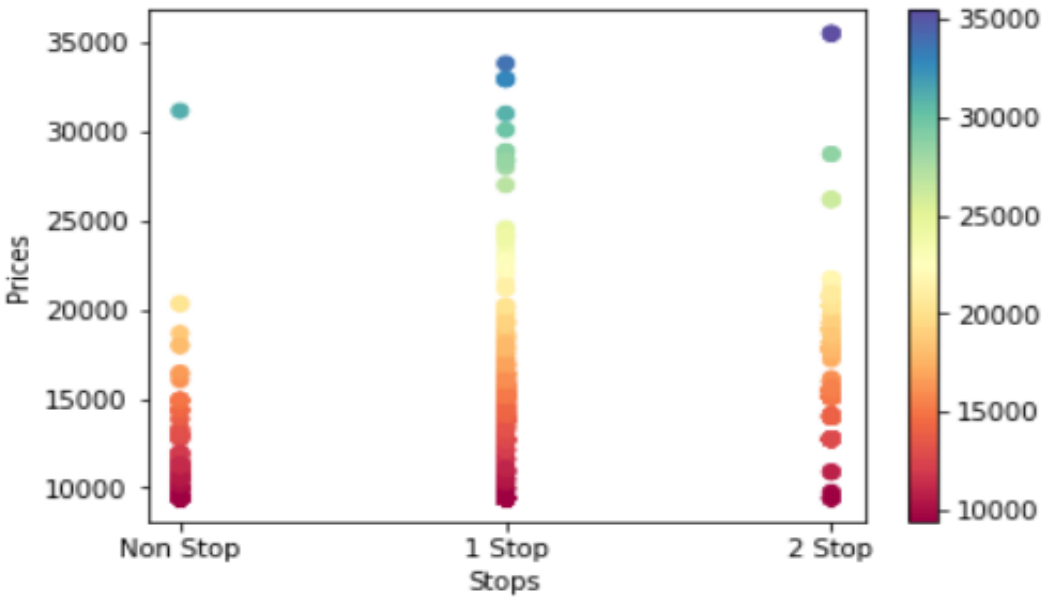
No. of flights in different dates



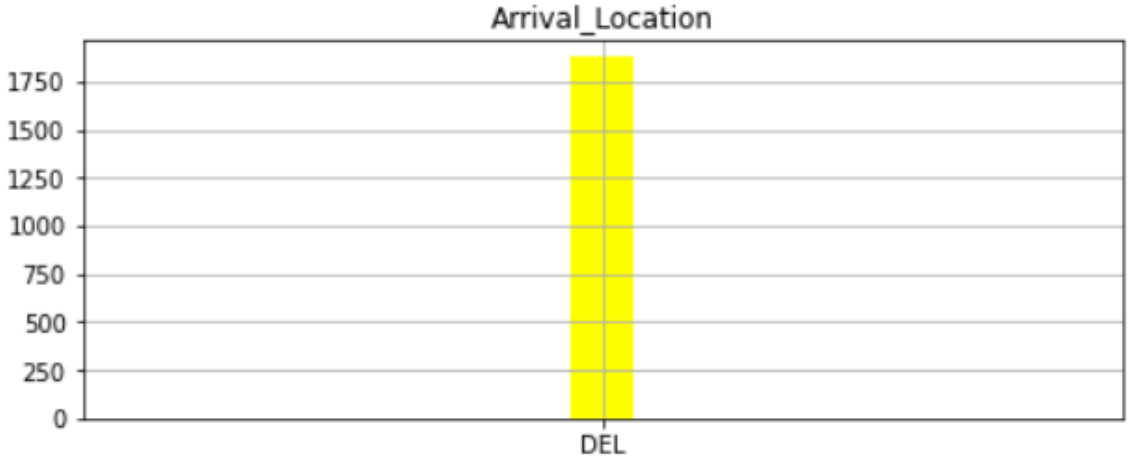
Bivariate Analysis

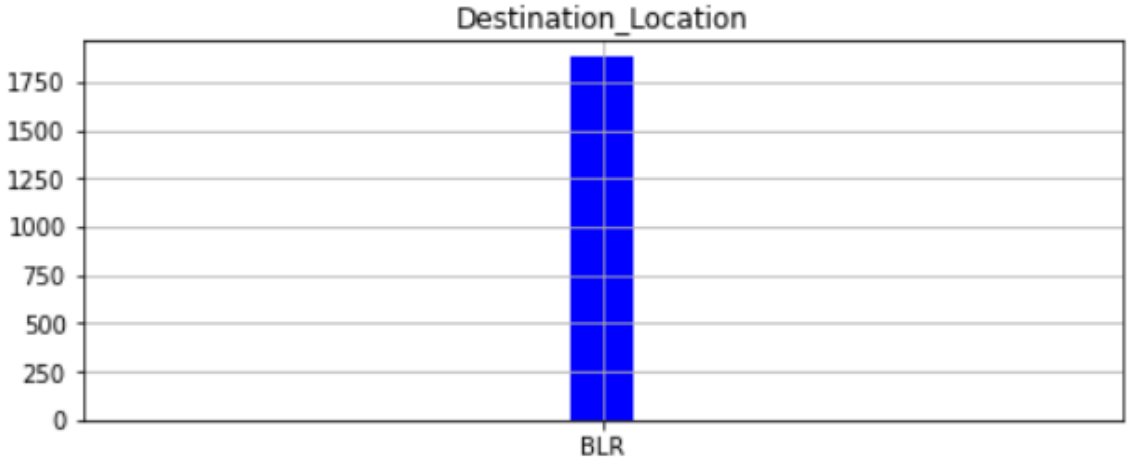
Price variation with number of stops





Arrival and Destination Location



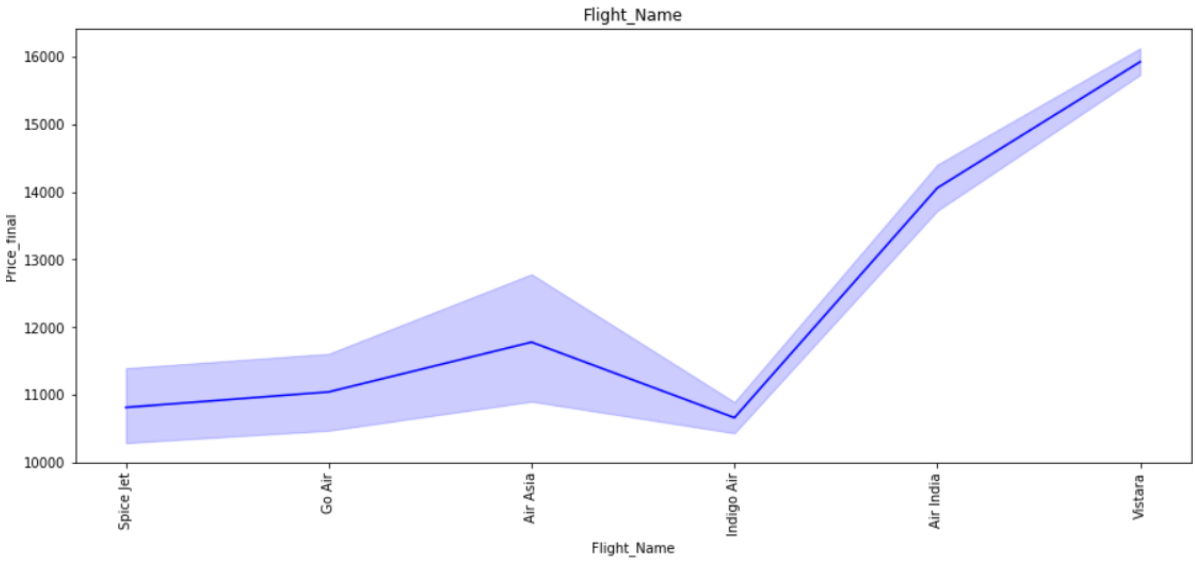


Observations

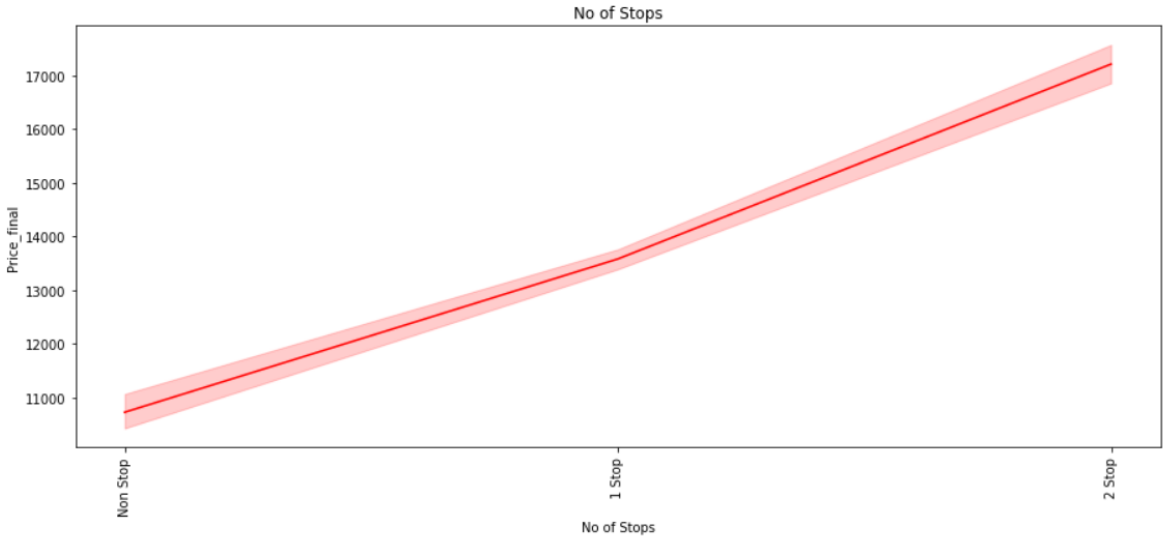
1. Vistara having very high no. of flights as per dataset.
2. 1 stop flights are more as per comparison to Non stops and 2 stops.
3. On Sundays there are more number of flights availables.
4. 1 Stop flights prices are very high as compared to others.
5. We have selected flight of arrival location for Delhi and destination location for Bangalore.

Line Plot Representation

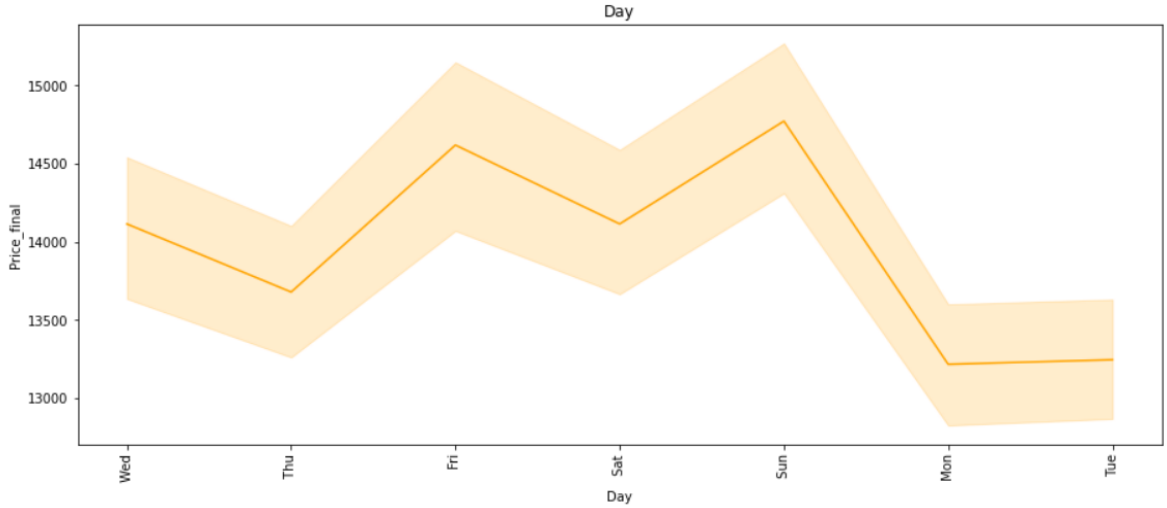
Price vs Flight_name



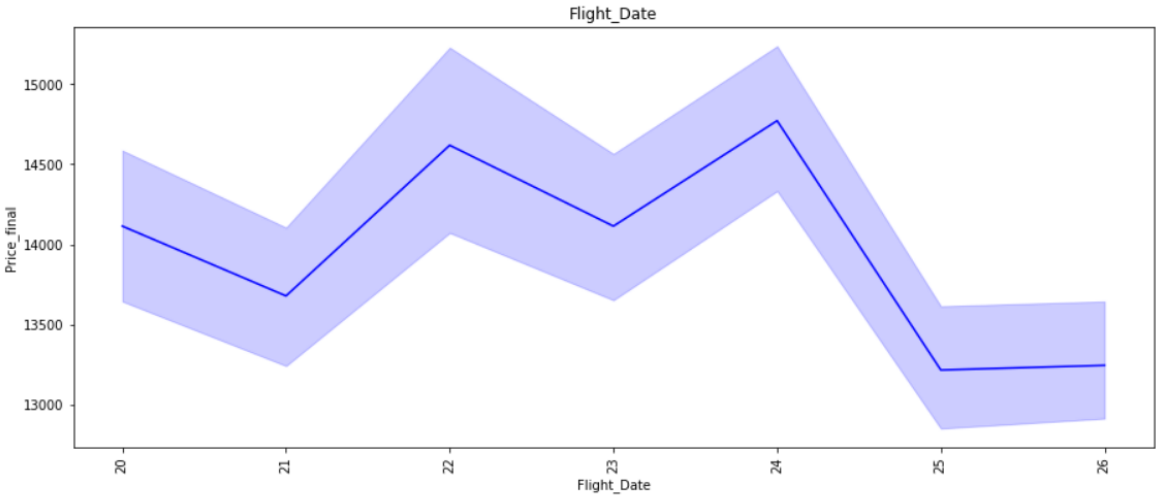
No of stops vs Price



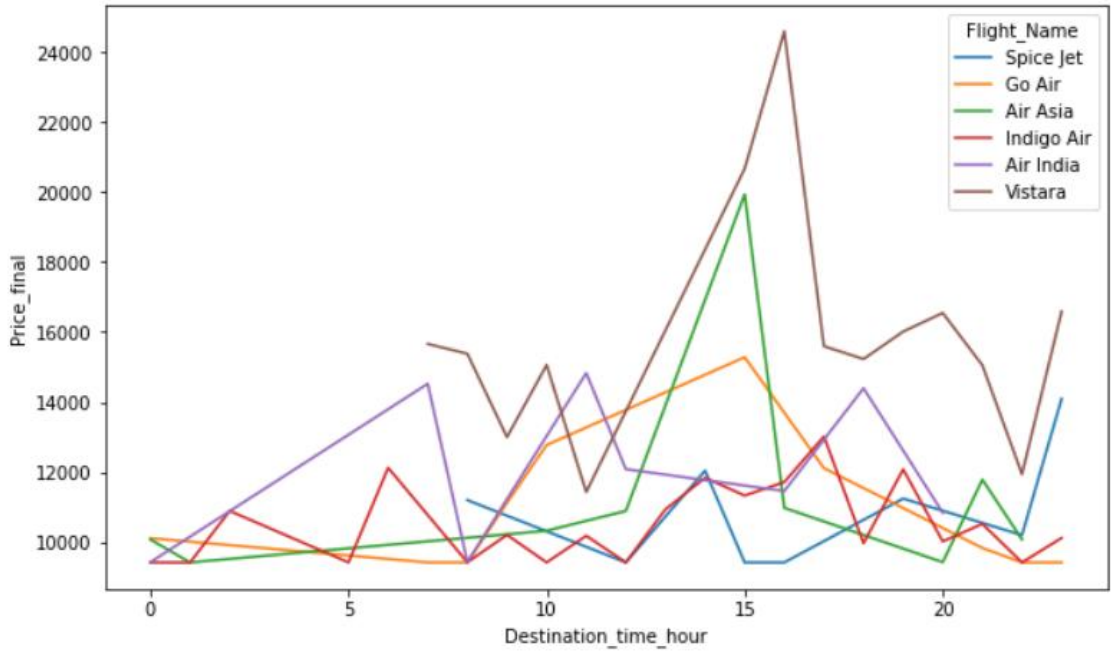
Price vs Week days



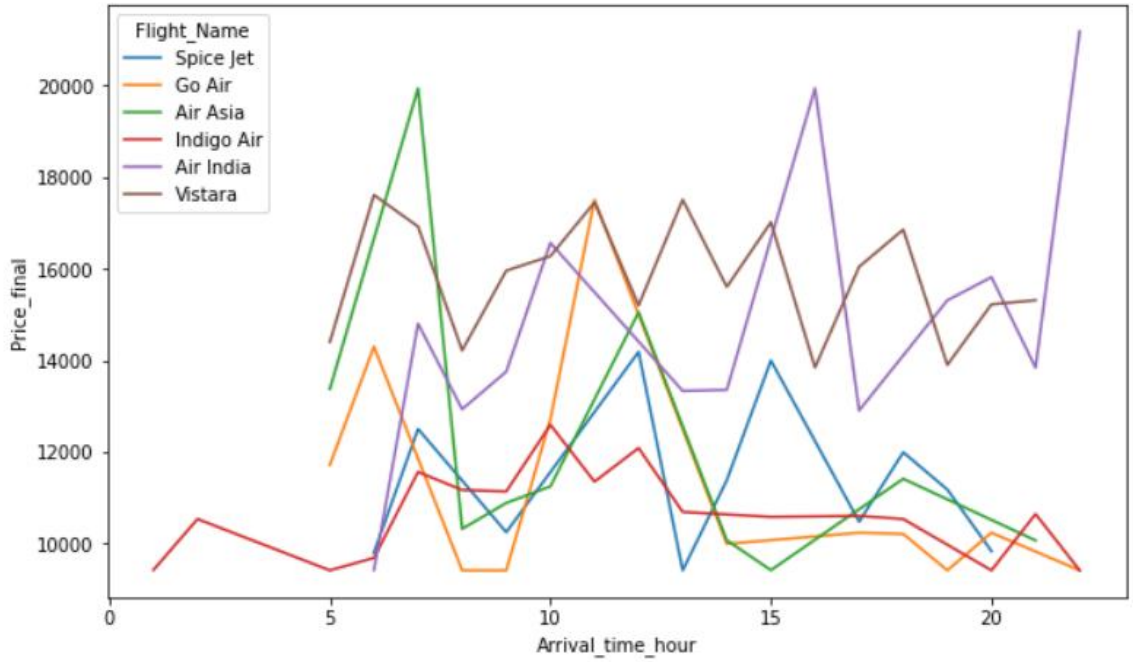
Brand vs Price



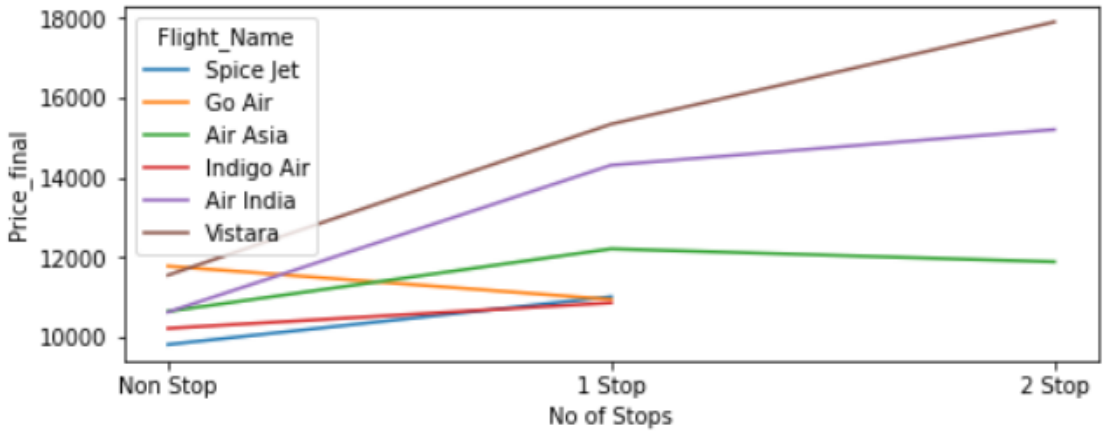
Destination time hour vs different flight price



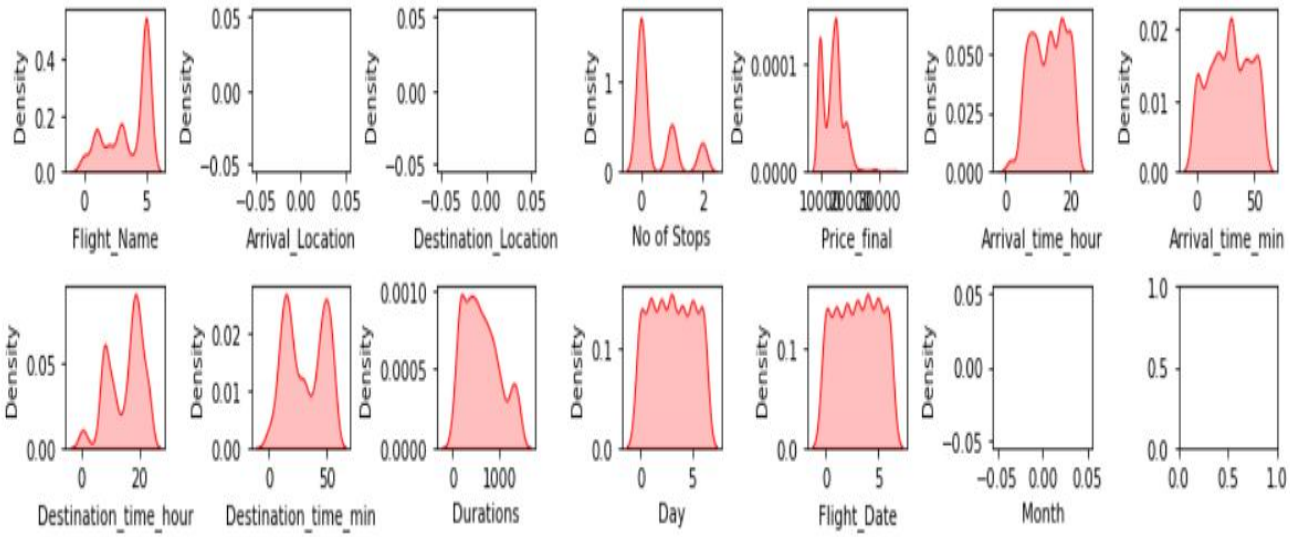
Arrival time hour vs different flight price



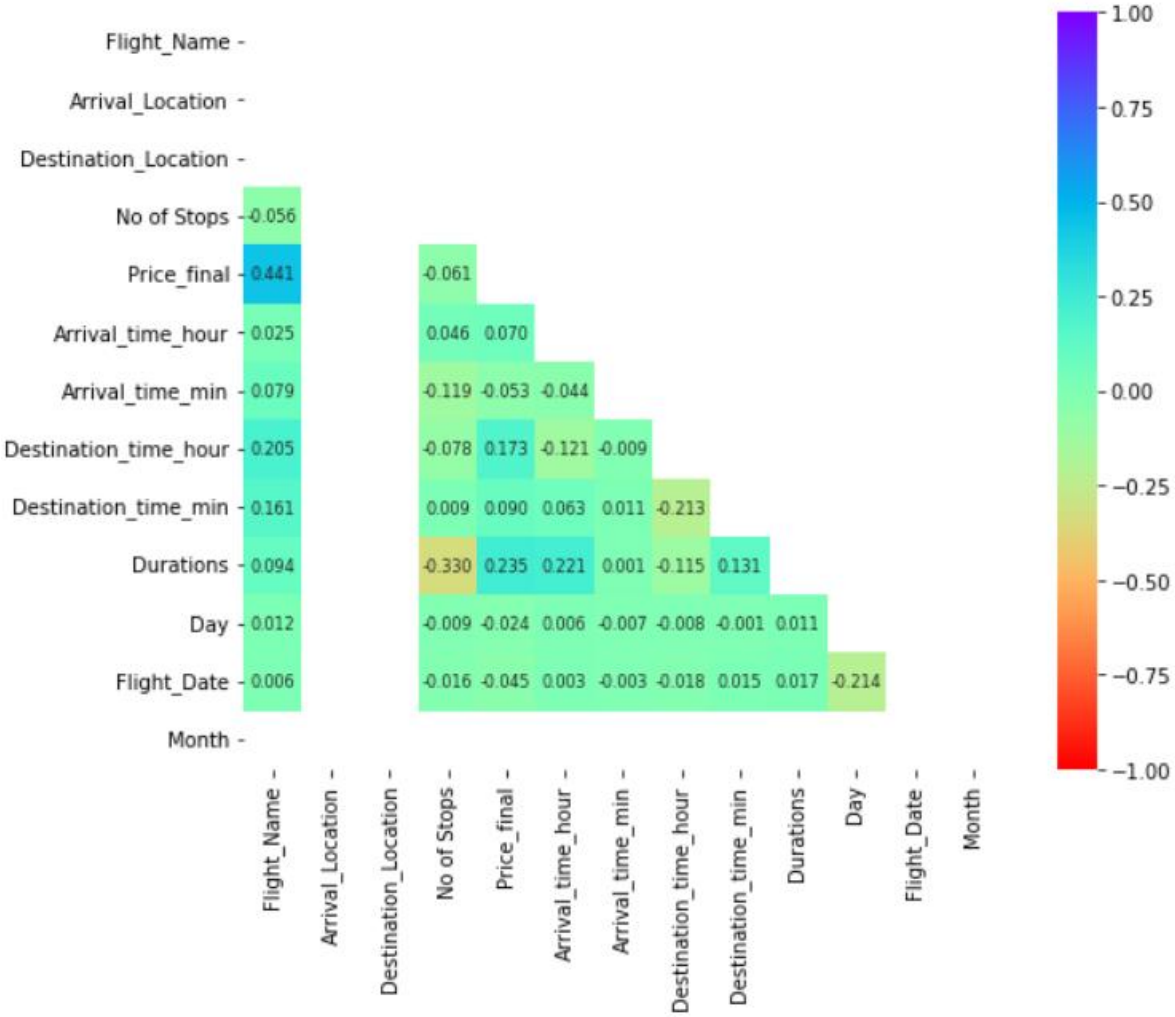
Price vs No of stops



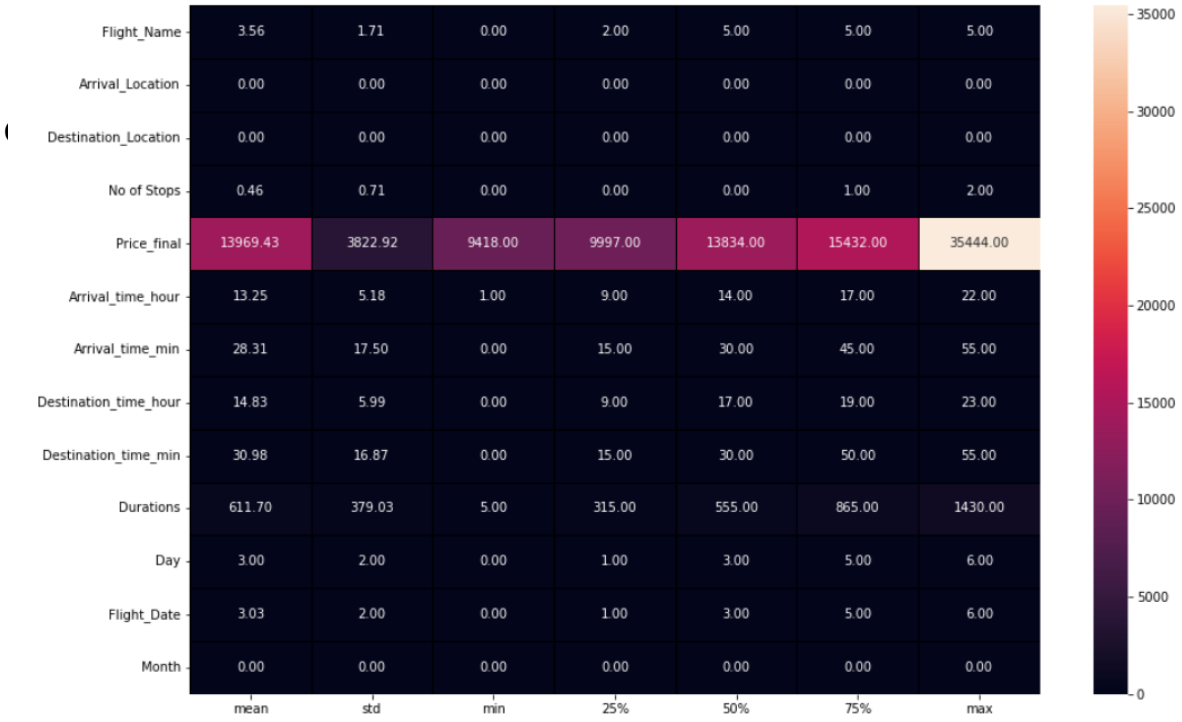
Density of different dataset



Correlation of the Dataset



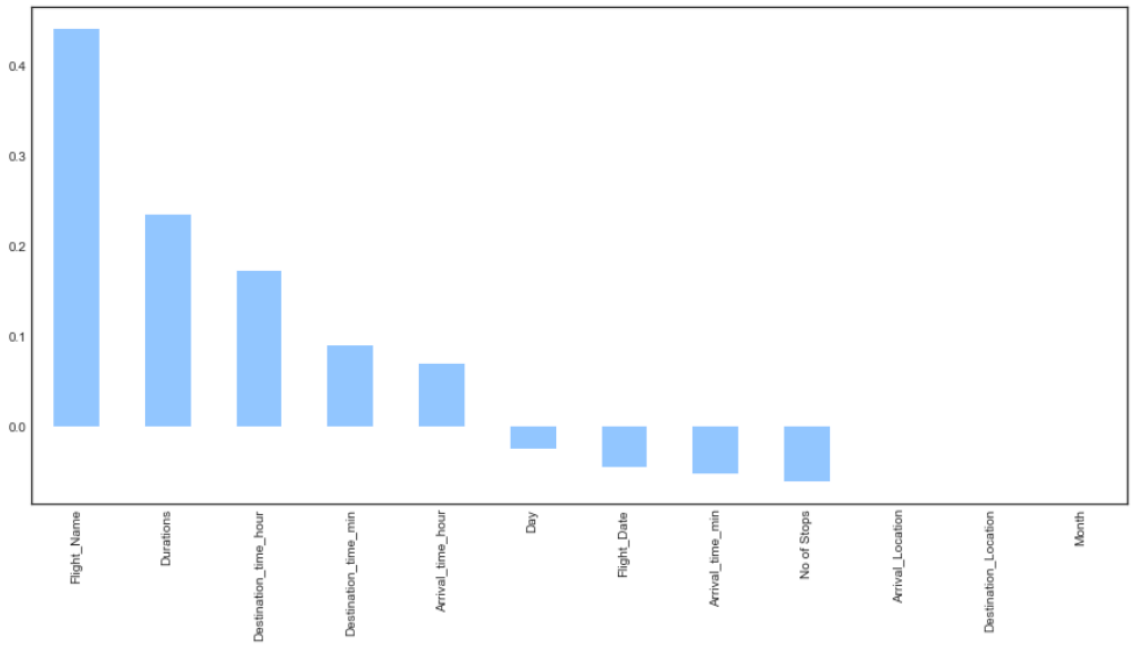
Describe of the Dataset



Outliers

We have applied Z score and Interquartile method for outlier removal and find that Interquartile gives lesser data lose hence we consider it.

Checking Positive and Negative Correlation



Dividing data for feature selection

```
#Splitting the indepedent and target variable in x and y
x= df_IQR.drop('Price_final',axis=1)
y= df_IQR['Price_final']
```

Checking Mutlicollinearity

	Variance	VIF Factor
0	Flight_Name	3.660836
1	Arrival_Location	NaN
2	Destination_Location	NaN
3	No of Stops	1.651636
4	Arrival_time_hour	1.050076
5	Arrival_time_min	1.021233
6	Destination_time_hour	1.135503
7	Destination_time_min	1.090194
8	Durations	1.267556
9	Day	2.457676
10	Flight_Date	2.459438
11	Month	NaN

Removing Skewness by 'yeo- Johnson' method.

Feature Scaling

Feature selection using Basic Linear Regression Model

	columns	importance
0	Location	0.540854
1	Fuel_Type	2.309869
2	Transmission	4.357363
3	Owners	0.039866
4	Driven_in_Thousand_km	0.593618
5	Brand	0.445940
6	Model	0.241050
7	Total_Years_of_Car	1.623697

Feature selection using Basic Ridge Regression Model

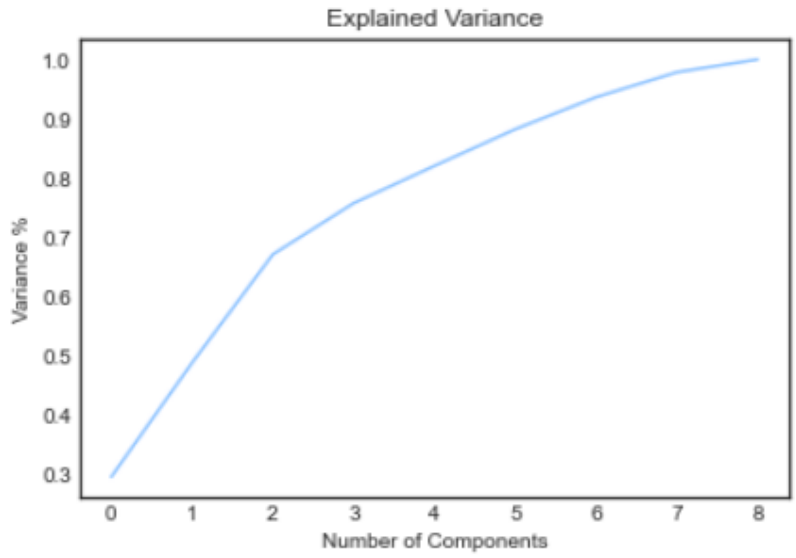
	columns	importance
0	Flight_Name	1327.859504
1	Arrival_Location	0.000000
2	Destination_Location	0.000000
3	No of Stops	132.076065
4	Arrival_time_hour	102.598490
5	Arrival_time_min	244.634047
6	Destination_time_hour	379.210534
7	Destination_time_min	80.919539
8	Durations	681.723047
9	Day	140.413713
10	Flight_Date	193.241120
11	Month	0.000000

Here we find that Month, Arrival_location and Destination feature has been discarded

Principle Component Analysis

```
from sklearn.decomposition import PCA
```

```
pca = PCA()
principleComponents = pca.fit_transform(x)
plt.figure()
plt.plot(np.cumsum(pca.explained_variance_ratio_))
plt.xlabel('Number of Components')
plt.ylabel('Variance %')
plt.title('Explained Variance')
plt.show()
```



All components explain around 95% variance in data

Selecting Kbest Features

```
: from sklearn.feature_selection import SelectKBest, f_classif
```

```
: bestfeat = SelectKBest(score_func = f_classif, k = 'all')
  fit = bestfeat.fit(x,y)
  dfscores = pd.DataFrame(fit.scores_)
  dfcolumns = pd.DataFrame(x.columns)
```

```
: fit = bestfeat.fit(x,y)
  dfscores = pd.DataFrame(fit.scores_)
  dfcolumns = pd.DataFrame(x.columns)
  dfcolumns.head()
  featureScores = pd.concat([dfcolumns,dfscores],axis = 1)
  featureScores.columns = ['Feature', 'Score']
  print(featureScores.nlargest(10,'Score'))
```

```
                Feature      Score
0           Flight_Name  62.352685
1           No of Stops  16.237275
6             Durations   5.656271
3      Arrival_time_min   3.654755
4  Destination_time_hour   2.897377
2      Arrival_time_hour   2.781764
5   Destination_time_min   1.827663
7                   Day   1.748782
8           Flight_Date   1.474312
```

Since all the dataset show some scores hence we are not dropping anyone of them

Model Building and Results

	Model	r2score	Cross_val_score	RMSE score	Difference between cv score and cross_val score
0	LinearRegression	32.820448	6.144071	2642.532989	26.676377
1	Ridge Regressor	32.723456	6.484572	2644.439904	26.238884
2	Lasso Regressor	32.814392	6.174359	2642.652087	26.640033
3	DecisionTreeRegressor	30.348945	-101.671832	2690.702694	132.020777
4	RandomForestRegressor	48.112343	35.323266	2322.383277	12.789077
5	KNeighborsRegressor	62.145320	-34.477613	1983.634516	96.622933
6	GradientBoostingRegressor	32.155566	-47.846051	2655.577475	80.001617
7	AdaBoostRegressor	51.855129	39.159140	2237.056123	12.695989
8	ExtraTreesRegressor	83.556638	47.428472	1307.365958	36.128166
9	XGBRegressor	85.571576	-0.126976	1224.648107	85.698552
10	LGBMRegressor	84.066664	43.276592	1286.930896	40.790072

We are selecting LGBM regressor as it gives less RMSE score with good R2 score rather than other models

Best models

- KNeighborsRegressor : Model shows low r2 score in training and testing accuracy hence we cannot consider it.

- DecisionTreeRegressor: Same as DecisionTreeRegressor shows very much difference in training and testing accuracy hence we cannot consider it. Model becames underfit.

- XGBRegressor: Same as above two model it shows similar in CV score and testing r2 score hence we can consider it.

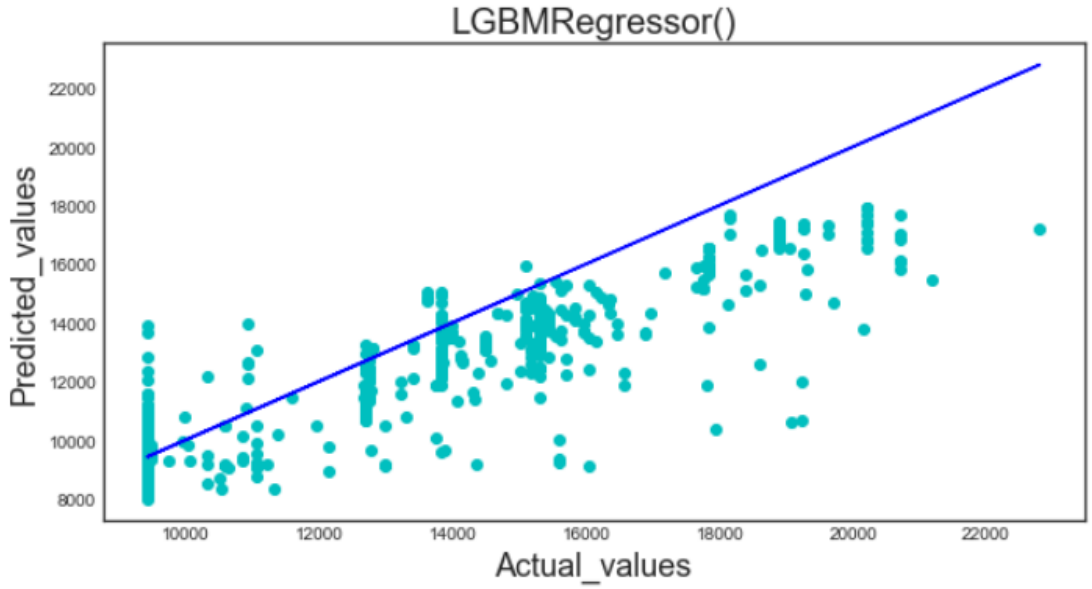
- LGBMRegressor : Model shows good results as it gives less RMSE score with good R2 score rather than other models.

Final Model LGBM Regressor

```
Training accuracy:-  43.98358076622812
Testing accuracy:-  42.14709638083677


--------------------------------------
Mean squared error:-  3880996.6486919206
Mean absolute error:-  1519.8785421946864
Root Mean squared error:- 1970.0245299721323
```

```
|: Text(0.5, 1.0, 'LGBMRegressor()')
```



Model Deployment

Deploy Model

```
import pickle

filename = "Flight.pkl"
pickle.dump(final_model, open(filename, 'wb'))
```

Loading Model

```
load = pickle.load(open('Flight.pkl', 'rb'))
result = load.score(x_test, y_test)
print(result)
```

```
0.6303105255246516
```

```
conclusion = pd.DataFrame()
conclusion['Predicted Flight price'] = np.array(final_model.predict(x_test))
conclusion['Actual Flight price'] = np.array(y_test)
```

```
conclusion.sample(10)
```

	Predicted Flight price	Actual Flight price
337	16795.186005	20209.0
46	9136.172583	9419.0
534	8713.313683	9419.0
467	9691.564778	9419.0
436	13634.462069	15300.0

➢ Hardware and Software Requirements and Tools Used

Operating System: Window 11

RAM: 8 GB

Processor: i5 10th Generation

Software: Jupyter Notebook

Python Libraries: Mainly

Pandas: This library used for dataframe operations .

Numpy: This library gives statistical computation for smooth functioning .

Matplotlib: Used for visualization.

Seaborn: This library is also used for visualization.

Sklearn: This library having so many machine learning module and we can import them from this library.

Pickle: This is used for deploying the model.

Xgboost: Extreme Gradient Boosting, is a scalable, distributed gradient-boosted decision tree (GBDT) machine learning library

Lightgbm: Light version of Gradient Boosting Machine.

CONCLUSION

➢ Key Findings and Conclusions of the Study

This project has built a model that can predict upcoming Prices of flight. For this company can reduces loses in Investment. The challenge behind booking flight Price finding in machine learning is the number of features in dataset. Also some other issues like imputation understandings and so many values are zeros.

➢ Learning Outcomes of the Study in respect of Data Science

Data cleaning is the most important part in this model building as we see above there are so many NULL values we fill with imputation and ranges some of column dataset for better observations. This project has gives so much information about parameters that how a single parameter can increase or decrease prices of house.

➢ Limitations of this work and Scope for Future Work

Model work with similar parameters as we build the whole model if some of the parameters missed then we need to train model with remains parameter after that we can predict upcoming flight booking for client hence we need to up to date all the parameters as per training dataset.