Understanding Travel Time to Airports in New York City

Sierra Gentry | Dominik Schunack

1 Introduction

Even with the rising competition of rideshare services, many in New York City still utilize taxis for their transportation needs, particularly when the airport is involved. However, as rideshare services continue to disrupt this aspect of the transportation industry, taxi services must adapt and transform to better serve their customers. They must be able to better predict traffic given various conditions which will be further discussed in the feature section.

This transportation analysis uses machine learning techniques to more accurately predict travel time and trip origin and destination densities to allow the taxi fleets to better serve their customers and distribute themselves throughout the city. In particular, this analysis focuses on trips to and from the John F. Kennedy International Airport [JFK]. While the overall analysis can be applied to the entirety of the fleet, this subset was chosen to allow for a more meaningful interpretation of results to occur.

The algorithm inputs were features associated with weather, travel, and time (see *Section 3.4* for a more in depth list of input features). The final models used linear regression and gradient boosting to predict demand density and travel time, respectively.

2 Related Work

Travel time prediction has always been of interest to transportation engineers. As transportation related data collection improves, whether it be from sensors, taxi meters, rideshare apps, or other sources, the attempt to better predict these times has as well. This is particularly apparent for companies such as Lyft and Uber[1]. They primarily rely on Long Short Term Memory architectures to create more complex models that better plan for abnormalities in their time predictions, particularly for holidays.

3 Data

3.1 Sources

Two primary datasets were utilized for this analysis: the *New York City Taxi and Limousine Commission*[2] [NYCTLC] provided transportation based information, while *Weather Underground*[3] [WU] provided weather attributed to New York City between July 2016 and June 2017.

3.2 Data Processing

All data has been cleansed and processed in R. The corresponding file has been uploaded to GitHub[4] and named *"data_preparation_jfk"*. One major task was to concatenate weather and taxi data. For both data sets, a single timestamp variable needed to be converted to three variables, namely hour, day and month. This way, weather information could be assigned to one specific taxi ride (data point). The level of granularity is thus on an hourly basis. It was decided to use the pick-up time of a passenger as the decisive timestamp in order to determine the weather conditions that would define that taxi ride. The matching of data was achieved by creating unique IDs for hour, day and month.

As a next step, data was filtered for rides that go to or leave from JFK. Exploratory data analysis was applied to identify potential outliers. It was determined that taxi rides longer than 240 minutes and shorter than 5 minutes should be excluded from the model. This appears to be reasonable since, for example, the average taxi ride from Manhattan to JFK is less than 60 minutes. Furthermore, trip distances longer than 50 miles and shorter than 1 mile were excluded.

3.3 Handcrafted Features

In order to generate a more accurate predictive model, handcrafted features were introduced. Specifically, average travel time for the *previous hour*, *two hours before*, and average trip duration the *day before* were calculated and applied to the correct time stamp and pick-up and drop-off location. The idea of those variable is to capture short-term contextual influences on travel time, such as traffic that is caused by an accident. Ideally, the travel time for the hours before would capture that information. The average travel time for the day before was meant to incorporate medium-term instances, for example if roads are closed for multiple days due to weather conditions. In the results section, the effect and limitations of those variables will be discussed.

Furthermore, national and NY-State specific holidays were added as a binary variable, as well as a variable that accounts for the two days before and after the holiday. The latter was thought to incorporate any change in traffic patterns that occur around holidays. Holidays are believed to change traffic conditions, since much of the work-related traffic, such as daily commutes, are altered during those days.

In similar fashion to the hour before variable, the *location density for the hour before* was calculated. In order to predict the taxi demand in a given district during a given hour, information on that variable during previous hours is assumed to correlate with actual demand. For example, an event that leads to taxi demand may stretch over several hours.

3.4 Final Features

After the data processing, a variety of different features were utilized in the analysis of travel time and demand.

Transportation related features included: trip distance, travel time one hour prior, travel time two hours prior, and trip density one hour prior.

Weather related features included: temperature, wind speed, precipitation, and factorized weather condition.

Time specific features included: hour, day, month, weekday/weekend, holiday, and a two day holiday buffer.

4 Methods

The analysis can be split into two sections. In the first part, we wanted to understand how predictable the demand was for taxis going to JFK at a given location and time. Knowledge of the density distribution can help in better coordinating the fleet of cars. The second part of the analysis examined travel time prediction under the light of certain factors. Models were tested on different subsets to gain more insight into the influence of time and weather for travel time predictions.

For density prediction, only a linear model was applied, since the results were considered acceptable, as will be discussed in the next section (see *Table 1* and *Figure 1*). For the time travel prediction, four different models were tested. They shall briefly be presented in the following.

Linear Regression

Linear regression minimizes the following cost function:

$$\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$$

Features were selected manually. For the final model, all features as described in *Section 3.4* were included.

Gradient Boosting [GBM][5]

GBM was selected for its regression abilities to combine multiple weak predictors to create a more robust model. This uses a decision tree for incorporating weak predictors.

Principal Component Regression [PCR][6]

Based on a standard linear regression model, PCR first applies an analysis by constructing principal components as the predictors in the regression model, which is fit using least squares.

Single Layer Hidden Neural Network [NN]

To better understand how a nonlinear model would interpret the data, a simple NN was utilized. Given depth and complexity of the dataset, only a single hidden layer was utilized.

5 Results

5.1 Density Predictions

Error Metrics

Prior to a rigorous analysis, the test and train errors needed to be compared to ensure the model did not have an inherent bias to the testing data. A simple root mean squared error [RMSE] metric was applied to the testing and training set, whose results are summarized in *Table 1*. These errors were comparable, implying that the model did not generate a strong bias on the training set.

Table 1 - Error Metrics for Demand Density

Train RMSE	Test RMSE
3.23	3.40

Density Analysis

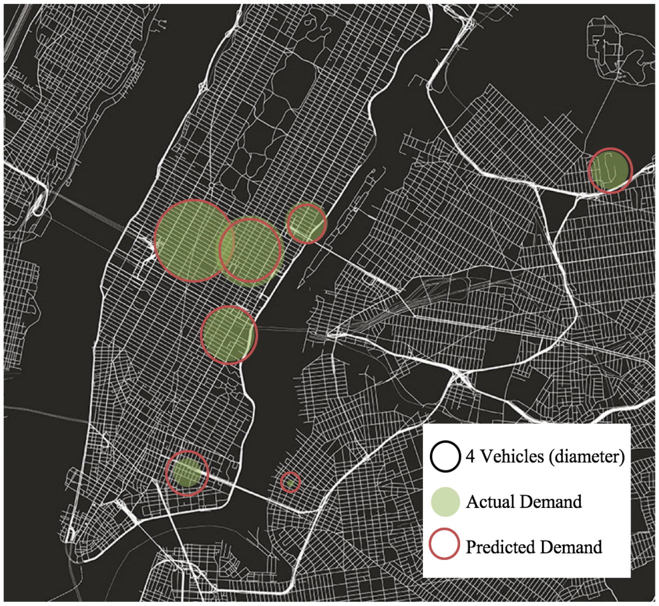

Figure 1 - Demand Density Map for 7 Districts in New York City For October 5th, 2016 at 3pm

Figure 1 displays a density map for 7 randomly selected locations for a likewise randomly selected day. The circles show that in general the model performs better in areas with higher demand. While the model could be further modified to better predict lower densities, it was not deemed necessary for the purpose of this project. The NYCTLC in particular may only be interested in areas with higher demand for airport travel simply because these trips generate the highest profits for the drivers.

5.2 Travel Time Predictions

Error Metrics

Once again, the RMSE metric was used, whose results are summarized in *Table 2*. These errors are reasonably close, thus supporting the idea that the model is robust and generalizable.

Table 2 - Error Metrics for Travel Time Prediction

Model	Test RMSE	Train RMSE	Test MRSE	Train MRSE
Linear Regression	10.87	9.95	7.41	7.17
Gradient Boosting	10.43	9.11	6.95	6.48
PCR	10.8	9.96	7.24	6.91
Neural Net	11.27	10.45	7.72	7.59

General Travel Predictions

Beyond the RMSE, a more interpretable metric was required. To determine a comparable time metric (meaning what the generalized window of accuracy was for travel time in terms of minutes), the mean square root error (MRSE) was used.

$$MRSE = \sqrt{\sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2 * 1/m}$$

This metric showed by how much time, on average, the model differed from reality. Again, *Table 2* highlights that in general, each model gave predictions that were on average off by 7 minutes. Considering that the average travel time was approximately 50 minutes, it seems these models performed reasonably well. These metrics show that the GBM Algorithm had the best performance, and was the chosen model for further analysis.

Time Influence

Given the practical implications of the model, we wanted to test error metrics on specific time ranges. This allowed us to see which hours of the day had better or poorer predictions.

Figure 2 outlines the predicted and actual travel times for a randomly chosen day using the GBM. This model does a fair job of picking up the overall travel trends throughout the day (red line). However, it seems to poorly interpret the travel times for the earlier hours.

Table 3 shows some of the results from the time based analysis. The results correspond to the following time blocks: 8 - 11 am, 4 - 7 pm, and 8 - 11 pm. These times were chosen to correspond to times of particular interest for transportation planning: morning rush hour, evening rush hour, and night hours respectively. Confirming what is seen in *Figure 2*, the morning hours were notably harder to predict, while the evening hours were easiest.

Table 3 - Error Metrics for Travel Time Predictions Given the Time of Day

Time Block	Test RMSE
8 - 11 am	11.92
4 - 7 pm	10.11
8 - 11 pm	7.33

It is assumed that these metrics differed so much by time block because there were model features that were not included that may have been able to better predict the morning hour travel times. Perhaps more features could be explored in future work.

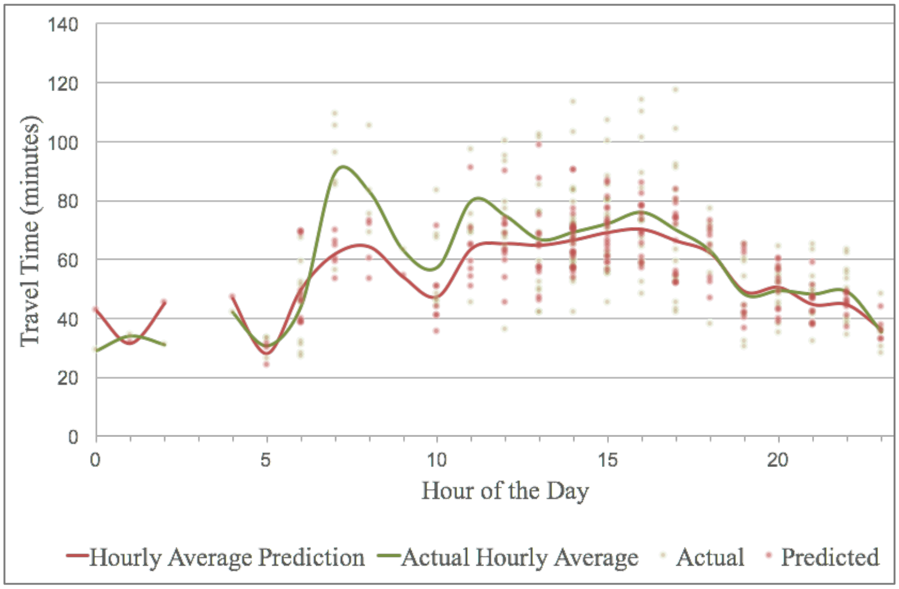

Figure 2 - Actual and Predicted Travel Time
From Midtown Manhattan to JFK in October 2016

Weather Influence

Likewise, there was value in determining how well the models performed when considering specific weather conditions. The weather dataset provided the four general weather conditions: fog, snow, rain, and sun/cloud. The data was then divided by these weather conditions, and were again input into the GBM.

Table 4 - Error Metrics for Travel Time Predictions Given the Time of Day

Weather	Test RMSE
Fog	4.95
Snow	8.87
Rain	10.24
Sun/Cloud	10.63

Surprisingly, the model had the best performance in conditions of fog, and the worst with the sun. Perhaps this is because more people tend to be outside in better weather conditions, thus adding an additional layer of unpredictability.

Hand Crafted Feature Influence

While all hand crafted features were highly significant for the final model, their influence on reducing the final RMSE values was minimal.

6 Conclusion and Future Work

While individuals may be unpredictable, their collective travel patterns are not. This project was able to both analyze and predict within reasonable accuracy the anticipated travel times and demand for airport based taxi rides in New York City. We saw that these models performed well on average, but did not have an appropriate degree of flexibility. Beyond providing insights to the NYCTLC, the City of New York could utilize the developed models and apply the predictions to the for better understanding of general traffic patterns. If implemented correctly, trip density could be further analyzed at the minute level, supporting better designed public transit routes, along with promoting actual ride share services that predict consumer demand to generate shared routes.

Contributions

For this project, we were always able to collaborate in person. While the workload was split evenly, Dominik focused on data cleansing and processing, while Sierra worked on model setup.

References

[1]N. Laptev, J. Yosinski, E. Li, and S. Smyl, "Time-series extreme event forecasting with neural networks at Uber," International Conference on Machine Learning, 2017.

[2]"NYC Taxi & Limousine Commission - Trip Record Data", *Nyc.gov*, 2017. [Online]. Available: http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml. [Accessed: 11- Dec- 2017].

[3]"New York, NY Forecast | Weather Underground", *Weather Underground*, 2017. [Online].Available: https://www.wunderground.com/weather/us/ny/new-york-city. [Accessed: 11-Dec- 2017].

[4] The link to the github code: https://github.com/sierradominik/R-codes]

[5] C. Li, (2017). A Gentle Introduction to Gradient Boosting. [online] *North Eastern University*. Available at: http://www.ccs.neu.edu/home/vip/teach/MLcourse/4_boosting/slides/gradient_boosting.pdf [Accessed 16 Dec. 2017].

[6] J. Gareth, D. Witten, T. Hastie, R. Tibshirani, "An Introduction to Statistical Learning: With Applications in R." 2013. Springer, New York. Print.

Libraries Utilized

The following libraries were used in R:
Readr
Readxl
Tidyverse
Dplyr
Stringr
Glmnet
AppliedPredictiveModeling
Caret
E1071
Earth
Leaps
Boot
Kernlab
HydroGOF