Capstone Project - 2

NYC Taxi Trip Duration Prediction

Saurabh Funde

About Dataset

NYC Taxi Trip dataset is released by the NYC Taxi and Limousine Commission, which includes pickup time, geo-coordinates, number of passengers, and several other variables. The dataset is based on the 2016 NYC Yellow Cab trip record data made available in Big Query on Google Cloud Platform.

Problem Statement

NYC is very prominent and populous city and its streets are always busy. It’s important for taxi company to know and understand the relationship between travelled distance and trip duration required for trips.

Data Preparation

1. Data Exploration : Explored the given NYC Taxi Trip dataset. Checked the shape of dataset, datatypes , NaN values and any duplicated records.

2. Data Processing : After initial data exploration did column by column data wrangling and changed the datatypes accordingly.

3. Feature Engineering : Created and added new required features to the dataset for model building.

4 EDA : To see the main characteristics about given dataset attributes visualized using Matplotlib and Seaborn library.

Dataset Features

Independent Features :
id : a unique identifier for each trip.
vendor_id : a code indicating the provider associated with the trip record.
pickup_datetime : date and time when the meter was engaged.
dropoff_datetime : date and time when the meter was disengaged.
passenger_count : the number of passengers in the vehicle (driver entered value).
pickup_longitude : the longitude where the meter was engaged.
pickup_latitude : the latitude where the meter was engaged.
dropoff_longitude : the longitude where the meter was disengaged.
dropoff_latitude : the latitude where the meter was disengaged.
store_and_fwd_flag : This flag indicates whether the trip record was held in vehicle.
memory before sending to the vendor because the vehicle did not have a connection.
to the server - Y=store and forward; N=not a store and forward trip.

Target Feature :
trip_duration : duration of the trip in seconds.

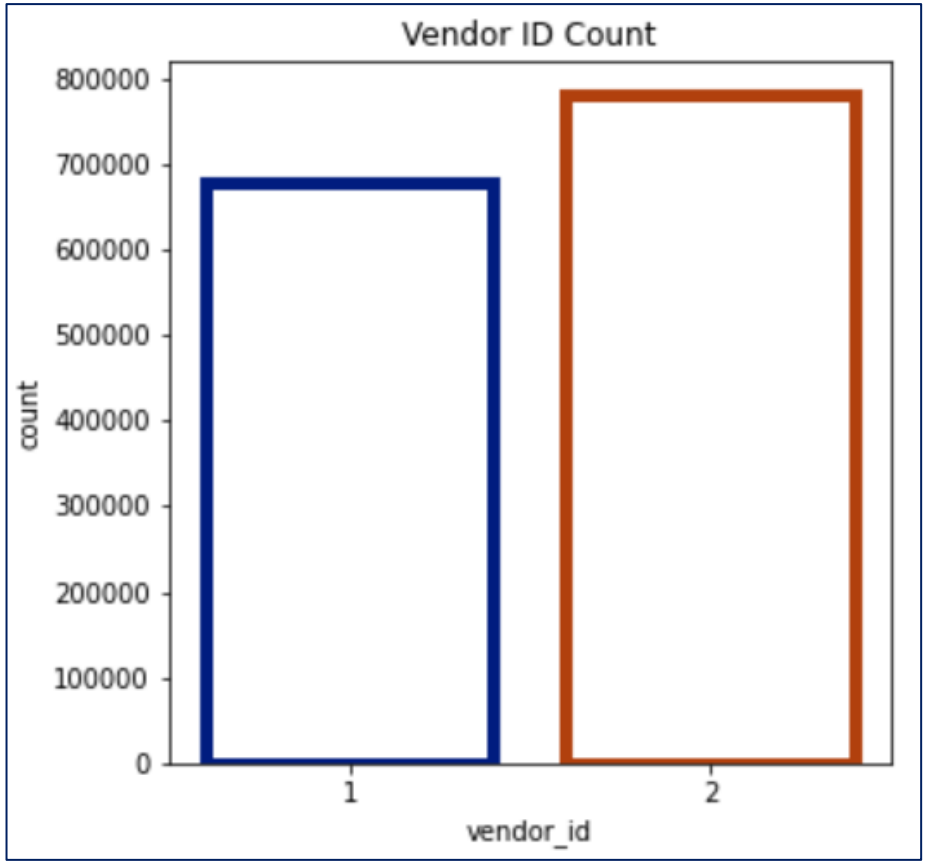
Descriptive Analysis

	vendor_id	passenger_count	pickup_longitude	pickup_latitude	dropoff_longitude	dropoff_latitude	trip_duration
count	1,458,644.00	1,458,644.00	1,458,644.00	1,458,644.00	1,458,644.00	1,458,644.00	1,458,644.00
mean	1.53	1.66	-73.97	40.75	-73.97	40.75	959.49
std	0.50	1.31	0.07	0.03	0.07	0.04	5,237.43
min	1.00	0.00	-121.93	34.36	-121.93	32.18	1.00
25%	1.00	1.00	-73.99	40.74	-73.99	40.74	397.00
50%	2.00	1.00	-73.98	40.75	-73.98	40.75	662.00
75%	2.00	2.00	-73.97	40.77	-73.96	40.77	1,075.00
max	2.00	9.00	-61.34	51.88	-61.34	43.92	3,526,282.00

- This describe table tell us about description of Numerical features.
- The given dataset contains 2 unique vendors.
- The passenger count from 0 to 9 in Taxi.(Driver Input)
- Trip Duration is target variable and its range is from 1 second to 3526282 seconds.

EDA

Vendor ID

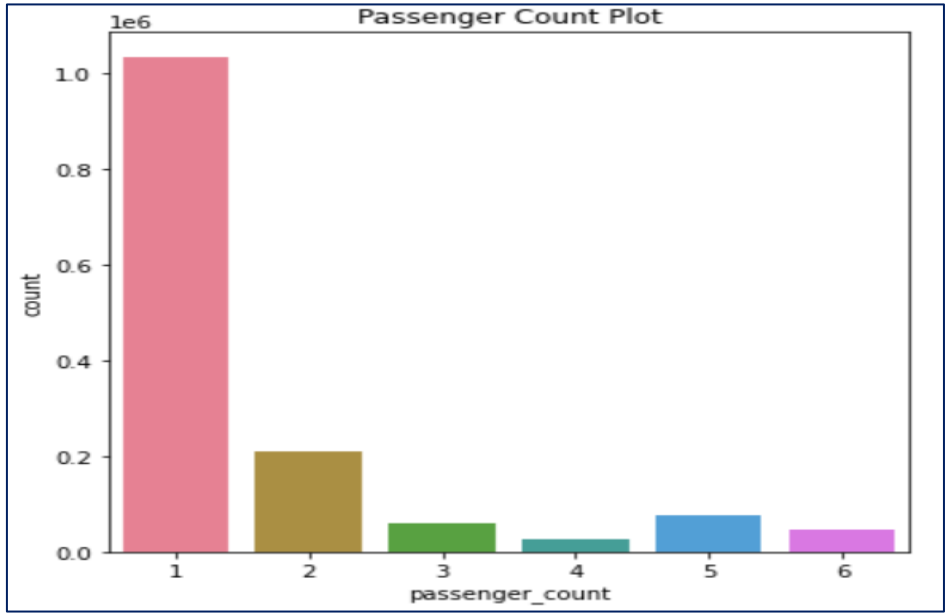


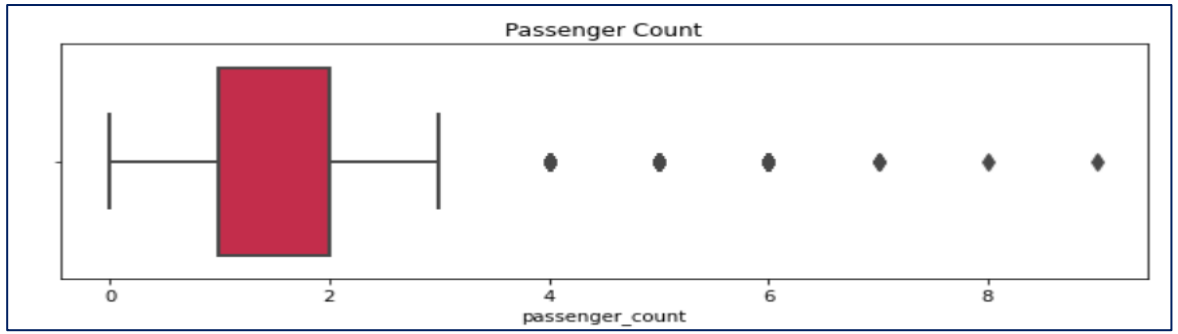
Vendor_ID 1 Count : 678342

Vendor_ID 2 Count : 780302

EDA

Passenger Count



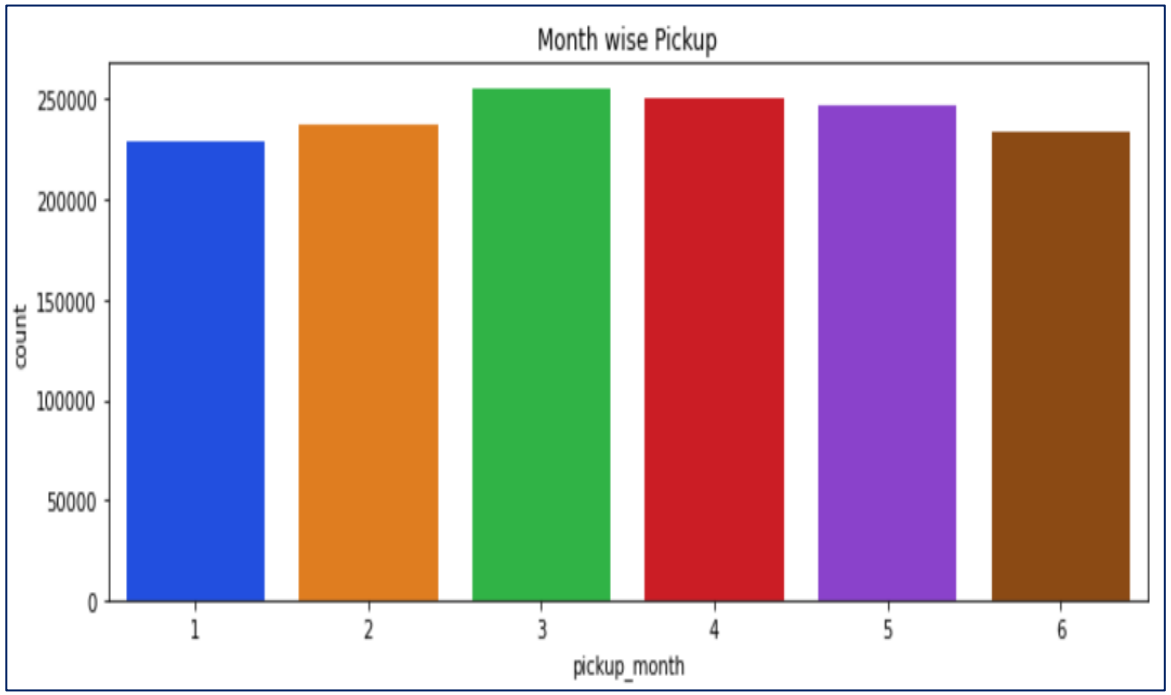


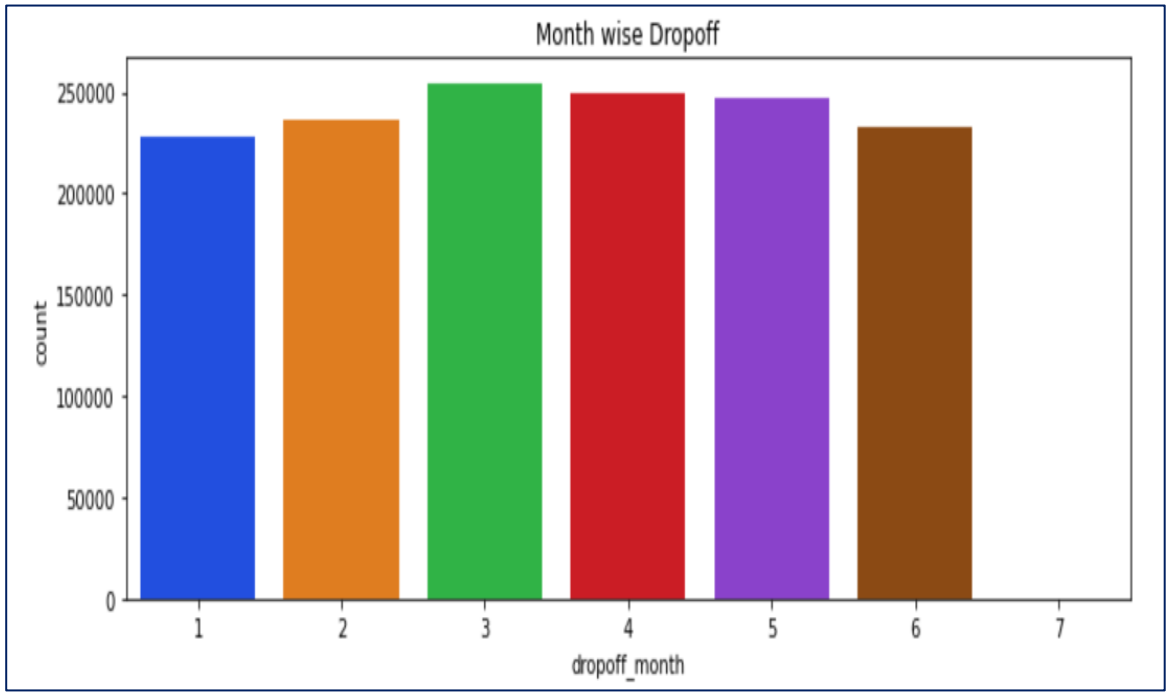
Passenger boxplot shows few Outliers.

Passenger Count	
1	1033540
2	210318
3	78088
4	59896
5	48333
6	28404

EDA

Pickup and Drop Off Month

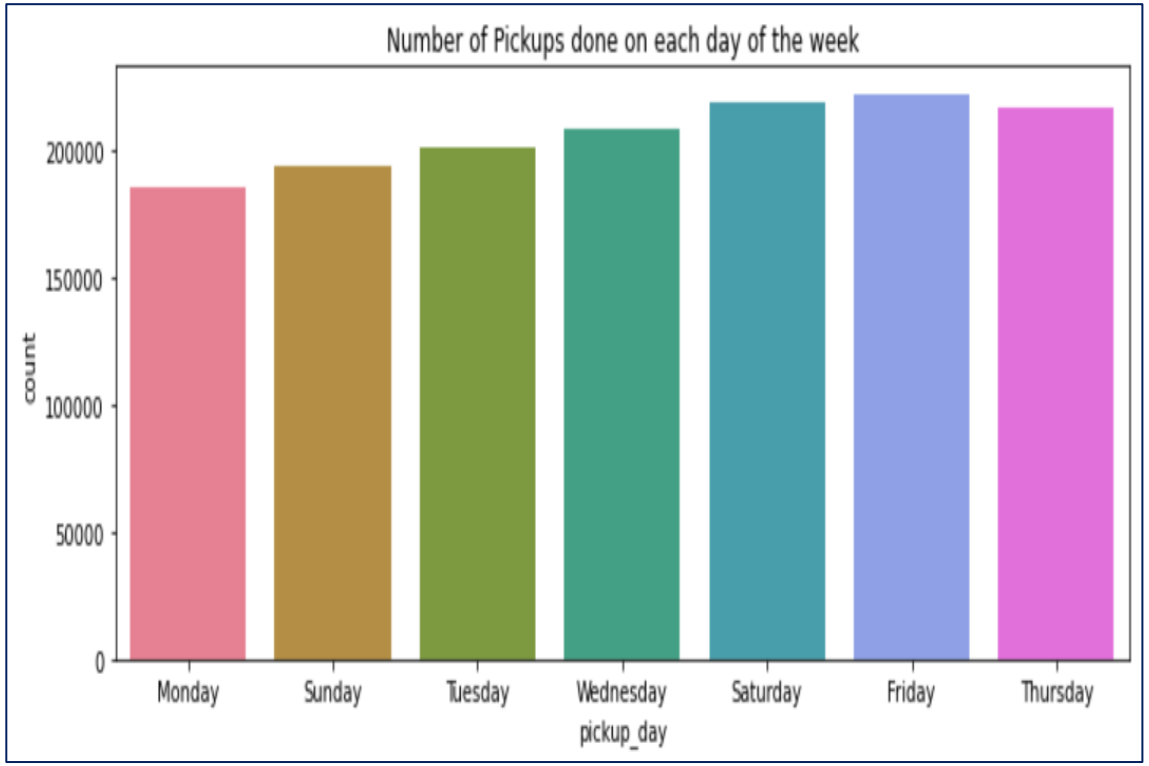


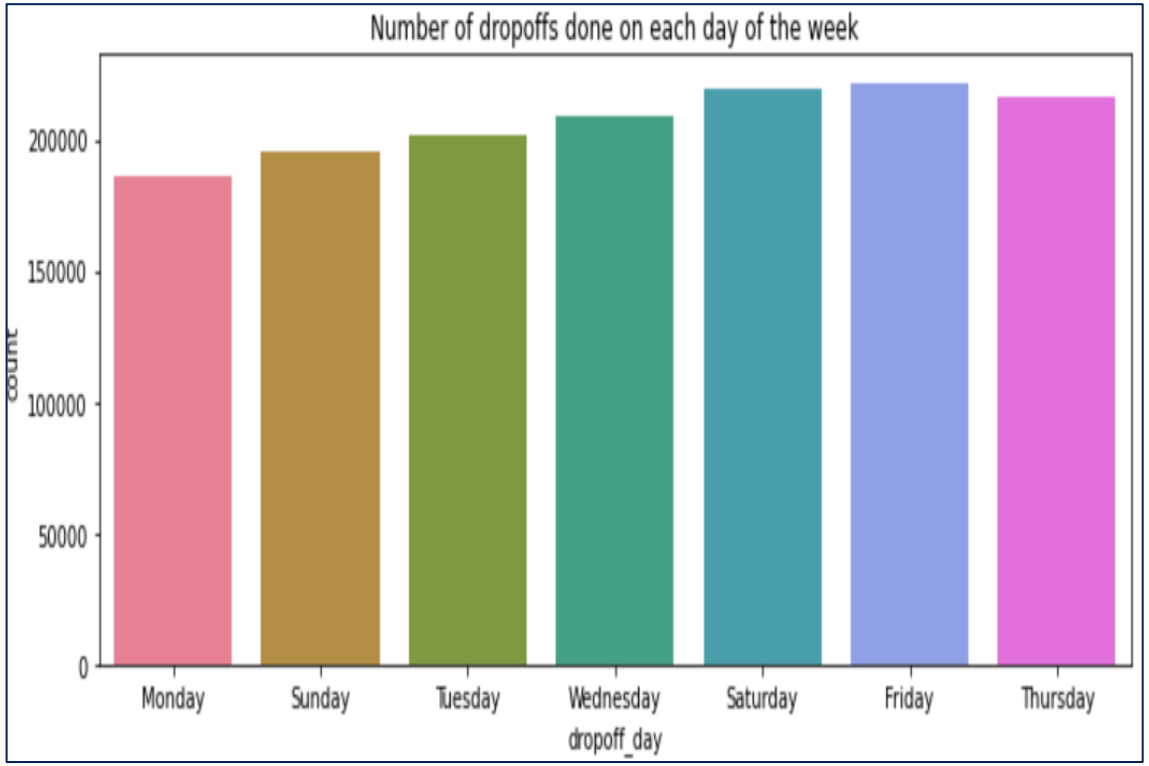


- Maximum Pickups and Drop-offs are from 3rd month.

EDA

Pickup Day

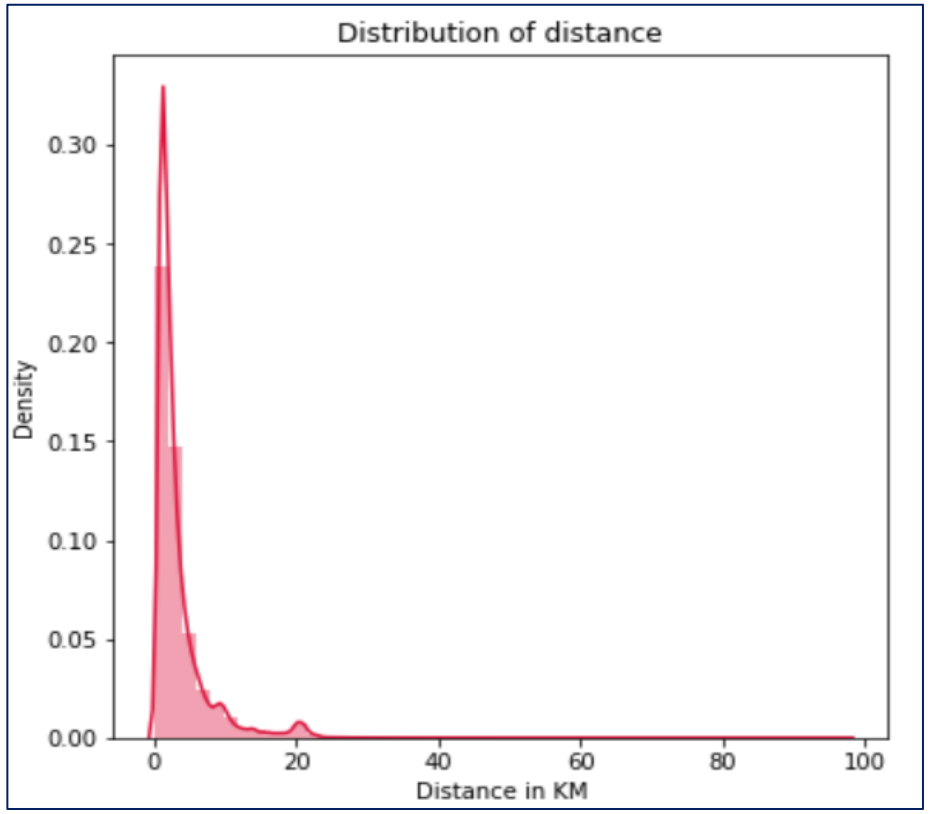


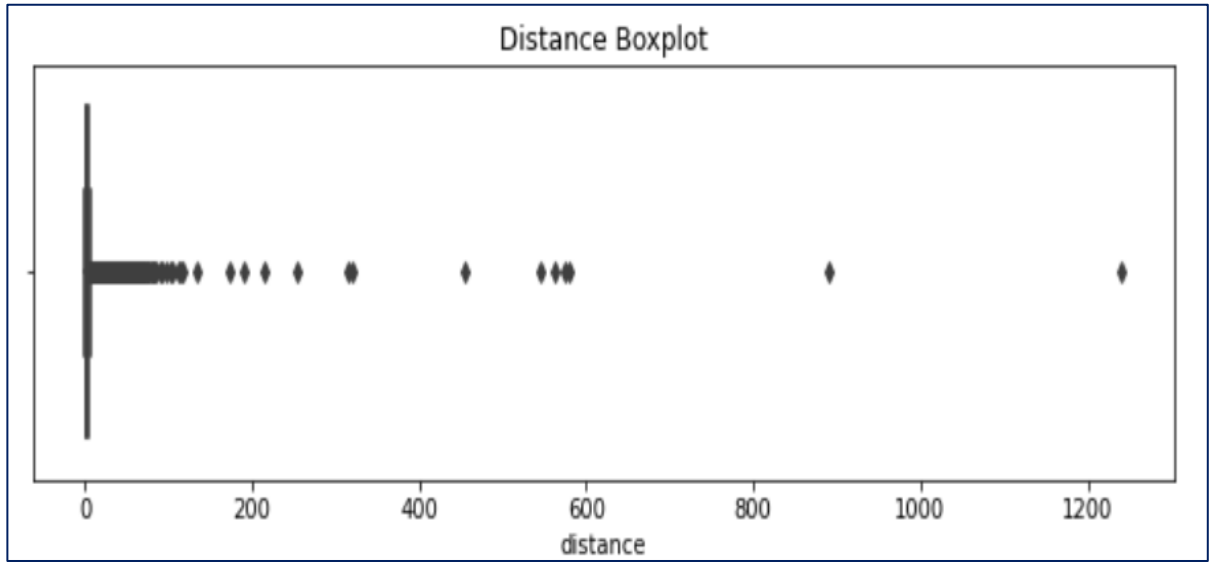


- This viz shows that on Saturday's pickups and Drop-offs are maximum.

EDA

Distance

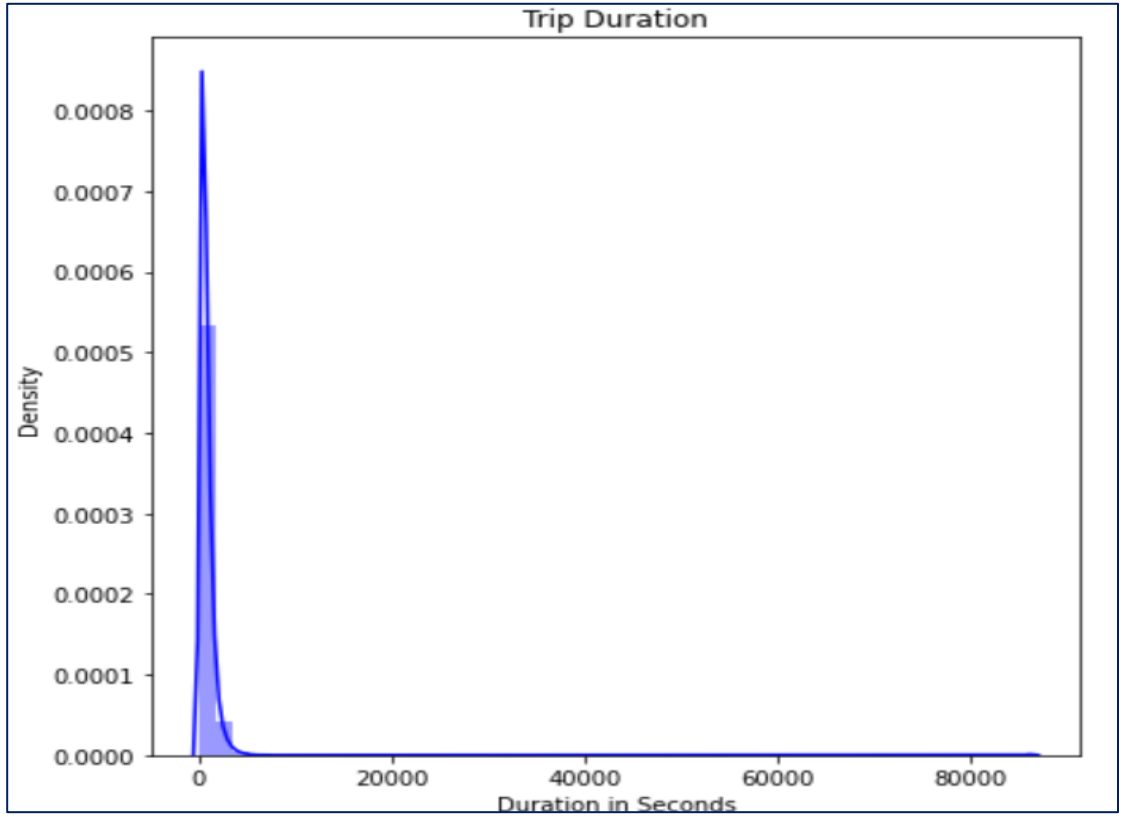


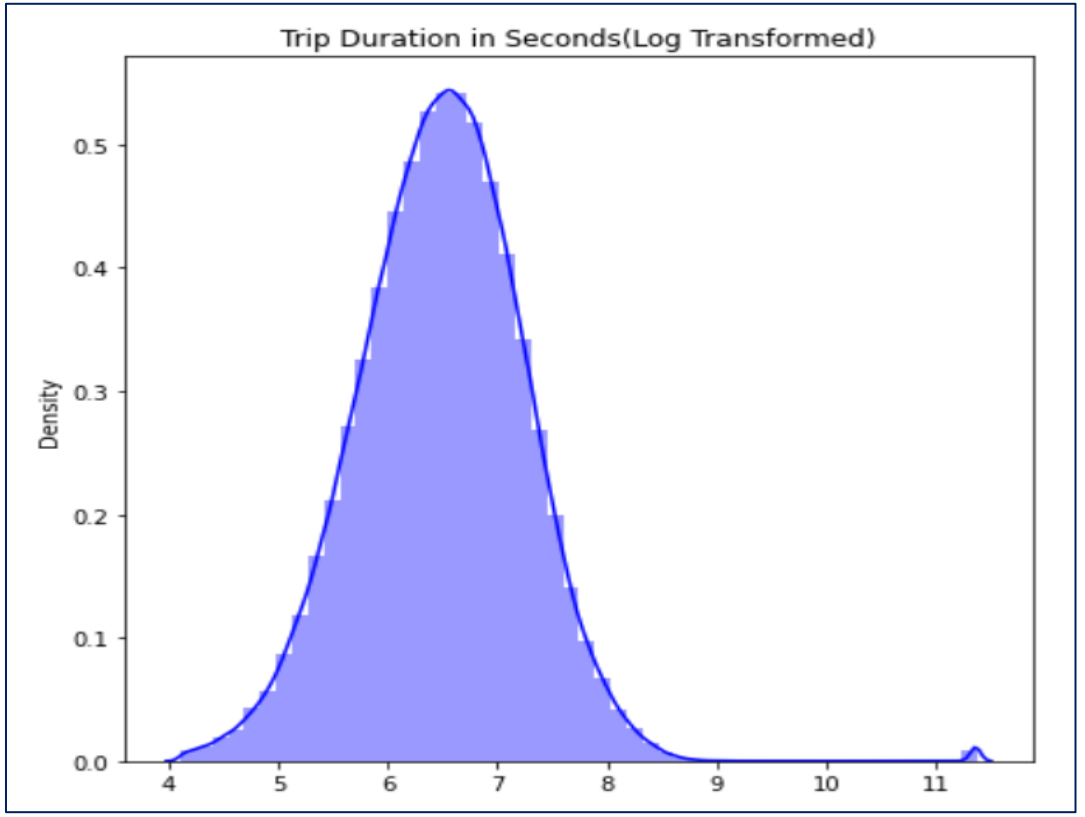


- Distribution of distance is highly right skewed.
- Distance boxplot showing there are many data points which covered big distance.

EDA

Trip Duration





- Trip duration given in seconds. It's right skewed distribution.
- Using log function it transformed into normal distribution.

EDA

Trip Duration Vs Distance

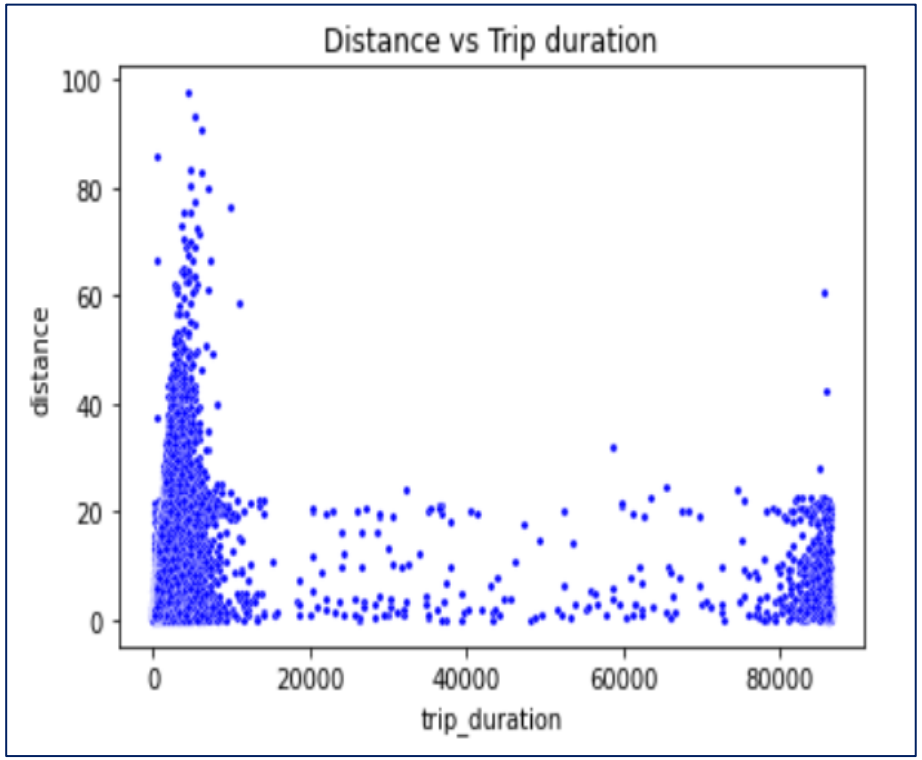


Fig. 1

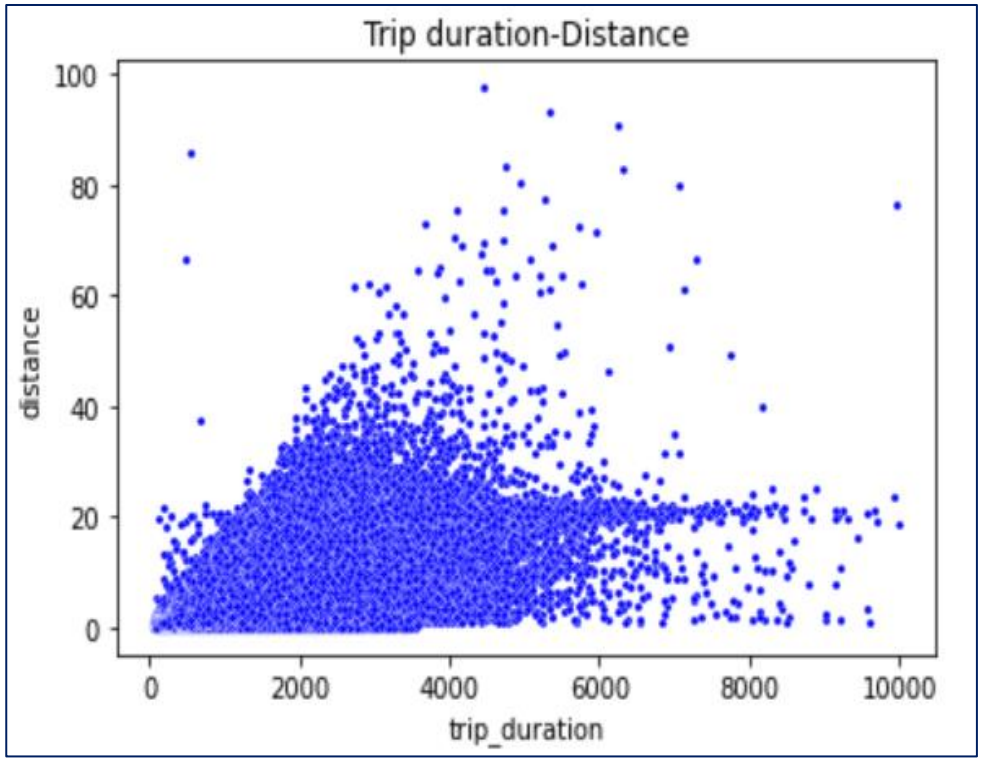


Fig. 2

- Trip duration vs distance (fig.1) shows that some of trips travelled very less distance but took lot of time.
- Fig. 2 after removing outliers and inconsistent records.

Applied Model's

Linear Regression : Linear Regression is a supervised machine learning algorithm, where the to be predicted output is continuous. This model has four assumptions. It's assuming a linear relationship between dependent and independent variables.

Decision Tree : Decision tree is a supervised learning algorithm that is mostly used in classification problems but it can also used for regression problems. Decision tree is considered to be a non-parametric method. It works for both categorical and continuous input and output variables.

Random Forest : Random forest is a Decision Tree based algorithm. It's a supervised learning algorithm. This algorithm can solve both type of problems i.e., classification and regression. Decision Trees are flexible and it often gets overfitted. To overcome this challenge Random Forest helps to make classifications more efficiently. It creates a number of decision trees from a randomly selected subset of the training set and averages the final outcome. Its accuracy is generally high. Random forest has ability to handle large number of input variables.

XG Boost : XG Boost is boosting algorithm based on decision Tree. Ensemble methods involve group of predictive models to achieve a better accuracy and model stability. XG Boost does parallel processing. It helps to reduce overfitting. XG Boost can handle missing values very well. It has inbuilt cross validation.

Evaluation Metrics

Mean Squared Error : It's most common and popular method. It finds averages of squared distance between the target variable and the value predicted by regression model.

MSE = 1/N (∑ (Truth Value – Predicted Value) ^2

RMSE : Root mean squared error corresponds to the square root of the average of squared difference between target values and values predicted by regression model.

RMSE = 1/N (∑ (Truth Value – Predicted Value) ^2

R2 Score : This metrics tell us that, 'How much of total variation in Y target variable is explained by the variation in X regression line.'

R2 = 1 – (Residual sum of squares / Total sum of squares)

Model’s Metric Evaluation

Model_Name	MSE_Train	MSE_Test	RMSE_Train	RMSE_Test	R2_Score_Train	R2_Score_Test
Linear regression	169569.92	168846.67	411.78	410.9	0.6054	0.6062
Decision Tree	235.63	216260.94	15.3	465.03	0.9994	0.4956
Decision Tree with Hyperparameters Tuning	108972.39	115174.7	330.1	339.37	0.7464	0.731
Random Forest	123321.15	123433.88	351.17	351.33	0.713	0.7121
XGBoost	98161.0	102544.87	313.3	351.33	0.7715	0.7608

- **XG Boost** predicted **good accuracy R2 score for both train and test,** comparatively to the other models.

Conclusion's:

1. Linear regression model performed poorly on both train and test dataset. It predicted 0.60 R2 score.

2. Decision Tree model overfitted and predicted 0.99 R2 score for Train and 0.49 for test dataset.

3. Decision tree model tuned using Hyperparameters and model's performance stabilized and it predicted 0.74 and 0.73 R2 score for train and test dataset.

4. Random forest predicted 0.71 and 0.71 R2 score for train and test.

5. **XG Boost** predicted good accuracy comparatively to the other models. It predicted **0.77 and 0.76 R2 score.**

End