Springboard Data Science Career Track

Capstone 1 Project

NYC Taxi Trip Durations

Created By: Andrew Milne

NYC Taxi Data - Capstone 1 Final Report

Problem Definition

The kaggle competition behind this is to build a model that predicts the total ride duration of taxi trips in New York City. The primary dataset is one released by the NYC Taxi and Limousine Commission, which includes pickup time, geo-coordinates, number of passengers, and several other variables.

Source:
https://www.kaggle.com/c/nyc-taxi-trip-duration

Client:

The literal client is Kaggle but the figurative client would be a taxi company looking to optimize their route selection. This kind of service provider could be an Uber/Lyft or traditional taxi company...perhaps even a technologically inclined rickshaw driver.

Data Cleaning/Wrangling:

The data needed to be cleaned to remove the following:

- Extremely long ride times, eliminated data outside +/- 3 std of mean or < 5 hrs duration
- Limited passenger count to <=6 as most cabs don't have capacity of a mini-van
- Direct route information, filled blanks with average durations and steps

Feature engineering was performed to incorporate the following

- Conversion of timestamp data into day, week, month, hour, year, date
- Checking of trip duration & date time values for consistency
- Haversine distances between pickup and drop off latitude & longitude
- Directional bearings calculated based upon latitude & longitude of pickups
- Average speed of trips
- Time series plots

I used a variety of tool bags and libraries to assist in this analysis. These included the following:

- Python 3.6.1 run in Jupyter Notebook
- Plotting capability provided by matplotlib, seaborn
- Analysis capability from pandas, numpy, datetime
- Machine Learning provided by SciKit Learn: MiniBatchKMeans, Linear Regression, Decision Tree Regressor, Random Forest Regressor

Other Datasets:

- Direct route information incorporated from an Open Source Routing Machine such as one found at project-osrm.org
- Weather related data could have been incorporated to review outliers for snow storms or flooding (*not included*)

Summary of Key Findings:

Through exploratory analysis and investigation we have determined the following:

- The average trip durations between each vendor are similar but statistically different
- The average trip lasts 840 seconds or ~14 minutes.
- The variables are generally independent of each other, meaning there are few pairs that demonstrate strong correlation to each other, save for trip duration and distance.
- Time of day, week, month can impact trip durations
- The lowest amount of traffic in the city occurs in the early morning hours and late evening.
- Machine Learning algorithms for Regression and Clustering were applied to reveal important features and neighbourhoods
- There is a strong response in trip duration based upon the following three variables: direct distance, hour of the day, direction and dropoff cluster.

Exploratory Data Analysis & Inferential Statistics Discussion:
The following represents a few key features about the data.

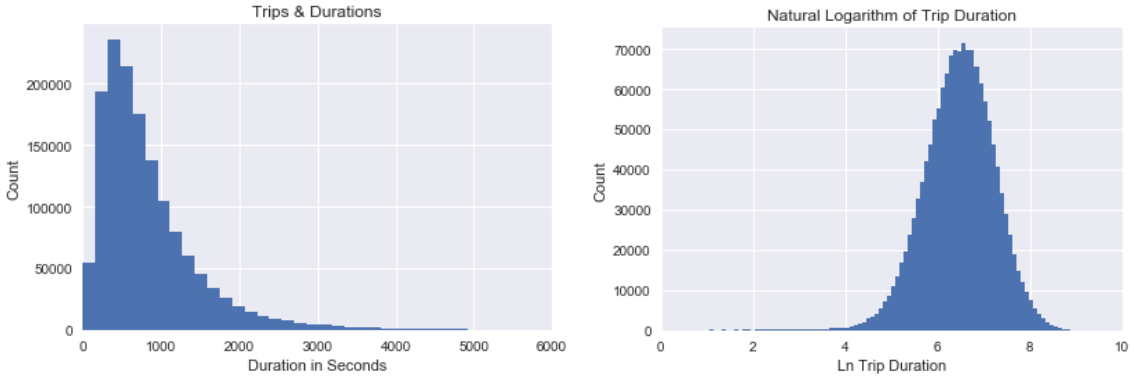

Fig. 1 - Trip Data Distributions

Key Finding: Statistical Distribution
The trip duration is normally distributed around of mean of 840 seconds or 14 minutes. The median trip is roughly 660 seconds or 11 minutes.

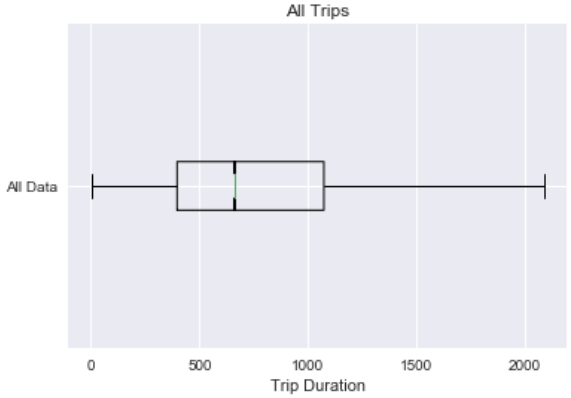

Fig. 2 - All Trips Boxplot

Split by vendor this makes for an interesting opportunity to test means using *hypothesis test for two samples*

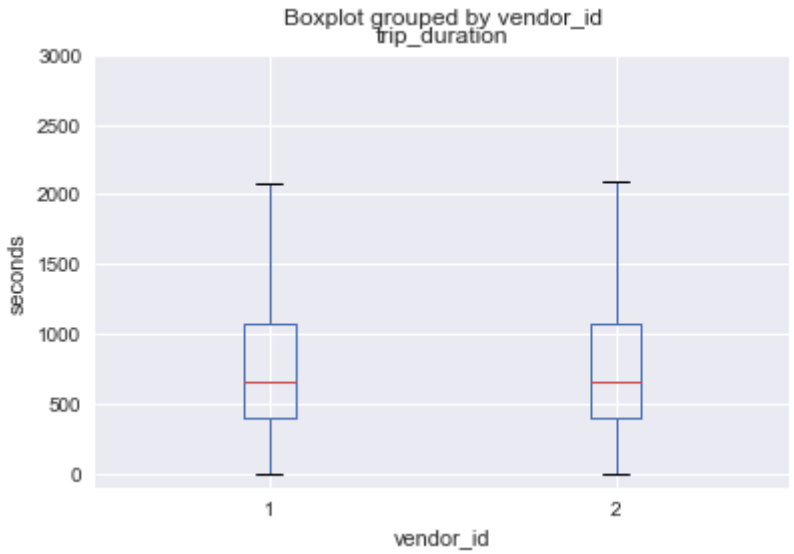

Fig. 2 - Average Trip Time by Vendor

These are huge samples at roughly 600k per vendor. As such we can determine the means and standard deviations to be representative of the populations. The Central Limit Theorem holds that since sample size is >40.

$$\text{Ho: } \overline{x} - \overline{y} = \Delta_o$$
$$\text{Ha: } \overline{x} - \overline{y} < \Delta_o$$

This is to say that for Ha, the mean trip duration of vendor 1 is less than mean trip duration of vendor 2. This inequality indicates a lower tailed test where we reject the null hypothesis if $z \leq -z_\alpha$.

For an alpha value of 0.95 the $z_{critical}$ value is 1.65 (Probability & Statistics for Engineers - Table A.3) meaning that in order to reject the null hypothesis our calculated statistic has to be less than or equal to -1.65.

$$z = \frac{\overline{x} - \overline{y} - \Delta_o}{\sqrt{\left(\frac{s_x^2}{n_x} - \frac{s_y^2}{n_y}\right)}}$$

Equation of Z Statistic

In this case our z statistic is -10.7 which is less than -1.65 and so we must reject the null hypothesis that the mean trip durations are equal and accept that vendor 1 has a lower mean trip duration than vendor 2.

The P-value is also 0 meaning Ho should be rejected for any reasonable significance level.

It is also fair to say that although the means are different, they are not different by much and this will likely not be a significant factor in determining trip duration.

Key Finding: Hypothesis Test
Vendor will probably not impact ride time significantly.

We investigate other features of the data including pickup times and distances.

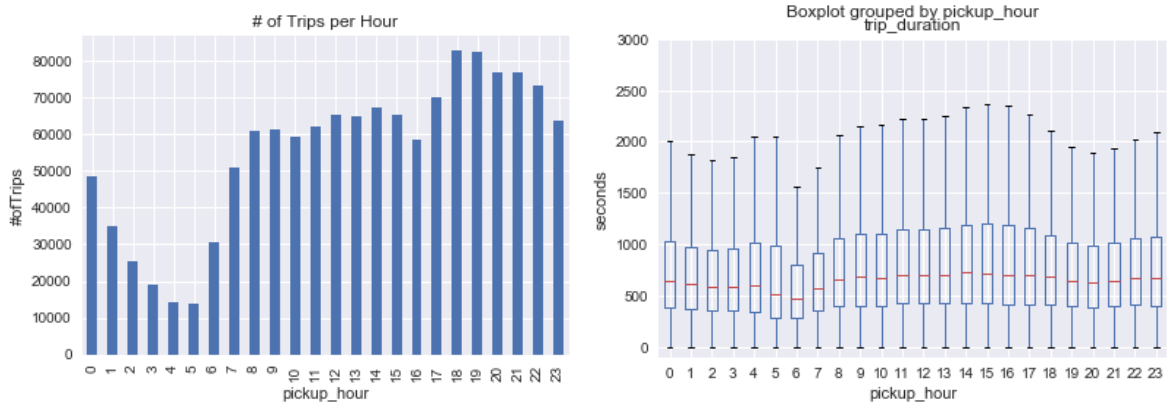

Fig. 3 - Trips & Duration by Hour

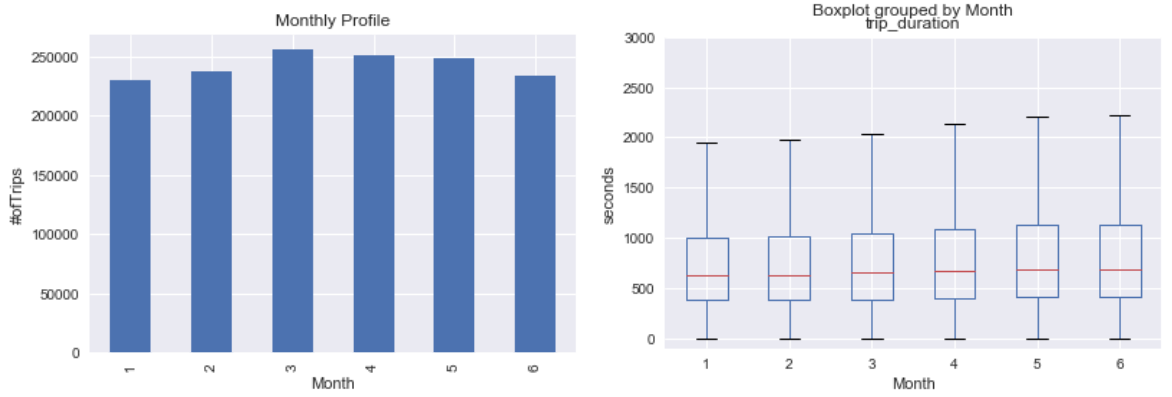

Fig. 4 - Trips & Duration by Month

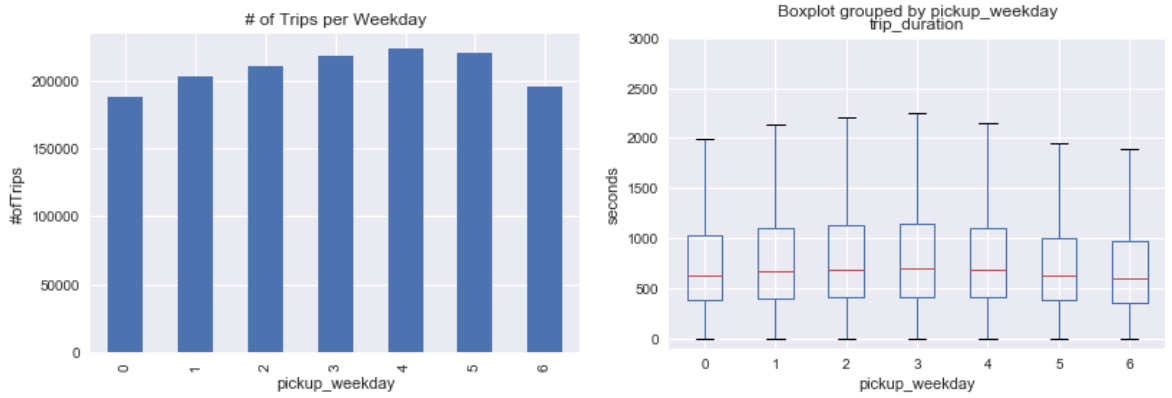

Fig. 5 - Trips & Duration by Weekday

In general we see that the profiles of figures 3,4 and 5 are supported by the trends in the boxplots for trip duration.

Key Finding: Data Trends
We see an increase in trip duration for the evening hours, the summer months and peak trip durations on Wednesday nights. We will incorporate these in our models.

Let us perform a time series plot on number of trips by date. The figure below shows a relatively consistent traffic pattern except for late January 2016.

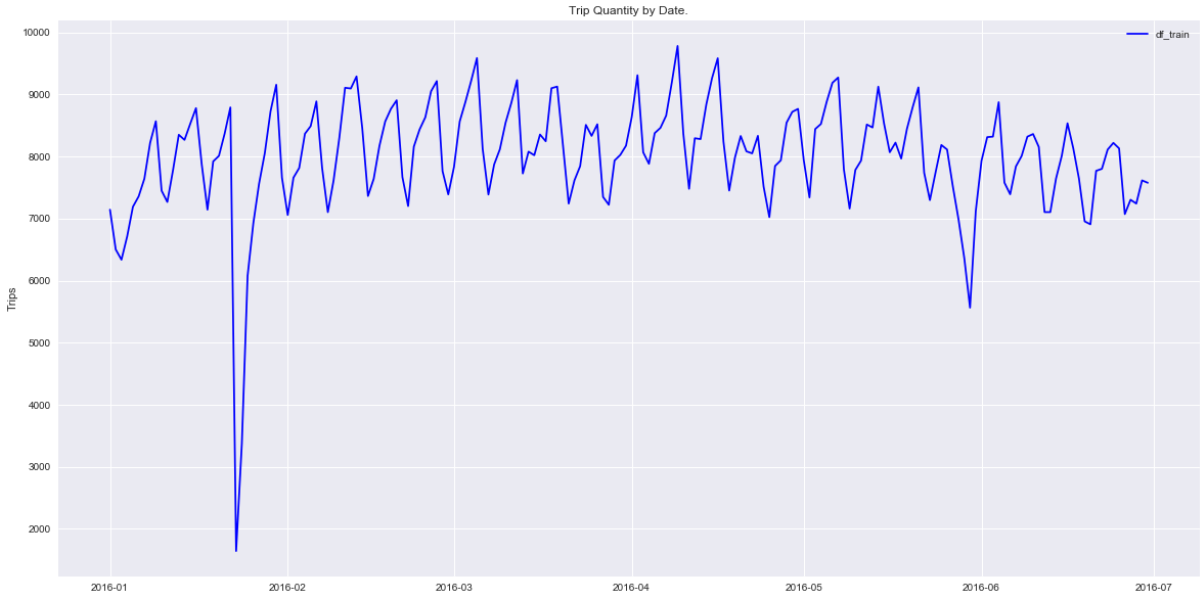

Fig. 6 - Daily Trips Profile

Since this date in late January is such an outlier we ask the question, what could have happened to cause such a drop in the number of taxi rides? My first thought was some form of disaster caused by humans or nature. As it turns out a quick google search for weather events yields:

Key Finding: Weather Event
The fact that there was a crippling blizzard event caused traffic to stop for an extended period.

As this is isolated we have elected to not pursue further investigation.

As a point of interest, the distance feature is 'direct distance' not accounting for one way streets and turns. Data related to shortest route was added as a bit of feature enrichment using the 'Open Source Route Machine'. More on this later.

We obtained the average velocity information by incorporating haversine distance calculations and dividing by units of time.

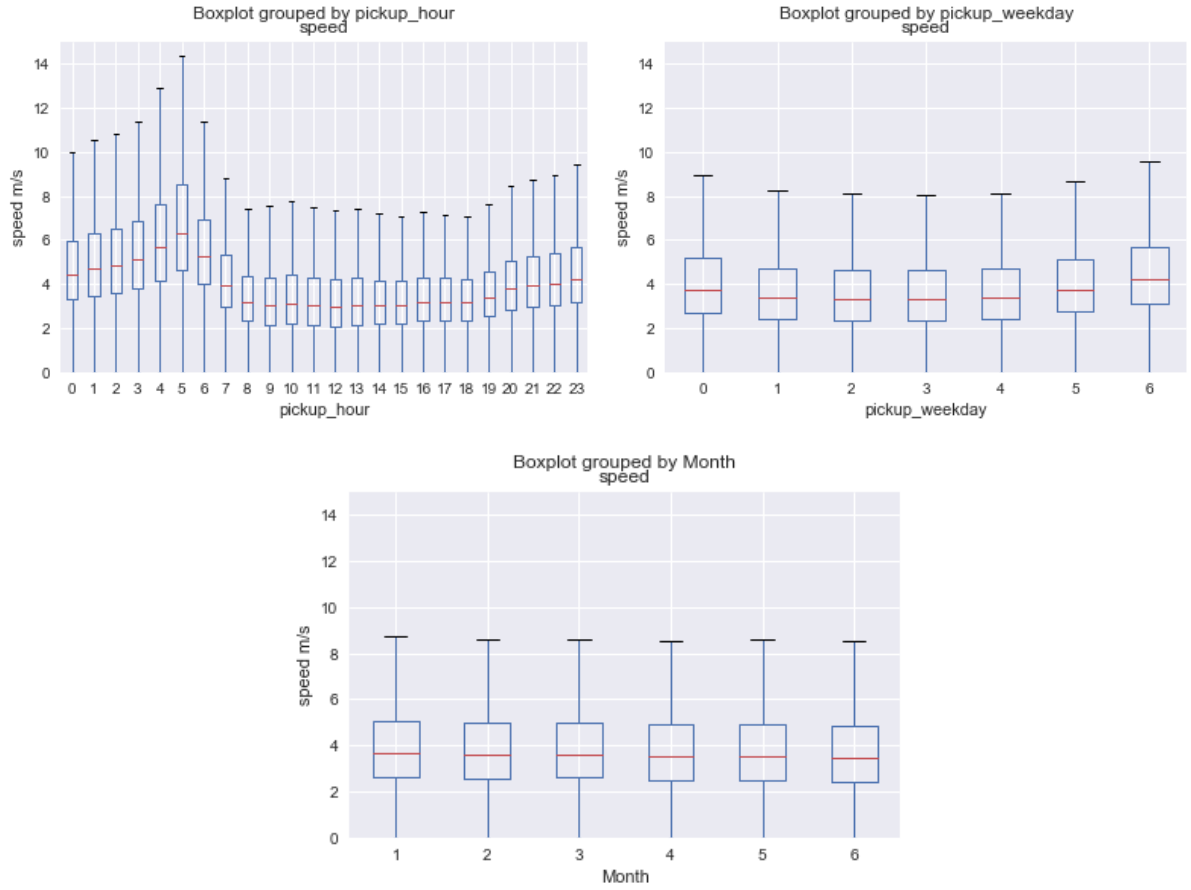

Fig. 7 - Trip Speed

The trip speed data reflects what we saw earlier in the plots of trip duration, Fig 3-5.

The number of passengers per trip may prove significant in our modelling. Let us plot it first to see what the distribution is like.

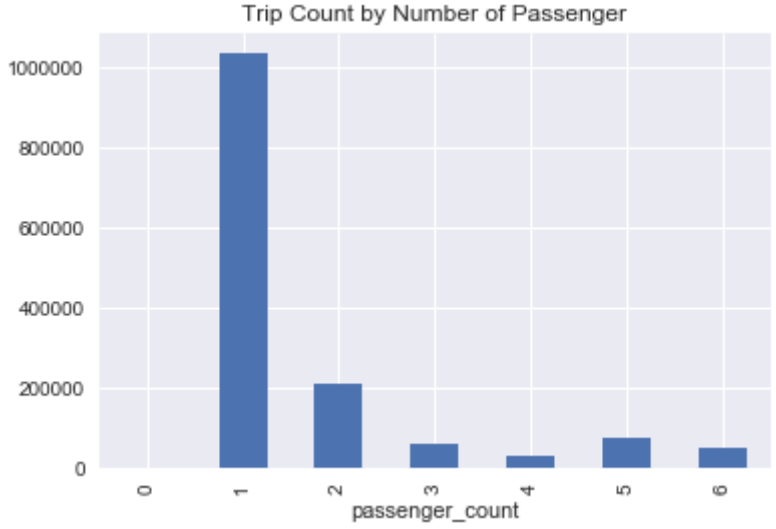

Fig. 8 - Passengers

Figure 10 shows that the majority of trips are for one passenger only. Relatively few trips occur with groups of riders.

Key Finding: Passenger Count
Logically, it follows that passenger count is not likely to be a big factor in determining ride time.

Machine Learning Regressions
Clustering with K-Means Outcomes:

As the data represents the cumulative set of taxi trips within a defined geographic area it makes sense to try and do some clustering algorithms. These clusters can be treated as analogues to the city neighbourhoods.

The first attempt at clustering was to use a number of clusters roughly equal to the number of neighbourhoods within NYC, ~100. In order to perform this analysis it was necessary to first create a numpy array of the available coordinates defined by the pickup and dropoff latitudes and longitudes.

We then created a random sample of the coordinates data and fit a small batch k-means clustering algorithm. The next step was to map the pickup and dropoff clusters by predicting the cluster off the actual latitude and longitude of the pickup and dropoff locations. This meant that the k-means model had been fit to a random sample and then used to predict clusters of unseen data.

An intuitively beautiful figure was produced by the first pass at clustering. While slightly disproportioned due to density of the data we can see distinct similarities to a map of the city.

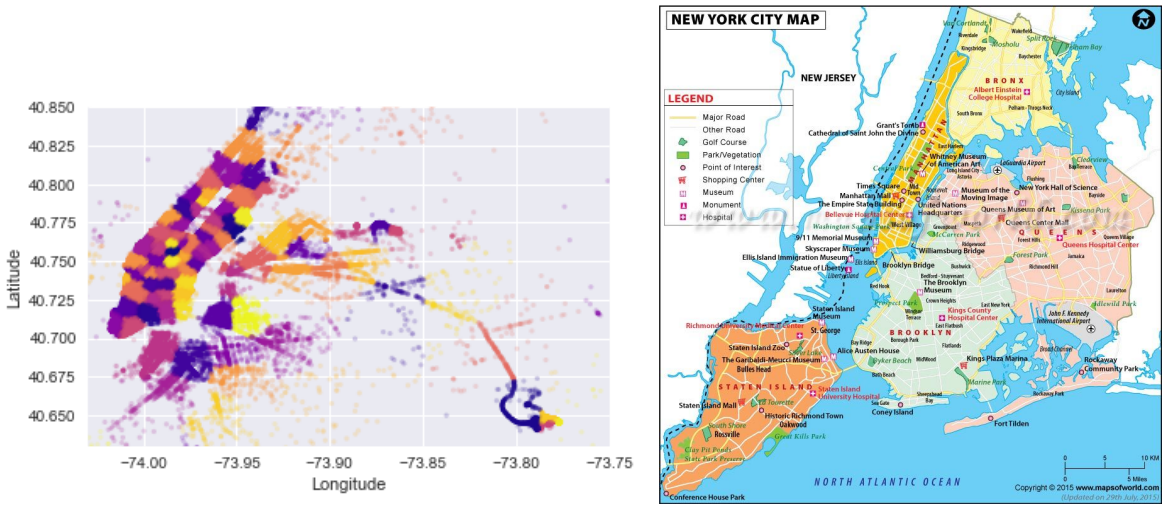

Fig. 9 - Neighbourhoods of NYC

We repeated the process but first employed the ‘elbow method’ to determine an optimum number of clusters. The point at which the sum of squared error begins to flatten is deemed the optimal value for clusters on the data.

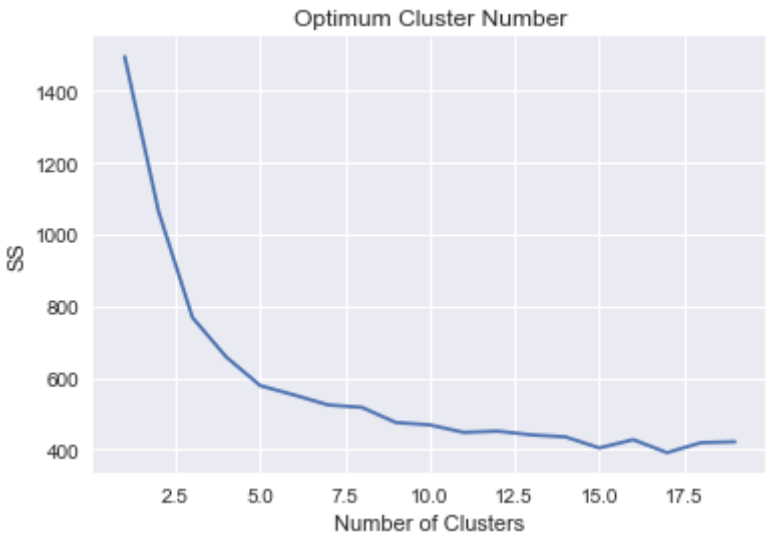

Fig. 10 - Optimum Cluster Number

Key Finding: Cluster Optimization
The optimum value for cluster quantity was determined to be 5.

This is a convenient result because there are 5 boroughs in New York City - Queen's, Harlem, The Bronx, Staten Island and Manhattan.

Here are the results of 5 clusters on the same plot of the pickup and dropoff locations.

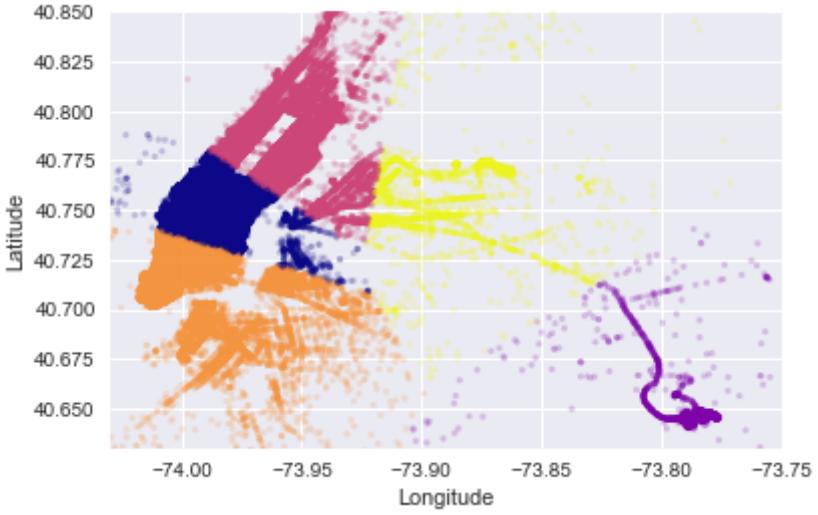

Fig. 11 - NYC Boroughs

Again, we take note of the fact that this silhouette bears resemblance to the city. We can see distinct areas such as central park, the rivers, the airport and the five boroughs.

Key Finding: Cluster Number
We will use the cluster number as a feature in our models.

An additional clustering was performed using the duration and distance of each trip. This produced another interesting finding that there are pickup areas which correspond to faster trips than others. Perhaps these trips are short in distance and therefore quick but the

number of clusters converges upon ~4 fairly convincingly. It is worthwhile noting this observation as a feature in our upcoming models.

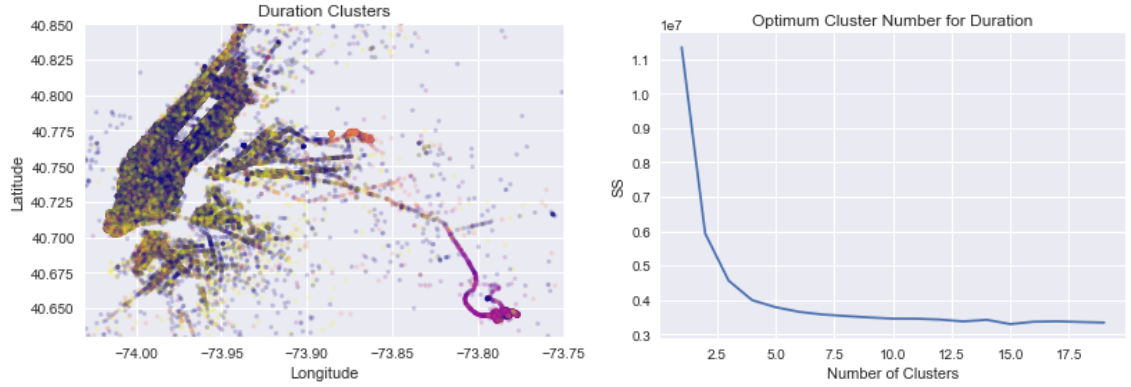

Fig. 12 - Clustering by Duration & Distance

We can see that the model is fairly successful at clustering the longer trips from the airport. The pickup locations within the city must correspond to longer duration trips and are likely related to different dropoff clusters.

Regression Models:

We can then move on to regression models using SciKit Learn. The following parameters were included in the model based on the key findings of our EDA:

- Month (1-6)
- Pickup hour of the day (0 to 23),
- Pickup weekday (0-6) with Monday as zero,
- Direct Distance in kilometres,
- Total Distance based upon OSRM in metres,
- Number of steps in ride, i.e. turns, based upon OSRM data,
- Pickup cluster & Dropoff cluster (0 to 4),
- Duration cluster (0 to 3),
- Passenger count (0 to 6),
- Vendor,
- Log Trip Duration

The ‘log trip duration’ variable was used as the dependent outcome as we are trying to predict trip durations. The natural logarithm was used as it was required for the submission to Kaggle and also for ease of calculations.

The regressions were carried out with and without dummy variables for categorical features. It proved to be true that the models performed better without dummies. This is most likely caused by the categorical features having numerical values to begin with.

We performed three iterations of regression making use of SciKit Learn’s Train, Test, Split moethodology. The performance results are as follows:

Table 1 - Regression Performance Metrics

	Linear Regression	Decision Tree	Random Forest
R-Squared	0.53	0.73	0.74
MAE	0.39	0.29	0.28
MSE	0.28	0.16	0.16
RMSE	0.53	0.40	0.39

Key Finding: Random Forest
Random Forest is our best model

It is demonstratively clear that the Random Forest regression model performed best when tested on unseen data. The plot of the residuals for the Random Forest model is shown below and profiles a normal distribution, indicating good fit.

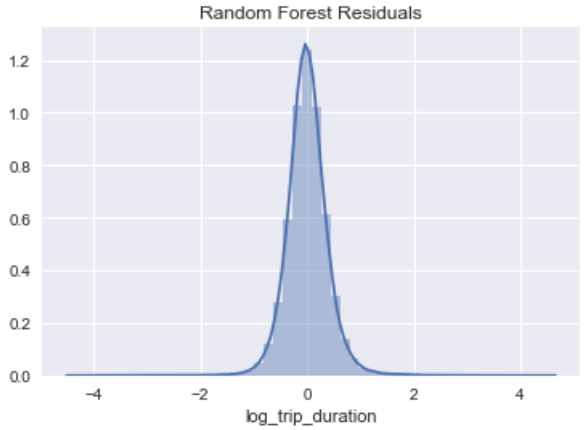

Fig. 13 - Random Forest Residuals

The maximum depth of the decision tree and random forest regressors was set to ten based upon n-1 number of features. This was done to limit the calculations and also avoid over fit of the data.

It is important that we address the fundamental performance difference between a linear model and a tree based model. Linear regression assumes a model of the form

$$f(X) = \beta_0 + \sum_{j=1}^{p} X_j \beta_j,$$

Whereas a regression tree,

$$f(X) = \sum_{m=1}^{M} c_m \cdot 1_{(X \in R_m)}$$

Contains partitions of the feature space in the form of R (Hastie et. al, 2013). Most importantly tree based models are capable of modelling nonlinear relationships between features.

There are some advantages to tree models such as their similarity to decision making, graphical interpretability and limited need for dummy variables. Random forests can be used to enhance the predictive limitations of decision tree models, as we have shown above.

Lastly, the relative importance of features in the random forest model are shown below in figure 15.

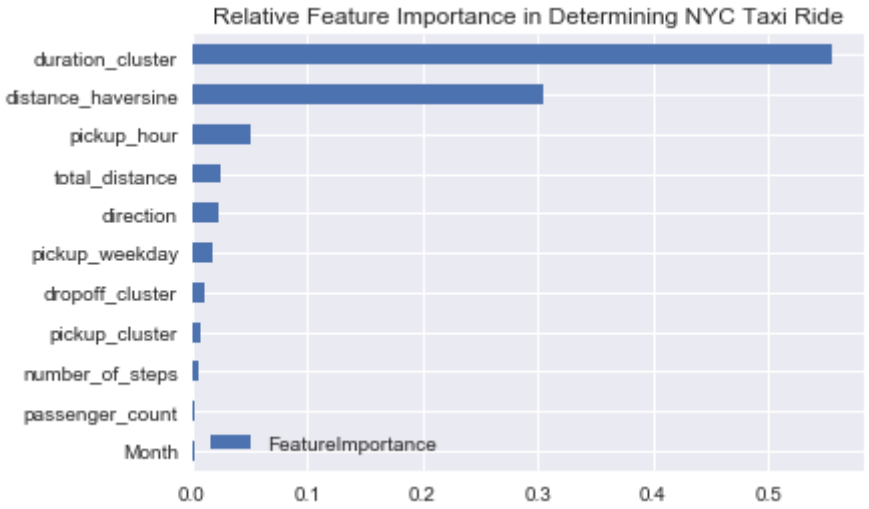

Fig. 14 - Relative Importance of Features in Determining Ride Time

The most prominent factors in determining ride time is the duration cluster and direct geographical distance. This is followed by the time of day, total distance and general direction of route.

Improving the Random Forest with Grid Search CV

The model was put through a grid search to tweak it for better performance. The grid search cross validation uses numerous iterations to test parts of the model and minimize error. The two parameters that we focussed on were ‘n_estimators’ and ‘max-depth’. These represent the number of trees and the number of branches per tree respectively.

At first the model was tested using a low number for max_depth so as to limit calculations. Once the model was run it became clear that optimal values for max_depth were nearer to the total number of features in the model.

The number of trees to be used in the random forest was optimized at 20 with nearly zero benefit of raising the value.

Results of Testing the Model

Finally, the model was tested using the test set of data from the Kaggle competition. The training and testing data were prepared using the feature engineering described earlier.

It is important to note that the training data held one more column than the testing data because of the presence of the response variable, trip duration.

The random forest regressor was called upon the test data and produced a mean trip duration of 792 seconds or 13.2 minutes. Very similar to the ~14 minute mean duration of the training data.

Conclusions & Recommendations:
EDA & Inferential Statistics

The data for trip durations was normally distributed before and after transforming it to the natural logarithm.

What is likely to play the biggest role in determining length of the time spent riding in a cab is the total distance travelled. Beyond that obvious conclusion we can expect to take longer rides in the summer months, when it is later in the day and if it is during rush hour. As the week goes on the trips generally decrease in duration.

After incorporating the distance calculations and the OSRM we were able to add a bit more context to the data.

Further exploration could be performed on the directional bearing of the trips, weather related data, annual city events, durations by landmark, driver ID, distance features, cost of the rides etc.

Statistically we could do more investigation. For example, ANOVA tests could be performed on features to determine correlation with an outcome.

Machine Learning Regressions

The regression models proved to be fairly successful after using a Random Forest. Further analysis would probably take the shape of an XGBoost combined with some Principal Component Analysis. Further manipulation of the model could also include the use of a Random Search CV.

The point to point distance haversine is the most important feature in determining trip duration. This comes as no surprise after exploring the data. Intuitively, distance and speed are the main contributors to elapsed time between points. In the absence of dramatically increased speeds, distance remains as the key contributor to duration. In a large and dense city like New York it would be impractical to expect that taxi's would be able to dramatically

increase their speed through minor differences in route selection. As such it is logical that geographic distance would make for the most important feature.

As seen in the EDA, pickup hour of the day showed significant impact on the other fundamental variable, speed. Ride times were dramatically reduced early in the morning and late at night. This is the same reason people drive to work before 7 am; to avoid traffic.

Ultimately the model was able to predict the trip durations of the test data with a reasonable degree of accuracy. The average trip duration fell within one standard deviation of the training mean.

Appendix:
Calculations
https://github.com/andrewcmilne/capstone1_taxi/blob/master/TaxiInferentialStatistics%26ML.ipynb

References
https://en.wikipedia.org/wiki/Haversine_formula
http://scikit-learn.org/stable/modules/model_evaluation.html