

BABS initial report

Georgie Knight

23 August, 2016

Introduction

We are analysing the Bay area bike share usage data. This is available here:

<http://www.bayareabikeshare.com/open-data>

The Bay area bike share is an easy to use bike rental scheme set up in the Bay area. The data set contains information about:

Bike number Trip start day and time Trip end day and time Trip start station Trip end station Rider type - Annual or Casual (24-hour or 3-day member) If an annual member trip, it will also include the member's home zip code The data set also includes: Weather information per day per service area Bike and dock availability per minute per station

The task is to use this data to understand how the system is currently being used, how it can be improved and anything else we can find with a view to a proposed expansion of the system.

The data

We are using the year 2 data which ranges from September 2014 - August 2015. This contains four separate .csv files. Station data contains the information about the individual docking stations like location, trip data contains information about every trip taken, status gives a minute by minute account of the status of each docking station and weather data contains the daily weather reports.

Preparing the data

Firstly we needed to either compress the status data or put it into a more manageable form as it contains 37 million observations. We decided to compress it as a lot of the observations are redundant. We selected only the observations where there is a change in the status. This gave us just over 1 million observations. This process is outlined in the accompanying *Filter_the_data* file.

We then had to prepare and combine the data so that we could work with individual databases. This meant preparing the station and weather data and combining with the trip and status data files. This gave us two databases containing all the information for trip and status. This process is outlined in the accompanying *Preparing_station_data*, *Preparing_trip_data*, *Preparing_weather_data* and *Combine_the_data* files.

Finally, some of the stations had been physically moved during the year. This meant that their longitude and latitude had changed at a particular date. This information had to be incorporated to the database. This process is outlined in the accompanying *_Update_the_data* file.

The result is two databases which we can analyse: *trip* and *status*

```
glimpse(trip)
```

```
## Observations: 354,152
## Variables: 40
## $ Trip.ID           <int> 913460, 913459, 913455, 913454, 9134...
## $ Duration          <int> 765, 1036, 307, 409, 789, 293, 896, ...
## $ Start_trip        <time> 2015-08-31 23:26:00, 2015-08-31 23:...
```

```

## $ Start.Station      <chr> "Harry Bridges Plaza (Ferry Building...
## $ Start.Terminal     <int> 50, 31, 47, 10, 51, 68, 51, 60, 56, ...
## $ End_trip           <time> 2015-08-31 23:39:00, 2015-08-31 23:...
## $ End.Station        <chr> "San Francisco Caltrain (Townsend at...
## $ End.Terminal       <int> 70, 27, 64, 8, 60, 70, 60, 74, 55, 6...
## $ Bike..             <int> 288, 35, 468, 68, 487, 538, 363, 470...
## $ Subscriber.Type    <chr> "Subscriber", "Subscriber", "Subscri...
## $ Zip.Code           <chr> "2139", "95032", "94107", "95113", "...
## $ Date               <date> 2015-08-31, 2015-08-31, 2015-08-31,...
## $ landmark           <chr> "San Francisco", "Mountain View", "S...
## $ start_lat          <dbl> 37.79539, 37.40044, 37.78898, 37.337...
## $ start_long         <dbl> -122.3942, -122.1083, -122.4035, -12...
## $ end_lat            <dbl> 37.77662, 37.38922, 37.78226, 37.330...
## $ end_long           <dbl> -122.3953, -122.0819, -122.3927, -12...
## $ Max.TemperatureF   <int> 78, 82, 78, 85, 78, 78, 78, 78, 78, ...
## $ Mean.TemperatureF  <int> 69, 72, 69, 72, 69, 69, 69, 69, 69, ...
## $ Min.TemperatureF   <int> 60, 61, 60, 59, 60, 60, 60, 60, 60, ...
## $ Max.Dew.PointF     <int> 58, 62, 58, 59, 58, 58, 58, 58, 58, ...
## $ MeanDew.PointF     <int> 57, 56, 57, 55, 57, 57, 57, 57, 57, ...
## $ Min.DewpointF      <int> 54, 52, 54, 51, 54, 54, 54, 54, 54, ...
## $ Max.Humidity        <int> 84, 84, 84, 84, 84, 84, 84, 84, 84, ...
## $ Mean.Humidity       <int> 67, 63, 67, 58, 67, 67, 67, 67, 67, ...
## $ Min.Humidity        <int> 50, 42, 50, 32, 50, 50, 50, 50, 50, ...
## $ Max.Sea.Level.PressureIn <dbl> 29.95, 29.97, 29.95, 29.95, 29.95, 2...
## $ Mean.Sea.Level.PressureIn <dbl> 29.91, 29.92, 29.91, 29.90, 29.91, 2...
## $ Min.Sea.Level.PressureIn <dbl> 29.87, 29.86, 29.87, 29.85, 29.87, 2...
## $ Max.VisibilityMiles <int> 10, 10, 10, 10, 10, 10, 10, 10, 10, ...
## $ Mean.VisibilityMiles <int> 10, 10, 10, 10, 10, 10, 10, 10, 10, ...
## $ Min.VisibilityMiles <int> 9, 10, 9, 10, 9, 9, 9, 9, 9, 9...
## $ Max.Wind.SpeedMPH   <int> 18, 22, 18, 20, 18, 18, 18, 18, 18, ...
## $ Mean.Wind.SpeedMPH  <int> 9, 6, 9, 6, 9, 9, 9, 9, 9, 9, ...
## $ Max.Gust.SpeedMPH   <int> 21, 25, 21, 24, 21, 21, 21, 21, 21, ...
## $ PrecipitationIn     <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ CloudCover          <int> 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ Events              <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
## $ WindDirDegrees      <int> 246, 6, 246, 308, 246, 246, 246, 246...
## $ Zip                 <int> 94107, 94041, 94107, 95113, 94107, 9...

```

```
glimpse(status)
```

```

## Observations: 1,135,974
## Variables: 33
## $ station_id         <int> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, ...
## $ bikes_available    <int> 15, 14, 15, 14, 15, 14, 15, 14, 15, ...
## $ docks_available    <int> 12, 13, 12, 13, 12, 13, 12, 13, 12, ...
## $ time               <time> 2014-09-01 00:00:03, 2014-09-01 02:...
## $ name               <chr> "San Jose Diridon Caltrain Station",...
## $ lat                <dbl> 37.32973, 37.32973, 37.32973, 37.329...
## $ long               <dbl> -121.9018, -121.9018, -121.9018, -12...
## $ landmark           <chr> "San Jose", "San Jose", "San Jose", ...
## $ installation       <date> 2013-08-29, 2013-08-29, 2013-08-29,...
## $ Date               <date> 2014-09-01, 2014-09-01, 2014-09-01,...
## $ Max.TemperatureF   <int> 86, 86, 86, 86, 86, 86, 86, 86, 86, ...
## $ Mean.TemperatureF  <int> 72, 72, 72, 72, 72, 72, 72, 72, 72, ...

```

```
## $ Min.TemperatureF      <int> 58, 58, 58, 58, 58, 58, 58, 58, 58, ...
## $ Max.Dew.PointF        <int> 60, 60, 60, 60, 60, 60, 60, 60, 60, ...
## $ MeanDew.PointF        <int> 54, 54, 54, 54, 54, 54, 54, 54, 54, ...
## $ Min.DewpointF         <int> 50, 50, 50, 50, 50, 50, 50, 50, 50, ...
## $ Max.Humidity          <int> 86, 86, 86, 86, 86, 86, 86, 86, 86, ...
## $ Mean.Humidity         <int> 59, 59, 59, 59, 59, 59, 59, 59, 59, ...
## $ Min.Humidity          <int> 31, 31, 31, 31, 31, 31, 31, 31, 31, ...
## $ Max.Sea.Level.PressureIn <dbl> 29.86, 29.86, 29.86, 29.86, 29.86, 29.86, 29.86, 29.86, 29.86, ...
## $ Mean.Sea.Level.PressureIn <dbl> 29.81, 29.81, 29.81, 29.81, 29.81, 29.81, 29.81, 29.81, 29.81, ...
## $ Min.Sea.Level.PressureIn <dbl> 29.75, 29.75, 29.75, 29.75, 29.75, 29.75, 29.75, 29.75, 29.75, ...
## $ Max.VisibilityMiles    <int> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, ...
## $ Mean.VisibilityMiles    <int> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, ...
## $ Min.VisibilityMiles    <int> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, ...
## $ Max.Wind.SpeedMPH      <int> 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, ...
## $ Mean.Wind.SpeedMPH     <int> 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, ...
## $ Max.Gust.SpeedMPH      <int> 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, ...
## $ PrecipitationIn        <chr> "0", "0", "0", "0", "0", "0", "0", "0", "0", ...
## $ CloudCover             <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ Events                 <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
## $ WindDirDegrees         <int> 296, 296, 296, 296, 296, 296, 296, 296, 296, 296, ...
## $ Zip                    <int> 95113, 95113, 95113, 95113, 95113, 95113, 95113, 95113, 95113, 95113, ...
```

Preliminary analysis

Bike availability

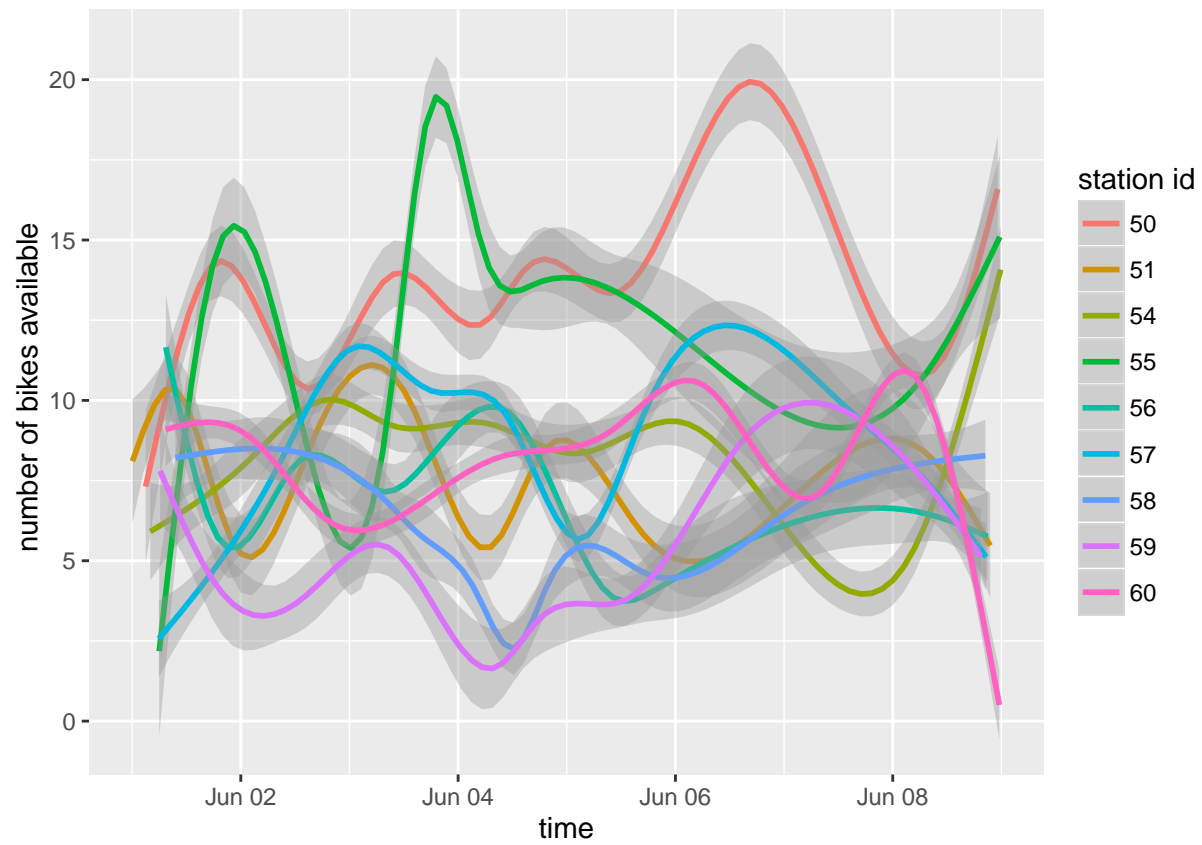
We can use our data to look at many aspects of the bike sharing system and how it functions. We can firstly answer a very simple question without doing anything fancy: Are there ever no bikes available?

```
min(status$bikes_available)
```

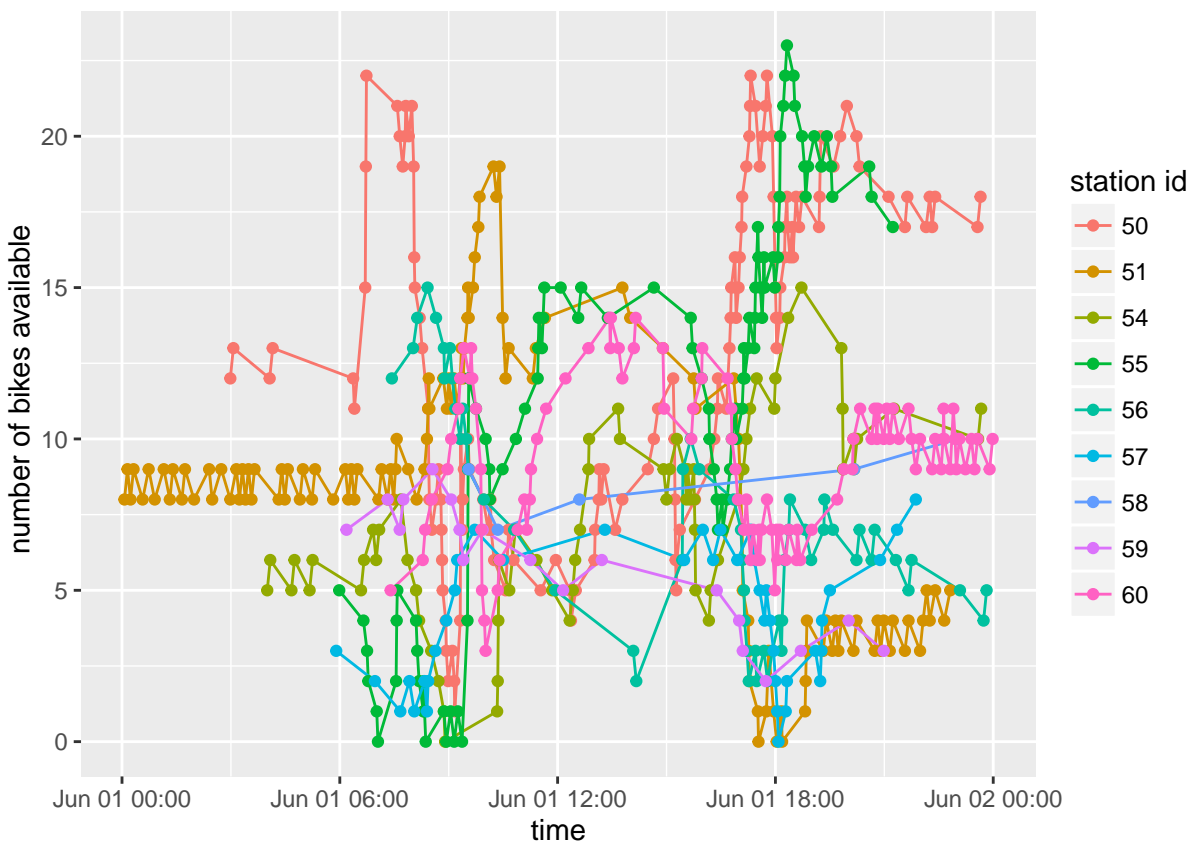
```
## [1] 0
```

Yes it happens. But that may not be the full story of course. Is this during operation hours? Is it due to maintenance? We will need to analyse further.

Let's take a look at how bikes available fluctuates on average in some select stations in San Francisco over a particular week.



Now let's look at the precise fluctuations on a particular day



We see that there are times when the number available goes to zero or is quite low.

Trips

We can look at a summary of the trip duration data:

```
##      Duration
##  Min.   :    60
## 1st Qu.:   342
##  Median :   511
##   Mean  :  1046
## 3rd Qu.:   739
##   Max.  :17270400
```

We see that the average trip length is $1046/60 = 17.4$ minutes, the minimum trip length was 1 minute and the max was nearly 200 days! Does this vary by location?

```
## # A tibble: 5 x 1
##   landmark
##   <chr>
## 1 San Francisco
## 2 Mountain View
## 3 San Jose
## 4 Palo Alto
## 5 Redwood City
```

```
##      Duration
## Min.   :    60
## 1st Qu.:   352
## Median :   519
## Mean   :   976
## 3rd Qu.:   740
## Max.   :17270400
```

```
##      Duration
## Min.   :    61
## 1st Qu.:   238
## Median :   294
## Mean   :  1430
## 3rd Qu.:   457
## Max.   :1852590
```

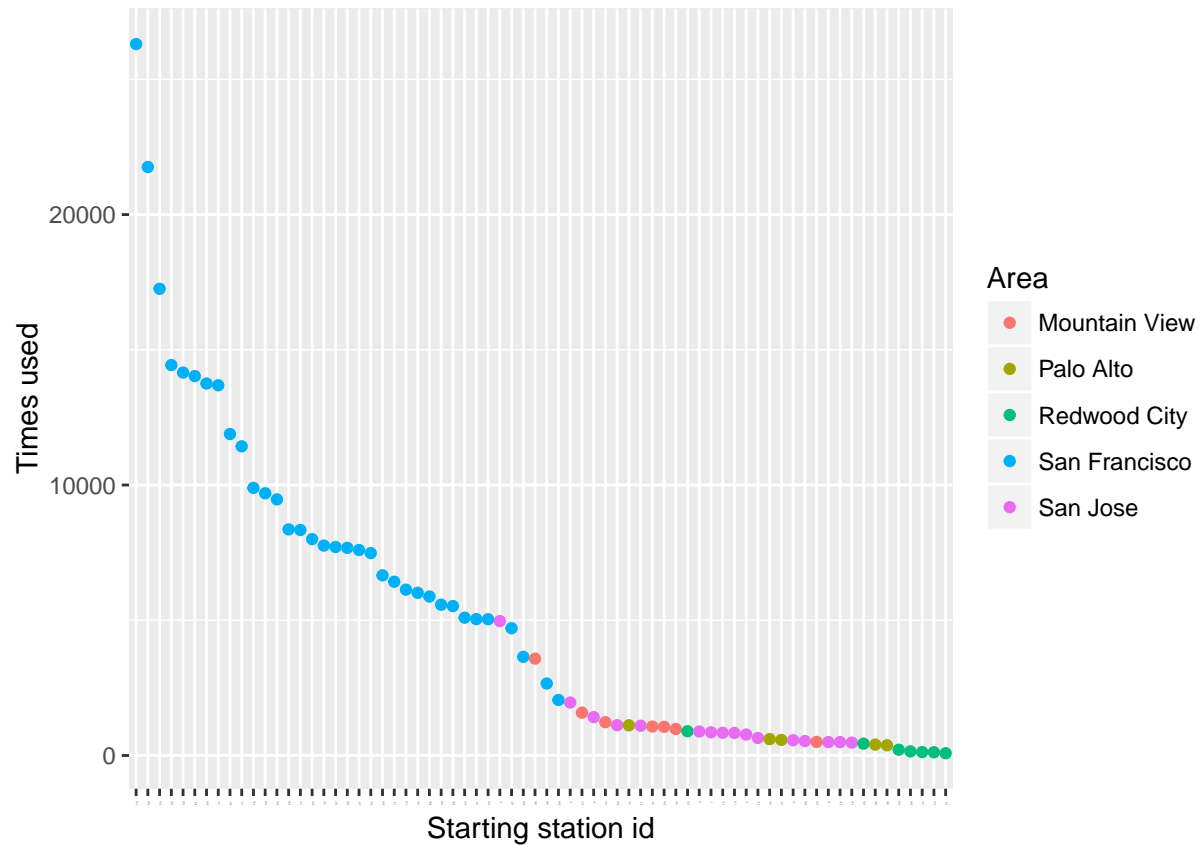
```
##      Duration
## Min.   :    62
## 1st Qu.:   312
## Median :   466
## Mean   :  1401
## 3rd Qu.:   663
## Max.   :655939
```

```
##      Duration
## Min.   :    66
## 1st Qu.:   288
## Median :   863
## Mean   :  4230
## 3rd Qu.:  2018
## Max.   :1133540
```

```
##      Duration
## Min.   :   68.0
## 1st Qu.:  273.5
## Median :  621.0
## Mean   : 2287.6
## 3rd Qu.:  863.5
## Max.   :720454.0
```

Station popularity.

We can find the most used starting points as a measure of station popularity

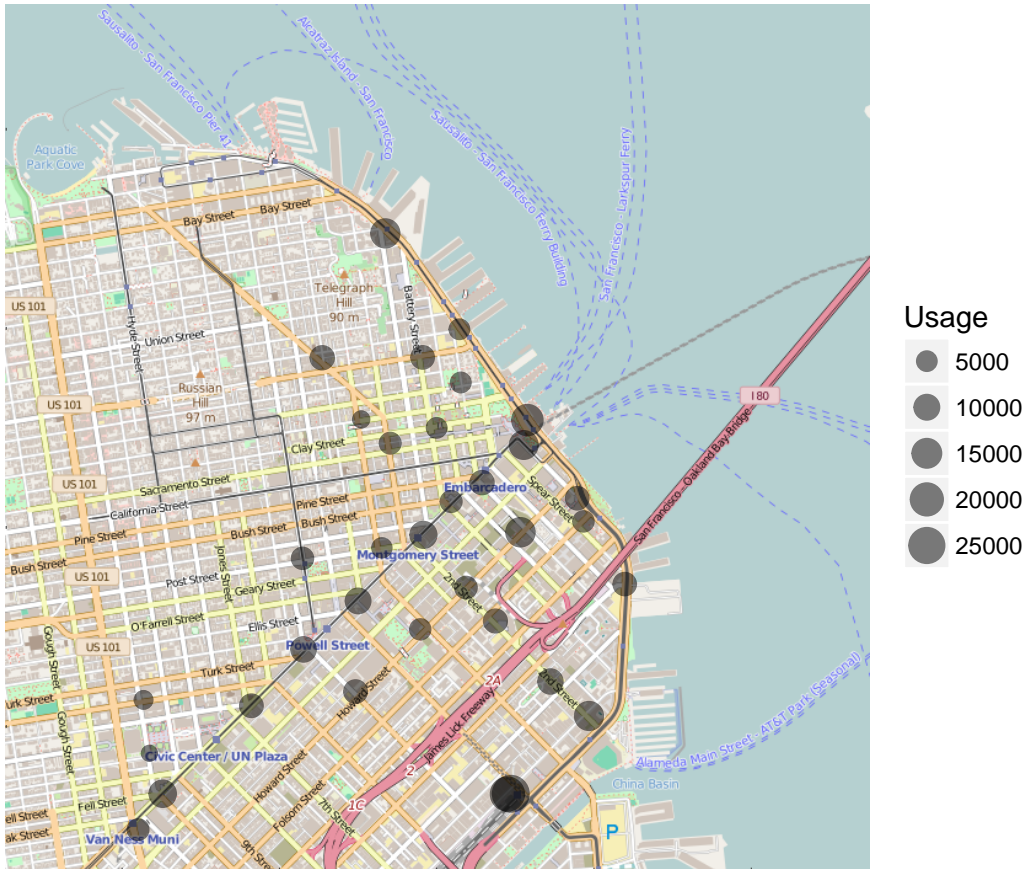


We can visualise the usage in a particular region like San Francisco

```
## Map from URL : http://maps.googleapis.com/maps/api/staticmap?center=san+francisco+financial+district
```

```
## Information from URL : http://maps.googleapis.com/maps/api/geocode/json?address=san%20francisco%20fi
```

```
## Warning: Removed 36 rows containing missing values (geom_point).
```



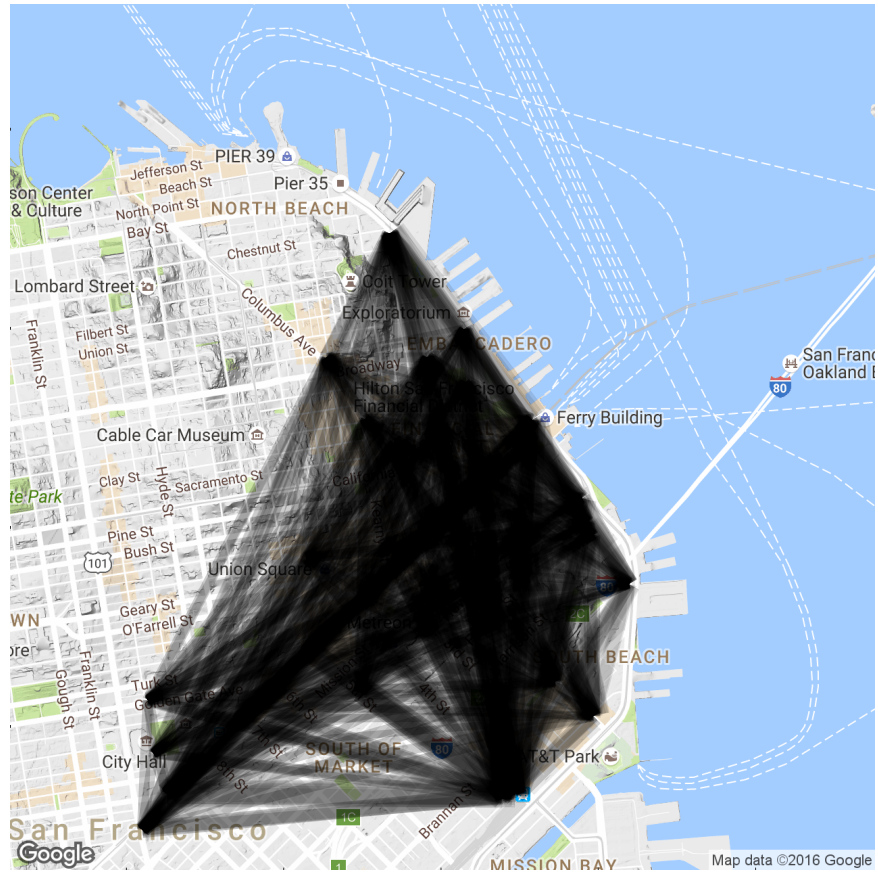
Trip popularity

We can visualise the trips in San Francisco to try to get an idea of how the system is used.

```
## Map from URL : http://maps.googleapis.com/maps/api/staticmap?center=san+francisco+financial+district
```

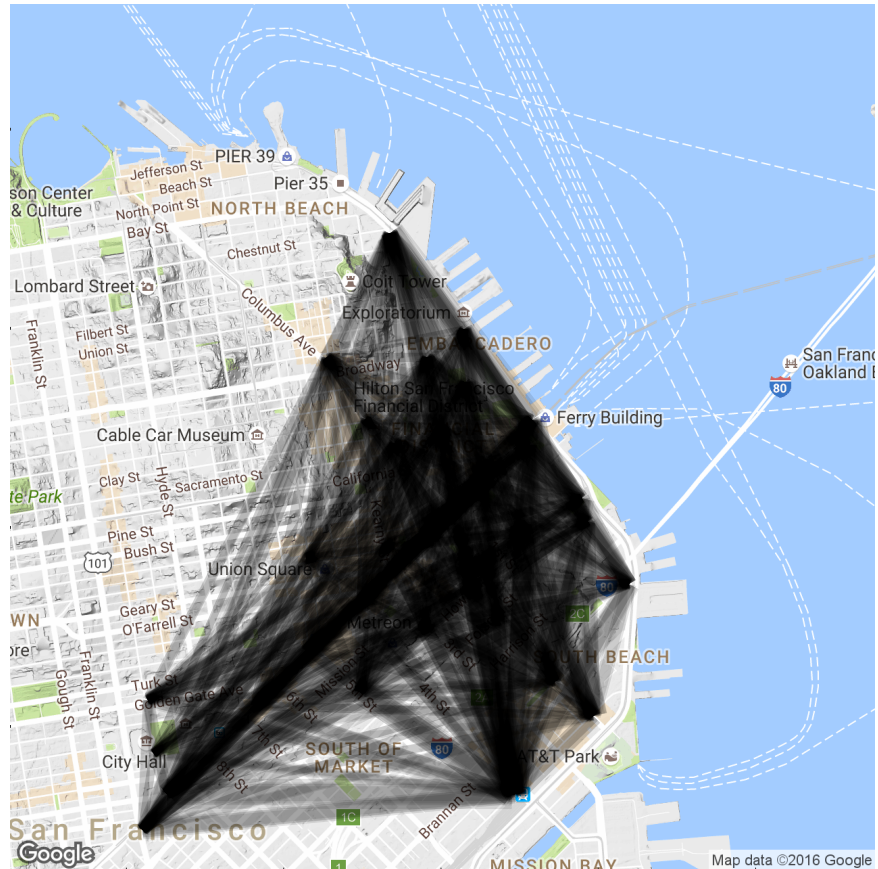
```
## Information from URL : http://maps.googleapis.com/maps/api/geocode/json?address=san%20francisco%20fi
```

```
## Warning: Removed 79 rows containing missing values (geom_segment).
```

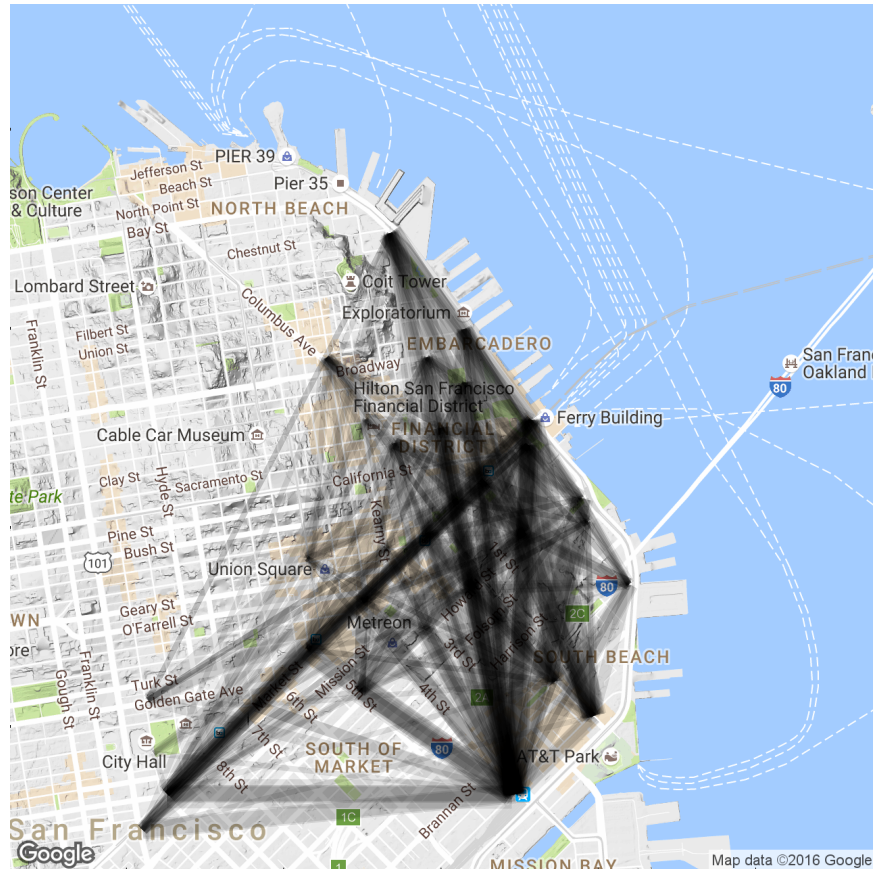
Let's look at all the trips in a given week

```
## Warning: Removed 69 rows containing missing values (geom_segment).
```



Or on a particular day

```
## Warning: Removed 43 rows containing missing values (geom_segment).
```



We start to see how the system is being used from these illustrations.

Future analysis

Now that we have the data prepared and have taken a look at it we can start our proper analysis.

We will need to obtain more temporal data such as how the stations are used on average at a particular time of day. How are they used throughout the week etc. We need to find out how long stations go without having any bikes or docking stations. We need to define some measures of overused and underused stations and find these stations. Performing a graph based analysis will allow us to pick out the important stations. Deriving certain centrality measures will help with this.

We will need to build some predictive models for bike use. This could involve correlating the weather data with bike use so that one can accurately estimate bike use from a weather report.

The data itself will not tell us *why* a particular station is so popular or not. Obtaining and combining with data about the Bay area such as shops, restaurants, offices will help us answer this question and potentially identify other popular spots.