Carto Take Home Test

Tip Prediction Model – NYC Yellow Taxi Data

Project by Bisan Alhmood

Bisan.Alhmood@columbia.edu
+1 (917) 415 – 3763

INDEX

Introduction

This report was written for the purpose of delivering insights into the transportation sector in New York for a new start up in the ride sharing industry. The company is looking to build a model that predicts tip amount and presents it in their app as a suggestion at the end of each trip. I will be using NYC Yellow Taxi data that has been published by NYC Taxi & Limousine Commission. The data includes various information about daily taxi trips including;

- VendorID A code indicating the TPEP provider that provided the record.
- tpep_pickup_datetime: The date and time when the meter was engaged.
- tpep_dropoff_datetime: The date and time when the meter was disengaged.
- Passenger_count: The number of passengers in the vehicle.
- Trip_distance: The elapsed trip distance in miles reported by the taximeter.
- PULocationID: TLC Taxi Zone in which the taximeter was engaged.
- DOLocationID: TLC Taxi Zone in which the taximeter was disengaged.
- RateCodeID: The final rate code in effect at the end of the trip.
- 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.
- Payment_type: A numeric code signifying how the passenger paid for the trip.
- Fare_amount: The time-and-distance fare calculated by the meter.
- Extra: Miscellaneous extras and surcharges. Currently, this only includes the $0.5 and $1 rush hour and overnight charges.
- MTA_tax: $0.50 MTA tax that is automatically triggered based on the metered rate in use.
- Improvement_surcharge: $0.30 improvement surcharge assessed trips at the flag drop.
- 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.

I will be examining three months, March, June, and November of 2017. However, knowing the limitation of available machines and the short time frame set for this project, I will examine each Month separately. This will not affect the overall result of my analysis.

I have chosen R as the programming language for this project, as I believe I have the knowledge and experience to carry out all of tasks successfully.

It is worth mentioning before we dig in with all the technical elements, that this report has some limitations. It can be summarized with three main points; computational power/capability, time frame, learner/student experience. I will be addressing these issues as I explain my process of thought while tackling each task.

Data exploration and cleaning

Data exploration is the initial step I took in my data analysis process, the intent was to explore this large data set in an investigative mindset to unveil surfacing patterns, characteristics, and points of interest, just to help me create a broad picture of important features and major points to study in more depth as I progress though the analysis. This process makes deeper analysis easier, as I get to brainstorm ideas on what techniques I will be using.

My initial plan to handle the data was to merge the three months of values. However, since the data is quite large and with the limited capabilities of my machine, I had to figure out a different way. One option was to load one-month data and start exploring, cleaning, modeling and then use the best performing model to train and test on the other two data sets. The other option, which what I hoped to do, was to use AWS or google cloud platform to enable running any code I saw fit. However, I had a few drawbacks with my AWS account which required customer service handling that is expected to take one to three days. That is when I jumped right into exploring the March dataset.

I started pulling column names, structures of variables, simple mathematical calculations that reflects statistical indicators, examined the data's completeness, cleanliness, and quality. As you can see in my code, the data was not too messy, it did not have any missing values, and I thought over all it is in a fairly good shape.

Though, as I was getting to know the data, I noticed a few odd values. First, few transactions had an enormous value for total_amount, it was due to either outrageously large tolls_amount, large fare_amount or tip_amount. It did not make sense to have a taxi trip transaction equaling to more than $171,000. The other odd part was negative total amounts, the most significant one was -$308. Even though it is unheard of, the negative values can be explained as return transaction for dissatisfied customers. I chose to look further into extreme outliers by using boxplot() and filtering, and since size of the odd data was quite small, I decided to leave the data untampered with for my initial model. After building a simple model and training/testing it, I would revisit the idea of treating outliers, by either imputing/replacing values with a more reasonable point such as a mean or removing them all together.

As will be explained later in the report, I ended up removing a very small number of rows (<20) with values (>$1000) in total_amount.

The other main issue I believe was the sparsity of the variables, which I attempted to solve by creating more variables using the ones already existed in the dataset. I brainstormed what I could add by reflecting on my own experience of what would affect my choice of tipping my driver in a taxi ride. I used the distance, pick-up and drop-off times to calculate speed and duration variables. I also added time of the day variable by extracting the time (hours and minutes) only and then adding definition grouping (rush_hour, late_morning, afternoon, night, post_midnight). I then added IsWeekend variable by looking up days of the week and applying a simple Yes/no to whether it is a weekend or not. Running these codes took sizable amount of time, and that was what kept me from adding more. I initially planned to also add weather variables, such as temperature and IsRaining/IsSnowing by loading weather dataset and maping it to the date variable, as people usually tend to be more generous during bad weather for services such as this one. Another potentially useful variable would be adding average income by locationID, as we can figure out commuters' traffic and see if tipping is affected by higher/lower income neighborhood, where customers are being dropped off.

As a final step in preparing the data for analysis, I converted the format of a few variables such as, Store_and_fwd_flag, tpep_pickup_datetime/tpep_dropoff_datetime and timeofday to numeric, datetime format, and numeric, respectively.

Data Summary

Please review the R markdown accompanied with this report, which includes notes and interpretations of trends, graphs, and calculations.

Model Building

Before discussing my model building process. I believe it is important to explain what I am aiming to do with my model. I am aware that the task mentioned that I can assume that the ride sharing company can provide data that has the same attributes as the taxi data for each trip. Meaning I can use the same attributes just as if it was the company's own data. However, that does not necessarily mean that the company's app cannot use or provide more attributes. As mentioned earlier I decided to push the app by using more data in the hopes of providing the most accurate predictions and a recommendation of

tip_amount, which will increase the likelihood of the customer agreeing to use the recommended amount, thus, increasing the chance of tipping.

As I got to examine different modelling techniques, I came to understand that there can be a trade-off between what is simple and efficient, and what is complexity and accuracy. Initially I ruled out the idea of choosing a simple linear regression for this particular project, as I wanted to showcase sophistication and knowledge. However, to my shock, after analysis, model building and performance evaluation, I chose the linear regression modelling technique. The reason is that linear regression is an extremely simple method. It is very easy, intuitive to use, easy to interpret, and would lower processing time to a minimum, which can fit our business case here. It is also great to use in our case because most of our variables' relationship with the prediction is known to be linear. For example, it is a known fact that longer distance will result in a higher charge, and higher fare usually increases tip_amount as it is the norm, at least in the US, to tip as a percent of charge. This explains how simple linear regression, for our purpose, might be the best fit.

I used a combination of feature selection methods and judgment to pick the best set of variables for the model. For the first model I used a correlation cutoff of 0.2 for positive correlation and a -0.2 for negative one. The variables that made it through are, trip_distance+RatecodeID+payment_type+fare_amount+tolls_amount+total_amount.

I used Root Mean Squared Error (RMSE) to evaluate the model's performance, which came out to be approximately 0.46.

However, after examining several combination on variables, I arrived at the bet performing model that included the variables, fare_amount, extra, mta_tax, tolls_amount, improvement_surcharge, total_amount, with an RMSE of 0.1024497.

It is worth mentioning, that even though I ended up not using the variables that I created in my best performing model, I think they helped clarifing the data, and could be useful in future uses and further analysis. Also, might turn out to be useful for a different modeling technique.

Although, linear regression is a good fit here, it is important to understand its limitations in case it becomes an issue when implementing it.

- Linear regression is limited to linear relationships; making it miss non-linear/ possibly beneficial relationships.
- Linear regression only looks at the mean of the dependent variable, which risks sometimes missing important extreme values of the dependent variable.
- Linear regression is sensitive to outliers.
- Data must be independent which might not always be the case

Moving forward, I would work on improving my model by including more variables such as the ones that were discussed earlier (weather, income, etc). If I had more time, I would cluster the data and tailor my model to each cluster. This will increase accuracy of predictions.

To turn my model into an API that makes it easy for other languages to use, I could establish a connection to the software using the plumber package in R. The package will turn existing R code as a webservice via an API; i.e HTTP APIs. A simple outline of what I need to do is define the output, add additional comments to customize input using #, output and other functionalities of your API. This way my R code can be called from other software, even if it is not written in R.

Conclusion

This model is expected to have a great impact on improving the company's application, will also improve the drivers' satisfaction and pay as it will increase customer's chance of tipping the driver. Thus, the company can increase its ability to retain employees, which will help mitigate one of Uber's biggest problems, low employee retention rate. Data science and machine learning can be used to take any industry to a whole new level, bettering its operations in every aspect. The Transportation industry was revolutionized when Uber decided to utilize a "blue ocean" methodology using data. The impact my field is doing and how we can make data have value and turn it into an asset instead of business "debris", is what triggered my passion.

Data cannot speak for itself, that's why I believe in the power of storytelling with data. This was my version of the NYC Yellow Taxi data story, and I hope to keep developing my storytelling and data analysis skills to better reach a wide spectrum of audience and help turn ideas into reality and in bettering the decision-making process.