Taxi Trip Time Prediction

NYMISHA BANDI

Hypothesis

To improve the efficiency of taxi electronic dispatch system by predicting the trip time and demand.

Data-Preprocessing

The original dataset consists of 12 files corresponding to taxi rides for 12 months in Chicago. For this analysis we are considering only January data which has 1705805 observations. The following steps were followed to preprocess the data:

- Impute missing values corresponding to pickup_community_area and dropoff_community_area
- Remove observations with trip distance = 0
- Remove outliers from Trip time and Trip distance measures: We considered any observation outside mean+1.5*SD as the range for removing outliers.
- Adding a new column called Speed
- Remove any observations with unusual speed, i.e., Speed>60 mph
- Add a column called Weekday which indicates if the trip happened on a weekday or not
- Hour of the day is obtained from pickup time and is used to sort time into 5 bins based on the avg trip time. This is saved in TimeStamp.
- The following columns are removed after obtaining the above derived columns:
    - pickup_census_tract, dropoff_census_tract, trip_end_timestamp, trip_start_timestamp, company, fare, tips, tolls, extras, trip_total, payment_type, pickup_latitude, pickup_longitude, dropoff_latitude, dropoff_longitude, time_hrs
- Convert all columns to numeric.

Refer to the appendix for relevant graphs for outlier removal and splitting time into bins.

Since the number of rows after the pre-processing is 1035066 and it is difficult to handle this data for computation on my laptop, I chose to analyse data for weekends and for the trips which started between 5PM and 9PM only.

Feature Engineering

Below is a list of feature templates we use to extract features from each data point:

- Hour of day $\in$ [0, 23]. We expect overall NYC taxi ridership to follow a daily cycle
- Day of week $\in$ [0, 6]. We expect day of week to correlate with taxi traffic.
- Speed of the taxi. This helps us filter out the data based on unusual speeds.

Exploratory analysis

1. The average trip time is observed to be quite constant throughout the day. The time slot corresponding to morning has a slightly higher trip time average. Refer to figure 8 in the appendix.
2. The distribution of trip time can be observed in the below figure. It can be inferred that most of the trips have a trip time of 300-450secs.

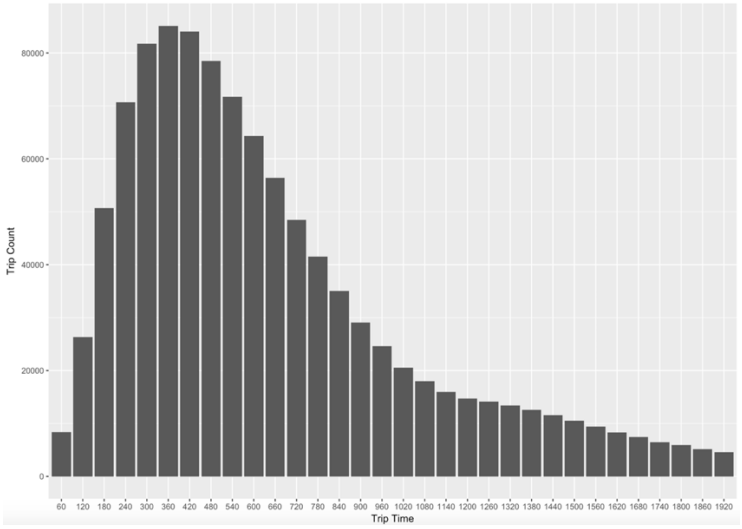


Figure 1: Trip time distribution

3. The heat map of the demand in Chicago is as seen below. This can help in further narrowing our analysis. But in this paper we will not consider any filtering by location.

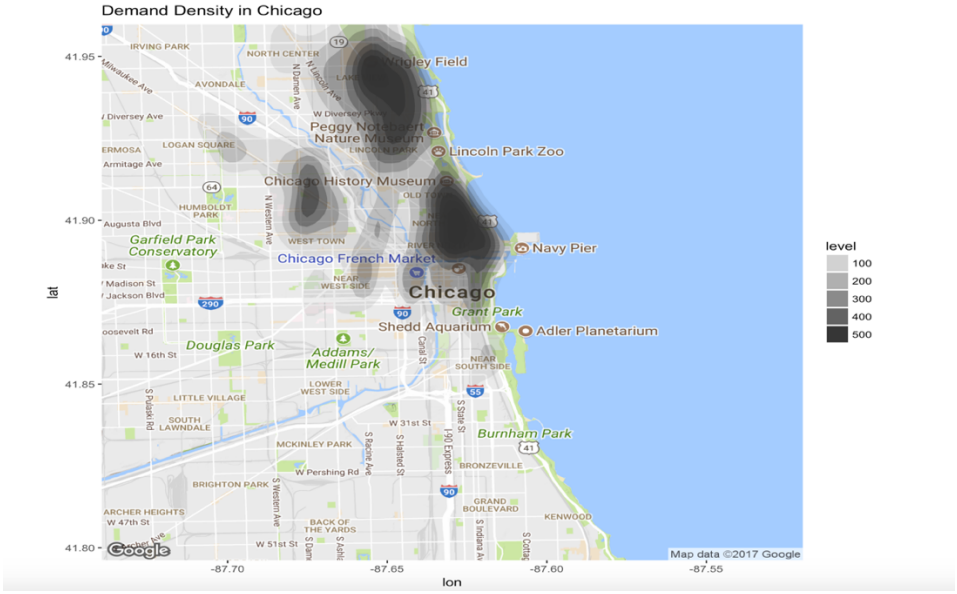


Figure 2: Demand distribution

Modeling

We will first study the correlation between our target variable “trip_seconds” and the other variables. It is evident that the correlation is maximum with trip_miles followed by pickup_community_area, Speed and dropoff_community_area. This can be seen in the figure 10 in the appendix.

Baseline Model

Baseline model means that we assume there are no predictors (i.e., independent variables). Thus, we have to make an educated guess (not a random one), based on the value of the dependent value alone. This model is important because it will allow us to determine how good, or how bad, are the other ones. In the absence of any other predictors, our best guess would be the mean of the trip times observed till date. The RMSE obtained using this as our model is 409.67 secs and RMSLE is 0.69.

Feature selection

The dataset is split into training and test with split ratio of 60%. Forward selection method is used to select the variables which give the best model. This lead to the selection of trip_miles, Speed, dropoff_community_area as the important variables. It is surprising that pickup_community_area is not an important variable.

Linear Regression

Linear regression is one of the simplest models and from the correlation plot we can see that there is some amount of linear relationship between speed and trip time. The number of data points are so huge that finding any relationship becomes extremely difficult. Also, Linear regression helps in understanding the statistical significance of the predictors and is highly interpretable. To avoid a sub-optimal model, results of the forward selection are used to select the subset of variables which would be the best.

Let us now check for multicollinearity using the VIF function. In our case, the VIF values are all less than 5 indicating moderate collinearity. Predicting the trip time for the test dataset using the above linear model, we get a RMSLE of 0.57. Also, the RMSE is 209.18 secs. All predictions with negative or zero trip time are considered as zero. This is done because a negative trip is doesn’t make sense.

Ridge and LASSO

To further optimize our coefficients, let’s try Ridge and LASSO. Ridge Regression is a remedial measure taken to alleviate multicollinearity amongst regression predictor variables in a model. In our model, we found that there is a moderate collinearity. When they are, the regression coefficient of any one variable depend on which other predictor variables are included in the model, and which ones are left out. (So the predictor variable does not reflect any inherent effect of that particular predictor on the response variable, but only a marginal or partial effect, given whatever other correlated predictor variables are included in the model). Ridge regression adds a small bias factor to the variables in order to alleviate this problem. Hence, we implement Ridge regression for

multiple lambda values varying between 10^{-5} to $10^{2.5}$. We use 10-fold cross validation to pick the best lambda which gives the least mean square error.

The RMSLE is 0.57 and the RMSE is 209.19 secs. This indicates that our feature selection with linear regression was as good as implementing a Ridge regression model.

The only difference of LASSO from Ridge regression is that the regularization term is in absolute value. But this difference has a huge impact on the model. Lasso method overcomes the disadvantage of Ridge regression by not only punishing high values of the model coefficients but actually setting them to zero if they are not relevant. Therefore, you might end up with fewer features included in the model than you started with, which is a huge advantage.

We implement Ridge and Lasso for multiple Lambda values. We go about the modelling in a similar fashion as in Ridge except that the alpha is set to 1 in case of LASSO. After cross validation, the best lambda obtained is used for predicting the trip time. The RMSLE obtained with LASSO is 0.56, which is slightly better than Ridge. The RMSE observed is 208.3 secs.

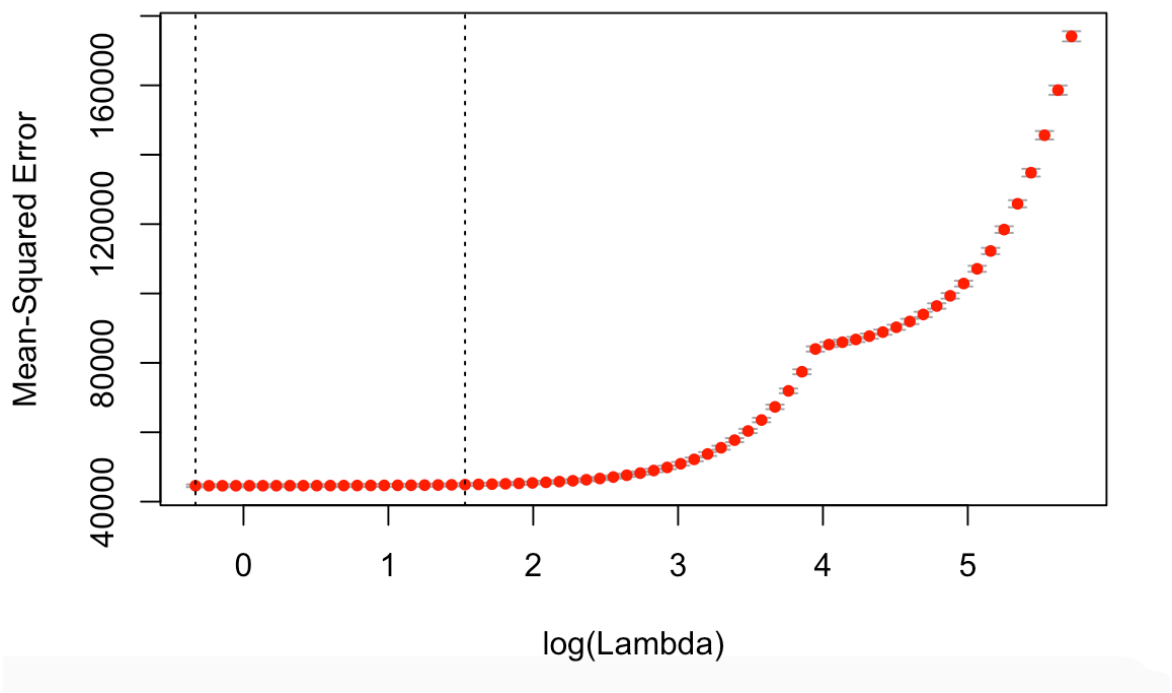


Figure 3: Cross validation output to pick the best lambda

Random Forest

As trip time is clearly not varying solely based on the magnitude of the coordinates and speed of the vehicle, the linear model fails to account for the nonlinear effects of the variables. A model which can capture such nonlinearity is Random forest.

We model a random forest with 100 trees and only the variables obtained from our forward feature selection. Number of variables randomly sampled as candidates at each split is set to 1. The default value is p/3 where p is the number of variables, which is 6 in our case. The RMSE obtained is 64.40 secs and the RMSLE is 0.16.

The importance obtained from random forest reinstates the variable selection which we obtained as a result of the forward selection. The %incMSE shows the highest for trip_miles which implies that any random permutation in this measure changes the output the most. Same is the same with IncNodePurity, which indicates the GINI index.

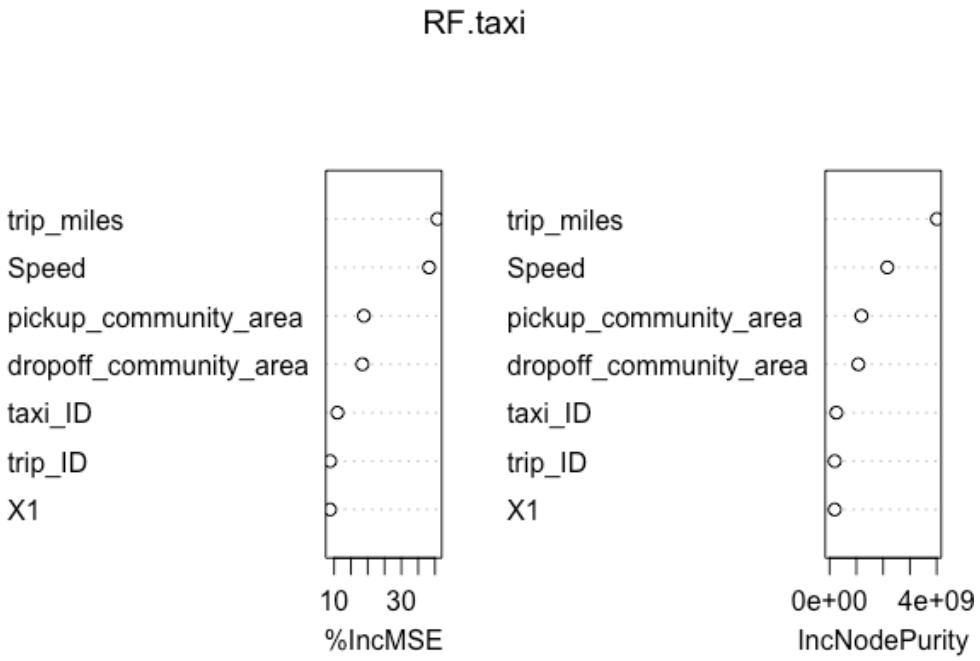


Figure 3: Variable importance from random forest

Boosting

Boosting is another algorithm which is primarily used for reducing bias, and also variance. It is mainly used when we have a set of weak learners to build a single strong learner. The problem with this is that the model obtained is not interpretable. We build a boosted model with the help of 100 trees, with Gaussian distribution and maximum depth of variable interaction set to 3. This implies that there can be a 3-way interaction possible between the variables. The model is built for an array of lambdas and the best lambda is selected which is gives the least error. Refer to figure 12 for the plot. The least RMSE obtained is 33.98 secs and RMSLE is 0.15

Inference and future work

Model	RMSE (secs)	RMSLE
Baseline	409.67	0.69
Linear	209.18	0.57
Ridge	209.19	0.57
LASSO	208.3	0.56
Random Forest	64.4	0.16
Boosting	33.98	0.15

Figure 4: Errors obtained from all the models

The above table summarizes our findings. The best model obtained is from Boosting as we can see from the table in figure 4. Using this model we can predict trip time for any of the taxi providing services so that they can better manage their pickup and drop services. This in turn will increase their profits.

There is still scope for further improvement of the models and we can use few more non linear models for the same. We can use ensemble methods to further improve the performance. Ensemble modeling is the process of running two or more related but different analytical models and then synthesizing the results into a single score or spread in order to improve the accuracy of predictive analytics and data mining applications. This can be done by combining 2 strong models which in our case are Random forest and boosting models.

We may be able to achieve good results using a neural network regression, since neural networks can automatically tune and model feature interactions. Instead of manually determining which features to combine in order to capture feature interactions, we could let the learning algorithm perform this task.

Appendix

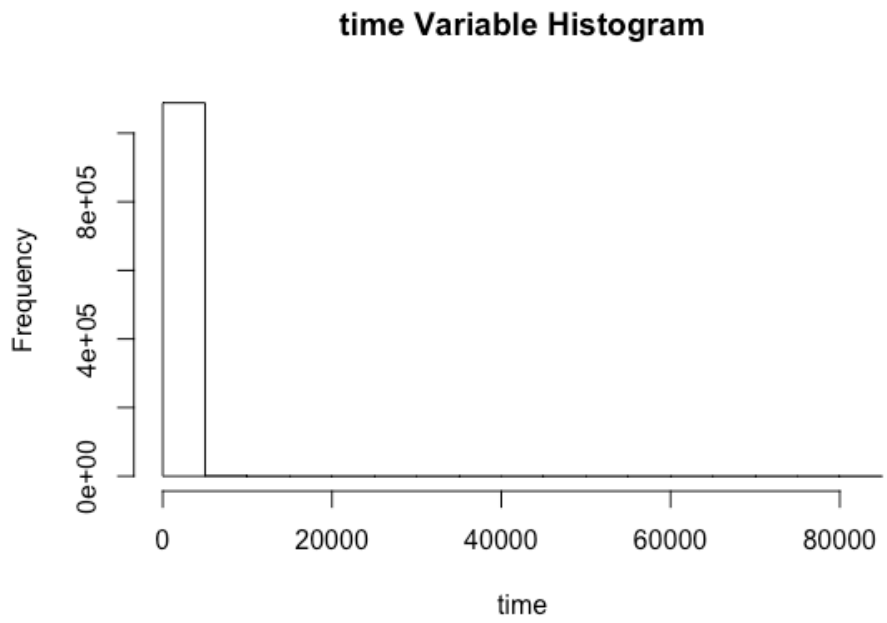


Figure 1: The histogram plot of time_seconds before outlier treatment

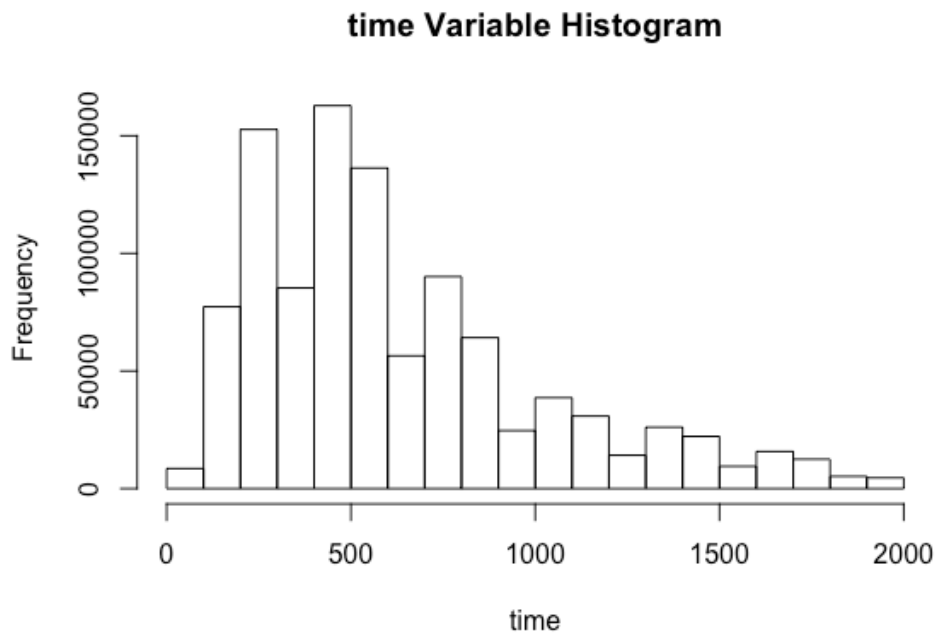


Figure 2: The histogram plot of time_seconds after outlier treatment

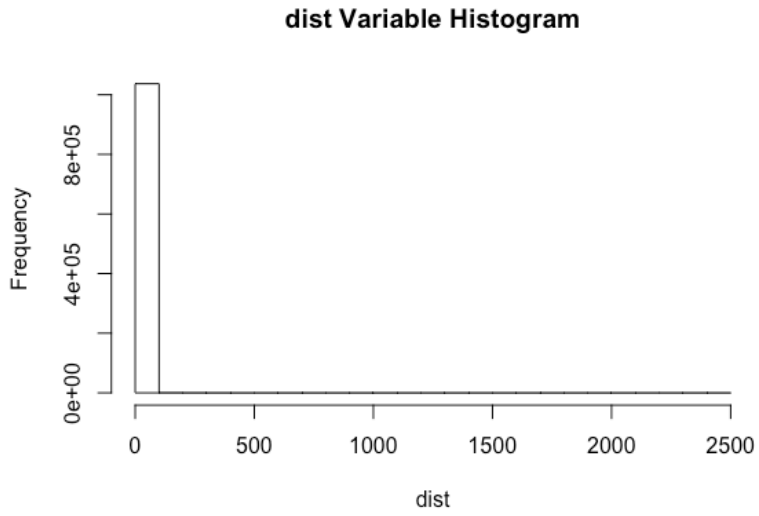


Figure 3: The histogram plot of time_miles before outlier treatment

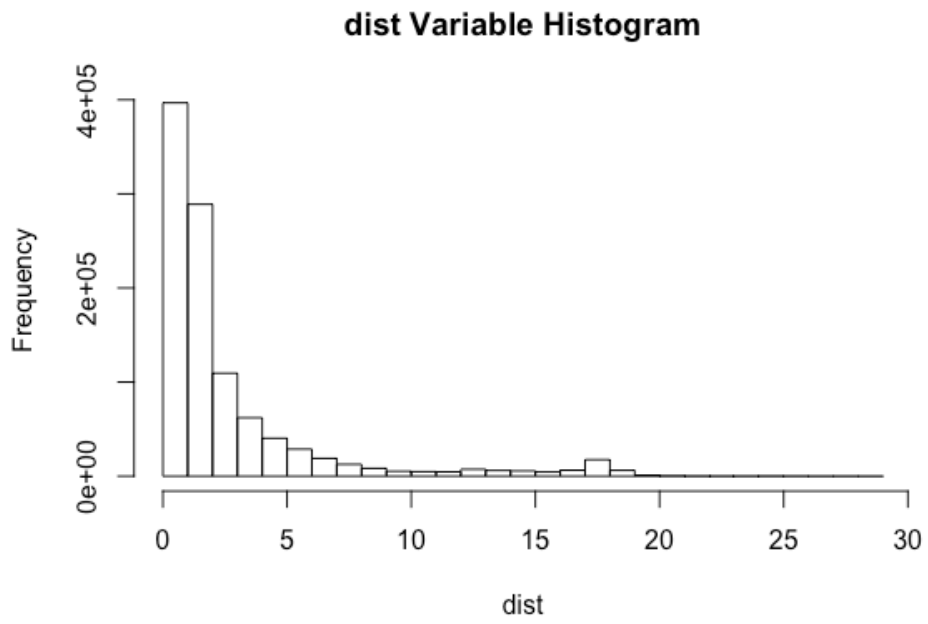


Figure 2: The histogram plot of time_miles after outlier treatment

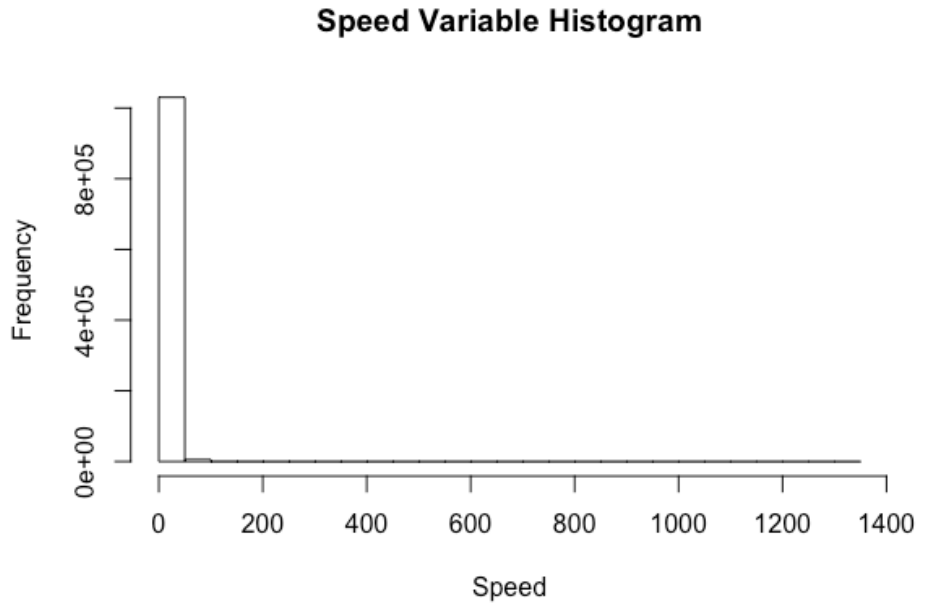


Figure 5: The histogram plot of Speed before outlier treatment

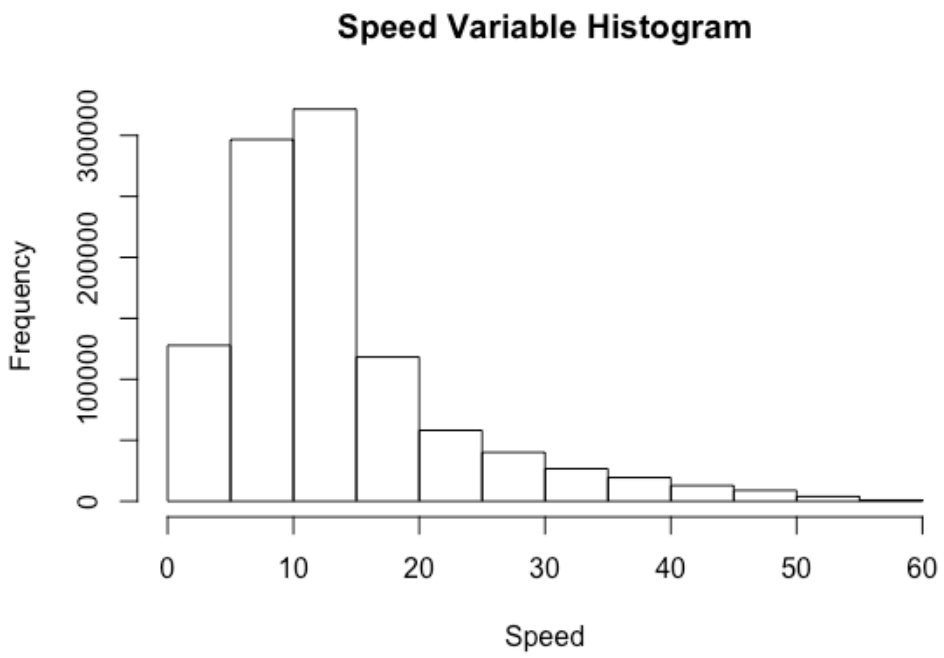


Figure 6: The histogram plot of Speed after outlier treatment

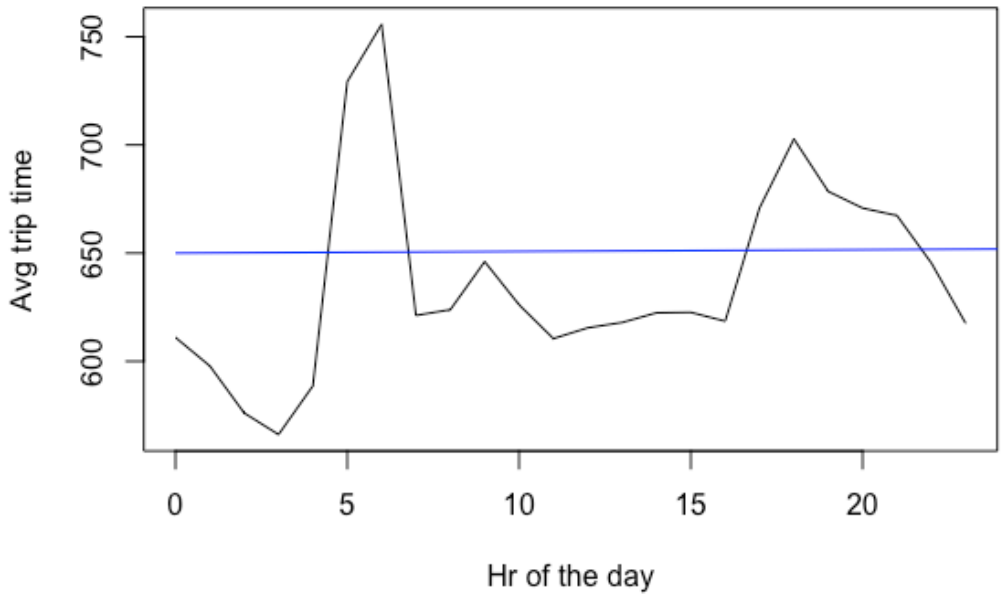


Figure 7: Avg trip time used to separate the hours of the day into bins

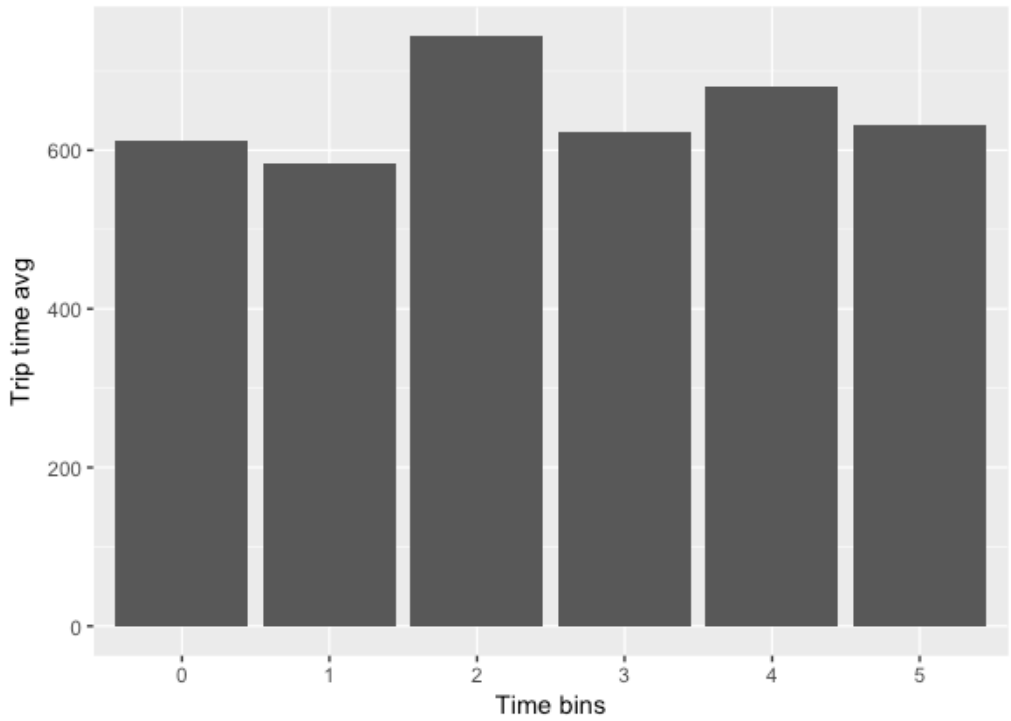


Figure 8: Exploratory analysis of average trip time with respect to day of the week

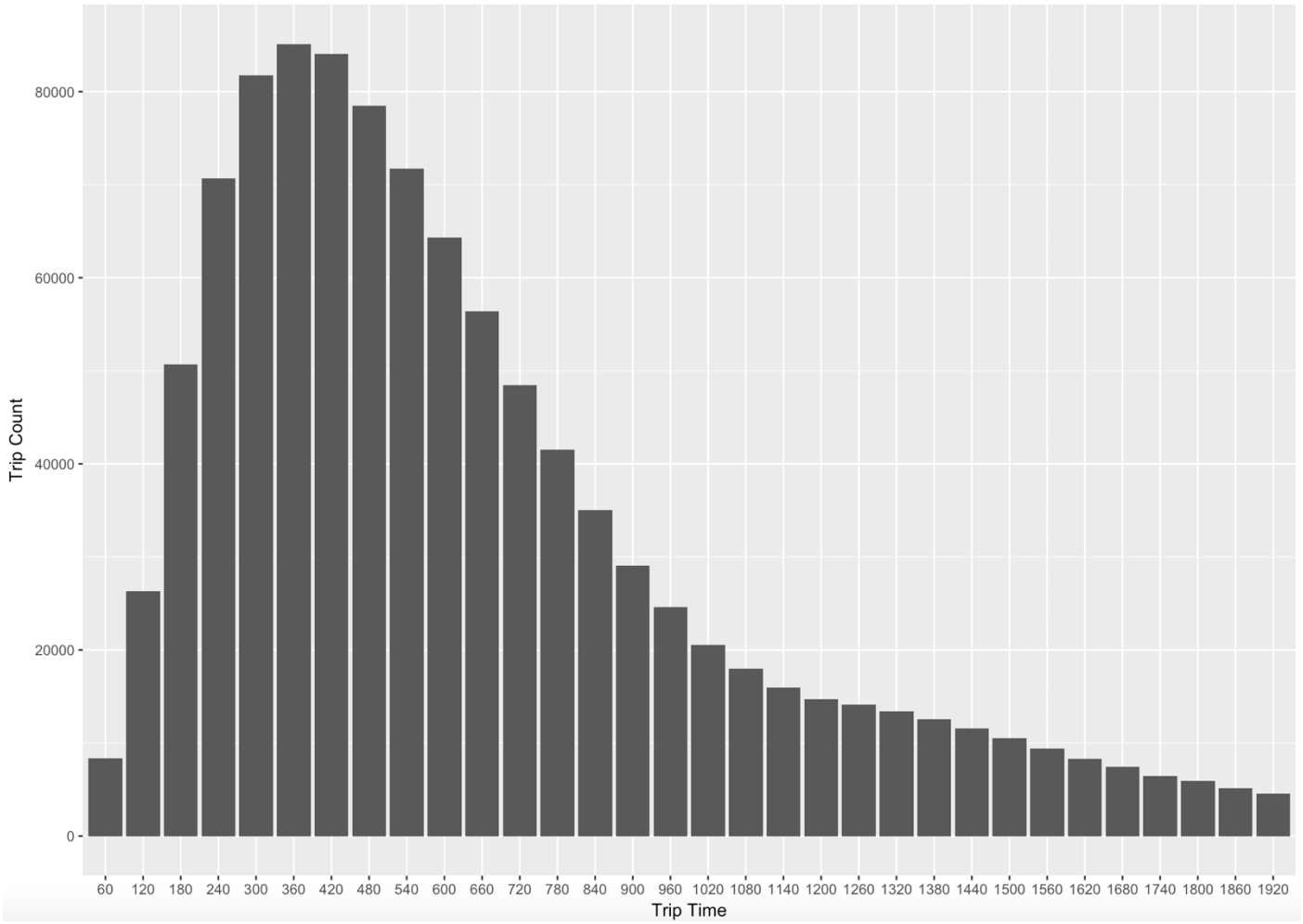


Figure 9: Distribution of ride duration

```
> cor(Taxi_New_Week)
                          taxi_id trip_seconds trip_miles pickup_community_area dropoff_community_area
taxi_id                1.000000e+00  0.009397483 0.01455055            0.00962234           5.607323e-05
trip_seconds           9.397483e-03  1.000000000 0.72218651            0.44271665           1.128999e-01
trip_miles             1.455055e-02  0.722186512 1.00000000            0.56142429           1.248376e-01
pickup_community_area  9.622340e-03  0.442716652 0.56142429            1.00000000           1.560859e-01
dropoff_community_area 5.607323e-05  0.112899897 0.12483756            0.15608591           1.000000e+00
Speed                  1.086467e-02  0.382337731 0.85621530            0.44204344           1.221044e-01
                              Speed
taxi_id                0.01086467
trip_seconds           0.38233773
trip_miles             0.85621530
pickup_community_area  0.44204344
dropoff_community_area 0.12210444
Speed                  1.00000000
```

Figure 10: Correlation table for the data after processing

Figure 11: Errors for all the models built

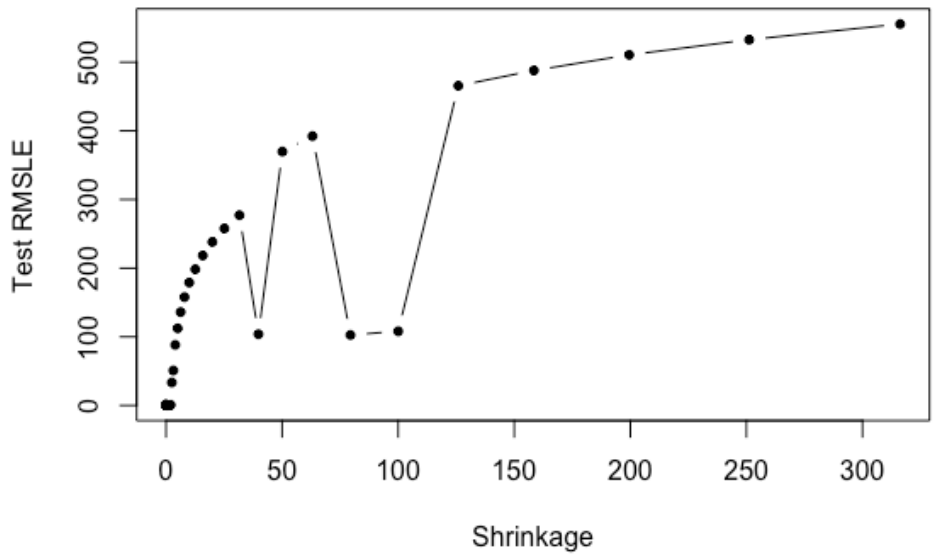
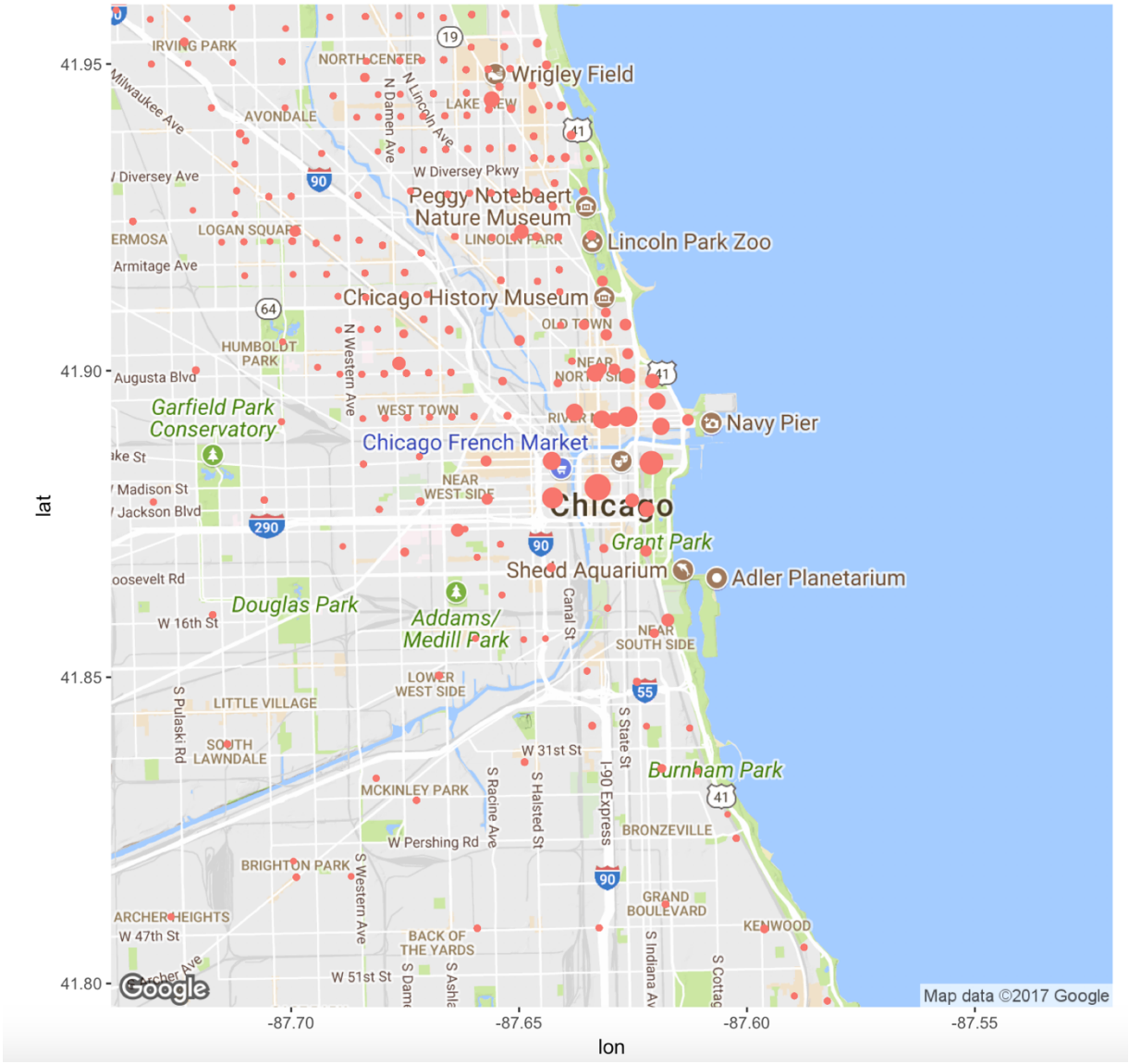


Figure 12: Test errors for different lambda in Boosting

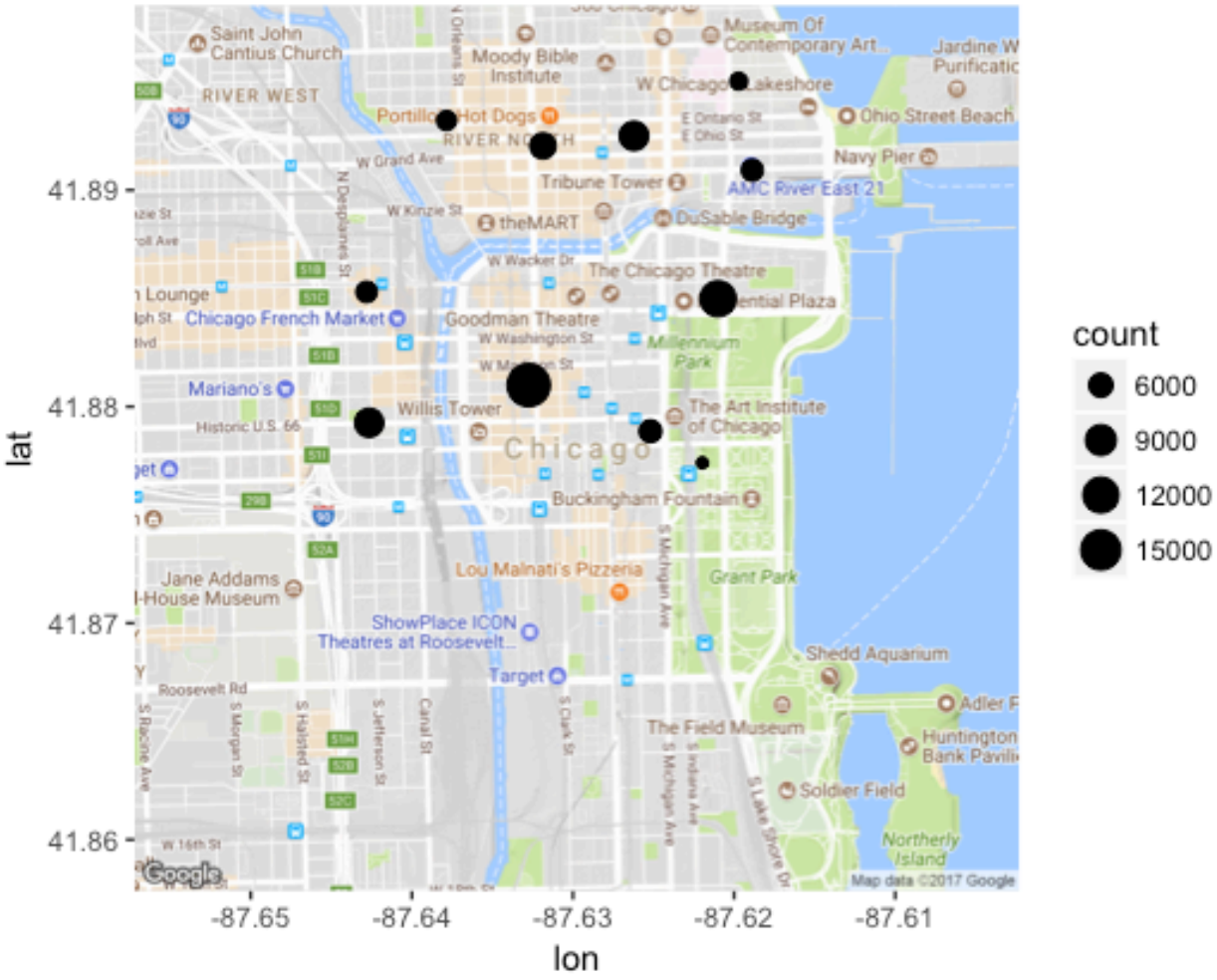
References

- https://www.kaggle.com/chicago/chicago-taxi-rides-2016
- http://www.transitchicago.com/maps/
- https://dspace.mit.edu/bitstream/handle/1721.1/99565/924315586-MIT.pdf?sequence=1
- https://arxiv.org/pdf/1706.06279.pdf
- http://cs229.stanford.edu/proj2016/report/AntoniadesFadaviFobaAmonJuniorNewYorkCityCabPricing-report.pdf

Demand by location for the month of Jan



Top 30 high demand areas


lat
41.89
41.88
41.87
41.86
lon
-87.65
-87.64
-87.63
-87.62
-87.61
count
6000
9000
12000
15000
Saint John Cantius Church
RIVER WEST
Moody Bible Institute
Museum Of Contemporary Art...
Jardine W Purificatic
W Chicago Lakeshore
Portillo Hot Dogs
Ohio Street Beach
Navy Pier
Tribune Tower
AMC River East 21
theMART
DuSable Bridge
The Chicago Theatre
Chicago French Market
Goodman Theatre
Millennium Park
Mariano's
Willis Tower
The Art Institute of Chicago
Chicago
Buckingham Fountain
Jane Addams House Museum
Lou Malnati's Pizzeria
Grant Park
ShowPlace ICON Theatres at Roosevelt...
Target
Shedd Aquarium
Roosevelt Rd
The Field Museum
Soldier Field
Northerly Island
Google
Map data ©2017 Google