Machine Learning Engineer Nanodegree

Capstone Project

Manoj Kumar Patra
04/11/2019

DEFINITION

Project Overview

New York Yellow Taxis do not offer a flat rate other than to certain airports. Fare charges are based on a meter counting time and distance. The meters are generally very accurate.
Base charges:

- Base fare – $2.50
- New York State tax surcharge – $0.50
- Additional charge from 4 pm to 8 pm on weekdays – $1.00
- Additional charge from 8 pm to 6 am everyday – $0.50

Once the meter is running:

- When the taxi is moving, the charge is $0.50 per one-fifth of a mile with speed being 6 miles per hour or more
- If the taxi is sitting still or must slow down due to traffic, the charge is $0.50 for 2 minutes of time stopped or travelling below 6 miles per hour

Airport fares:

- Flat fares from JFK to Manhattan – $52
- Fare to Newark Airport – Metered fare in addition to extra $17.50 surcharge, and also tolls for a tunnel

Therefore, it is always safe to have a good prediction of the taxi fare in New York.

Problem Statement

Given a training set of 55 million Taxi trips in New York since 2009 in the training data and 9914 records in the testing data, the goal is to predict the fare of a taxi trip with the help of information such as pick-up and drop-off locations, the pick-up date and time, and number of passengers travelling.

For the data, refer to point [3] under the section **REFERENCES**

Metrics

The metric used for this problem is Root Mean Square Error.

$$RMSE=\sqrt{\frac{\sum_{i=1}^{N}(Predicted_i-Actual_i)^2}{N}}$$

ANALYSIS

Data Exploration

There are 55M rows but for our purpose, we will consider 6M rows out of 55M

Available data fields are:

Input variables

ID

- ***key*** – Unique **string** identifying each row in both the training and test sets. Comprised of ***pickup_datetime*** plus a unique integer

Features

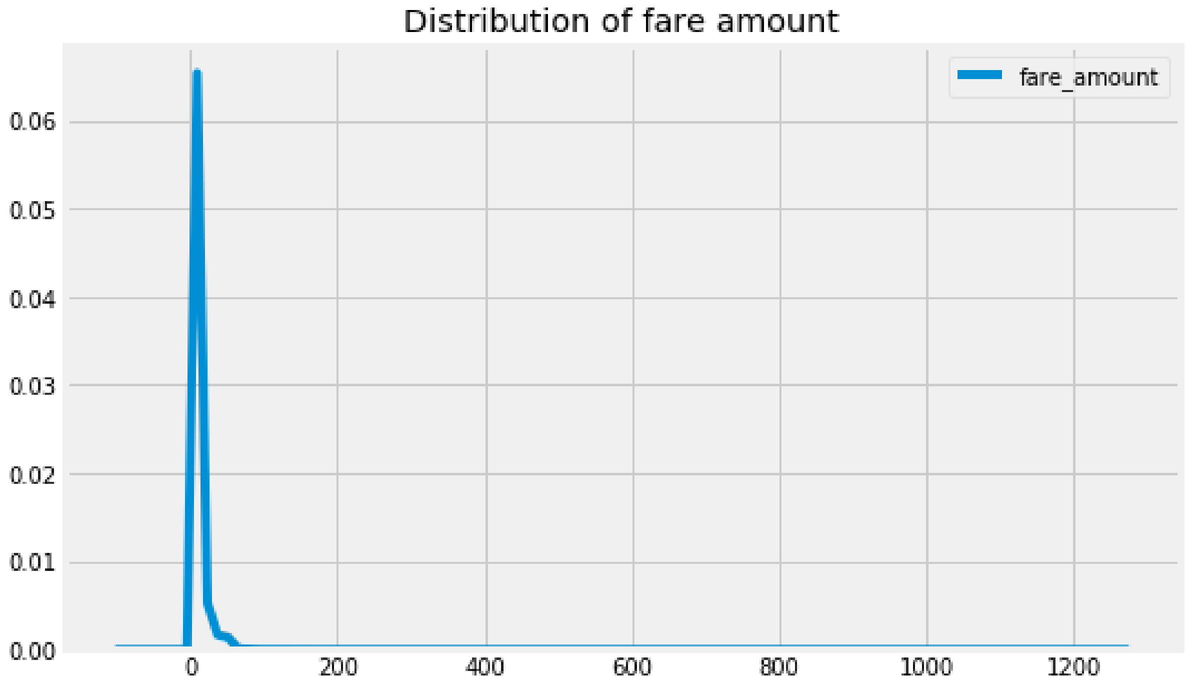
- ***pickup_datetime*** - **timestamp** value indicating when the taxi ride started.
- ***pickup_longitude*** - **float** for longitude coordinate of where the taxi ride started.
- ***pickup_latitude*** - **float** for latitude coordinate of where the taxi ride started.
- ***dropoff_longitude*** - **float** for longitude coordinate of where the taxi ride ended.
- ***dropoff_latitude*** - **float** for latitude coordinate of where the taxi ride ended.
- ***passenger_count*** - **integer** indicating the number of passengers in the taxi ride.

Target variable

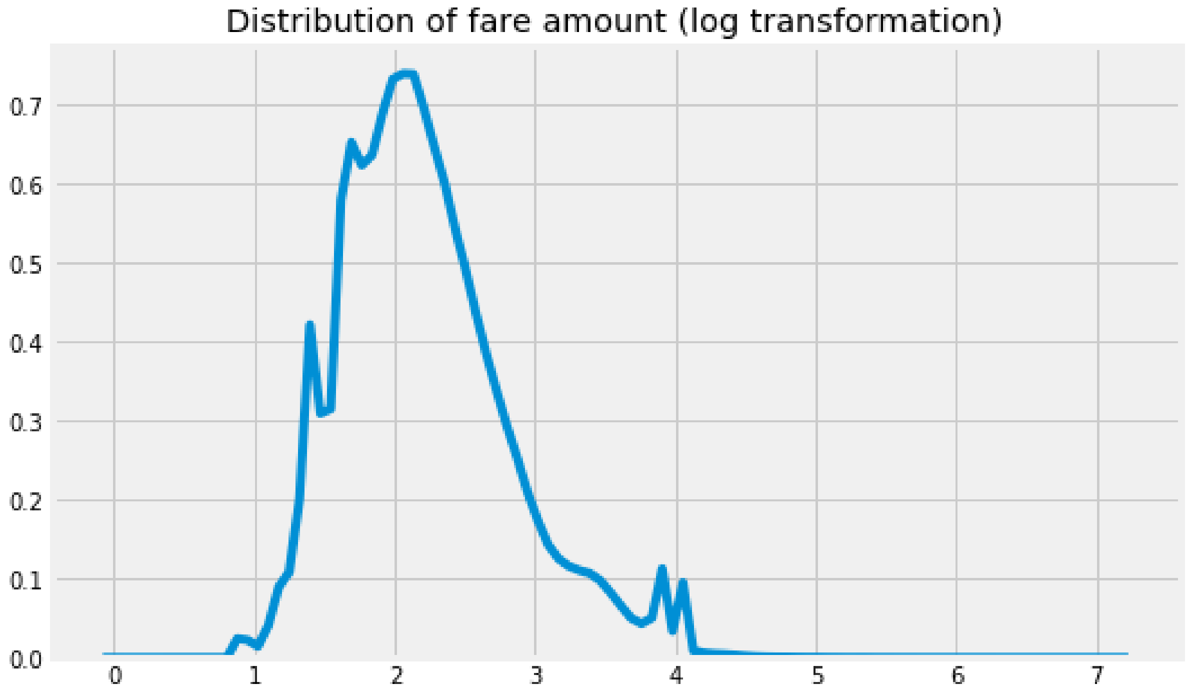
- ***fare_amount*** - **float** dollar amount of the cost of the taxi ride. This value is only in the training set; this is what you are predicting in the test set and it is required in your submission CSV.

Exploratory Visualisation

ANALYSIS OF FARE AMOUNT DISTRIBUTION



We found 262 records with negative fare amount in the data. Since, fare cannot be negative, we remove all records with negative fare amount or with fare amount of zero. Also, we did a log transformation on the distribution of fare amount to make it close to a normal distribution.



ANALYSIS OF TRIP FARE BASED ON PICKUP AND DROPOFF LATITUDES AND LONGITUDES

The central coordinates of New York are (**40.771133, -73.974187)**.

```
Range of coordinates for training set

-------------------------------------------
Range of Pickup Latitude is  (-3488.079513, 3344.459268)
Range of Dropoff Latitude is  (-3488.079513, 3345.9173530000003)
Range of Pickup Longitude is  (-3426.60895, 3439.425565)
Range of Dropoff Longitude is  (-3412.6530869999997, 3457.62235)


Range of coordinates for testing set

-------------------------------------------
Range of Pickup Latitude is  (40.573143, 41.709555)
Range of Dropoff Latitude is  (40.568973, 41.696683)
Range of Pickup Longitude is  (-74.252193, -72.986532)
Range of Dropoff Longitude is  (-74.263242, -72.990963)
```

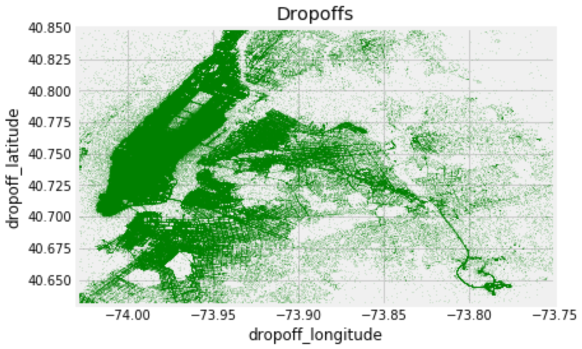
The range of coordinates as mentioned above indicate a lot of outliers in the training data. So, we removed all outliers from the training data based on the boundaries of the test data.

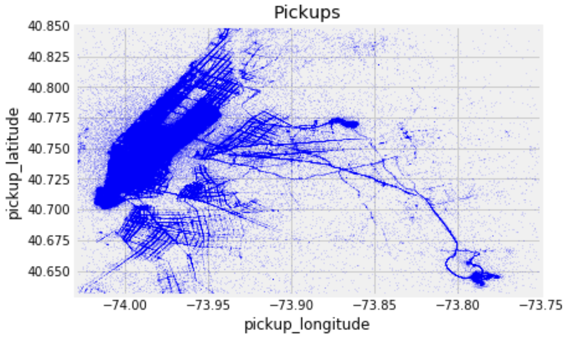
We found 114137 records with pick-up/drop-off location having latitude/longitude set to 0 and 1283 28 records with latitude and longitude outside the boundary of the test set. All such records were dropped from the training data.

ANALYSIS OF TRIP FARE BASED ON MAJOR AIRPORTS IN NEW YORK

Coordinates of major airports in New York are as follows:

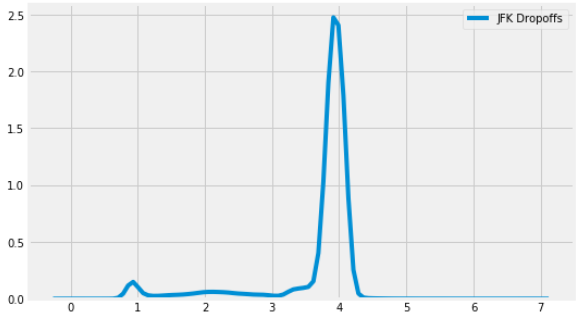
- Coordinates of Newark Airport = 40.6895 degrees North, 74.1745 degrees West
- Coordinates of JFK Airport = 40.6413 degrees North, 73.7781 degrees West
- Coordinates of La Guardia Airport = 40.7769 degrees North, 73.8740 degrees West

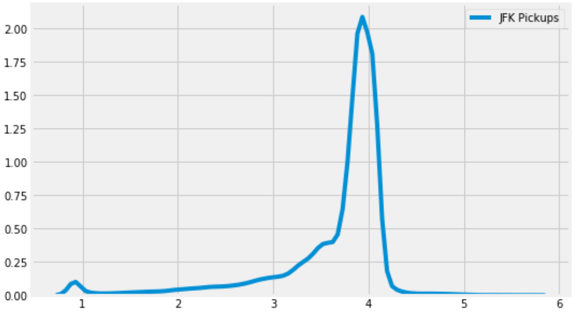




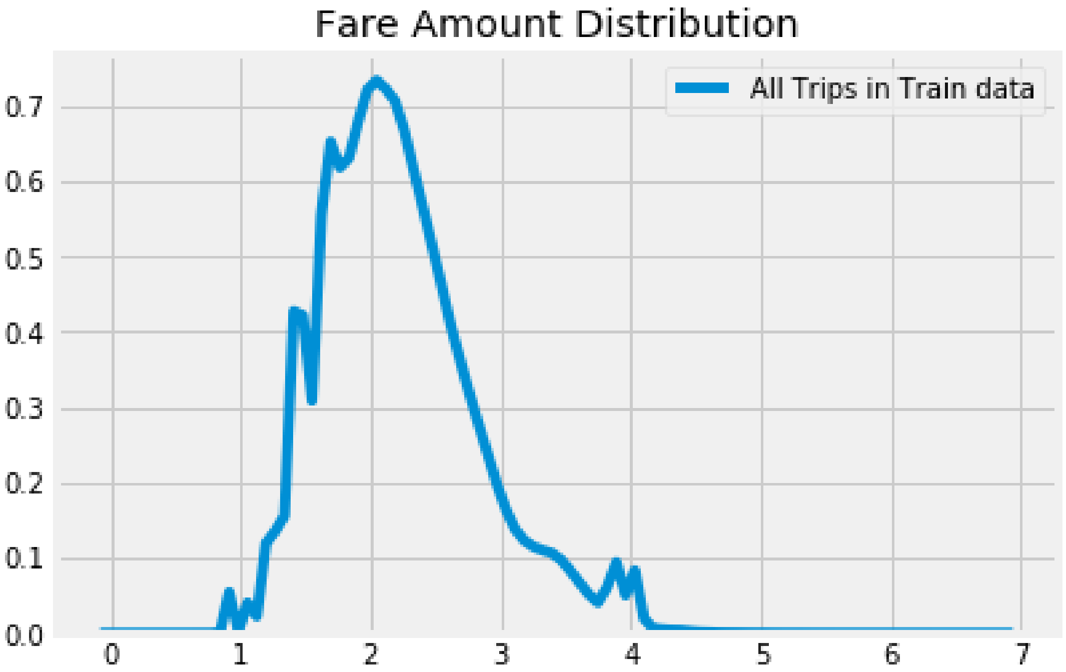
From the above visualizations, we realise heavy pick-up and drop-off near the three airports.

We also tried to check if the fare to/from the airport is always high. We did this visualization for the JFK airport.



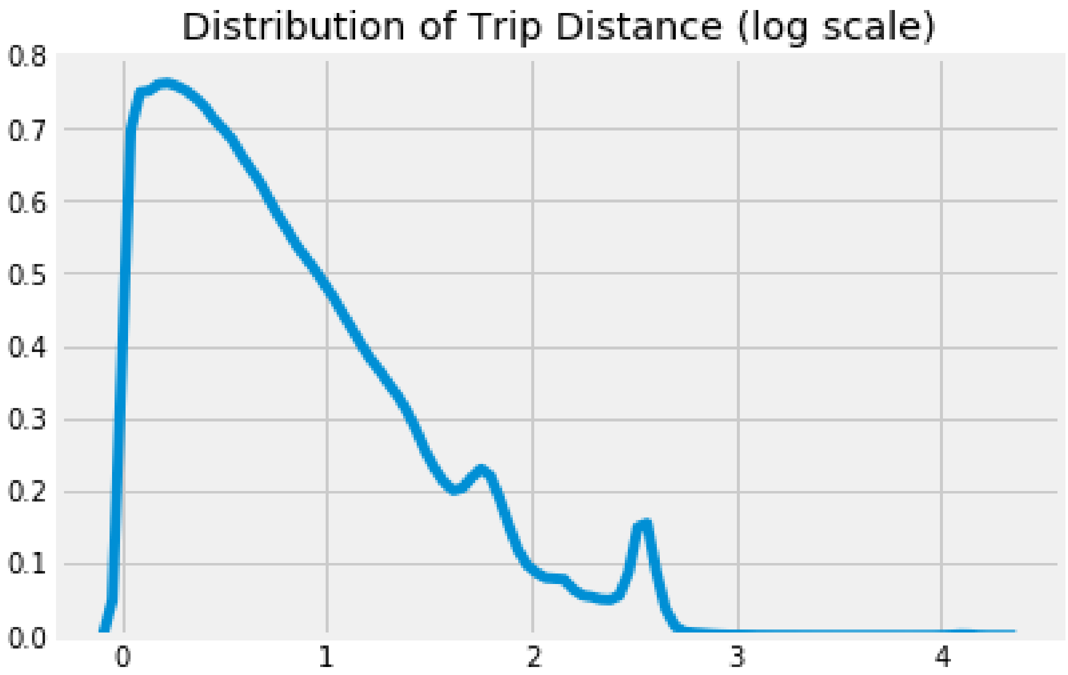


Then we tried comparing fares related to JFK airport with fare amount distribution across all trips in train data.



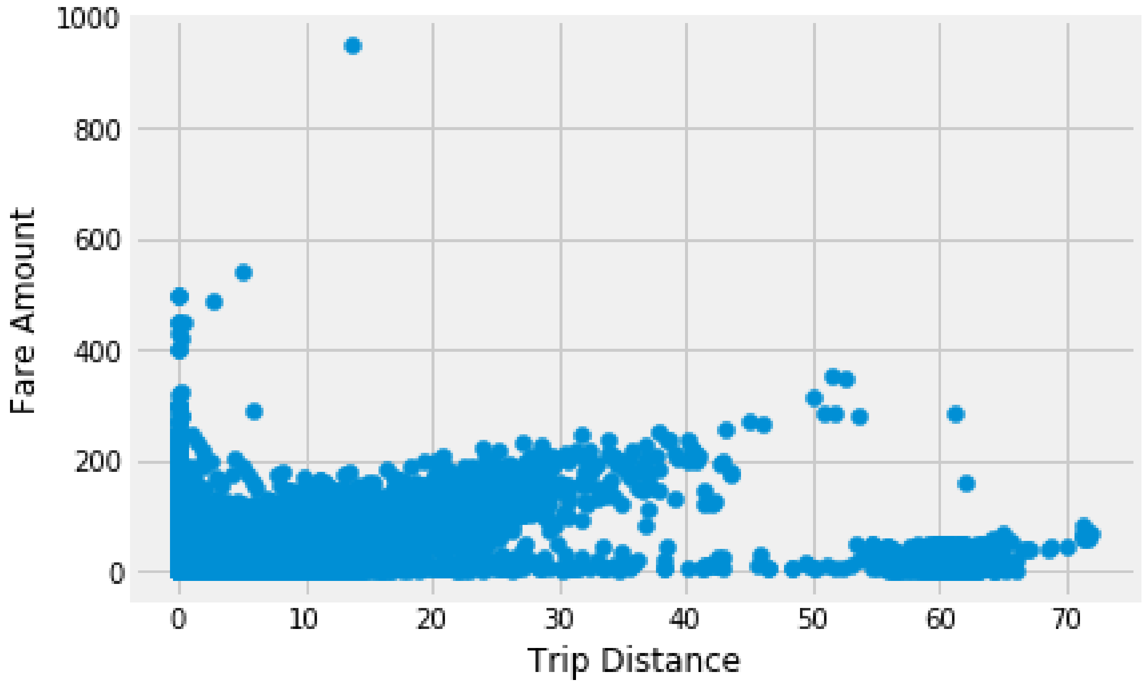
We realised that the average fare amount is much higher when pick-up and drop-off is related to JFK airport.

ANALYSIS OF TRIP FARE BASED ON TRIP DISTANCE



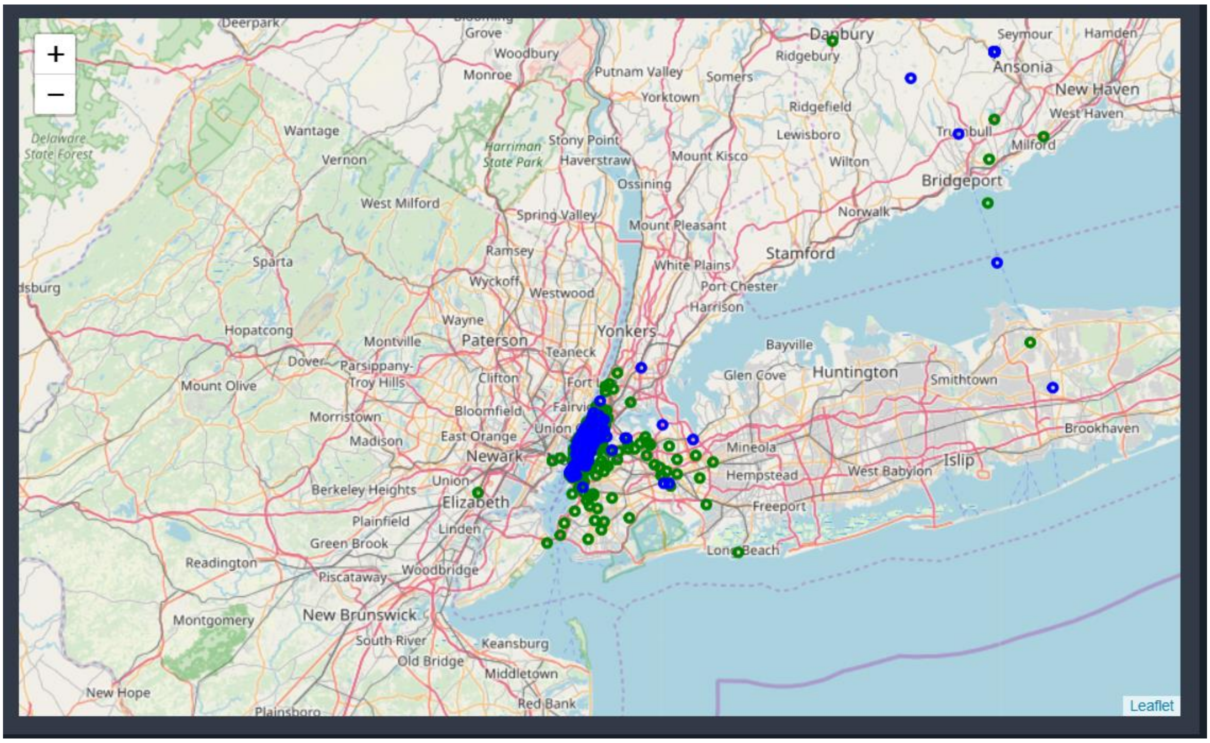
Trips within 0 to 1 mile have the highest distribution.

We visualised a scatter plot of *fare amount* vs. *trip distance*.

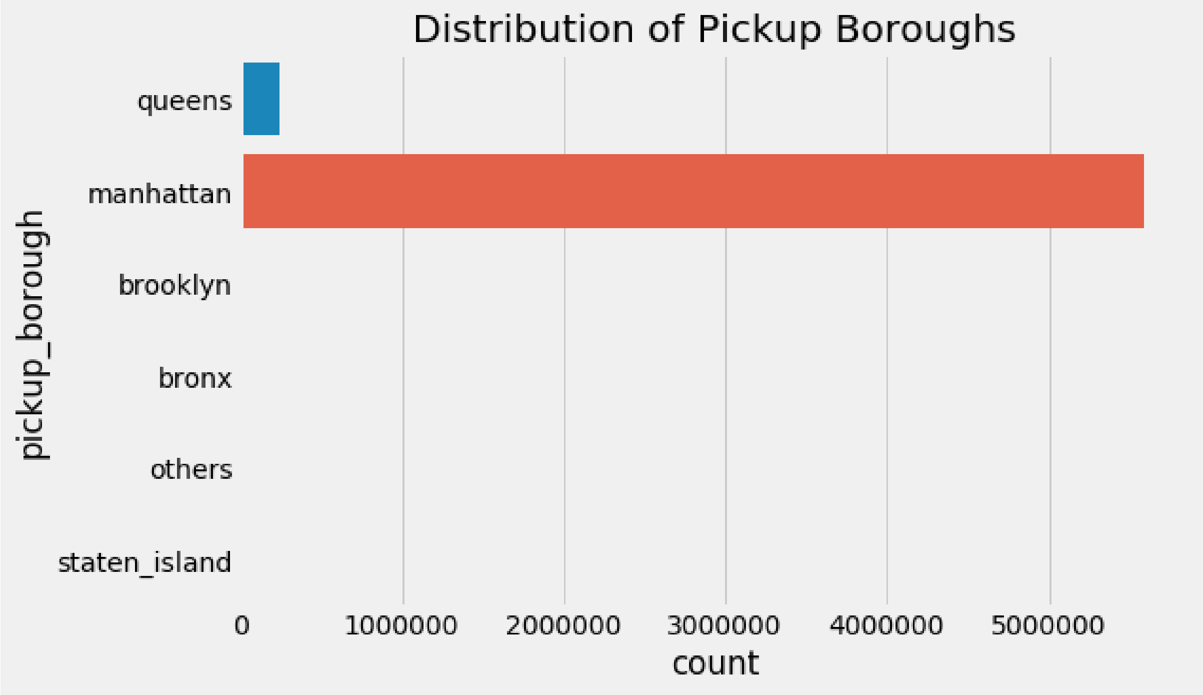


From the scatter plot above, we conclude that the fare is approximately fixed for trip distances greater than 50 miles.

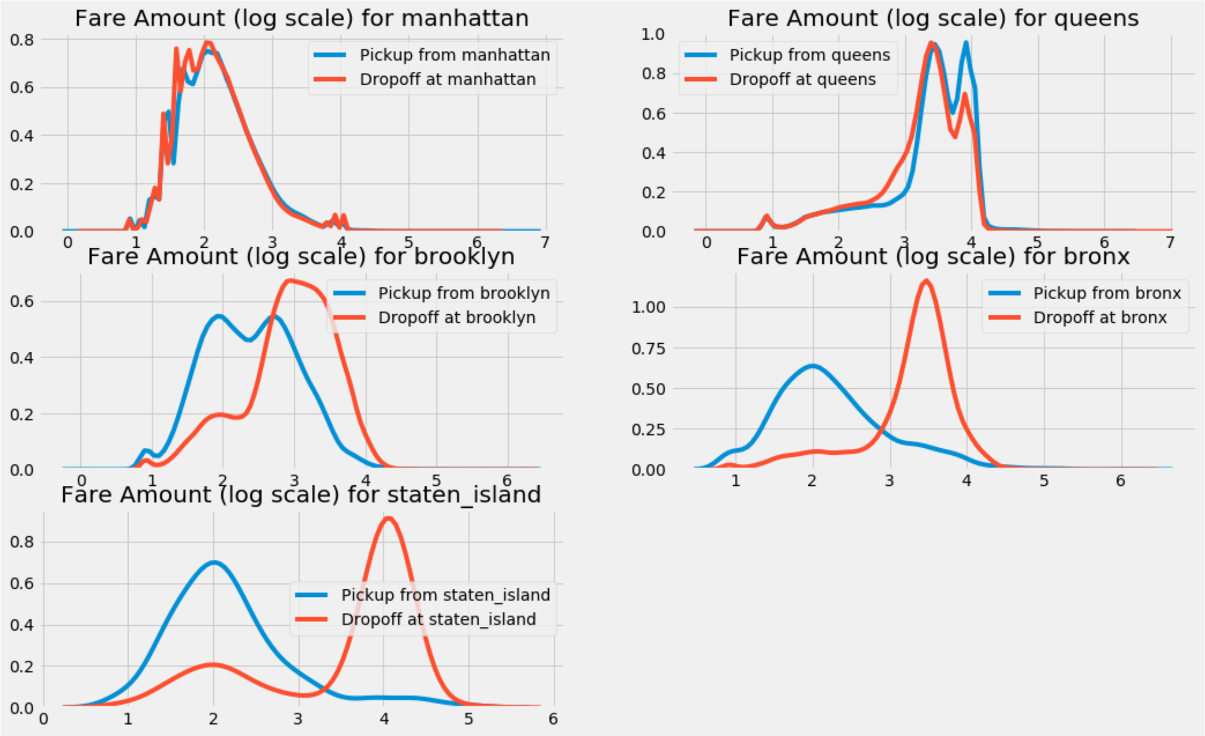
When we visualised the data on a map, we realised that the trips are mostly distributed across lower Manhattan.



New York encompasses five county-level administrative divisions called **boroughs**: *The Bronx, Brooklyn, Manhattan, Queens,* and *Staten Island.* So, we tried visualizing the data by grouping into each borough.



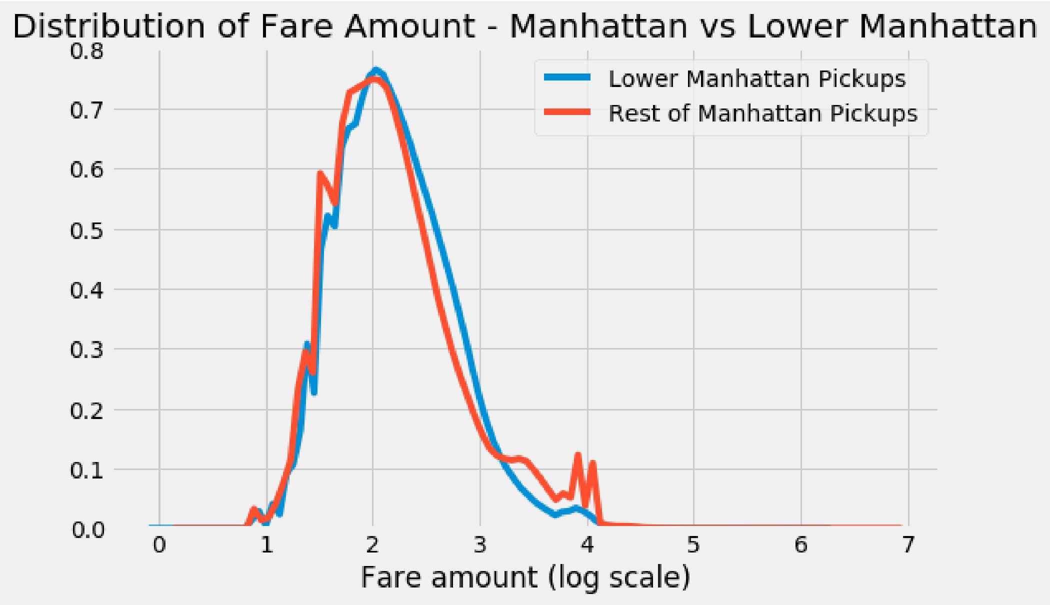
As we see, Manhattan has the largest number of trips whereas Queens has the lowest.



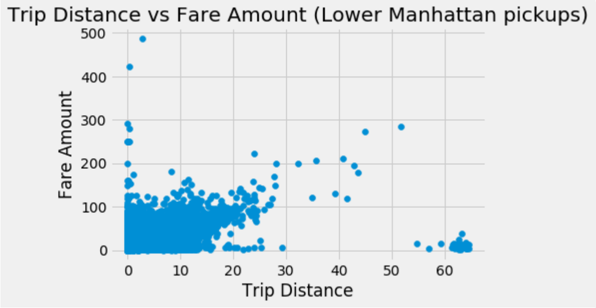
We conclude the following from the above graphs:

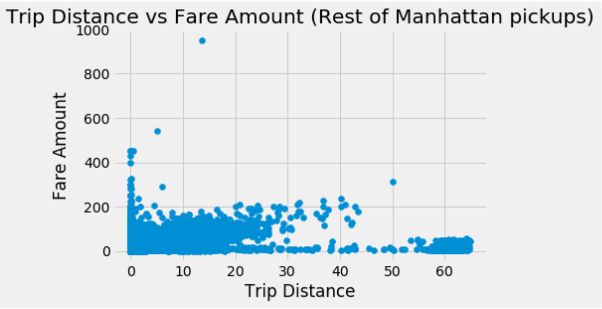
1. Pick-up and drop-off fare amounts have a similar distribution for Manhattan.
2. Drop-off fare amount has higher distribution for Staten Island and Bronx, which means trips to these places take more time.

ANALYSIS OF TRIP FARE BASED ON WHETHER PICK UP AND DROP OFF LOCATION IS IN LOWER MANHATTAN OR IN REST OF MANHATTAN



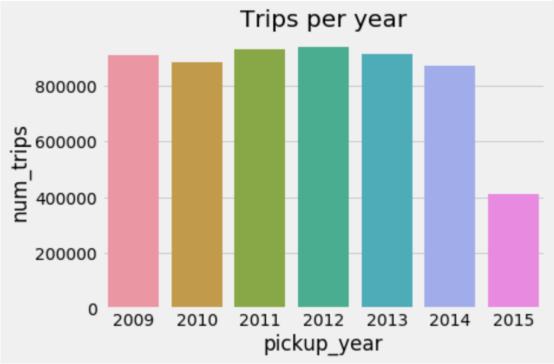
Distribution of fare is almost the same for trips in lower Manhattan and the rest of Manhattan.

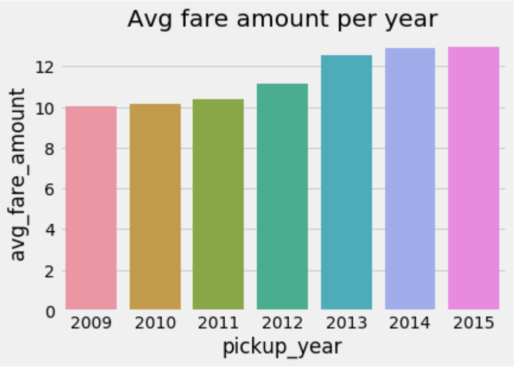




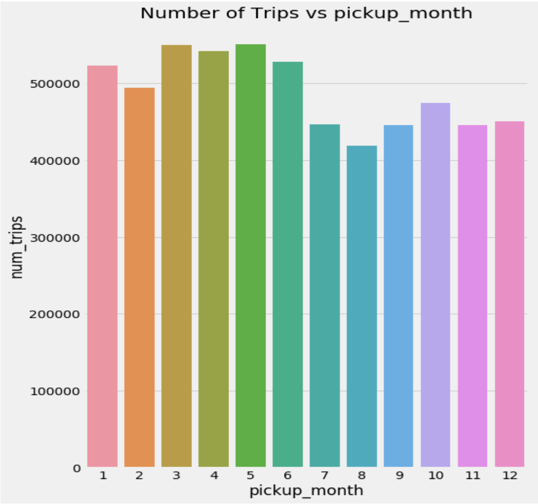
We conclude that the trip cost is relatively lower for the same distance in rest of Manhattan.

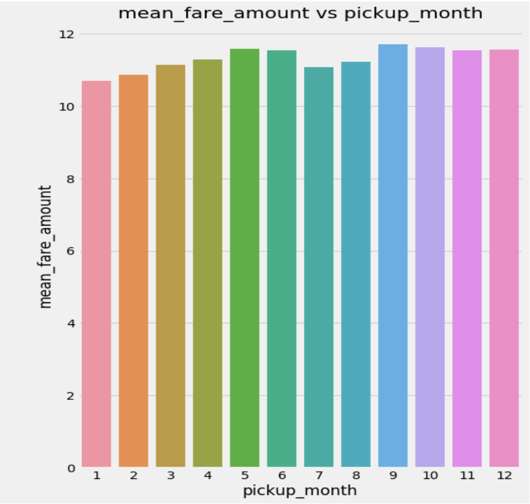
ANALYSIS OF TRIP FARE BASED ON DATE AND TIME FEATURES





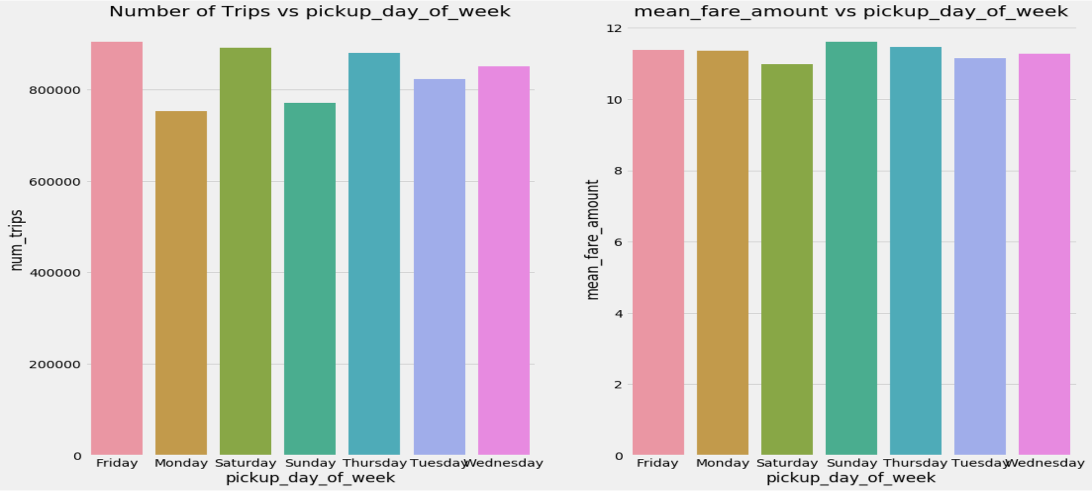
The most number of trips were in the year 2011 and 2012, in both cases for a distance greater than 0.8 miles. The average fare continues to increase over the years.





From the above graphs, we conclude:

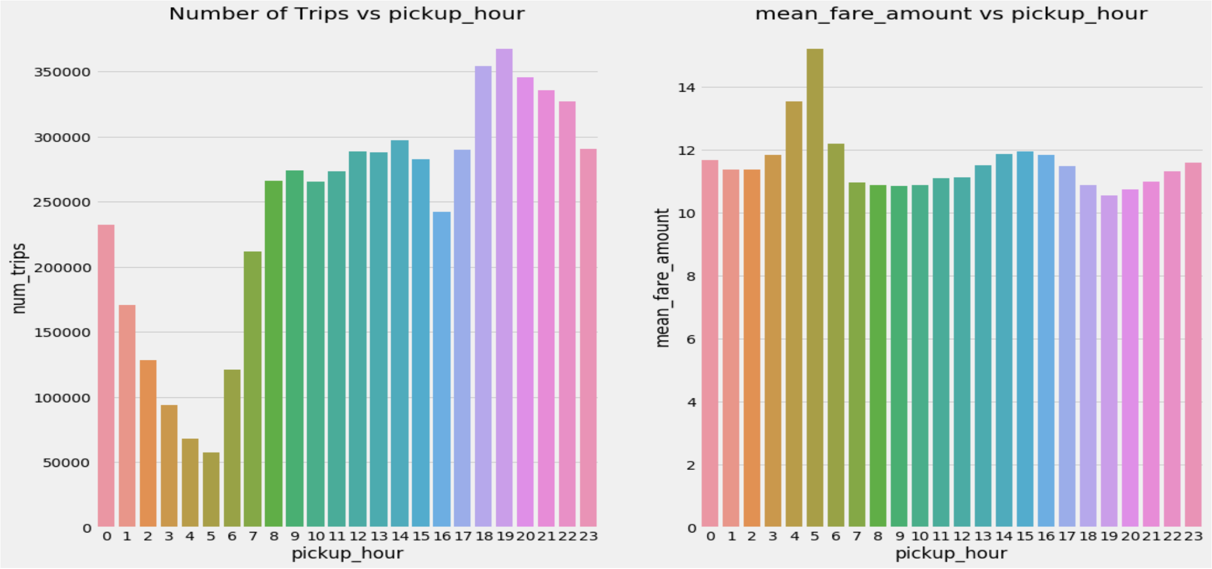
1. Number of trips are less from June till December
2. Trips are high in the months of March and May
3. The fare distribution is approximately consistent



From the above graphs, we conclude:

1. Saturday has the lowest average fare amount but also has the highest number of trips.

2. On Monday and Sunday, the number of trips are low, but the average fare amounts are higher.



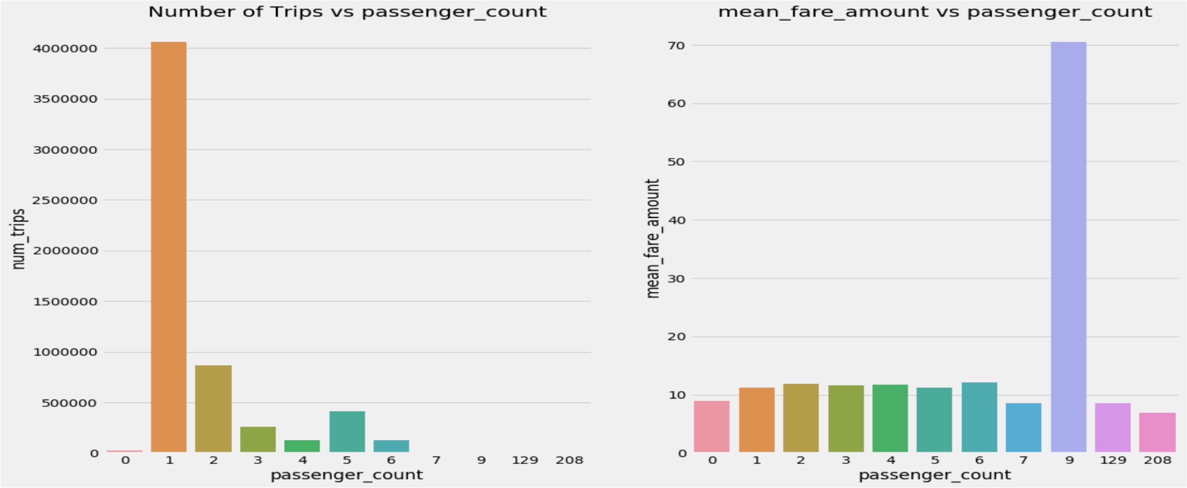
From the above graphs, we conclude:

1. The number of trips are least at 5 in the morning but the average trip fare is the highest at the same time.

2. Between 1 am to 5 am, the number of trips decrease and the average fare amount increases.

3. The highest number of trips is at 7 in the evening but the average fare amounts are the lowest.

ANALYSIS OF TRIP FARE BASED ON NUMBER OF PASSENGERS



```
Trips with 0 passengers =  20719
Trips with 9 passengers =  2
Trips with 129 passengers =  1
Trips with 208 passengers =  2


Trips with 9 passengers in test data =  0
Trips with less than 9 passengers in test data =  9914
```

From the above graphs, we conclude:

1. There are 20719 records where number of passengers is 0 but the trips have a fare more than 0. This might be because the passenger was charged a cancellation fee.

2. The price is highest when there are 9 passengers and only two such records exist.

3. Trips with 129 and 208 passengers are definitely outliers.

4. The maximum number of passenger count in the test data is 8 per trip. So, we can safely remove all trips with passenger count more than 8 from the training data set.

Algorithms and Techniques

The given dataset is a typical supervised learning problem. Algorithms used are:

1. **Linear Regression** – For model evaluation and comparison
2. **Random Forest** – For model evaluation and comparison
3. **LightGBM** – For model evaluation and comparison

Benchmark

A baseline model is a solution to a problem without applying any machine learning techniques. For our problem, we calculated the average fare amount which came out to be **9.61117614**. Our goal was then to come up with a model that had a lower RMSE score.

METHODOLOGY

Data Pre-processing

The data contains abnormalities. So, the pre-processing steps carried out include:

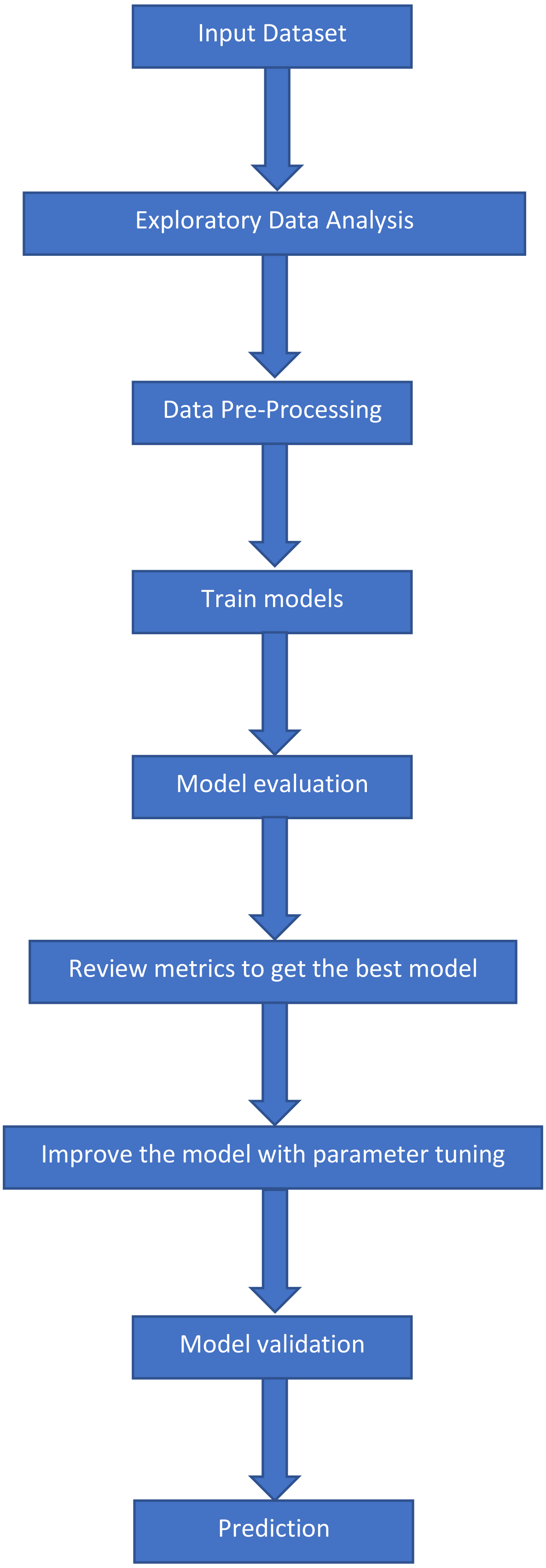
1. Removal of records with fare amount less than zero
2. Removal of records with passenger count more than 8
3. Removal of records which lie outside the boundaries of NYC

   We define the boundaries of NYC as:

```
Boundary = {
         'min_lng': -74.263242,
         'min_lat': 40.573143,
         'max_lng': -72.986532,
         'max_lat': 41.709555
}
```

Implementation

The project follows typical predictive analysis hierarchy as shown below:


Input Dataset
Exploratory Data Analysis
Data Pre-Processing
Train models
Model evaluation
Review metrics to get the best model
Improve the model with parameter tuning
Model validation
Prediction

Refinement

We will prepare the data by splitting feature and target columns. The split is done with 80% of the data to be used for training and 20% of the data for validation.

We used three types of models:

- **Linear Regression**

```
Test RMSE for Linear Regression is  8.14028848424604
Train RMSE for Linear Regression is  8.20954077875561
Variance for Linear Regression is  0.06925229450956927
RMSE for Linear Regression is  8.14028848424604
```

- **Random Forest**

```
Test RMSE for Linear Regression is  3.722274
Train RMSE for Linear Regression is  1.412137
Variance for Linear Regression is  -2.310136
RMSE for Random Forest is  3.722273669463352
```

- **LightGBM**

```
Test RMSE for Linear Regression is  3.793030
Train RMSE for Linear Regression is  3.071476
Variance for Linear Regression is  -0.721555
RMSE for Light GBM is  3.7930303182354526
```

In order to understand the variance in the model and get the best model, we compare the train RMSE and test RMSE for each model:

MODEL	**TEST RMSE**	**TRAIN RMSE**	**VARIANCE**
Linear Regression	8.140288	8.209541	0.069252
Random Forest	3.722274	1.412137	-2.310136
LightGBM	3.793030	3.071476	-0.721555

We conclude from the above table, that Linear Regression did not perform very well. Random Forest performed better on the train data among the three, but it’s performance is comparable to

LightGBM's for test data. The variance is higher for Random Forest which indicates it is overfitting. So, we went ahead with LightGBM.

LIGHTGBM

Light GBM is a gradient boosting framework that uses tree based learning algorithm.

Light GBM grows tree vertically (leaf-wise) rather than horizontally (level-wise). It will choose the leaf with maximum delta loss to grow. When growing the same leaf, leaf-wise algorithm can reduce more loss than a level-wise algorithm.

Reasons for choosing Light GBM:

1. Performed better than other models
2. High speed
3. Handle large size of data
4. Takes lower memory to run
5. Focus on accuracy of results
6. Supports GPU learning

FEATURE ENGINEERING

Feature engineering is the process of transforming raw data into features that are input to the final model.

We did feature engineering by creating new columns for *train* and *test* data that indicate:

7. Pick up from any of the airports (La Guardia, EWR, JFK)
8. Drop off at any of the airports
9. Pick up from lower Manhattan
10. Drop off at lower Manhattan
11. Pick up distance to any of the airports
12. Drop off distance from any of the airports

13. Pick up distance to any of the five boroughs (Queens, Brooklyn, Bronx, Staten Island, Manhattan)

14. Drop off distance from any of the five boroughs

CROSS VALIDATION

We trained the model with a 5-fold cross validation, 5000 boosting iterations and early stopping rounds set to 20. Early stopping is used to stop training if one metric of one validation data doesn't improve in last ***n*** rounds(***n*** is the number of early stopping rounds).

TRAINING PARAMETERS

```
param = {'num_leaves':31, 'num_trees':5000, 'objective':'regression'}

param['metric'] = 'l2_root'
```

Training after feature engineering along with above mentioned parameters gave the following results:

```
RMSE for Light GBM with Feature Engineering is 3.6292832017557037

Train RMSE for Light GBM with Feature Engineering is 2.871387100198267

Variance of Light GBM with Feature Engineering is -0.7578961015574368
```

MODEL TUNING

We use grid search to find the best hyper parameters for Light GBM.

PARAMETER	DESCRIPTION	VALUES	BEST VALUE
learning_rate	Determines the impact of each tree on the final outcome. GBM works by starting with an initial estimate which is updated using the output of each tree. The learning parameter controls the magnitude of this change in the estimates. Typical values: 0.1, 0.001, 0.003...	[0.1, 0.75, 0.005, 0.025]	0.1
num_leaves	Number of leaves in full tree	[31, 60]	60
baggin_freq	Frequency for creating a new bag	[10, 20]	10

bagging_fraction	The fraction of data to be used for each iteration and is generally used to speed up the training and avoid overfitting	[0.85, 1, 0.9, 0.95]	0.85
boosting_type	Type of boosting (gbdt stands for gradient boosting)	[‘gbdt’]	‘gbdt’
max_depth	Max depth of the tree	[-1, 6, 5]	-1

RESULTS

Model evaluation and Validation

```
Validation RMSE after tuning 3.617727396580376

Train RMSE after tuning 3.084503221454753

Variance of model -0.533224175125623
```

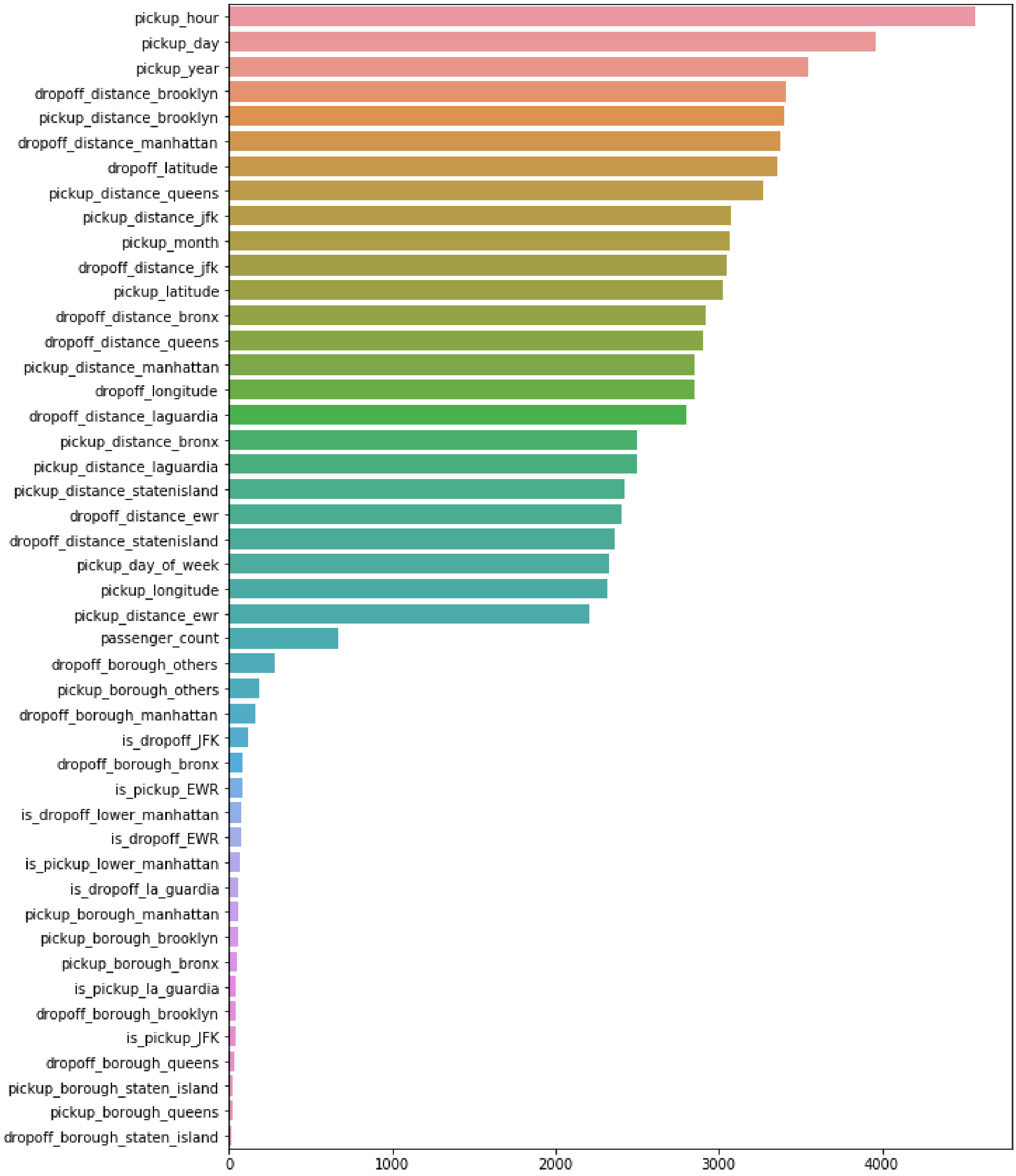
Justification

The variance in the model improved from 0.72155 to 0.53322. The test RMSE score also improved from 3.793030 to 3.61772.

There is room for improvement but it’s out of the scope of this report.

CONCLUSION

Free-form Visualization



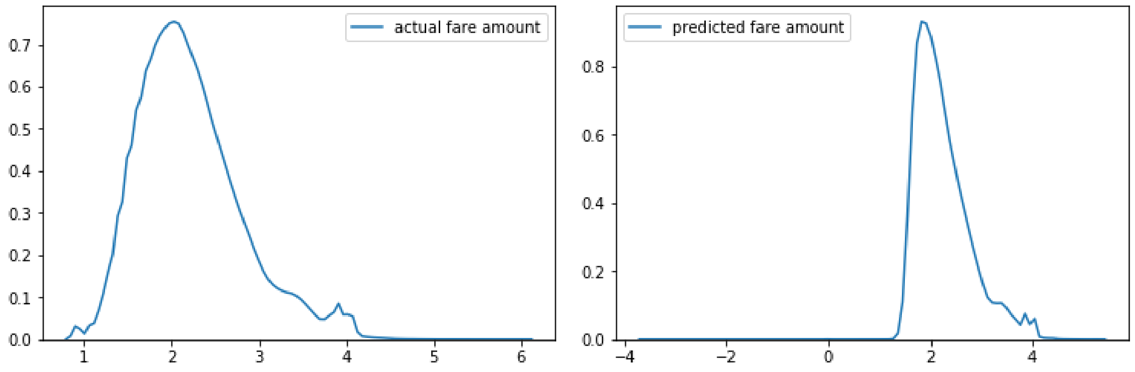
Reflection

Data pre-processing took a long time but it was also the most important part of the problem. We then evaluated three types of models. On comparing these models, we found Light GBM to outperform others. So, we continued with Light GBM. We improved the model with feature engineering and parameter tuning.

Improvements

We chose to use Light GBM and improved upon using feature engineering and parameter tuning with the use of Grid search.

FINAL PREDICTIONS



REFERENCES

[1] https://www.kaggle.com/c/new-york-city-taxi-fare-prediction

[2] https://lightgbm.readthedocs.io/en/latest/

[3] Data for this problem: https://www.kaggle.com/c/new-york-city-taxi-fare-prediction/data