**Analysis of the factors affecting the Fare amount, in New York City Yellow cabs**

Aditya Gautam

Date: 10/12/2018

**Abstract**

New York City taxis are often cited as the most expensive in the country and not yet particularly profitable for the driver. After paying a fee to the cab leasing company, covering gas and tolls and finally waiting in the queues, a driver may only take about $100 for a 12-hour shift. The objective of this study is to determine ways for an individual cab driver to be profitable. To accomplish this, we used data set for yellow cabs for the first 4 days of June 2018. The dataset includes a variety of information for each ride such as fare, tip, payment method, no. of passengers, surcharge, tax, tolls, etc. Using Linear regression and Statistical Machine Learning techniques we try find the factors affecting the fare.

*Keywords:* Fare amount, pickup time, pickup location, trip distance, trip duration

**Introduction**

**Data Preparation**

The dataset was selected from *http://www.nyc.gov/html/* for yellow cabs for the month of June 2018. For the first four dates the dataset gave us more than a million observations for the variables which amounts to about 35,000 observations per day. The dataset contained the variables ‘pickup\_time’ and ‘dropoff\_time’ in the standard time stamp format. So, for computational purposes these variables were converted to minutes of the day for e.g., if a taxi picked up a passenger at 00:15:40 it is represented as 15.67, which is 15.67th of that particular day. Here is the data description:

|  |  |
| --- | --- |
| **Field Name** | **Description** |
| VendorID | A code indicating the TPEP provider that provided the record.  **1= Creative Mobile Technologies, LLC; 2= VeriFone Inc** |
| Pickup\_time | The date and time when the meter was engaged |
| Dropoff\_time | The date and time when the meter was disengaged |
| Passenger Count | Number of Passengers in the cab. This is a driver entered value. |
| PULocation ID | The zone where the meter was engaged |
| DOLocation ID | The zone where the meter was disengaged |
| RatecodeID | |  | | --- | | The final rate code in effect at the end of the trip.  **1= Standard rate**  **2=JFK**  **3=Newark**  **4=Nassau or Westchester**  **5=Negotiated fare**  **6=Group ride** | |
| Store\_and\_fwd\_flag | This flag indicates whether the trip record was held in vehicle memory before sending to the vendor, aka “store and forward,” because the vehicle did not have a connection to the server.  **Y= store and forward trip**  **N= not a store and forward trip** |
| Payment\_type | A numeric code signifying how the passenger paid for the trip.  **1= Credit card**  **2= Cash**  **3= No charge**  **4= Dispute**  **5= Unknown**  **6= Voided trip** |
| Fare\_amount | The time and distance fare calculated by the meter. |
| Extra | Miscellaneous extras and surcharges. Currently, this only includes the $0.50 and $1 rush hour and overnight charges. |
| MTA\_tax | $0.50 automatically applied |
| Improvement\_surcharge | $0.30 improvement surcharge assessed trips at the flag drop. The improvement surcharge began being levied in 2015. |
| Tip\_amount | Tip amount – This field is automatically populated for credit card tips. Cash tips are not included. |
| Tolls\_amount | Total amount of all tolls paid in trip. |
| Total\_amount | The total amount charged to passengers. Does not include cash tips. |

*Table 1: data dictionary for trip records.*

To capture the duration of the trip a variable ‘trip\_duration’ was created by subtracting ‘pickup\_time’ from ‘dropoff\_time’. The date (day of the month) was separated from these variables and were made into features.

**Methods Applied**

**Linear Regression**

Linear regression was applied with the following formula:

*reg='fare\_amount~minute\_pickup+PULocationID+tip\_amount+tolls\_amount+VendorID+*

*DOLocationID+trip\_duration+trip\_distance'.*

The dependent variable here is ‘fare\_amount’ instead of ‘total\_amount’ because these two are highly correlated. The correlation value between the two came out to be 0.983168, which justified our choice. Same was the reason why only ‘minute\_pickup’ was included and ‘minute\_dropoff’ was not due to high correlation value of 0.933388. ‘RatecodeID’ and ‘Payment\_type’ were not chosen due to their discrete nature. ‘Extra’, ‘Improvement\_surcharge’ and ‘MTA\_tax’ were not chosen because these are constant values which are added to the fare due to regulations. ‘Store\_and\_fwd’ was not considered for the model because it is a manual process which doesn’t affect the fare amount.

**Results of the Linear Regression**

The following results were obtained after applying Linear Regression:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | coef | std err | t | P>|t| | [0.025 | 0.975] |
| Intercept | 4.3696 | 0.046 | 94.996 | 0 | 4.279 | 4.46 |
| minute\_pickup | 0.0005 | 2.36E-05 | 20.779 | 0 | 0 | 0.001 |
| PULocationID | 0.0005 | 0 | 3.886 | 0 | 0 | 0.001 |
| tip\_amount | 0.514 | 0.004 | 124.288 | 0 | 0.506 | 0.522 |
| tolls\_amount | 0.3881 | 0.007 | 56.473 | 0 | 0.375 | 0.402 |
| VendorID | -0.0194 | 0.018 | -1.077 | 0.282 | -0.055 | 0.016 |
| DOLocationID | -0.0007 | 0 | -5.578 | 0 | -0.001 | 0 |
| trip\_duration | 0.0008 | 6.42E-05 | 13.01 | 0 | 0.001 | 0.001 |
| trip\_distance | 2.4584 | 0.003 | 790.293 | 0 | 2.452 | 2.464 |

|  |  |  |  |
| --- | --- | --- | --- |
| Dep. Variable: | fare\_amount | R-squared: | 0.58 |
| Model: | OLS | Adj. R-squared: | 0.58 |
| Method: | Least Squares | F-statistic: | 1.81E+05 |

Prob (F-statistic): 0.00

Only the variable VendorID was not significant in predicting the ‘fare\_amount’

The model also demonstrated collinearity between the features which prompted the use of other techniques for explanation. The variable ‘trip\_distance’ was the most influential in the model, as it had the highest coefficient. Which makes sense that the distance being the most important factor in predicting the fare.

**Neural Networks**

To get a better explanation the factors affecting the fare we applied Neural Networks to the dataset. The complicated nature of the working of the Neural Networks technique, the dataset was split in small parts containing only 14,000 rows of observations. The method applied here was Stochastic Gradient Descent. The test-train split was set at 30% for the dataset with maximum iterations equalling 2500 and hidden layers equalling 300. The following plot shows the comparison between predictions of the Neural Networks and actual values of ‘fare\_amount’ in the datasets.

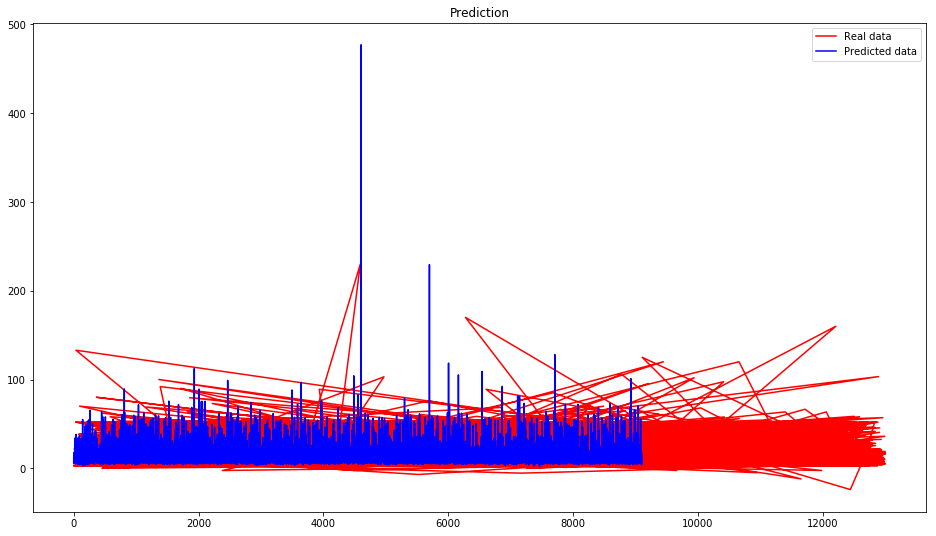


Figure 1: Prediction vs actual values

The MSE score for this technique equalled 54.783 and the R-squared value was 0.66265.

This method was applied to 20.5% rows of the total observations for cross-validation and to check if it can be helpful in understanding the factors affecting the outcome. For computational purposes the method was applied to sections of dataset each containing 14000 rows.

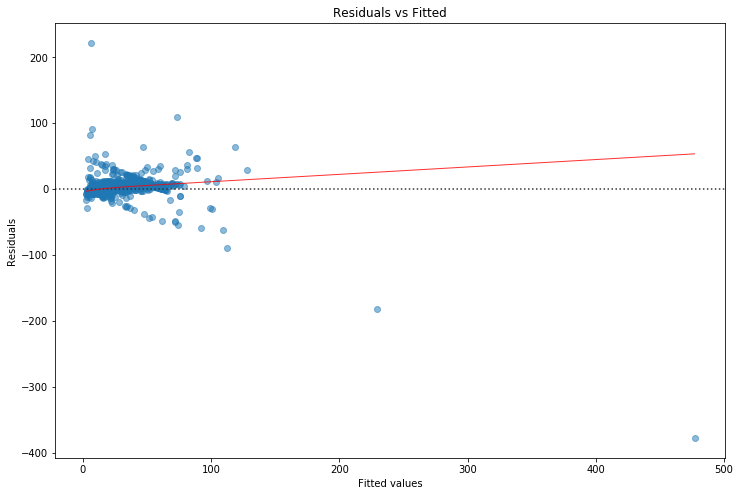


Figure 2:Residuals vs Fitted values plot for Neural Networks.

The following results were obtained.

|  |  |
| --- | --- |
| **Dataset** | **R2 score** |
| Df2 | 0.8204749229768078 |
| Df3 | 0.9113437167857258 |
| Df4 | 0.8696416035823988 |
| Df5 | 0.8603467211362432 |
| Df6 | 0.9366760168960606 |
| Df7 | 0.928237529271955 |
| Df8 | 0.8910535082474211 |
| Df9 | 0.9496613674119596 |
| Df10 | 0.5129237521199951 |
| Df11 | 0.8801910490439734 |
| Df12 | 0.8994861680563097 |
| Df13 | 0.9072205823798998 |
| Df14 | 0.7967239834836015 |

*Table 1: Neural Networks results.*

The following is the summary of parameters used in this method.

MLPRegressor(activation='relu', alpha=0.0001, batch\_size='auto', beta\_1=0.9, beta\_2=0.999, early\_stopping=False, epsilon=1e-08, hidden\_layer\_sizes=300, learning\_rate='constant', learning\_rate\_init=0.001, max\_iter=2500, momentum=0.9,nesterovs\_momentum=True, power\_t=0.5, random\_state=None, shuffle=True, solver='sgd', tol=0.0001, validation\_fraction=0.1, verbose=False, warm\_start=False)

This method performed very well on the dataset and the R2 values confirm that. The Stochastic Gradient Descent (SGD) method is used here, as it is the prime algorithm used on Neural Networks, it also provides a simple way to produce accurate results in deep networks. The stochastic gradient descent can give an approximate estimate of loss using a small set of samples lessening the amount of compute required. However, due to the black box nature of neural networks the interpretation of the model is very difficult. One way of seeing what a neural network is doing would be to probe the network using test inputs and measuring the impact of input variables on the outputs where the anticipated results are known. A bit like the interrogation technique of asking a subject a question to which you know the answer to test the accuracy of the subject. This could be done by tracking error terms during the backpropagation step and then measuring the amount that each input impacts the output. The rules that the network uses can be extracted by studying the weights that are assigned to the hidden layers in the path between inputs and outputs. The knowledge obtained from the network can lead to new insights into patterns and dependencies within the data.

**Principal Component Analysis**

For dimensionality reduction PCA is applied after standardizing the data such that it has unit variance. This is necessary because fitting algorithms highly depend on the scaling of the features. This method was chosen because, by projecting our data into a smaller space, we’re reducing the dimensionality of our feature space, but because we have transformed our data in these different “directions”, we have made sure to keep all original variables in our model.

After removing the features, as per the reasons justified before, the following explained variance and eigen values were observed:

|  |  |  |
| --- | --- | --- |
| Principal Component | Explained Variance | Eigen Value |
| PC1 | 0.19548491 | 1501.59392436 |
| PC2 | 0.11253581 | 1139.30830274 |
| PC3 | 0.11226302 | 1137.92657878 |
| PC4 | 0.10582509 | 1104.81664241 |
| PC5 | 0.09495049 | 1046.51264864 |
| PC6 | 0.0917884 | 1028.93935187 |
| PC7 | 0.08079846 | 965.37808133 |
| PC8 | 0.069583222 | 895.54816508 |
| PC9 | 0.06933613 | 894.28442945 |
| PC10 | 0.03962039 | 676.01329528 |
| PC11 | 0.02786509 | 566.92563882 |

*Table 2: Explained variances and Eigen Values.*

The first seven principal components explain 79.23% of the variability. After applying ‘inverse pca transform’ and ‘inverse standard normal transform’ on the data, regression was performed. The results of the regression were not that different from when it was performed earlier. This could be attributed to the multicollinearity within the data. No threshold was set, regarding the total explained variance for feature selection, because the intention here was to get a full picture painted by all the features.

The following are the regression results obtained after applying PCA:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | coef | std err | t | P>|t| | [0.025 | 0.975] |
| Intercept | 4.17 | 0.047 | 89.237 | 0 | 4.078 | 4.262 |
| minute\_pickup | 0.0005 | 2.39E-05 | 20.423 | 0 | 0 | 0.001 |
| pickup\_day | -0.0441 | 0.008 | -5.549 | 0 | -0.06 | -0.029 |
| PULocationID | 0.0005 | 0 | 4.032 | 0 | 0 | 0.001 |
| tip\_amount | 0.5506 | 0.004 | 134.608 | 0 | 0.543 | 0.559 |
| VendorID | -0.0193 | 0.018 | -1.066 | 0.286 | -0.055 | 0.016 |
| trip\_duration | 0.0009 | 6.43E-05 | 14.359 | 0 | 0.001 | 0.001 |
| trip\_distance | 2.5431 | 0.003 | 929.281 | 0 | 2.538 | 2.548 |

R-squared: 0.579

Prob (F-statistic): 0.00

There is no improvement in the model after selecting the features as per the explained variance by the PCA. Still the variable ‘trip\_distance’ remains the most influential one, and the variable ‘VendorID’ is not significant in predicting the fare amount for the trip. This can be related to the fact that, there are only two providers for the yellow cabs in New York City. The multicollinearity issue needs to be examined further because it is for this reason that we are getting low explained variance ratio values.

**Other Methods for consideration**

Forcing a quadratic function on the variables related to location identification seems like an obvious approach for optimization using Newton Raphson method. But, the discrete nature of the variable prevents us from forcing a parabolic curve on the data to search for the minima.

Genetic algorithm seems to be the best bet to get more insight regarding the factors affecting the fare of the trip. A heuristic approach can surely get us to the optima, as it is doing the same thing as calculus-based method. But, the cost parameters regarding the trip distance, fuel consumption, mechanical wear and tear of the vehicle, the fluctuation of fare during the time of the day and pickup location are needed to perform this method. Knowledge of these parameters will help us in manufacturing of an objective function which can be minimized. This is a shortcoming of the dataset.

**Conclusion**

From the analysis conducted on this dataset there is clear evidence of multicollinearity within the predictor variables. Performance of Neural Networks is the best from the rest, but it is the black box nature of the method that ties our hands. It presents a similar kind of dilemma that do we need understand electricity to turn on a light bulb. Sobol indices may help in the sensitivity analysis that will help us to get past the Bryman’s paradox.