Citi Bikes Analysis and Bike Demand Prediction

Julian DeGroot-Lutzner & Vikramaditya Salwan

12/14/2017

Background and Motivation

Our project is inspired by the Kaggle bike sharing demand competition. Bike sharing is a service that has become popular within the last few years in many cities across the United States. Bikes are checked in and out of a network of stations that keep track of each ride, its start and end time, its start/end location, as well as information about subscribers such as gender and age. The Kaggle competition challenged users to predict the hourly bike usage of a test dataset by building a model on a training set combing hourly weather data with hourly bike share usage in Washington, DC.

We planned to first visualize and understand the data and then try to find if there are any inconsistencies in bike stations traffic. Is there a bike station that people take bikes from but do not take bikes back to? If there was an bike station that is a net exporter - meaning it has an excess in demand of bikes - we wanted to make recomendations on how often CitiBikes should manually transport bikes back to the station.

Instead of using the Kaggle data, we wanted to wrangle bike sharing data from a different city's bike share dataset so that we could get familiar with the process of preparing a dataset for analysis. We decided on the Citibikes bike sharing dataset from New York City because there is data available on every ride from January 1st, 2013 to September 31, 2017. However, the dataset is very large, over 40 million observations, and the complete set cannot be ran in R. Our new goal became learning how to work with Big Data in R.

Working with Big Data in R

Big Data in R is when the data cannot fit in to memory. Instead, we stored the data in a SQL database and received the data using 'RMySQL' and 'dplyr.' We based our work process off of the Working with Big Data in R webinar.. The life cycle of a big data analysis project usually involves five parts. Subset (extract data to explore and work with), clarify (become familiar with the data and template a solution), develop (create a working model), productize (automate and integrate), and publish. In other words, first work with a smaller sample, then scale up the work to a larger dataset.

We worked on Dr. Jo Hardin's sever at the Pomona math department using MySQL. Since the server is within the Pomona network we needed to navigate both the Pomona ITS and Department firewall. Additonally, our limited permissions on our personal accounts made it difficult to update needed packages within R. These limitations forced us to learn, adapt, and change our project as needed.

The first difficulty in the project was uploading the Citibikes data to a SQL database. Citibikes publishes a zip file for each month of data so there were over 50 seperate files to download. We wrote scripts to download, alter, and write csv files from the the Citibikes server, to our server, and then onto the MySQL database. The difficulties served as a learning experience as we developed better skills in the command line as well as in writing Shell, Python, and SQL code. You can see some of this code in the setup-code folder. Unfortunately, as of now our complete process is not reproducible by a single code but we can come back and add more explanation on how to use this code later.

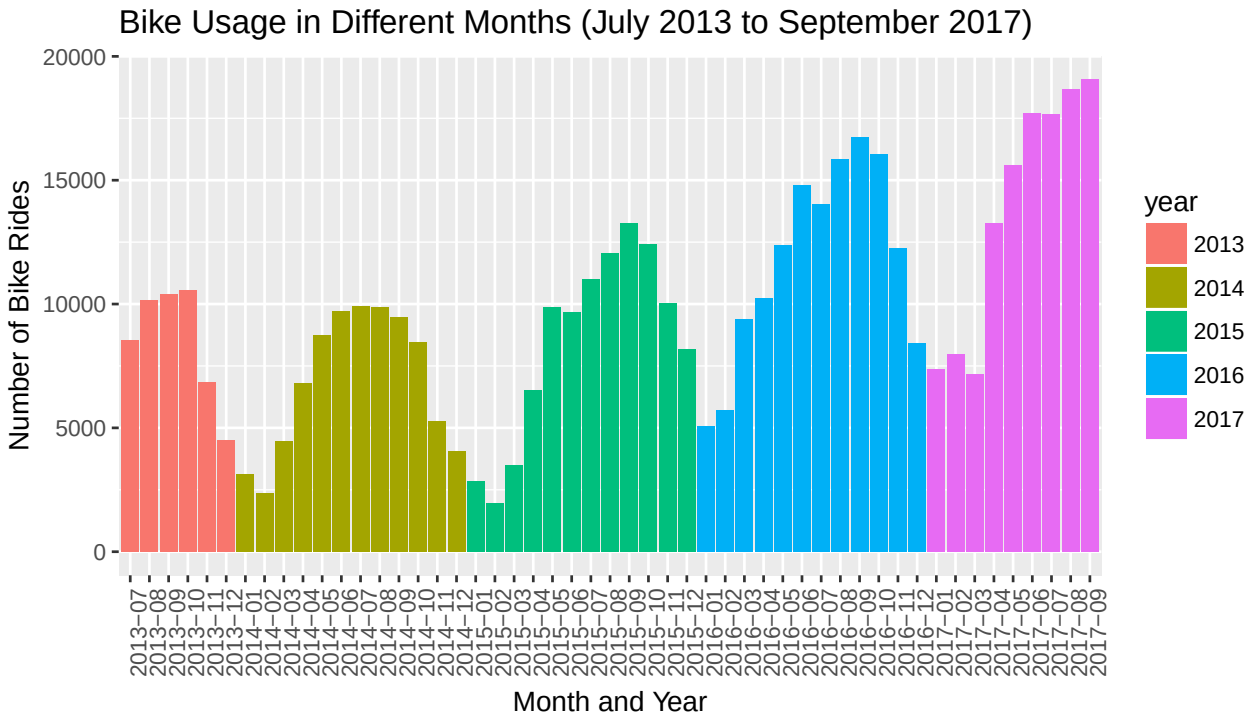
Dplyr allows integration with many different databases. We originally tried running code on the complete >40 million dataset. However, the SQL queries took too long for us to create a reasonable project within our relatively short time frame. For example, some queries would take more than a day. In the future we could use other big data techniques like MapReduce to work more quickly. Instead we used the aforementioned big data work schedule of subsetting and scaling. First we took a random sample of 500,000 observations - approxiametly 1 % - from the original dataset.

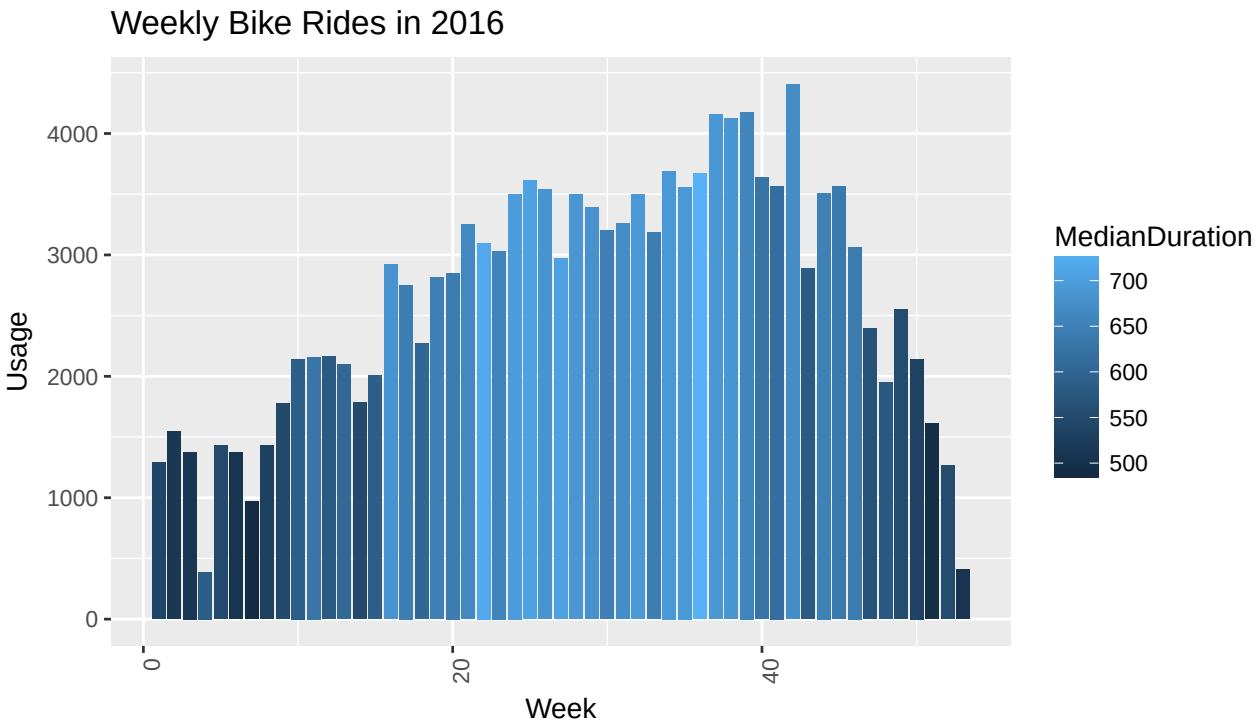
Understanding the Random Sample

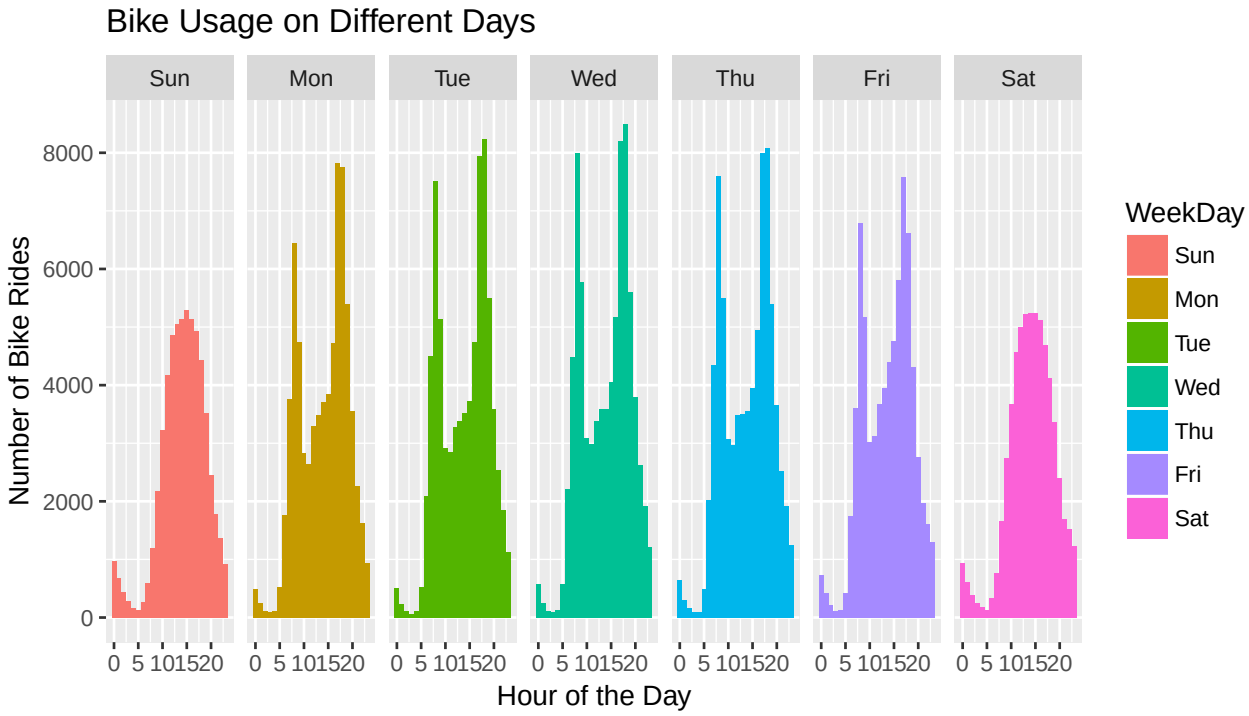
Exploratory Graphs on Ride Use

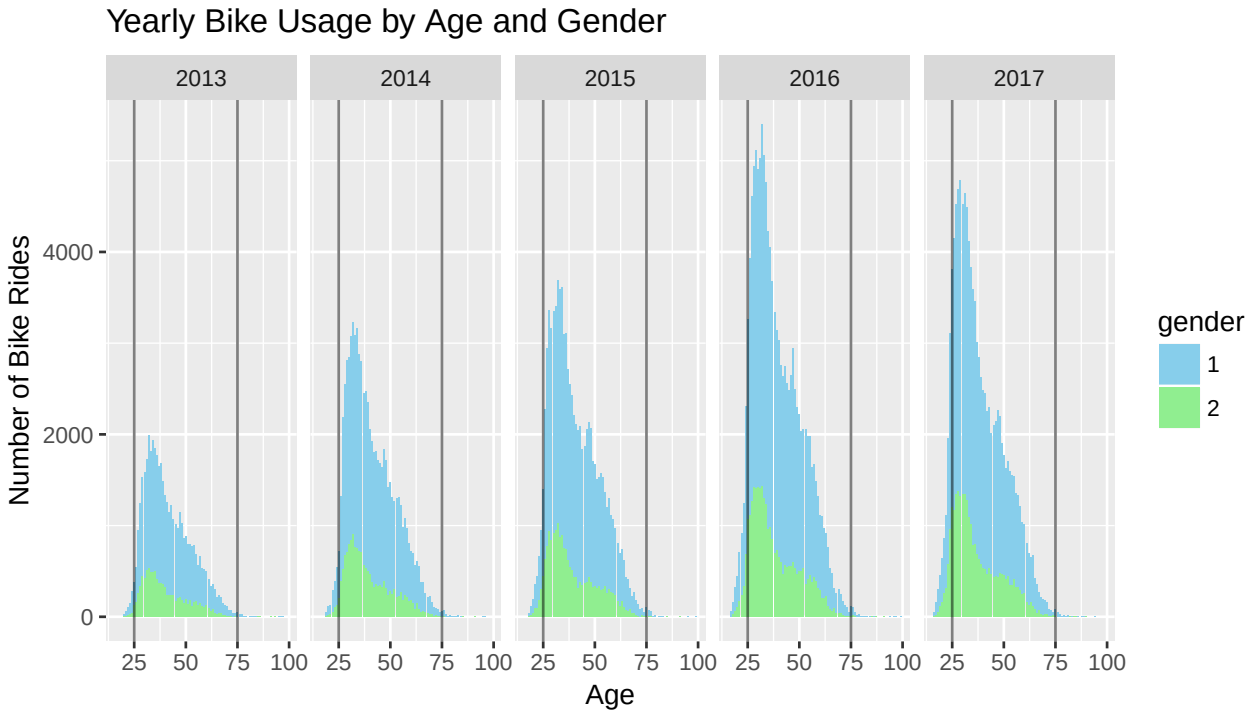
We wrote a lot of code, but we decided to use as little code as possible in our write up so that it is easier to read. If you're interested in learning how to make cool graphs like these please download our Rmd file and take a look for yourself.

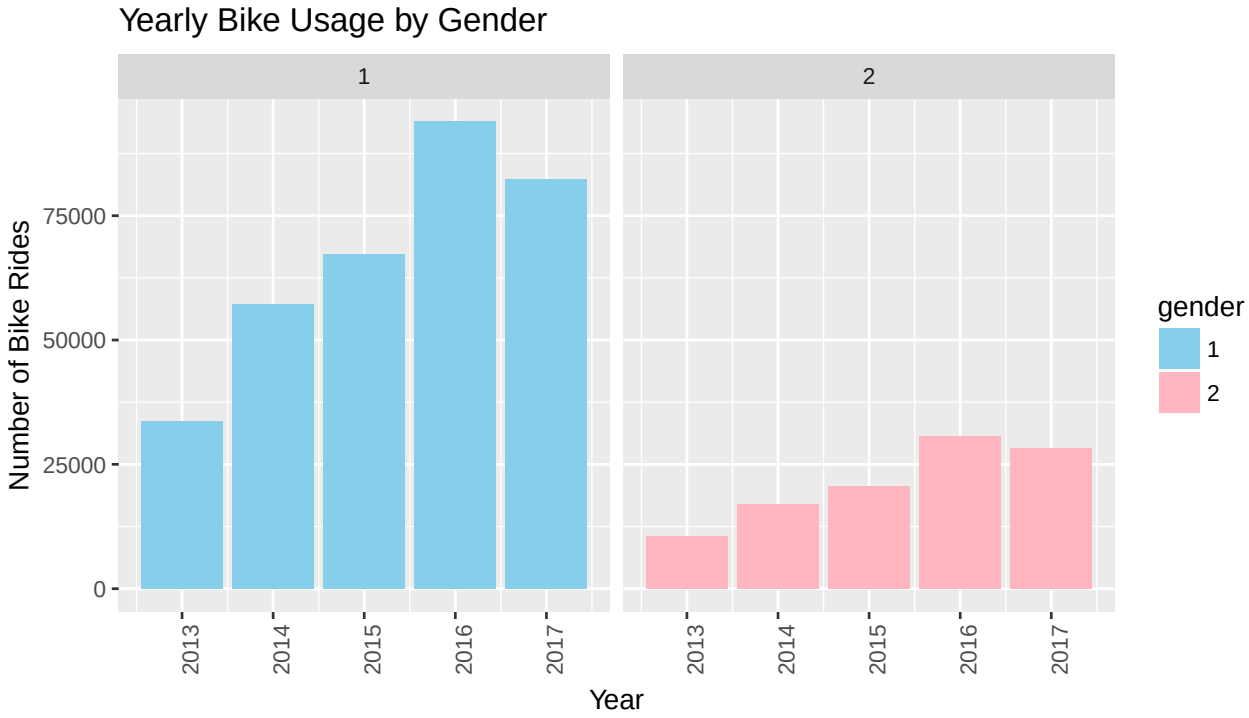
Here are some graphs that should help you get an understanding of CitiBike usage. After the graphs is a write up about some of the takeaways we found.




Weekly Bike Rides in 2016
Usage
4000
3000
2000
1000
0
0
20
40
Week
MedianDuration
700
650
600
550
500


Bike Usage on Different Days
Sun
Mon
Tue
Wed
Thu
Fri
Sat
Number of Bike Rides
8000
6000
4000
2000
0
0 5 101520
Hour of the Day
WeekDay
Sun
Mon
Tue
Wed
Thu
Fri
Sat





Based on a preliminary exploratory analysis the following insights were found which could be useful for Citi Bikes:

1) In the graph above blue represents males and pink represents females. It was found that bike usage for both females and males has increased each year since the initiation of this bike sharing project in May 2013 and since the data became available in July 2013. After the missing observations for gender were removed, it was found that bikes are used predominantly by males. We believe this trend is unexpected and needs to be corrected. We posit that female prefer walking, which is why the bike usage for females is not very high as compared to bike usage for males. The 2017 data is only available till September,

which is why when compared to 2016 it shows a decline.

2) It was found that bike usage has increased the most for the 25-50 years age group. The bike usage for the age group of 50-75 has increased significantly too, while the usage for the age group less than 25 increased too. The bike usage for the age group of 75 years or older has not increased much. Based on this information, we believe that the target market for Citi Bikes is 25 to 50 year olds.

3) In the year 2016 bike usage peaked from week 21 to week 44. This corresponds to the dates from May 16 to October 31. This is because bike usage is generally higher during the summer and fall months. It was also found that the median bike ride is longer during these months as the weather does not act as an impediment for bikers. The highest median bike ride was in week 36 (12 minutes and 9 seconds), while the lowest was in week 7 (8 minutes and 8 seconds). Hence weather does play a part in biker's decision whether to bike or not. However, it remains to be seen which of the six weather predictors play the most important role in a user's decision to bike or not.

4) Based on the first graph, it can be seen that the overall popularity of this bike sharing system has surged significantly each year. It has increased each year since its inception. The number of bike rides continue to grow, but the number of bike rides in the last four months in 2013 are higher than the last four months in 2014, albeit not by a big amount. This may have been because of the weather conditions. The average temperature in September 2013 was 67.2 degree Farenheit, 2 degrees higher than the average temperature in September 2014, which could have translated in the marginally lower number of bike rides in in September 2014 than in September 2013. Similarly, the average temperature in December 2013 was 2 degrees lower than the average temperature in December 2014. We hypothesize that weather conditions could be the reason for this slight decrease in bike rentals. However, it is still not quite clear about why the number of bike rides rented in October and November 2014 were marginally lower than in October and November 2013.

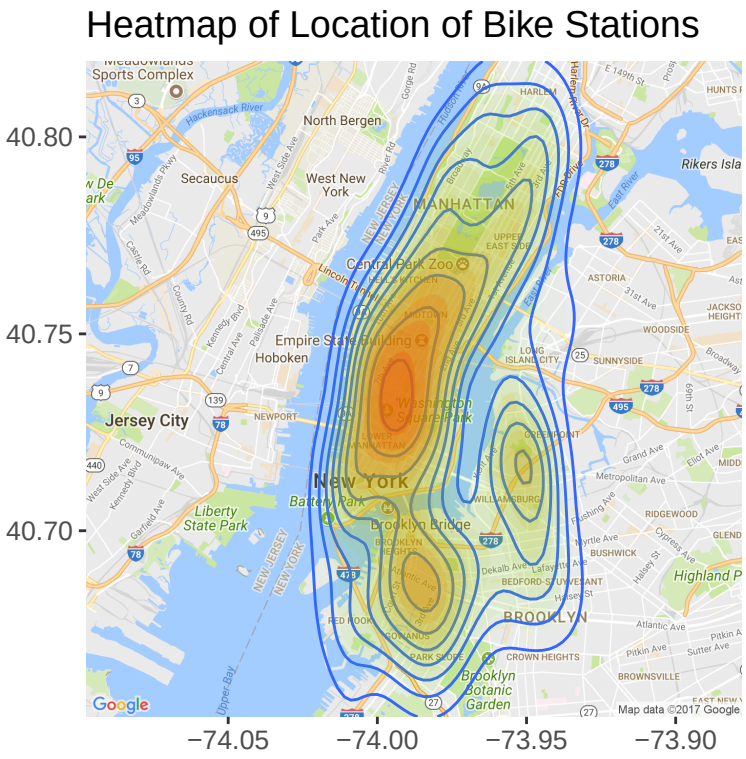
5) The bike usage is significantly higher during the weekdays than during weekends. This could be because the bikes are predominantly used by commuters commuting to work on weekdays. This drop is the highest during the commuting hours from 7am to 9am and from 4pm to 7pm, which makes us believe that commuters account for a majority of the bike rides in New York City. Based on this information, Citi Bikes could shift its major focus to commuters instead of any other group. This could be undertaking or promoting campaigns to bike to work, which would increase their annual memberships.

It is important to note that the exploratory analysis is based on a sample of 500,000 observations, which is 1% of the total population. This sample is representative of the population because of its large size.

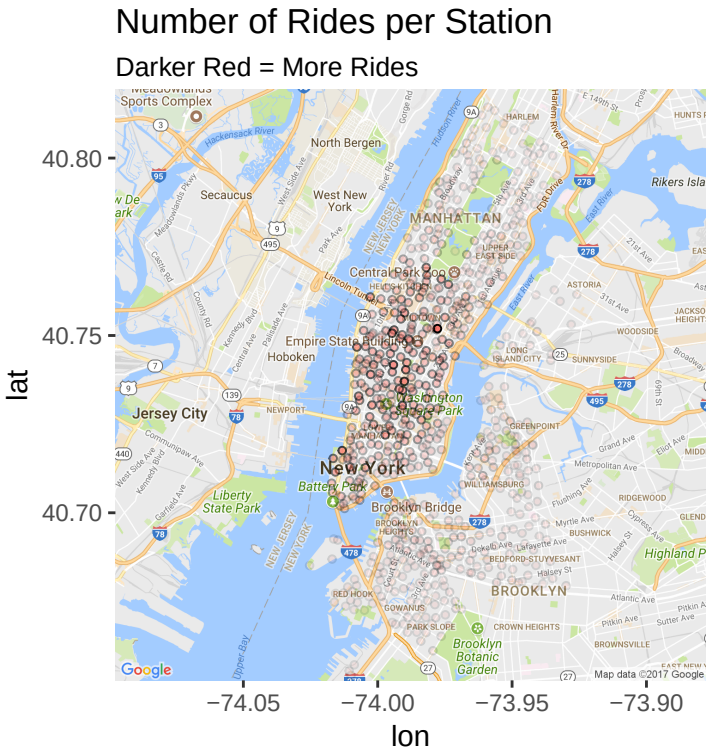
Station Analysis

After getting a better sense of the data, we will return to one of the orignal questions we had. Are there any bike stations that have inconsistencies between bike supply and bike demand? This part of the exploratory analysis moves to the stations.

First lets see where the stations are located in New York City.

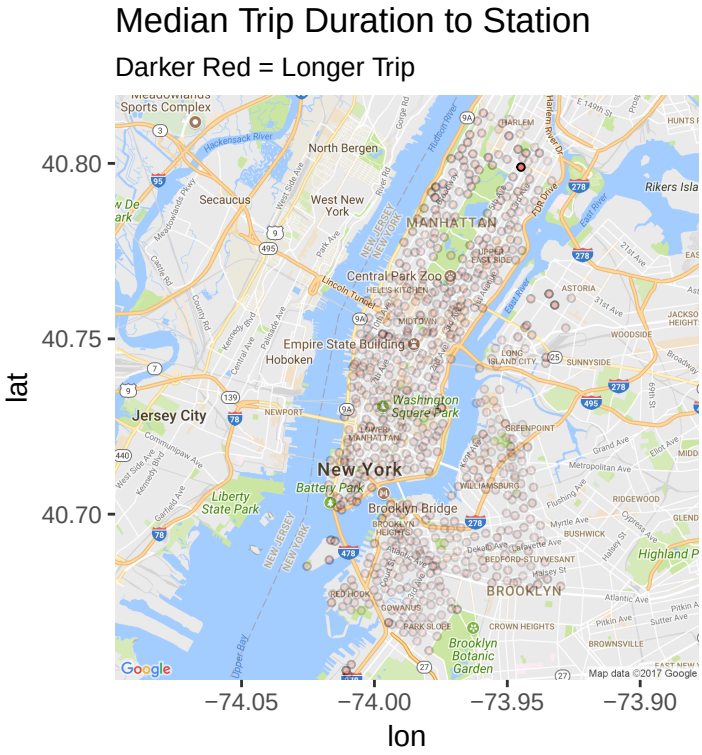


As you can see most bike stations are centered in lower Manhatten. Now lets get a different but related view of Citibikes as shown by the number of rides per station.



As we can see most bike rides happen in downtown Manhatten. The rides really slim out over the east and west side of Central park. We can see a slight increase of red in Brooklyn on the opposite side of the Brooklyn Bridge. We must keep in mind that we did not account for the number of years that the station was around.

Now let's take a look at the median trip duration.



The one red dot could be an outlier and is worth investigating more but we chose to go in a different direction.

At this point we moved in to our investigation of whether a station was a net importer or exporter. To figure this out we calculated the sums of the number of trips in and out of a station for each station. We then found the difference between the start and the finish or the net change. The difference (start - finish) is positive if more bikes start at a station than end there, and is negative if more bikes end at a station than start there. *I confused myself and probably should have rethought about the change as in and out and then I would have gotten finish - start or in - out. Anyways, please deal with this confusion*

We normalized the difference by dividing it by the total station trips and we filtered the stations to only look at the top ten percent of the most popular stations (the stations with the most bike rides). Looking back at our metric, we could have adjusted for how long the bike station has been around and looked at how the usage has changed overttime. For example, did extreme popularity of one station spark a new station near it that helped provide more supply of bikes? That's a question we should look at in the future.

Here is a list of the top exporters - more bikes start from these stations than end at these stations. They need more bikes.

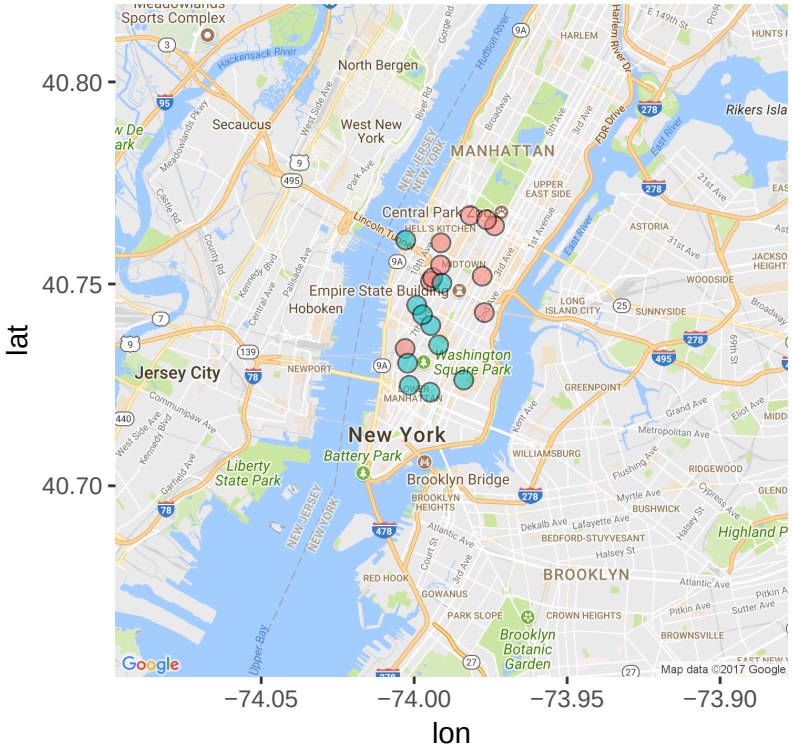
```
## # A tibble: 10 x 5
##       id start.count end.count difference normalized.difference
##    <int>       <int>     <int>      <int>                 <dbl>
##  1   521        2649      2380        269               0.05349
##  2   519        5367      4835        532               0.05215
##  3   281        2493      2279        214               0.04484
##  4   490        3264      3064        200               0.03161
##  5   479        2026      1904        122               0.03104
##  6   457        2287      2160        127               0.02856
##  7   523        2631      2496        135               0.02633
##  8  2006        2706      2574        132               0.02500
##  9   380        2199      2105         94               0.02184
```

```
## 10   528        1889      1811         78               0.02108
```

Here is a list of the top importers - more bikes **end** at these stations than start in these stations. They have excess bikes.

```
## # A tibble: 10 x 5
##       id start.count end.count difference normalized.difference
##    <int>       <int>     <int>      <int>                 <dbl>
##  1   432        2114      2330       -216               -0.04860
##  2   492        2384      2602       -218               -0.04372
##  3   382        2728      2966       -238               -0.04180
##  4   514        2632      2849       -217               -0.03959
##  5   348        1768      1910       -142               -0.03861
##  6   453        1847      1971       -124               -0.03248
##  7   251        1973      2105       -132               -0.03237
##  8   168        2558      2724       -166               -0.03143
##  9   368        2921      3110       -189               -0.03134
## 10   334        1886      1995       -109               -0.02809
```

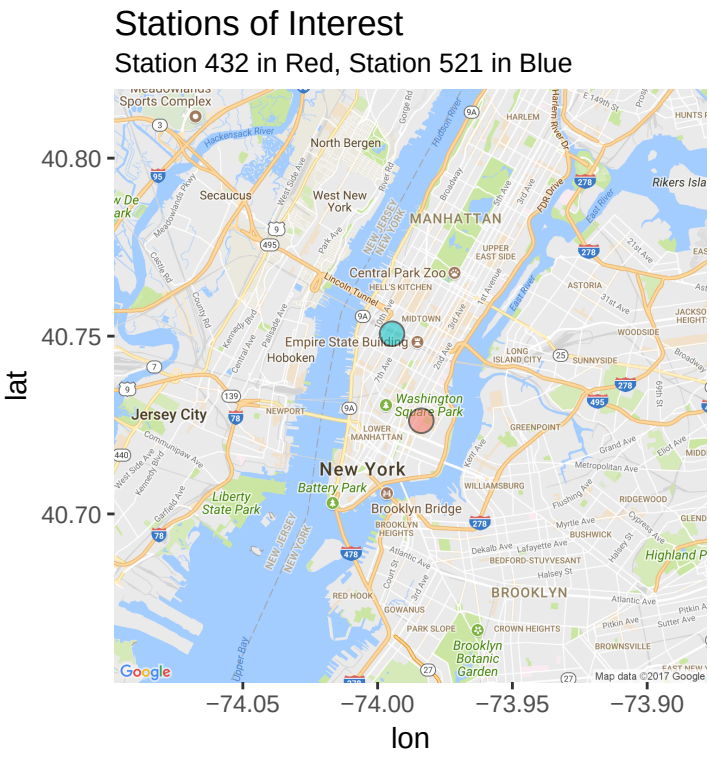
Tables of data are no fun! Let's see the data visualized.



As you can see most of the most popular importers are in the lower side of Manhatten. Most of the popular exporters are closer to time square. Although the analysis is not perfect. We decided to move on and make a predictive models on the top two stations. We chose our importer to be station 432, which is in Alphabet City (at E 7 St & Avenue A near Washington Square Park), and our exporter to be station 521, which is in Midtown between the Empire State Building and Lincoln Tunnel (at 8 Ave & W 31 St N). Shown below:

```
## # A tibble: 6 x 11
##      id median.time.out start.count median.time.in end.count difference
##   <int>           <dbl>       <int>          <dbl>     <int>      <int>
## 1    72           851.0        1419            944      1383         36
## 2    79           664.0        1093            640      1145        -52
## 3    82           547.0         471            647       441         30
```

```
## 4    83           613.0         567           733        589        -22
## 5   116           521.5        2124           518       2153        -29
## 6   119           502.0          98           587         98          0
## # ... with 5 more variables: station.latitude <dbl>,
## #   station.longitude <dbl>, total.ride.count <int>,
## #   normalized.difference <dbl>, positive.difference <lgl>
```



Let's move on to our model.

Building a Model

We went back to our SQL server and now pulled a new subsample: all rides that went to and from our selected stations. At this point we will now combine the data from Citibikes and historic Weather Data to create predictive models for stations changes. We combined the data by hour so that each hour has the

We wrote code to wrangle the data so that it is ready to make interesting graphs about each station but we never had time to make the graphs. The start times, end times and dates were in different formats. We made the Citi Bikes date time objects and the weather data date time objects into one format using the tidyverse package lubridate. The code can be found in station-analysis.Rmd.

```
# Choosing important variables
Weather_NYC <- Weather_NYC %>%
  select(valid, tmpf, dwpf, relh, vsby)
```

Rationale for Choosing the weather data variables: The variables relating to the weather data chosen are air temperature in Fahrenheit(tmpf), Dew Point Temperature in Fahrenheit (dwpf), Relative Humidity in percentage (relh), and visibility in miles (vsby). We chose these variables because each of these variables can have an impact on a biker's decision to choose or not to choose to bike and can significantly affect bike usage. We chose these variables to build a preliminary model and make predictions based off the model. The other variables like wind direction in degrees from north, sky level coverages, sky level altitudes, and wind gust had

some missing observations. We believe that these variables don't affect bike usage much and not including them would not affect the model.

Based on the graphs we made for the original exploratory analysis we decided to group hour into categories of early morning, morning commute, daytime, and evening commute, and night.

We also engineered variables for weekday vs. weekend and all of the seasons.

A small part of the weather data was missing fields - roughly 1% - so we decided to remove these rows.

Random Forest Regression and Classification

```
## [1] 16.86
```

Addtionally, let's see let's look at how accurate the model was at classifying if the difference was positive or negative (export or import).

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE TRUE
##      FALSE  2232  918
##      TRUE   1239 1941
##
##                Accuracy : 0.659
##                  95% CI : (0.647, 0.671)
##     No Information Rate : 0.548
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.319
##  Mcnemar's Test P-Value : 5.58e-12
##
##             Sensitivity : 0.643
##             Specificity : 0.679
##          Pos Pred Value : 0.709
##          Neg Pred Value : 0.610
##              Prevalence : 0.548
##          Detection Rate : 0.353
##    Detection Prevalence : 0.498
##       Balanced Accuracy : 0.661
##
##        'Positive' Class : FALSE
##
```

Does our model do better than predicting all of the differences as negative ? (All of the hourly changes as import for station 521)

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE TRUE
##      FALSE  3471 2859
##      TRUE      0    0
##
##                Accuracy : 0.548
##                  95% CI : (0.536, 0.561)
##     No Information Rate : 0.548
```

```
##     P-Value [Acc > NIR] : 0.505
##
##                   Kappa : 0
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 1.000
##             Specificity : 0.000
##          Pos Pred Value : 0.548
##          Neg Pred Value :   NaN
##              Prevalence : 0.548
##          Detection Rate : 0.548
##    Detection Prevalence : 1.000
##       Balanced Accuracy : 0.500
##
##        'Positive' Class : FALSE
##
```

Yes, our classification error reat is better than baseline (classifying all the hourly changes for station 521 as net import). However, the RMSE is still very high. 16.86 is too high of a Root Mean Squared Error to be effective in predicting when a station might need bikes moved to it. The RMSE means that the errors are off by almost 17 bikes. That means that the prediction could predict 16 bikes left a station but actually no bikes left the station. In a future model, we should just predict the hourly demand (not hourly change) for each station and then create information from that.

SVM Linear Classification

```
## # A tibble: 6 x 16
##   net.change.521  tmpf  dwpf  relh  vsby summer spring winter  fall
##            <chr> <dbl> <dbl> <dbl> <dbl>  <dbl>  <dbl>  <dbl> <dbl>
## 1              N 73.94 69.98 87.45     9      1      0      0     0
## 2              P 75.02 69.98 84.34     7      1      0      0     0
## 3              N 75.02 71.06 87.50     6      1      0      0     0
## 4              P 75.92 71.96 87.55     7      1      0      0     0
## 5              P 73.04 71.06 93.52     6      1      0      0     0
## 6              N 73.04 71.06 93.52     8      1      0      0     0
## # ... with 7 more variables: week.day <lgl>, weekend.day <lgl>,
## #   EarlyMorning <dbl>, Commuting <dbl>, DayTime <dbl>, Evening <dbl>,
## #   Night <dbl>
```

```
## # A tibble: 6 x 16
##   net.change.521  tmpf  dwpf  relh  vsby summer spring winter  fall
##            <chr> <dbl> <dbl> <dbl> <dbl>  <dbl>  <dbl>  <dbl> <dbl>
## 1              N 75.02 69.98 84.34     8      1      0      0     0
## 2              P 73.94 69.98 87.45     9      1      0      0     0
## 3              N 75.02 73.04 93.57     6      1      0      0     0
## 4              N 75.02 73.04 93.57     4      1      0      0     0
## 5              P 73.04 71.06 93.52     8      1      0      0     0
## 6              P 73.04 71.06 93.52     7      1      0      0     0
## # ... with 7 more variables: week.day <lgl>, weekend.day <lgl>,
## #   EarlyMorning <dbl>, Commuting <dbl>, DayTime <dbl>, Evening <dbl>,
## #   Night <dbl>
```

```
## Confusion Matrix and Statistics
##
##           Reference
```

```
## Prediction    N    P
##          N 2614 1451
##          P  858 1408
##
##                Accuracy : 0.635
##                  95% CI : (0.623, 0.647)
##     No Information Rate : 0.548
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.25
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.753
##             Specificity : 0.492
##          Pos Pred Value : 0.643
##          Neg Pred Value : 0.621
##              Prevalence : 0.548
##          Detection Rate : 0.413
##    Detection Prevalence : 0.642
##       Balanced Accuracy : 0.623
##
##        'Positive' Class : N
##
```

The accuracy rate for the Support Vector Machines model was slightly lower than Random Forest by about 2 % but still better than the base line prediction.

Final Remarks

Our last modeling step was rushed. We did not spend the time graphing the data to better visualize how weather interacts with bike demand. Additionally, we did not spend enough time choosing variables and optimizing the models. Building models on large sets of data is time consuming! However, we still wanted to incorporate it to show how the data analysis life cycle might work. **We started with a very large dataset, then made a random sample to learn about the overall data. From the randomized sample we discovered that we wanted to look into a new subset of data (on stations we thought were of interest). We began to build a model on this data but found that it was not very effective at predicting our desired information (when to move bikes from one station to another).** The next step would be to go back to the drawing board and come up with a better way to approach our orignal goal. Once our model works we could scale it up to the larger dataset and then finish the cycle of the data analysis! Overall, this project, despite its limitations, was a great learning experience for the both of us. We ended up learning five new things throughout the course of this project namely big data analysis, SQL, visualizations using ggmaps, shell, and wrangling using lubridate.