Taxi Demand Prediction in New York City

Done by **Vishal Aggarwal**
Guided by **Karan Keswani**

Objective

In this project, we have the task of **predicting the number of taxi pickups in the respective region for a time period** (say 10 min) where end user will be taxi driver.

Constrains:

1. We expect ML model predict in adjoining region in few seconds.
2. Relative percentage error between actual and predicted.

Data Information

Get the data from:
http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml (2016 data). The data used in the attached datasets were collected and provided to the NYC Taxi and Limousine Commission (TLC).

Information on taxis:

Yellow Taxi: These are the famous NYC yellow taxis that provide transportation exclusively through street-hails.

For Hire Vehicles (FHVs): FHV transportation is accessed by a pre-arrangement with a dispatcher or limo company (not permitted to pick up passengers).

Green Taxi: Street Hail Livery (SHL): The SHL program will allow livery vehicle owners to license and outfit their vehicles with green borough taxi branding, meters, credit card machines, and ultimately the right to accept street hails in addition to pre-arranged rides.

Footnote:

In the given notebook we are considering only the yellow taxis for the time period between Jan - Mar 2015 & Jan - Mar 2016

ML Problem Formulation

Time-series forecasting and Regression

1. To find a number of pickups, given location coordinates(latitude and longitude) and time, in the query region and surrounding regions.
2. To solve the above we would be using data collected in Jan — Mar 2015 to predict the pickups in Jan — Mar 2016.

Performance metrics

1. Mean Absolute percentage error.
2. Mean Squared error.

Data Cleaning

In this section we will be doing univariate analysis and **removing outlier values** which may be caused due to some error.

Pickup location outside New York

It is inferred from the source https://www.flickr.com/places/info/2459115 that New York is bounded by the location cordinates(lat,long) - (40.5774, -74.15) & (40.9176,-73.7004) so hence **any coordinates not within these coordinates are outliers** as we are only concerned with pickups which originate within New York.

Drop-off location outside New York

It is inferred from the source https://www.flickr.com/places/info/2459115 that New York is bounded by the location cordinates(lat,long) - (40.5774, -74.15) & (40.9176,-73.7004) **so hence any coordinates not within these coordinates are outliers** as we are only concerned with drop-offs which are within New York.

Trip Durations

According to NYC Taxi & Limousine Commision Regulations the **maximum allowed trip duration in a 24 hour interval is 12 hours**. It is found using subtracting drop-off time with pick time.

Trip time not in the range of 0-12 hrs are considered as outliers according to TLC regulations.

Speed

Speed is calculated by dividing trip distance with trip time. It was observed that **speed of some data is very much high which is a false data** and hence can be considered as outlier. To remove this type of outlier I had used percentile method. It was observed that 99.9 percentile value of speed was 45.31 while considering other speed data greater than 45.31 miles/hr as outliers.

Average speed of cleaned speed is 12.45 miles/hr, so a cab driver can travel a distance of 2 miles in 10 minutes. So we can divide region according to above calculations.

Trip Distance

It was observed that 99.9 percentile value of trip distance was 22.57 while **considering other distance greater than 22.57 miles/hr as outliers**.

Total Fare

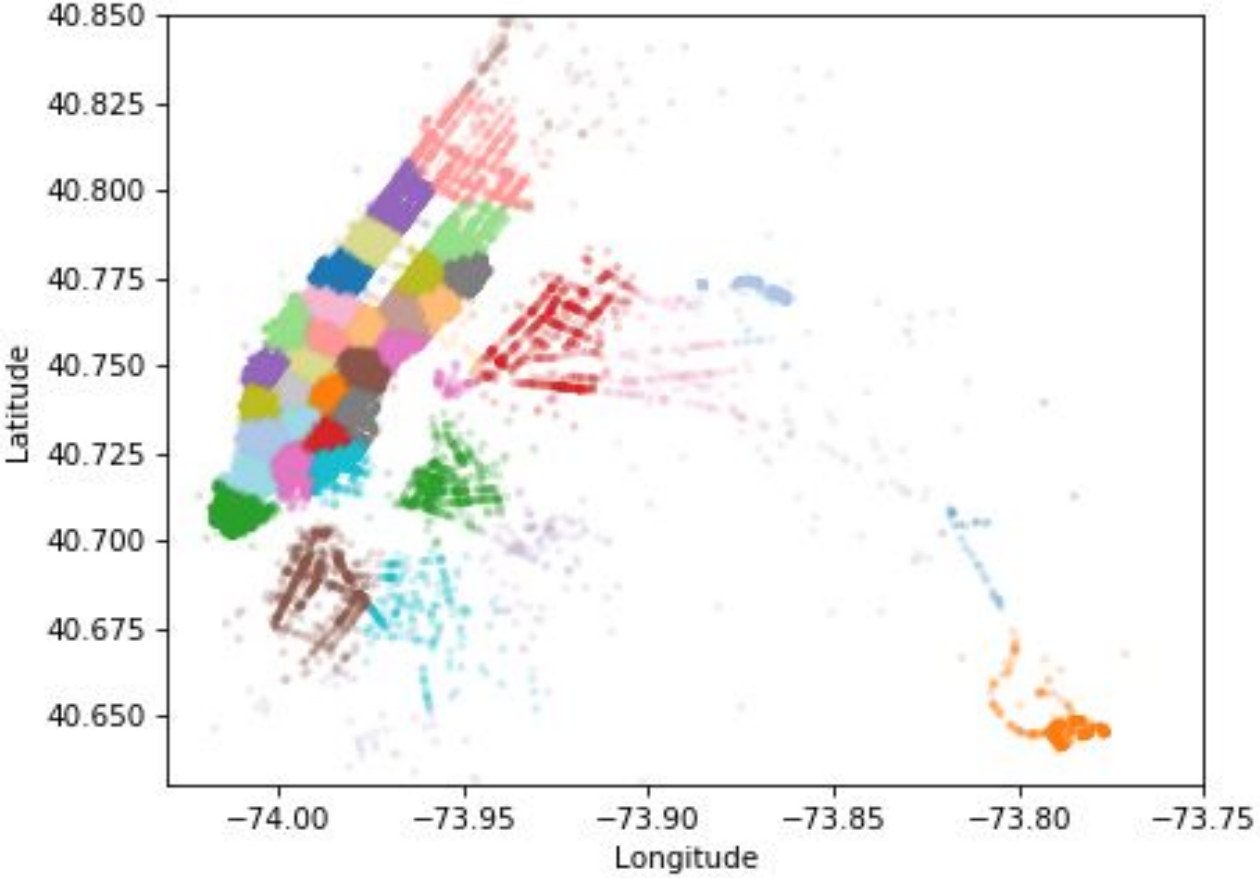
By graphical analysis we observed that **there is a drastic increase at 1000 fare value**.

Clustering and Segmentation

We will apply **clustering k-mean based on pickup size** and divide latitude and longitude of area. So the problem is how to find k value in k-mean.
I want intercluster distance less than 2 miles but region can’t be less than 0.5 miles because cluster will become too small.(distance measure from center of cluster)

For K=40,


Fairview River Road
Tonnelle Avenue
Guttenberg
Weehawken
Hudson River
East River
Broadway
31st Street
Flushing Meadows-Corona Park
Queens Boulevard
Union Turnpike
Grand Central Parkway
Myrtle Avenue
Fulton Street
Leaflet | Map tiles by Stamen Design, under CC BY 3.0. Data by © OpenStreetMap, under


40.850
40.825
40.800
40.775
40.750
40.725
40.700
40.675
40.650
Latitude
−74.00
−73.95
−73.90
−73.85
−73.80
−73.75
Longitude

We finally choose **k=40** because 9 cluster is within 2 miles and minimum intercluster distance is 0.5 miles. The main objective was to find an optimal minimum distance between the clusters which we got was 40. When checked for the 50 clusters it was observed that there are two clusters with only 0.3 miles apart from each other. **So we choose 40 clusters for solving the further problem**.

Time-Binning

Now here we have taken time in regular format and converted that time into Unix timestamp then divided that time by 600 to make 10 min bins. For January 2015 data, each time bin is a 10 minute time interval which is equal to the time elapsed — in seconds — since midnight Jan 2015, divided by 600(so as to make it in 10minute bin). Therefore, **there will be a total of 4464, 10 minute time bins in the month of January-2015** (Refer:https://www.unixtimestamp.com)/

Smoothing

When plotting between number of pickup and time bins, there is a chance that some value may hit zero for particular 10 min bins. that will create problem in ratio feature(divide by zero).

Smoothing is nothing but to avoid zero (ex. 100,0,50 pickup can be converted as 50,50,50)

Modelling: Baseline Models

Now we get into modelling in order to forecast the pickup densities for the months of Jan, Feb and March of 2016 for which we are using multiple models with two variations

1. Using **Ratios** of the 2016 data to the 2015 data i.e $R_t = P2016_t / P2015_t$
2. Using **Previous known values** of the 2016 data itself to predict the future values

Simple Moving Averages

For Ratio Feature: The First Model used is the Moving Averages Model which uses the previous n values in order to predict the next value. Using Ratio Values

$$R_t = (R_{t-1} + R_{t-2} + R_{t-3} \ldots . R_{t-n})/n$$

For Previously Known Values: We use the Moving averages of the 2016 values itself to predict the future value using $P_t = (P_{t-1} + P_{t-2} + P_{t-3} \ldots . P_{t-n})/n$. In this case, take average value of n previously know value to predict next known value.

Weighted Moving Averages

For Ratio Feature: It is found that the window-size of 5 is optimal for getting the best results using Weighted Moving Averages using previous Ratio values, therefore, we get $R_t = (5*R_{t-1} + 4*R_{t-2} + 3*R_{t-3} + 2*R_{t-4} + R_{t-5})/15$

For Previously Known Values: It is found that the window-size of 2 is optimal for getting the best results using Weighted Moving Averages using previous 2016 values, therefore, we get $P_t = (2*P_{t-1} + P_{t-2})/3$

Exponential Weighted Moving Averages

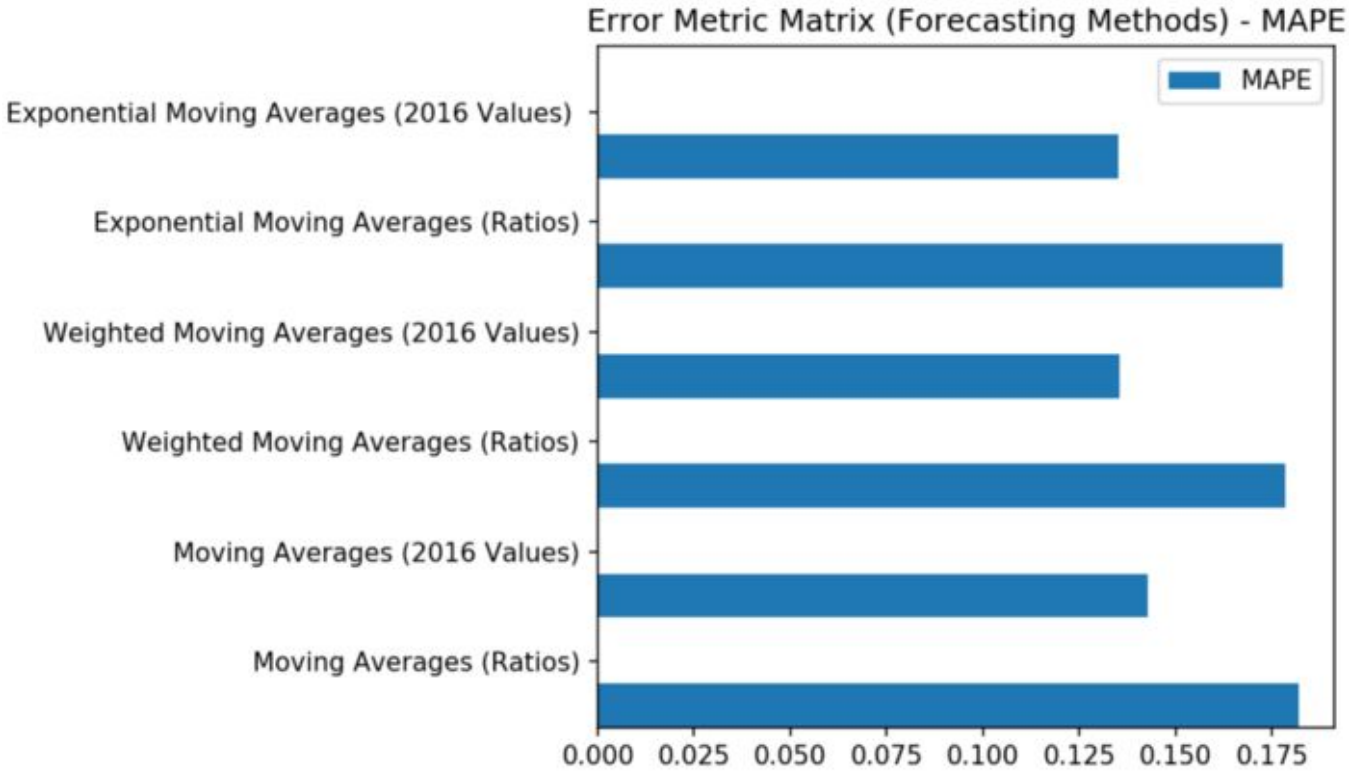
In exponential moving averages, we use a single hyperparameter alpha (α) which is a value between 0 & 1 and based on the value of the hyperparameter alpha the weights and the window sizes are configured. Also, the weights are assigned using $2/(N+1)=0.182/(N+1) =0.18$, where N = number of prior values being considered, hence from this it is implied that the first or last value is assigned a weight of 0.18 which keeps exponentially decreasing for the subsequent values.

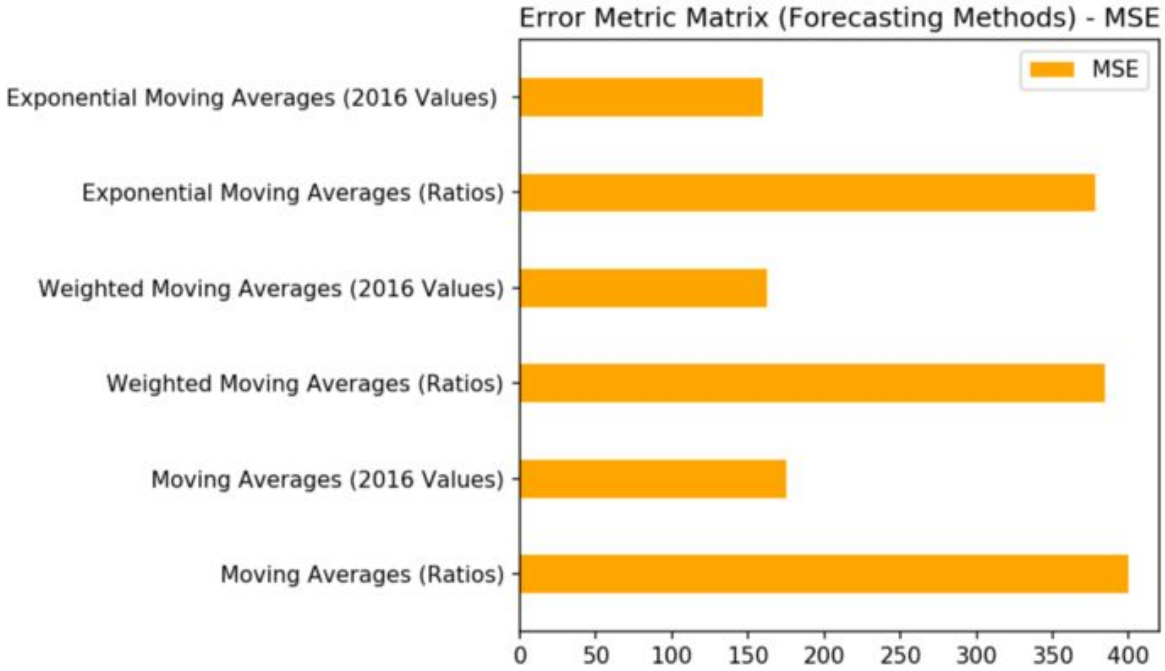
For ratio feature $R'_t=\alpha*R_{t-1}+(1-\alpha)*R'_{t-1}$ where R′ is predicted value at t time.

For known previous value $P'_t=\alpha*P_{t-1}+(1-\alpha)*P'_{t-1}$

Comparison between baseline models

We have chosen our error metric for comparison between models as **MAPE (Mean Absolute Percentage Error)** so that we can know that on an average how good is our model with predictions and **MSE (Mean Squared Error)** is also used so that we have a clearer understanding as to how well our forecasting model performs with outliers so that we make sure that there is not much of a error margin between our prediction and the actual value.


Error Metric Matrix (Forecasting Methods) - MAPE
MAPE
Exponential Moving Averages (2016 Values)
Exponential Moving Averages (Ratios)
Weighted Moving Averages (2016 Values)
Weighted Moving Averages (Ratios)
Moving Averages (2016 Values)
Moving Averages (Ratios)
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175



From the above information it is inferred that the best forecasting model for our prediction would be **$P'_t = \alpha * P_{t-1} + (1-\alpha) * P'_{t-1}$ i.e Exponential Moving Averages using 2016 Values**

Features Selection

1. **Latitude** of cluster center for every cluster.
2. **Longitude** of cluster center for every cluster.
3. **Weekday** of pickup (ex: sunday=0, monday=1, etc.)
4. **Number of pickups** that happened in last 5 10 minutes interval.
5. From previous observations we observed that exponential weighted moving averages gives us the best error, so we will try to add the same **exponential weighted moving average at t** as a feature to our data exponential weighted moving average.

So, we have total 9 features for predictions.

Train-test split and model implementation

Before we start predictions using the tree based regression models we take 3 months of 2016 pickup data and split it such that for every region we have **70% data in train and 30% in test**, ordered date-wise for every region.
Here, we had tried three models: **Linear regressor, Random forest regressor, XGBoost regressor** and predicted the pickups for both test and train data and then finally we calculated **Mean Absolute Percentage Error(MAPE)** for all of these models.

Observations

1. Random Forest Regression seems to be best model where MAPE of train value decrease below **12%**
2. There is not any sign of overfitting or underfitting but Random forest model seems little bi overfitting.
3. All model have test MAPE in range of **12.6 to 13.6%** if we avoid overfitting, XGBoost is best model.