# MACHINE LEARNING ENGINEER NANODEGREE

## CAPSTONE PROJECT

Satya Raghava Dilip Bhattiprolu
November 10th, 2018

# I. DEFINITION

## PROJECT OVERVIEW

Supervised learning is defined as a mapping function for set of input variables to a given output variable. The goal behind obtaining the mapping function is to predict any output for given set of new input. There are 2 kinds of supervised learning problems:

Classification – A classification problem is to categorize the outputs into set of categories such as is the person “short” or “tall”.

Regression – A regression problem is to predicting actual value of the output such as predicting actual height of person.

New York City is one of the most populous cities in the United States. For an overly crowded place like this, traffic has been a major concern for over several years now. The transportation system here is a network of complex infrastructure that has one of the largest subway systems in the world, an aerial tramway, a vehicular tunnel etc. Taxis account for a significant contribution towards the daily commute of people residing in NYC. There was a survey conducted to know the number of taxi trips per day and it turned out to be a whopping number of around 850,000 trips per day through taxis. This number accounts to approximately $1/10^{th}$ of the total population in New York City.

This project is a regression problem which predicts the fare amount for a taxi ride in New York City based on set of inputs.

## PROBLEM STATEMENT

This project aims to predict the estimated fare for a taxi trip in New York City for the given pickup and drop off locations. There are several factors that could influence the estimated fare and some of these include the point of pickup, destination, peak hours, high booking requests at the respective location, low availability of drivers, weather, number of passengers, time of booking request etc.

Following are the tasks involved in this project:

- Downloading the train and test dataset from Kaggle.
- Data preprocessing.
- Dividing the input data into training and validation set.
- Training the classifier to predict the taxi trip fare.
- Calculating the Root Mean Square Error for training and validation set.
- Applying the model to predict the fares from test file.

## METRICS

Mean Absolute Error (MAE) and Root mean squared error (RMSE) are two of the most common metrics used to measure accuracy. Both MAE and RMSE express average model prediction error in units of the variable of interest. Since the errors are squared before they are averaged, the RMSE gives a relatively high weight to large errors. This means the RMSE should be more useful when large errors are particularly undesirable.

Since we don't want large errors we use RMSE as the metric to choose the model.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2}$$

# II. Analysis

## Data Exploration

Following are the input features and target variable from the given test data set:

Features:

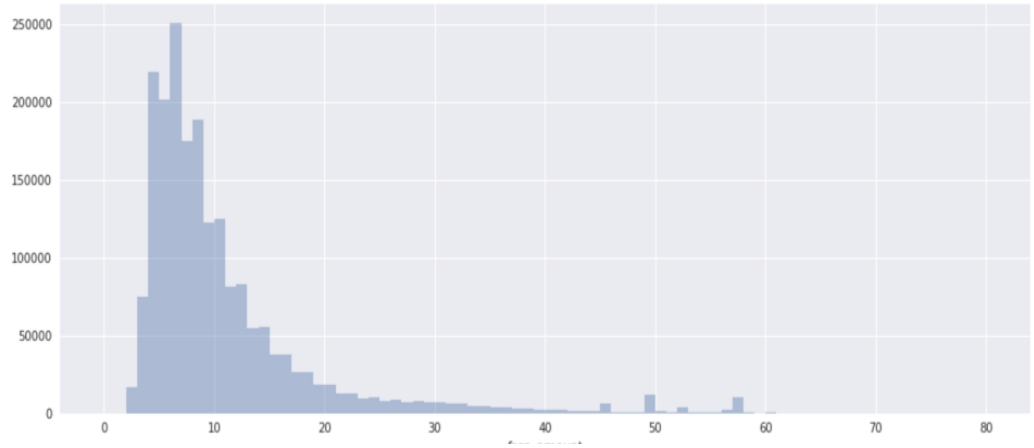
- pickup_datetime - timestamp value indicating when the taxi ride started.
- pickup_longitude - float for longitude coordinate of where the taxi ride started.
- pickup_latitude - float for latitude coordinate of where the taxi ride started.
- dropoff_longitude - float for longitude coordinate of where 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:

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

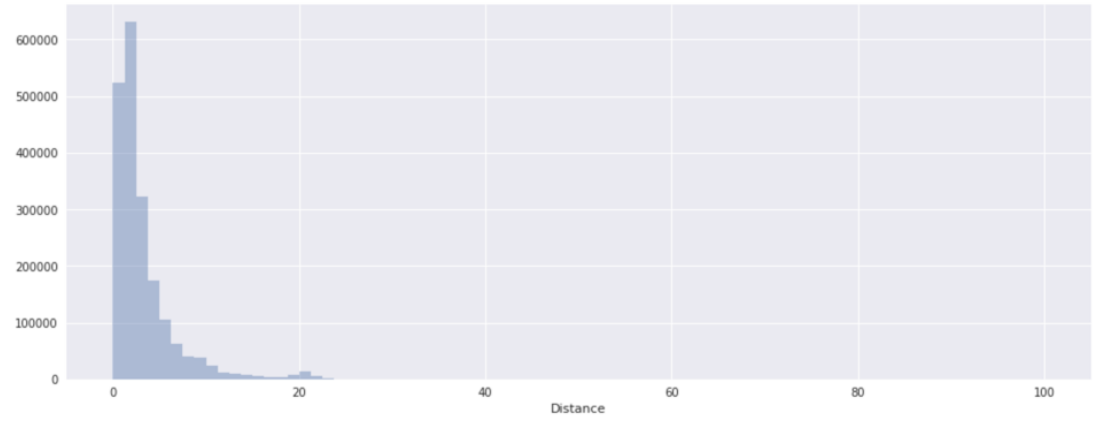
The input training data set contains 55 Million rows. Two million rows are used in the training model.

## EXPLORATORY VISUALIZATION



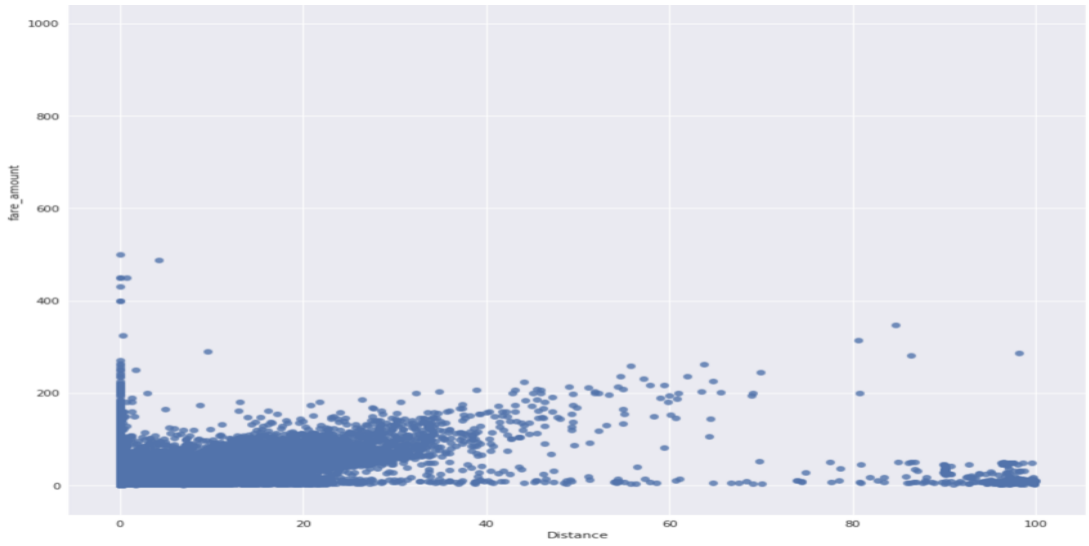

The above visualization depicts the distribution of fare_amount for 2 million rows. Most of the fare_amount is under 10$.

Certainly, the fare_amount depends on distance. The input features consists of pickup latitude, pickup longitude and drop off latitude and dropoff longitude. To obtain the approximate distance based on the latitude and longitudes, geopy library was used. Following visual depicts the distribution by distance.



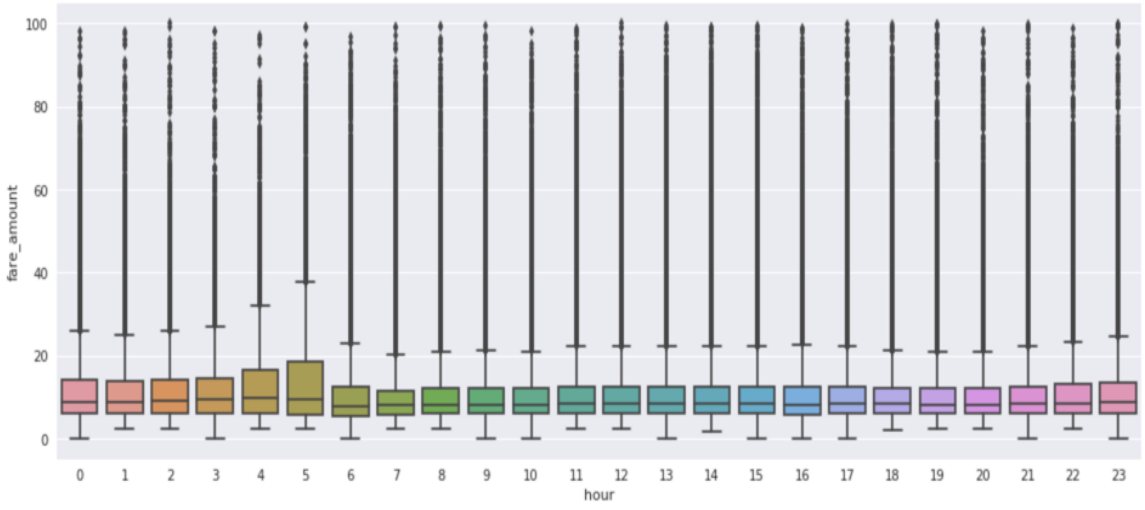

It is evident that most of the input training set have a distance under 10 miles.

The scatter plot below is the distribution between fare_amount and distance.

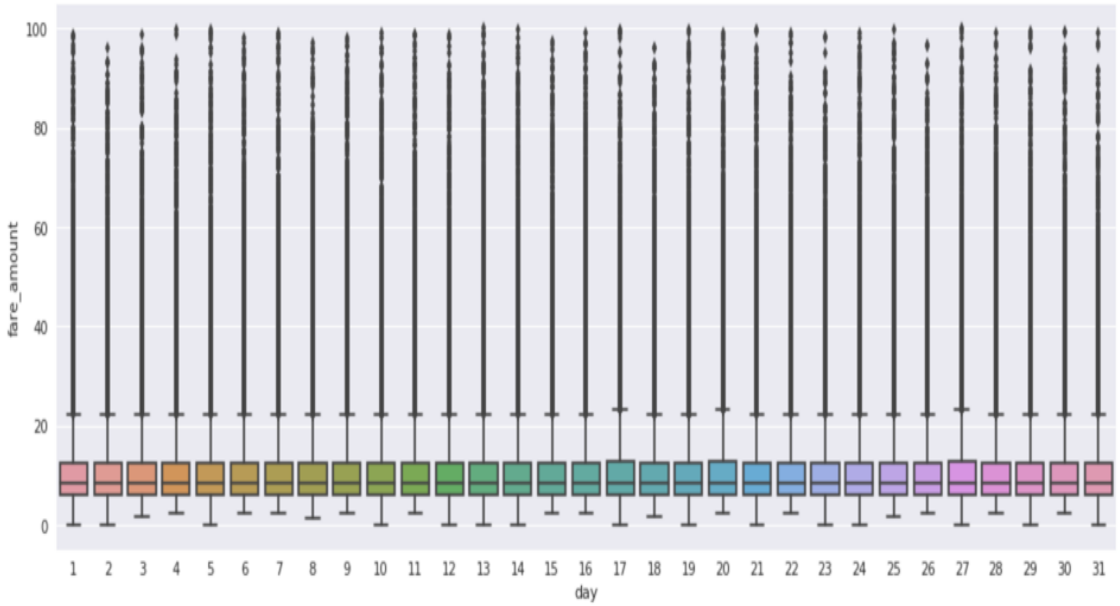



The fare_amount also depends on the date and time of pickup for a given day. The input timestamp is divided into day, month, year, hour and day of the week.

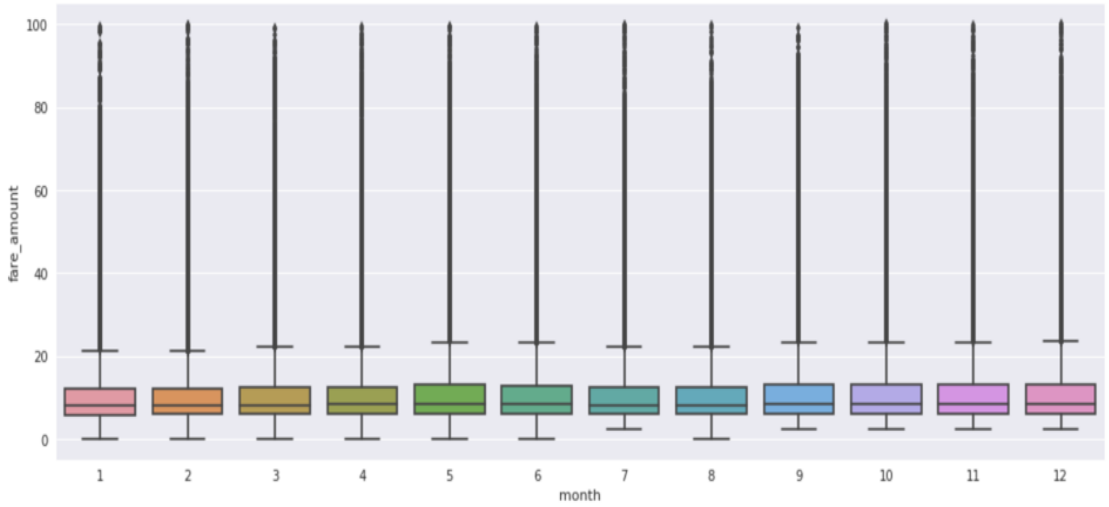
Box plots are used to observe the variation in fare_amount with respect to day, month, hour, and day of the week.



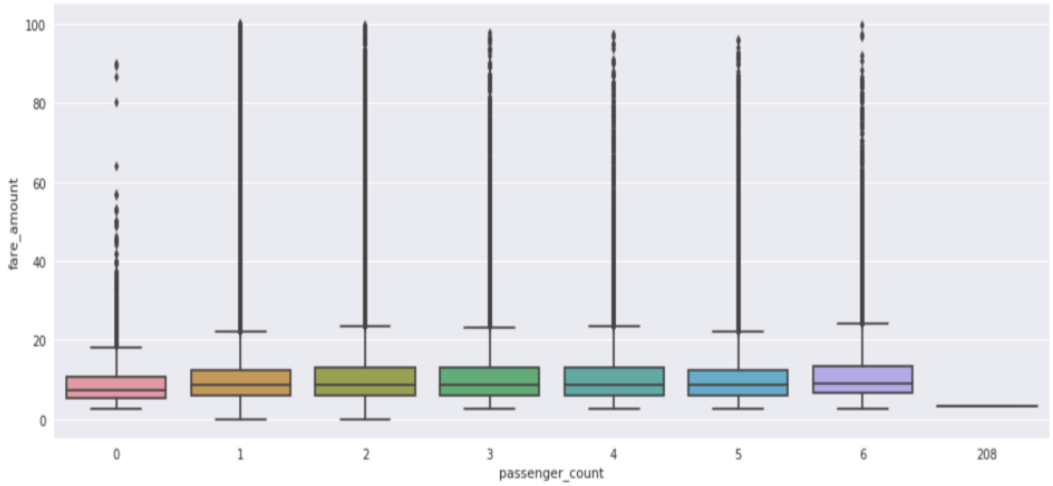

The distribution of fare_amount is higher during early hours and late hours of a day. This could be due to the limited availability of taxis at the time.




The fare_amount distribution looks similar in most of the days in a given month.




The fare_amount distribution looks higher from months 3-12.




The fare_amount distribution for passengers 1 to 6 resembles the same.

Data assumptions are discussed in the Data Preprocessing section.

### Algorithms and Techniques

This clearly is a supervised machine learning regression task. To resolve this problem the following models are used:

- Linear Regressor
- Random Forest Regressor
- XGBoost

### Benchmark

The author of this dataset in Kaggle calculated the basic estimate based on merely the distance between two points which resulted in an RMSE of $5-$8. Linear Regression has given a train and test RMSE of $6. The project utilizes $5 as a benchmark to make a decision on the model.

## III. Methodology

### Data Preprocessing

The primary goal of data cleaning is to detect and remove errors as well as anomalies. Following are the steps used for data preprocessing:

- Few rows contained fare_amount < 0. These rows have been removed.
- Removed the rows containing cells with null values.
- Latitude range is -90 to 90 and Longitude range is -180 to 180. Rows falling out of range have been removed
- Distance is calculated using geopy library based on pickup latitude, pickup longitude, dropoff latitude, and dropoff longitude.
- Distance travelled in taxis is usually under 300 miles. There are around 4k records which are over that. Clearly, this is because of an incorrect data entry. These are numerous records to be dropped. An attempt was made to correct the approximate distance for these values based on the fare_amount. Following assumptions were made based on an online search.

    - Minimum fare amount in nyc = 2.5
    - Cost for every 1/5 mile = 40
    - Cost for 1km = 40*5/1.6 = 125 cents = 1.25$
    - fare_amount = 2.5 + (dist * cost for 1km)
    - dist = (fare_amount - 2.5)/1.25

- Passenger count varies from 0 to 208. In general a maximum of 6 passengers can be accommodated in a big taxi. Considering the count exceeding 6 as an outlier, all the values beyond 6 were dropped. There were a few with count of 0. Online search showed that these could be charges for those who booked a taxi and cancelled them last min.
- The pickup date time is converted into day, month, year, hour, day of week. The pickup_datetime was dropped from the input data set.
- Key has been dropped since it does not provide any relevant information that can help in the prediction.

## IMPLEMENTATION

The input data has been classified into testing set and validation set where 10% of the data has been used as a validation set. This was achieved after utilizing the Sklearn's train_test_split. Sklearning provides various algorithms for addressing concerns related to regression. The Linear and RandomForest regression have been applied on the dataset. XGBoost is stands for eXtreme Gradient Boosting. It was

built to attain performance and computational speed. XGBoosting has also been applied for the dataset.

## IV. Results

### Model Evaluation and Validation

RMSE scores obtained by using the following models are as follows:

- Linear Regression: This method tries to fit a simple line for the input data set. The scores obtained by using this model are as follows:

    - Training-error: 6.66
    - Testing-error: 6.74

    Clearly this is above the benchmark of RMSE $5 and therefore it is not the correct model.

RandomForest Regression: The Random Forest is an ensemble of Decision trees which produces great results even without hyper-parameter tuning. It is a bagging algorithm that reduces variance. The following scores are obtained using this algorithm:

- Training-error: 2.08
- Testing-error: 4.63

The training and testing errors meet the benchmark.

- XGBoost: XGBoost is an implementation of gradient boosted decision trees designed for speed and performance. The following scores are obtained using this algorithm.

    - Training-error: 3.47
    - Testing-error: 4.26

Clearly the benchmark has been met. Training score for this is greater than RandomForest. However, a slight improvement in the testing score was noticed.

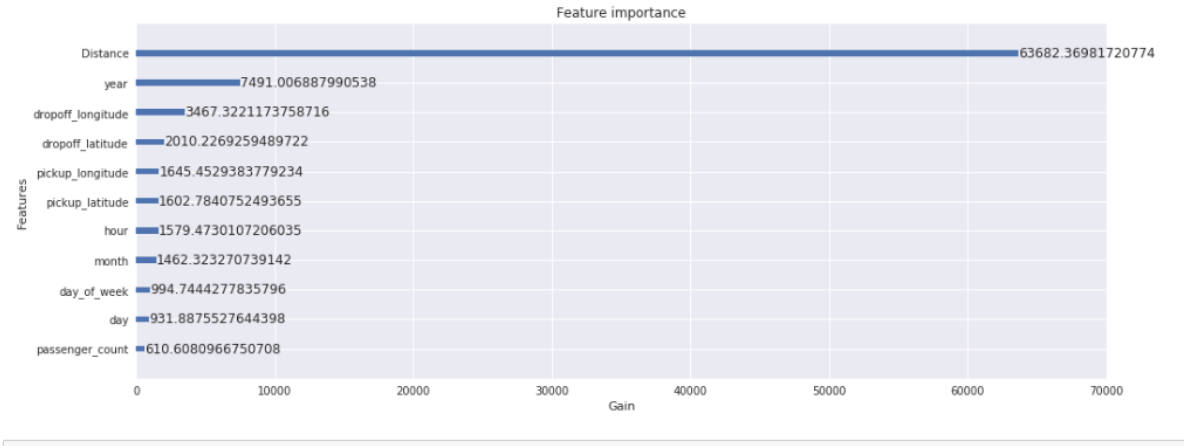
## JUSTIFICATION

The proposed benchmark was achieved by trying RandomForest and XGBoost that were among the initial guesses to predict the model. The RMSE for both the models have achieved the benchmark. Therefore we can go with one of those models.

## V. CONCLUSION

Linear Regression doesn't meet the given benchmark. RandomForest Regression and XGBoost meet the benchmark. RandomForest performs best on the training data. However, it could be overfit the training data. XGBoosting is fast and it performs well on training and testing data. It is possible to fine tune the hyper parameters and enhance the training, testing scores for XGBoosting using Grid Search.

The following is the feature importance obtained from XGBoost. As expected distance was the most critical feature in determining the fare-amount.




## REFLECTION

The following are the high level steps involved:

- Data Preprocessing.
- Model Selection.
- Predicting Values for the test set using the model.

Crucial piece for the project was data cleaning and preprocessing. This was the first time where I did extensive data preprocessing. Visualizations were helpful in understanding the data set. They also helped in preprocessing of data.

Linear Regression and RandomForest Regression were applied as mentioned in Proposal. Reviewer recommended to try XGBoost. This is the first time I used XGBoost and found it very useful in training supervised learning models in general.

## IMPROVEMENT

Hyper-parameter tuning was not used since the initial values used achieved good training and testing scores. Grid Search could be applied to XGBoost to get more improvement.

## REFERENCES


- https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/
- https://medium.com/human-in-a-machine-world/mae-and-rmse-which-metric-is-better-e60ac3bde13d
- https://elitedatascience.com/python-seaborn-tutorial
- http://nymag.com/nymetro/urban/features/taxi/n_20286/