New York City Taxis:

Prediction of Trip Duration & Trip Cost

Business Understanding

We are living in a world, where technological innovation is quickly immersed into our daily lives. Regardless of people's perception, this global phenomenon improved quality of life and even created diverse emerging markets for "newcomers".

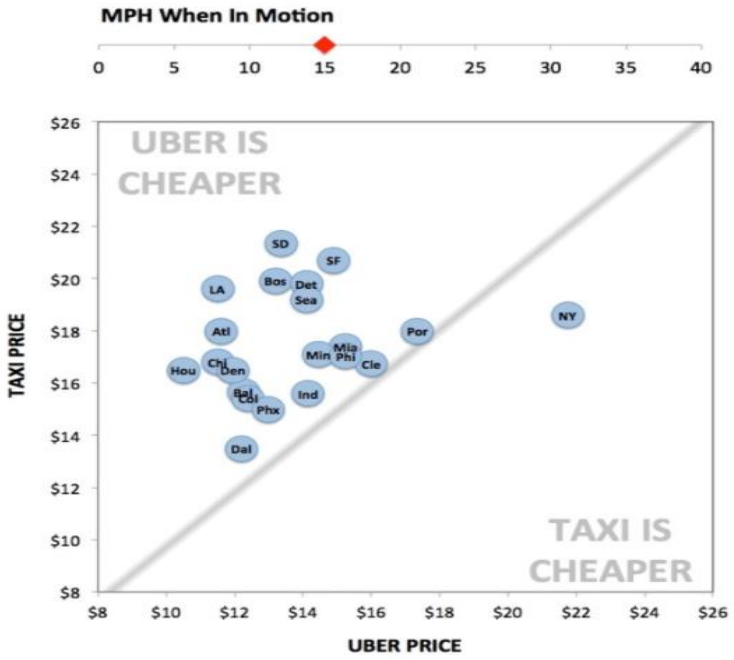
However, as Austrian-American economist Schumpeter insisted in 1950s about creative destruction, or the "process of industrial mutation that incessantly revolutionize[d] the economic structure from within, incessantly destroying the old one, incessantly creating new ones." - technological innovation destroys the precedent market economy. In transportation industry, the "newcomers", are Uber and Lyft and those "newcomers" begun to reign over the traditional taxi service operating market as they brought in on demand real time ridesharing (O2O) services in its businesses operating system. Therefore, traditional taxi business operators are now seeking better ways to maintain their customers to survive in the law of the jungle.

The most popular traditional taxi business operator would be New York City Taxicabs. From the late 19th century, New York City taxicab have earned its global fame as it was frequently exposed to various Hollywood film scenes taken in metropolitan of New York city. Especially it is rather known as a "Yellow Cab" and it is now an iconic feature of the New York City.

New York taxicabs operates through five boroughs of New York City, which are Upper Manhattan, the Bronx, Brooklyn, Queens and Staten Island. And these taxicabs are licensed by single private companies, which is a New York City Taxi and Limousine

Commission(TLC). New York City Taxi Cab business began to shrink as the culture of "creative destruction" penetrated the precedent taxi cab market in New York city since 2011. (Wikipedia, https://en.wikipedia.org/wiki/Taxicabs_of_New_York_City)

According to the article "These Animated Charts Tell You Everything About Uber Prices In 21 Cities" pressed by business insiders in 2014, New York City is the only city throughout the US that has lower average taxi fees compared to Uber.



(http://www.businessinsider.com/uber-vs-taxi-pricing-by-city-2014-10)

Our goal for this project is to explore and gather insights from the New York Yellow Cab Trip data and find informative values by applying raw data into the data mining algorithms and testing through models to propose any actionable solutions to optimize the New York Yellow Cab company operation.

Business Context & Business Problem

Our team was approached by a venture capitalist to see the possibility of creating an app like 'Kayak', but for local transport within New York City. This app would allow a user to enter a

pickup point (say point A) and a drop off point (say point B), and give the comparative fares of all taxi services that were available to get him from A to B. Now, scraping this data from Uber, or Lyft or Via or any other real time ridesharing app was possible, but this was not possible for the oldest of the services present – the New York City yellow taxis. The client wanted us to provide reliable information to passengers in terms of what was the projected cost and duration of a trip in a NYC yellow taxi.

So, **the business problem** is as follows: Is it possible to predict the trip cost, and the trip duration, for a New York City yellow cab, as is done by apps like Uber or Lyft? To solve this problem, we looked at the data from the licensing service – the New York Taxi and Limousine Commission (TLC). This data is available at this location: http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml

Data Understanding

The Trip Record Data that was available with the NYC TLC had was not created by them but was collected by technology providers (meters) licensed under Taxicab Passenger Enhancement Programs (TPEP) or Livery Passenger Enhancement Programs (LPEP).

The available data had records for yellow cabs, green (boro) cabs, and For Hire Vehicles (FHV's), but we considered only the dataset for yellow taxi trip records for the year 2016. The total data available or all months was as follows:

Month	Records
January'16	10,906,859
February'16	11,382,050
March'16	12,210,953
April'16	11,934,339
May'16	11,836,854
June'16	11,135,471
July'16	10,294,081
August'16	9,942,264
September'16	10,116,019
October'16	10,854,627
November'16	10,102,129
December'16	10,449,409
Total	**131,165,055**

This data had a total of 131 million observations (rows) with 19 different features. These 19 features are detailed listed out here:

- VendorID
- tpep_pickup_datetime
- tpep_dropoff_datetime
- passenger_count
- trip_distance
- pickup_longitude
- pickup_latitude
- RateCodeID
- store_and_fwd_flag
- dropoff_longitude
- dropoff_latitude
- payment_type
- fare_amount
- extra
- mta_tax
- improvement_surcharge
- tip_amount
- tolls_amount
- total_amount

The schema and data dictionary for the larger data set is available here: http://www.nyc.gov/html/tlc/downloads/pdf/data_dictionary_trip_records_yellow.pdf. A brief explanation of the variables in our file is as follows:

1. **Vendor ID (integer, categorical):** A code of TPEP provider. There are two possible values for this variable:
    a. 1 for Creative Mobile Technologies LLC
    b. 2 for VeriFone Inc.
2. **tpep_pickup_datetime (timestamp):** The datetime when meter was engaged.
3. **tpep_dropoff_datetime (timestamp):** The date and time when meter was disengaged.
4. **passenger_count (numeric):** Number of passengers boarded in taxi. This is a value that is entered manually by the driver, and hence can have erroneous values.
5. **trip_distance (numeric)**: the elapsed trip distance in miles reported by taximeter from starting point of the meter engagement to the meter disengagement.
6. **pickup_longitude (float)**: Longitude of the pickup location where the meter was engaged.
7. **pickup_latitude (float)**: Latitude of the pickup location where the meter was disengaged.
8. **RateCodeID (integer, categorical)**: the final rate code influencing the rate at the end of the trip. There are separate rate codes for Airports, Nassau or Westchester, or Group rides. This variable can take on six different values:
    a. 1 for Standard rate
    b. 2 for JFK
    c. 3 for Newark
    d. 4 for Nassau or Westchester
    e. 5 for Negotiated fare
    f. 6 for Group ride

9. **store_and_fwd_flg (binary):** this flag indicates the trip record was collected in vehicle memory before sending to the vendor due to the lost in connection to the server. This is just a flag and does not influence anything on the trip. This variable can have values of yes or no.
10. **dropoff_longitude (float):** Longitude of the drop location where meter was disengaged.
11. **dropoff_latitude (float)**: Latitude of the drop location where meter was disengaged.
12. **payment_type (integer, categorical):** This is a numeric code indicating the payment type availed of by the passenger. There are six possible values for this variable:
    a. 1 – Credit Card
    b. 2 – Cash
    c. 3 – No Charge
    d. 4 – Dispute
    e. 5 – Unknown
    f. 6 – Voided Trip
13. **fare_amount (numeric):** The overall fare calculated by the meter, including cost of idling time and distance fare.
14. **Extra (numeric):** Miscellaneous extras and surcharges in $. As of now, this only includes two possible values, for rush hour rides and overnight rides.
    a. 0.50 – rush hour charge
    b. 1.00 – overnight ride charge
15. **mta_tax (numeric):** a $0.50 MTA tax automatically triggered based on metered rate in use.
16. **improvement_surcharge (numeric):** a $0.30 improvement surcharge applied to trips at flag drop. The improvement surcharge began to be applied since year 2015.

17. **tip_amount (numeric):** automatically charged for total credit card at the end of the trip. Cash tips are not included.
18. **tolls_amount (numeric):** total toll fees paid in trip.
19. **total_amount (numeric):** total amount charged to passengers before tip (including tolls_amount, improvement_surcharge, MTA_tax , extra, fare_amount).

Data Preparation

We saw that some files in the data (July'16 to December'16) had a different schema as compared to the other files (January'16 to June'16) – the variables 6 and 7 were combined to make one 'pickup location ID' and the variables 10 and 11 were combined to make one 'drop off location ID'. To make the schema uniform, we had to drop all these variables – pickup and drop off latitude, pickup and drop off longitude, and pickup and drop off location ID.

Also, given we had a total dataset that had 131 million rows, we did not want to handle this entire data all at once. So, we started our work only on the data for January 2016, which included about 11 million data points. We checked to see if there were any errors in the data but did not find any NA's in the data.

Once this was done, we looked at descriptive summary statistics.

	count	mean	std	min	0.25	0.50	0.75	max
VendorID	10,906,860	1.54	0.50	1.00	1.00	2.00	2.00	2.00
passenger_count	10,906,860	1.67	1.32	0.00	1.00	1.00	2.00	9.00
trip_distance	10,906,860	4.65	2981.10	0.00	1.00	1.67	3.08	8000010.00
RatecodeID	10,906,860	1.04	0.52	1.00	1.00	1.00	1.00	99.00
payment_type	10,906,860	1.35	0.49	1.00	1.00	1.00	2.00	5.00
fare_amount	10,906,860	12.49	35.56	-957.60	6.50	9.00	14.00	111270.90
extra	10,906,860	0.31	0.42	-42.61	0.00	0.00	0.50	648.87
mta_tax	10,906,860	0.50	0.05	-0.50	0.50	0.50	0.50	89.70
tip_amount	10,906,860	1.75	2.62	-220.80	0.00	1.26	2.32	998.14
tolls_amount	10,906,860	0.29	1.69	-17.40	0.00	0.00	0.00	980.15
improvement_surcharge	10,906,860	0.30	0.01	-0.30	0.30	0.30	0.30	0.30
total_amount	10,906,860	15.64	36.41	-958.40	8.30	11.62	17.16	111271.60

To make for easier modeling, we imported a sample of 10% of the total 10.9 million rows to explore the dataset and test the model. As seen in the snapshot above, there were multiple variables that had outlier values, so there were multiple types of data cleaning that we did once we extracted this sample to ensure we had a clean sample to model from.

Firstly, we had to select target variables for the data we were going to be predicting – total trip cost, and total trip duration. We already had trip cost in the dataset but did not have a variable giving total trip duration. However, we did have the trip start time and trip end time. Given this, we created a target variable "total_duration" by converting pickup and drop off datetime variables into numeric variables and calculating the total duration in minutes that a certain trip took. This was calculated using the formula (("tpep_pickup_datetime" – "tpep_dropoff_datetime") / 60).

We also did not require some of the variables which would not add to predictive value – like VendorID, or improvement_surcharge (that had only values of 0.3 or 0), or payment_type. Hence, we removed these from the dataset used for modeling.

Secondly, in our initial data exploration, we had seen a lot of outliers, which would have skewed our model. Hence, we filtered the dataset to remove outliers. Assuming operation around the areas of NYC, we filtered to use dataset to make sure that the total duration was within range of 1 minute to 180 minutes. This is because we felt that it was unrealistic to have any trip ending in less than one minute after pickup and lasting longer than 3 hours after pickup. This was calculated using the range 1 minute < "total_duration" < 180 minute, and this removed all outliers in the duration variable – there were some values that had a trip duration of 23 hours that were removed. These were because the dates (trip pickup date and trip drop off date) were misaligned.

Another variable we cleaned was the total_amount, which we felt should not cross $500, and hence we filtered all values of "total_amount" that were over $500. There were

also a lot of negative values in the fare_amount, the extra and tax variables, which did not logically make sense, hence we removed them from the dataset. There was one single data point which had a trip distance of 8,000,000 (8 million miles!) which was also removed.

Thirdly, we factorized categorical variables “Vendor ID”, “RatecodeID”, “store_and_fwd_flag”, and “payment type” to convert categorical variables into numeric form. This was to ensure that the values in these variables did not adversely affect our model.

Finally, we checked to see if there were any null or N/A values existing in the filtered dataset and found there was no null or N/A values in our dataset. Once this entire cleaning was done, we checked summary statistics again, to see if the data we had was clean.

	count	mean	std	min	0.25	0.50	0.75	max
passenger_count	1,079,685	1.67	1.33	0.00	1.00	1.00	2.00	9.00
trip_distance	1,079,685	2.92	3.63	0.00	1.00	1.70	3.10	208.10
fare_amount	1,079,685	12.45	10.58	0.01	6.50	9.00	14.00	500.00
extra	1,079,685	0.31	0.37	0.00	0.00	0.00	0.50	1.50
mta_tax	1,079,685	0.50	0.03	0.00	0.50	0.50	0.50	2.22
tip_amount	1,079,685	1.75	2.45	0.00	0.00	1.28	2.34	450.00
tolls_amount	1,079,685	0.29	1.59	0.00	0.00	0.00	0.00	755.54
total_amount	1,079,685	15.61	13.06	0.31	8.30	11.62	17.16	795.84
duration	1,079,685	13.26	10.26	1.00	6.42	10.53	16.93	176.77

Modeling & Evaluation

Given that we are trying to predict a numerical value of Trip Duration and Trip Cost, we needed to run models that would give us a numeric output and not a classification result. We ran the following models on the sample data set to get an idea of which is the best model to run on the bigger, overall data set.

- Linear Regression
- Lasso Regression
- Ridge Regression
- Regression Trees
- Neural Nets

- AdaBoost Regression

In terms of features that should go into the model, since we decided to build two models and test explanatory variables on two target variables, we did feature selections individually for two different target variables which are “total_amount” and “total_duration”. For the feature selection, we used stepwise method: both backward and forward selections to determine the most statistically significant variables that should go into the models. The parameters used to judge the model results was RMSE at a 95% confidence interval. We used the RMSE as a percentage of the average value of the target variable to determine how accurate the model was in terms of predicting the Trip Cost and Trip Duration. For the sample that was used (10% of the overall data), the average values of Trip Cost and Trip Duration were:

Sample Average	Trip Duration	13.3 mins
	Trip Cost	15.6 USD

The results of the models are given below:

Model	Parameters	RMSE (Trip Duration)	Percentage of Average	RMSE (Trip Cost)	Percentage of Average
Linear Regression		6.29	47.3%	3.30	21.2%
Lasso Regression	alpha = 0.1	6.31	47.4%	3.49	22.4%
Ridge Regression	alpha = 0.1	6.29	47.3%	3.30	21.2%
Regression Trees	max_depth = 3, min_samples_split=10, splitter = 'best'	5.99	45.0%	4.85	31.1%
	max_depth = 5, min_samples_split=10, splitter = 'best'	5.76	43.3%	3.63	23.3%
	max_depth = 9, min_samples_split=10, splitter = 'best'	5.60	42.1%	2.90	18.6%
AdaBoost Regression		7.18	54.0%	2.77	17.8%
Neural Nets	Max_iter = 200, hidden_layer_sizes=20, activation = identity, learing_rate = constant	6.29	47.3%	2.56	16.4%
	Max_iter = 100, hidden_layer_sizes=20, activation = logistic, learing_rate = constant	5.59	42.0%	2.61	16.7%
	Max_iter = 100, hidden_layer_sizes=20, activation = logistic, learing_rate = invscaling	5.59	42.0%	2.53	16.2%
	Max_iter = 100, hidden_layer_sizes=20, activation = logistic, learing_rate = adaptive	5.58	42.0%	2.54	16.3%

Once we tested the model on the sample set of 10% of the January 2016 data and got results in terms of the best model, we ran it on the whole data for the month of January (10.9 million

rows) to see what results we got. We ran one iteration for each model – the parameters taken were those for the best result given by that model as per the sample. The results were as follows:

With Total Data

Model	RMSE (Trip Duration)	Percentage of Average	RMSE (Trip Cost)	Percentage of Average
Linear Regression	6.35	47.7%	3.32	21.3%
Lasso Regression	6.31	47.4%	3.50	22.4%
Ridge Regression	6.35	47.7%	3.32	21.3%
Regression Trees	5.60	42.1%	2.76	17.7%
Neural Nets	5.60	42.1%	2.60	16.7%

Points of Modeling:

- Neural Nets gave us the best value in terms of RMSE for both trip duration as well as trip cost. This was achieved by using different values and parameters for the learning rate. This was true for both the smaller sample data set as well as the larger complete dataset for January 2016.
- For lasso and ridge regression in the sample, we tried different values of alpha (1, 0.5, 0.3, 0.1) before arriving at the best value to give us optimal RMSE values. This was done manually, because the program kept crashing when we tried to do a grid search to get the best value of alpha.
- For regression trees in the sample, we got lower values of RMSE as and when we increased the depth of the tree. There were magnitudes of difference in the RMSE between the initial depth considered and the final depth taken.
- Once we got the parameters for the best regression tree, we applied the same parameters to AdaBoost, to see if we could get a better value. We managed to get a value very soon (this was the fastest of the lot), but not a better value. We did not run AdaBoost for the final larger dataset because we did not get any benefit from running that with the smaller sample dataset.

- We took both a regular sample as well as a shuffled sample – to see the difference in results, and the best results were published.

Conclusion

While we got a decent value – low RMSE as compared to average value – for the trip cost, we did not get a similar result for trip duration. This was true for the models run both on the sample set of 1 million rows, as well as the final set of 11 million rows.

We feel, given our accuracy in terms of trip cost, that there is value in doing this same exercise with a much larger data set (say the entire year of 2016) to see if we an get a better result – smaller RMSE and more accurate predictions. However, that requires both time, effort as well as cost in terms of computing power. Some more analyses we would like to do going forward is to check if the algorithm is different for different months, or different times of day, or for different days of the week – but this would be more descriptive analysis than predictive analysis.

In conclusion, we would like to say that despite the data that we have and the results that we have got, a lot more exploration needs to be done before we can get a model that would satisfy the needs of consumers in NYC.