Expansion of a bike share.

Georgie Knight

28 September, 2016

Contents


```
library("dplyr")
library("tidyr")
library("lubridate")
library("readr")
library("ggplot2")
library("ggmap")
library("igraph")
library("popgraph")
library("zipcode")
```

```
library("pander")
library("reshape2")
library("knitr")
trip_read    <- read_csv("trip_full_updated2.csv")
status_read  <- read_csv("status_full_updated.csv")
station      <- read_csv("station_data.csv")
weather_read <- read_csv("201508_weather_data.csv")
trip         <- dplyr::tbl_df(trip_read)
status       <- dplyr::tbl_df(status_read)
station      <- dplyr::tbl_df(station)
weather      <- dplyr::tbl_df(weather_read)
```

Figure 1: The Bay Area Bike Share.

1 Introduction

1.1 Bay area bike share

The Bay Area Bike Share is a bike rental system consisting of 700 bikes and 70 stations located throughout 5 cities ;

- Mountain View
- Palo Alto
- Redwood City
- San Jose
- San Francisco.

Anyone can purchase a 24 hour or 3 day pass from staion kiosks. 1 year memberships are also available online. Users can then hire and drop off bikes at any station. The system is designed to encourage short trips, hence trips over 30 minutes incur late fees. $4 is charged for a 30-60 minute trip, $7 is added for each additional 30 minutes.

1.2 Data

1.2.1 Open data set

The Bay Area Bike Share released data sets on how the system is being used. This can be found here. It covers the period from September 2014 to August 2015 and consists of four separate .csv files; station data contains information about each docking station, 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 daily weather reports.

We analyse this data to understand how the system is being used and to make recommendations for a proposed system expansion. We look at usage patterns and statistics and attempt to make recommendations for where new stations could best be located.

1.2.2 Preparing the data

The first job was to compress the status data to put it into a more manageable form. It contains 37 million minute-by minute observations. We compressed the data by selecting only the observations where there is a change in the status. This gave us just over 1 million observations. Note that we lost no information by compressing in this way.

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.

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.

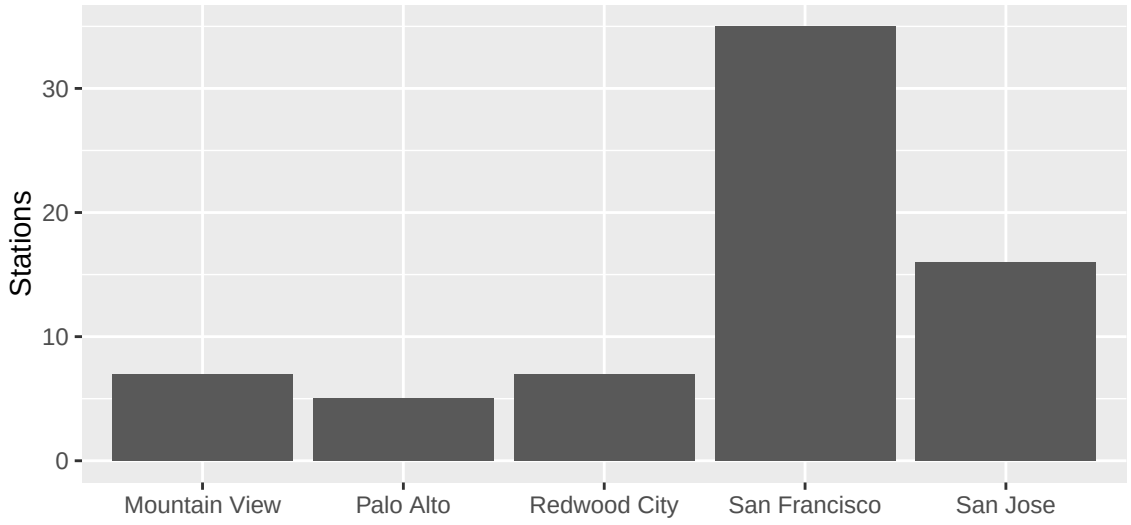
2 System statistics

The first task was to get a general idea of how the system is used. For this we calculated the basic statistics about the system.

2.1 Location

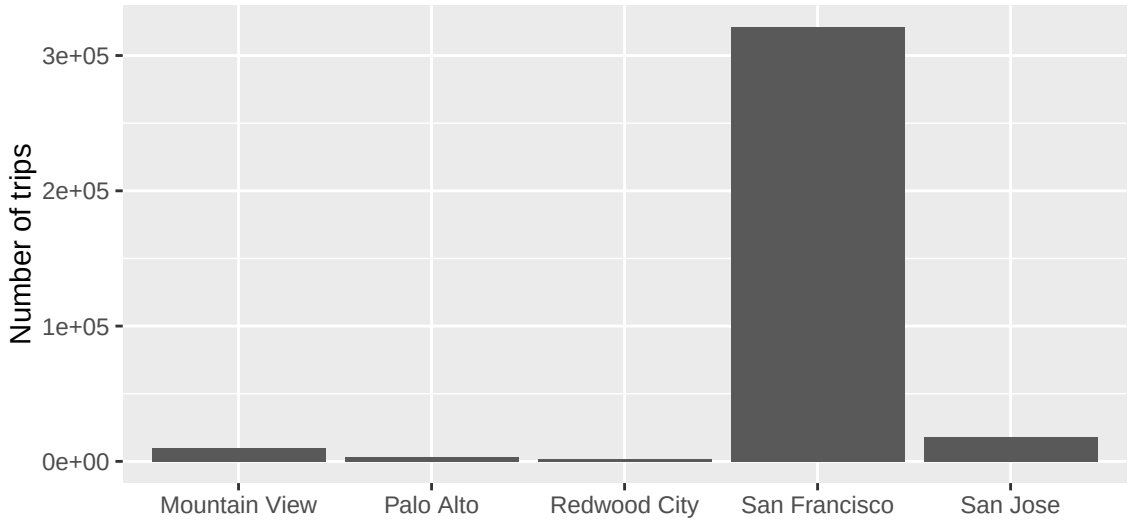
- Half of the stations are located in San Francisco.

```
ggplot(station, aes(x = landmark))+ geom_bar()+labs(x="", y="Stations")
```



- In total there were 354152 bike trips made. Over 90 per cent of these start in San Francisco.

```
ggplot(trip, aes(x = startLandmark))+ geom_bar()+
  labs(x="", y="Number of trips")
```

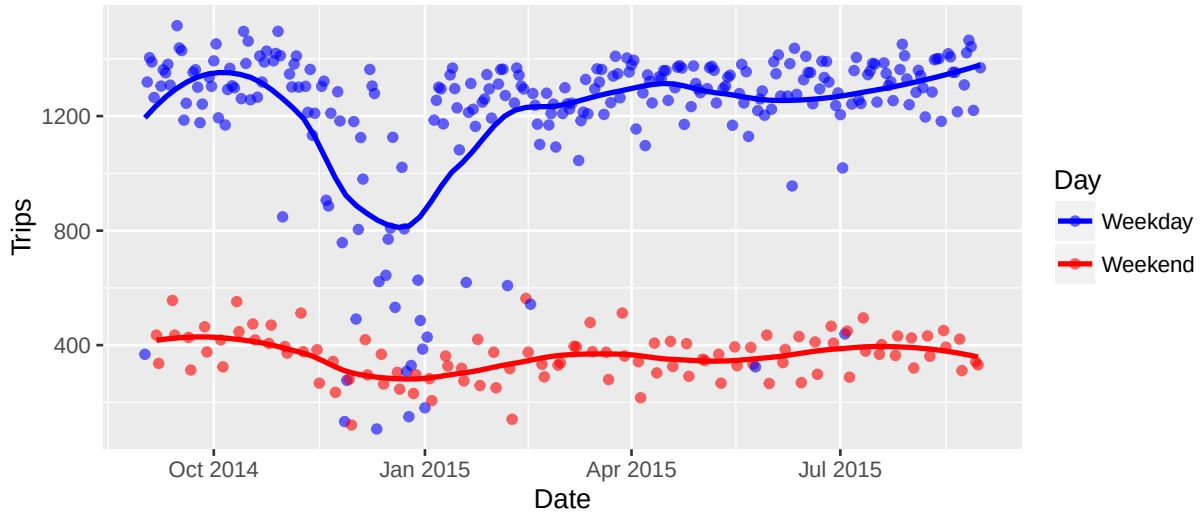


2.2 Trip time

- On average over 1200 trips per day are made during the week.

- Over 360 per day are made on weekends.
- Usage drops around January, the time with the most rain.

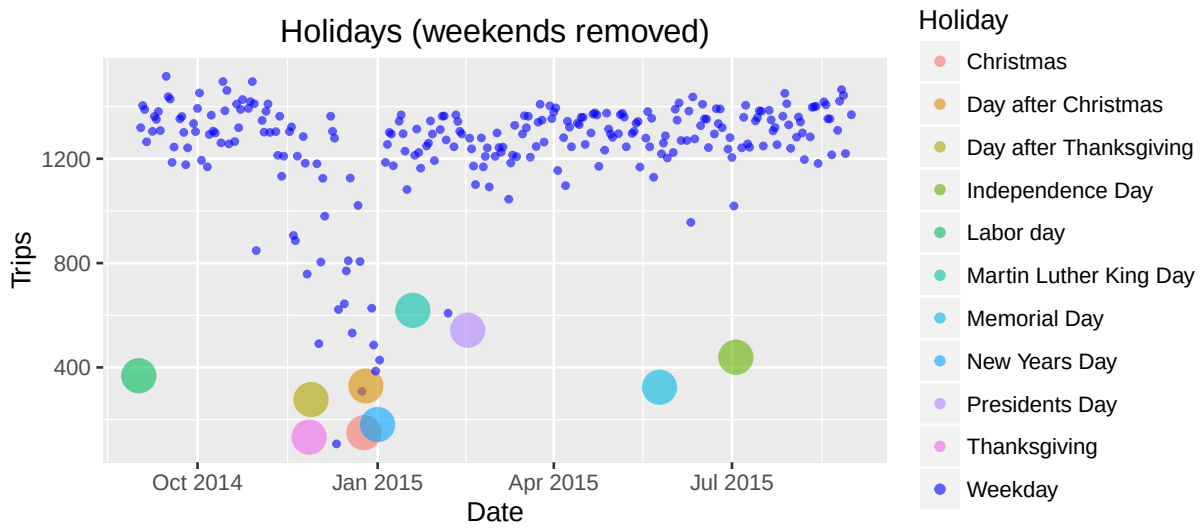
```
trip_per_day <- trip %>%
  group_by(Date) %>%
  summarise(count = n()) %>%
  ungroup() %>%
  mutate(weekday = wday(Date)) %>%
  mutate(weekend = ifelse(weekday == 1 | weekday == 7, "Weekend","Weekday"))

ggplot(trip_per_day, aes(x = Date,
                         y = count))+
  geom_point(aes(col = as.factor(weekend)),
             alpha = 0.6)+
  geom_smooth(aes(col = as.factor(weekend)),
              span = 0.4,
              se= FALSE,
              show.legend = TRUE)+
  labs(x = "Date", y = "Trips", col = "Day")+
  scale_color_manual(values = c("blue", "red"))
```



- Usage also drops on holidays. (Note that Independence day fell on a saturday so was observed on friday the 3rd.)

```
trip_per_day <- trip %>%
  group_by(Date) %>%
  summarise(count = n()) %>%
  ungroup() %>%
  mutate(weekday = wday(Date)) %>%
  mutate(weekend = ifelse(weekday == 1 | weekday == 7, "Weekend","Weekday"))

trip_holiday <- trip_per_day  %>%
  filter(weekend == "Weekday") %>%
  mutate(Holiday = ifelse(as.Date(Date) == "2014-09-01", "Labor day",
```

```
                  ifelse(as.Date(Date) == "2014-11-27", "Thanksgiving",
                  ifelse(as.Date(Date) == "2014-11-28", "Day after Thanksgiving",
                  ifelse(as.Date(Date) == "2014-12-25", "Christmas",
                  ifelse(as.Date(Date) == "2014-12-26", "Day after Christmas",
                  ifelse(as.Date(Date) == "2015-01-01", "New Years Day",
                  ifelse(as.Date(Date) == "2015-01-19", "Martin Luther King Day",
                  ifelse(as.Date(Date) == "2015-02-16", "Presidents Day",
                  ifelse(as.Date(Date) == "2015-05-25", "Memorial Day",
                  ifelse(as.Date(Date) == "2015-07-03", "Independence Day", "Weekday"
                  ))))))))))

gg_color_hue <- function(n) {
  hues = seq(15, 375, length = n + 1)
  hcl(h = hues, l = 65, c = 100)[1:n]
}

ggplot(trip_holiday, aes(x = Date,
                         y = count,
                         col = as.factor(Holiday)))+
  geom_point(aes( size = ifelse(Holiday == "Weekday",1,2)),
             alpha = 0.6)+
  labs(x= "Date",
       y= "Trips",
       col = "Holiday",
       title = "Holidays (weekends removed)")+
  guides(size=FALSE)+
  scale_color_manual(
    values=c("#F8766D",
             "#DB8E00",
             "#AEA200",
             "#64B200",
             "#00BD5C",
             "#00C1A7",
             "#00BADE",
             "#00A6FF",
             "#B385FF",
             "#EF67EB",
             "blue"))
```



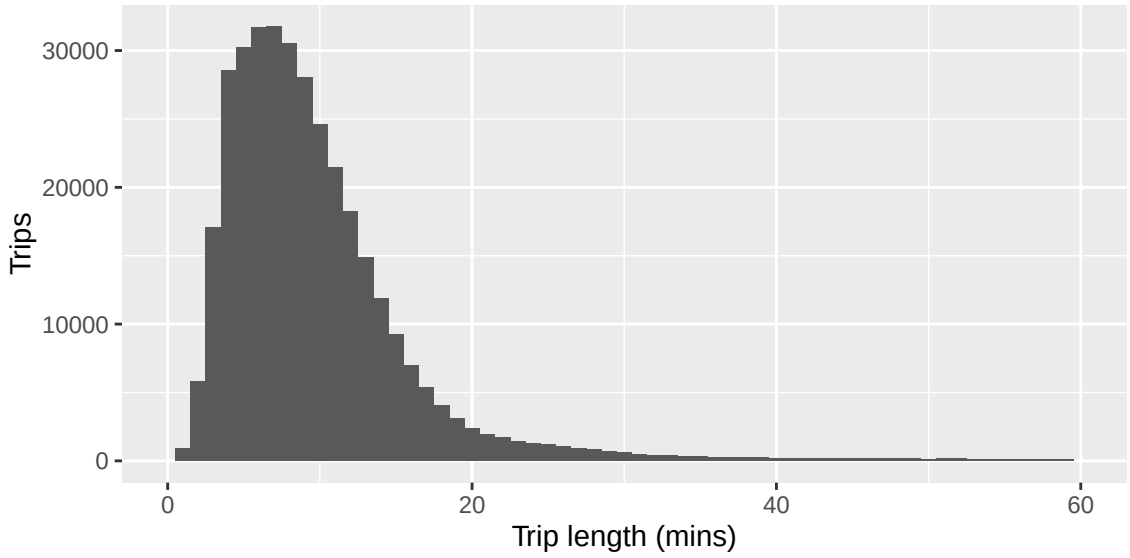

```
trip_l <- trip %>%
  mutate(Duration = Duration/60)

trip30 <- filter(trip_l, Duration <= 30)

# Calculate distance in kilometers between two points
earth.dist <- function (long1, lat1, long2, lat2)
{
rad <- pi/180
a1 <- lat1 * rad
a2 <- long1 * rad
b1 <- lat2 * rad
b2 <- long2 * rad
dlon <- b2 - a2
dlat <- b1 - a1
a <- (sin(dlat/2))^2 + cos(a1) * cos(b1) * (sin(dlon/2))^2
c <- 2 * atan2(sqrt(a), sqrt(1 - a))
R <- 6378.145
d <- R * c
return(d)
}

trip_dist <- mutate(trip, distance = earth.dist(start_long, start_lat, end_long, end_lat))
```

2.3 Trip length

- The average trip lasts approximately 17.43 minutes
- The average distance covered is approximately 1.35 km.
- Most trips (95.5 per cent) are made within the 30 minute time limit incurring no late fees.

```
trip_l <- trip_l %>%
  mutate(Duration = ifelse(Duration >60,61, Duration))

ggplot(trip_l, aes(x= Duration))+
```

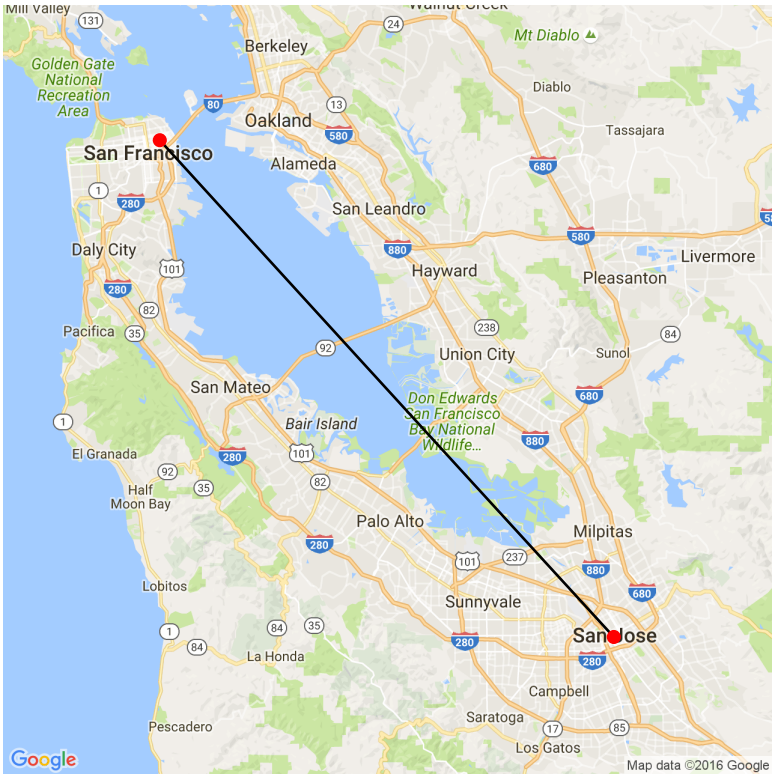
```
  geom_histogram(binwidth = 1)+
  labs(x= "Trip length (mins)", y = "Trips")+xlim(0,60)
```



- The longest trip made lasted 200 days! This trip would have technically incurred late fees of $67153. Perhaps this user didn't understand the *sharing* system. Or perhaps this was a recovered stolen bike?
- The furthest distance travelled between stations was 68 km, from San Jose to San Francisco, a trip of more than 8 hours:

```
longest_trip <- trip_dist %>%
  filter(distance > 60)

stations2 <- filter(station, name == "MLK Library" |
                            name == "Market at 4th")
location <-
  c((longest_trip$start_long + longest_trip$end_long)/ 2.0,
    (longest_trip$start_lat + longest_trip$end_lat)/ 2.0)
BABSmap <- get_map(location,maptype ="roadmap", zoom=10)
ggmap(BABSmap)+
  geom_segment(data = longest_trip,
               aes(y= start_lat,
                   x = start_long,
                   yend =  end_lat,
                   xend = end_long),alpha = 1.0)+
  geom_point(data = stations2, aes(y= lat,
                                   x = long),
             col = "red",
             size = 2)+
  theme(axis.line=element_blank(),
        axis.text.x=element_blank(),
        axis.text.y=element_blank(),
        axis.ticks=element_blank(),
        axis.title.x=element_blank(),
        axis.title.y=element_blank(),
```

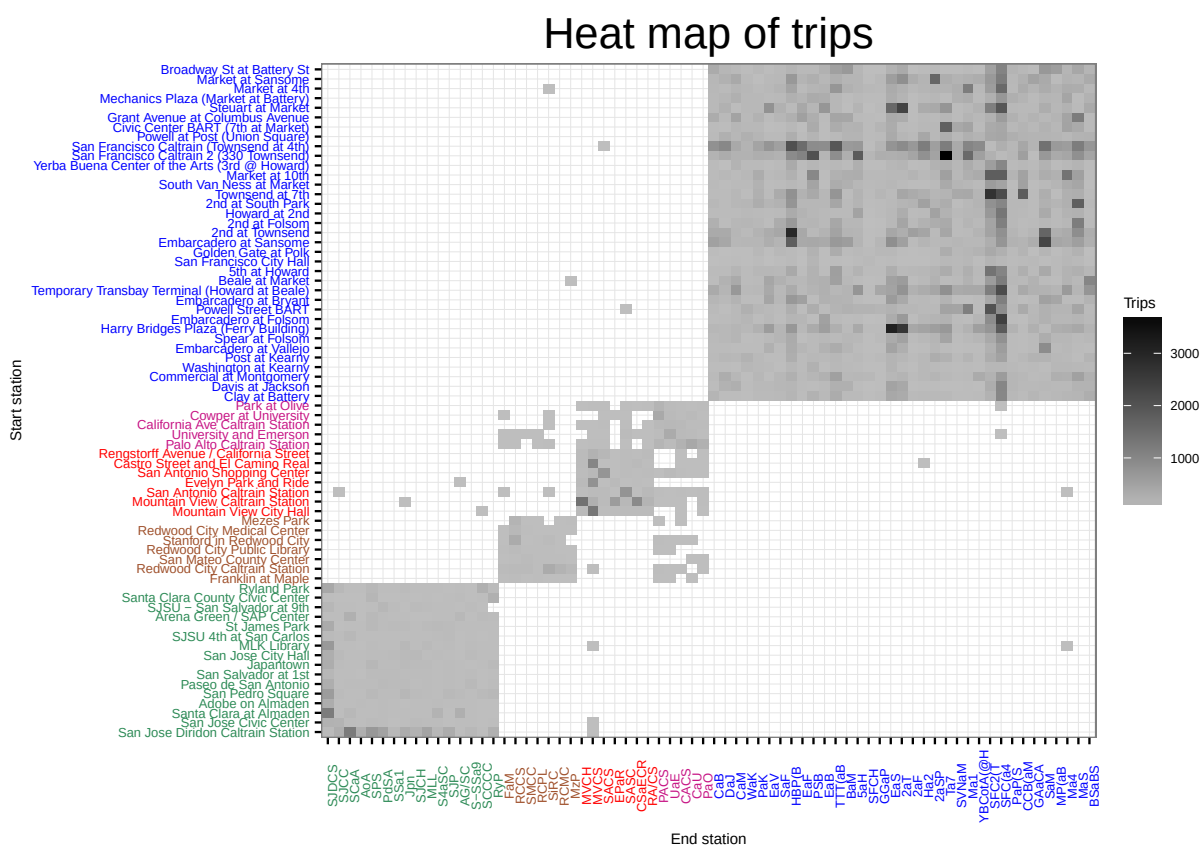
```
        legend.position="none",
        panel.background=element_blank(),
        panel.border=element_blank(),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        plot.background=element_blank())
```



- Most trips are of course intra city, but people do use the bikes to journey between cities which we can see from the following heat map of trips:

```
station2 <- station %>%
  select(station_id, name, landmark) %>%
  mutate(landmark =
           factor(landmark,
                  levels =
                    c("San Jose",
                      "Redwood City",
                      "Mountain View",
                      "Palo Alto",
                      "San Francisco"))) %>%
    arrange(landmark)

stationLabels <- station2$name
stationidsy <- stationLabels
stationidsx <- abbreviate(stationLabels,3)
```

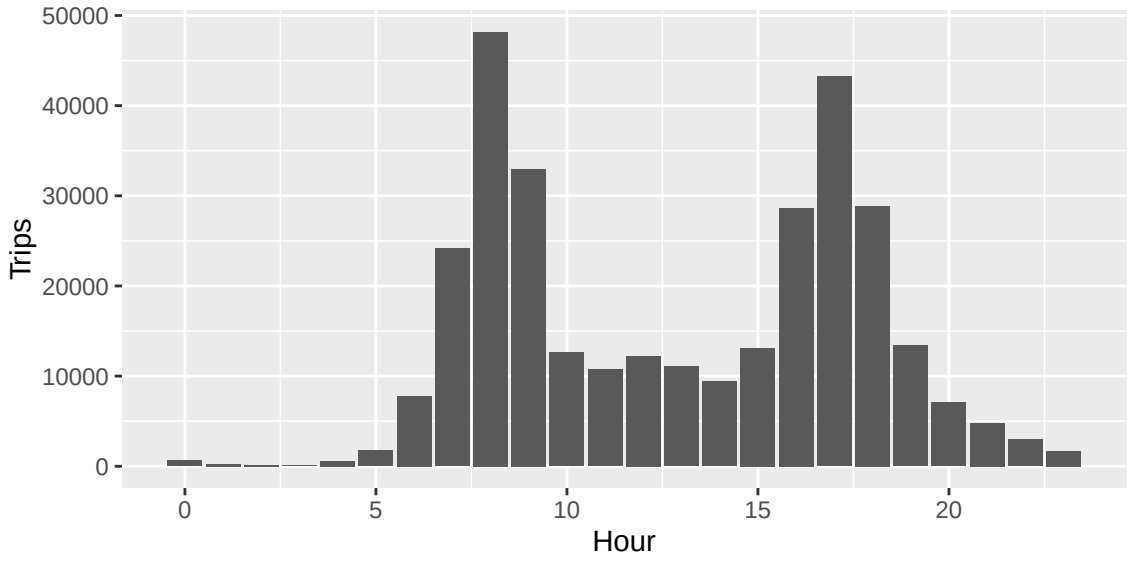
```
trip_numbers <- trip  %>%
  group_by(Start.Station, End.Station)  %>%
  summarise(trips = n())

myPalette <- c("SeaGreen", "Sienna", "red", "MediumVioletRed", "blue")
names(myPalette) <- levels(station2$landmark)

ggplot(trip_numbers, aes(y=Start.Station,
                         x=End.Station))+
    geom_tile(aes(fill = trips))+
    scale_fill_gradient2(low ="yellow", mid = "grey", high = "black")+
  theme_bw()+
  theme(text = element_text(size=7),
        axis.text.x = element_text(angle=90, vjust=1))+
  scale_x_discrete(limits=stationLabels,
                   labels = stationidsx)+
  scale_y_discrete(limits=stationLabels,
                   labels = stationidsy)+
  labs(x="End station",
       y= "Start station",
       fill = "Trips",
       title = "Heat map of trips")+
  theme(axis.text.x = element_text(colour=myPalette[station2$landmark]),
        axis.text.y = element_text(colour=myPalette[station2$landmark]),
        plot.title = element_text(size=20))
```



2.3.1 Weekday trips

- During the week trips are commonly made during commuting hours.

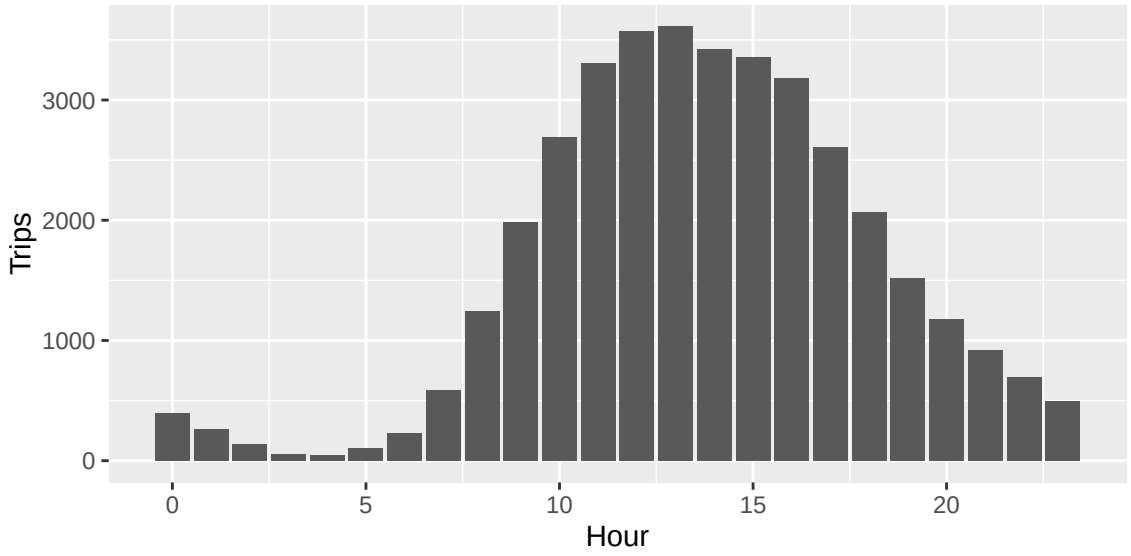
```
trip_per_hour <- trip %>%
  mutate(Hour = hour(Start_trip))

trip_per_hourWeek <- trip_per_hour %>% filter(Weekday != "Saturday") %>%
  filter(Weekday != "Sunday")

ggplot(trip_per_hourWeek, aes(x = Hour))+
  geom_bar()+
  labs(x= "Hour", y="Trips")
```



2.3.2 Weekend trips

- During the weekend trips are made during midday.

```
trip_per_hourWeekend <- filter(trip_per_hour, Weekday == "Saturday" | Weekday == "Sunday")

ggplot(trip_per_hourWeekend, aes(x = Hour))+
  geom_bar()+
  labs(x= "Hour", y="Trips")
```



2.3.3 Visualisation

For a visualisation of the system in action on a typical weekday click **here.**

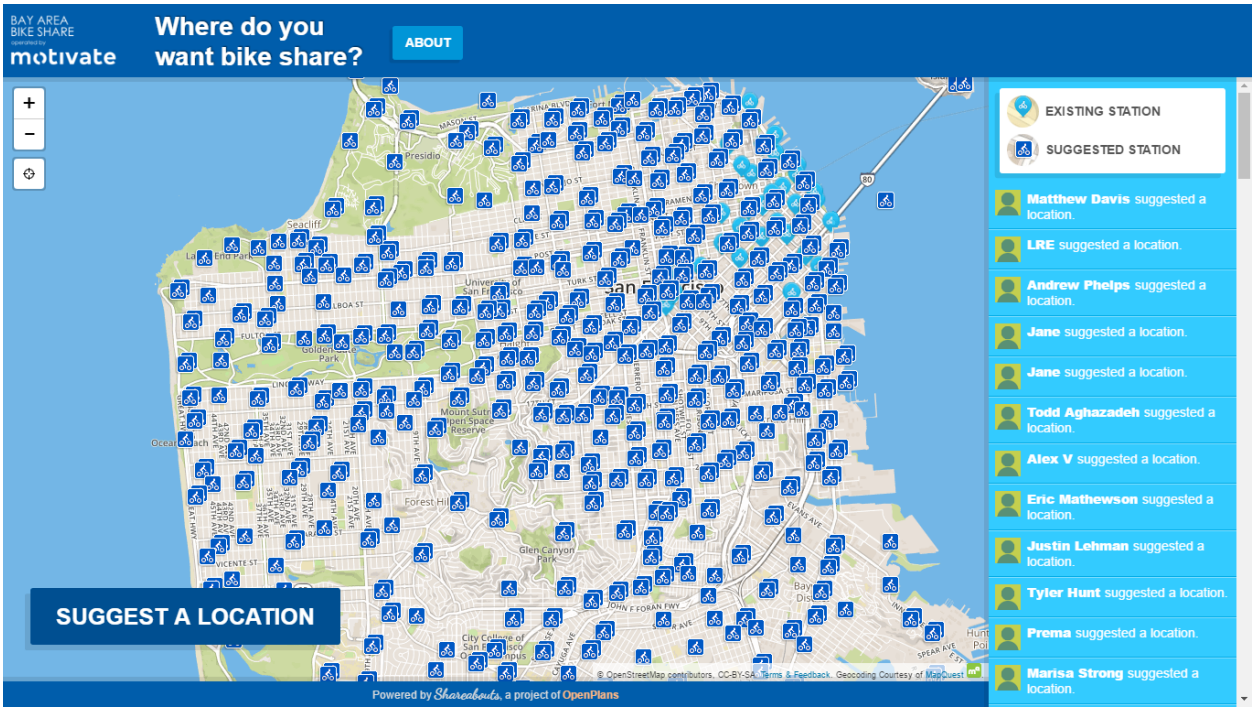


Figure 2: User suggestions for new station locations in San Francisco.

3 Expansion

As mentioned there is a planned expansion of the bike system. It will increase ten-fold to 7000 bikes with new stations. Where to place these new stations is a tricky question. Users can submit suggestions for locations online. Over 5000 suggestions have so far been made. See Figure 1 for the suggestions for San Francisco. It's of course great that people are so enthusiastic, suggesting plenty of locations for new bike stations, but it makes it a bit tough to choose! We'll look at the data in order to suggest some potential locations

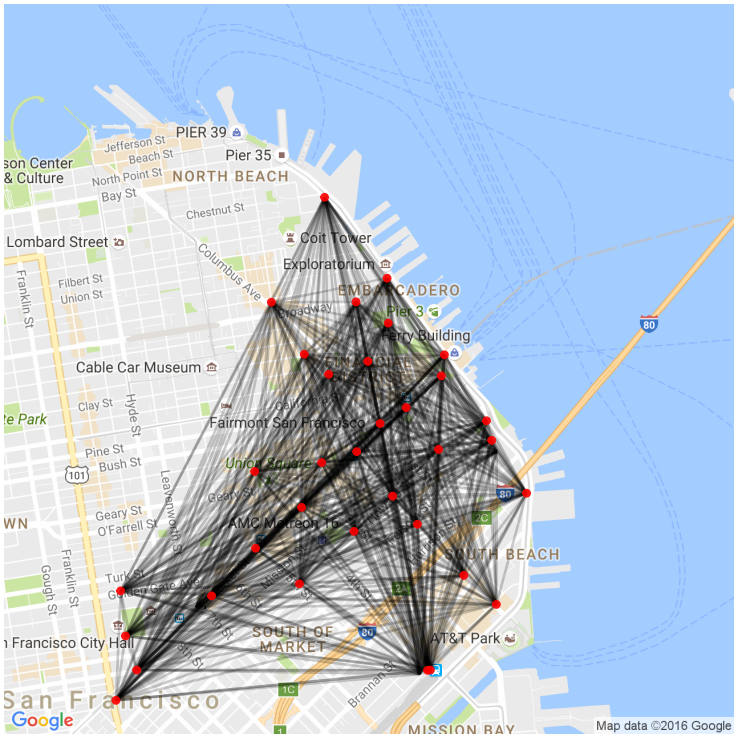
3.1 Busiest stations

Firstly we want to find out where the busiest stations are. A good tool for this comes from graph theory. We can use a graph ranking algorithm in order to sort the busiest stations. We firstly have to convert the bike sharing system into a *network*. In our network the stations are *nodes*. Two nodes are connected by an edge when a bike trip is made between them. Furthermore weights on the edges give the number of trips made. The result is a network of nodes and weighted edges.

```
tripNumbers <- trip %>%
  group_by(Start.Station,End.Station) %>%
  summarise(weight = n())  %>%
  ungroup() %>%
  mutate(weight = (weight - min(weight))/(max(weight)-min(weight)))
bikeGraph <- graph_from_data_frame(tripNumbers, directed=TRUE) %>%
set_vertex_attr("latitude", value = station$lat) %>%
set_vertex_attr("longitude", value = station$long) %>%
set_vertex_attr("city", value = station$landmark)
location <- c(-122.3999,37.7946)
```

```
BABSmap <- get_map(location,maptype="roadmap", zoom=14)

bayTrips <-trip %>%
  select(Start.Station, End.Station, start_lat, start_long,end_lat, end_long) %>%
  count(Start.Station, End.Station,start_lat, start_long, end_lat, end_long)

ggmap(BABSmap)+
  geom_segment(data = bayTrips,
               aes(y= start_lat,
                   x = start_long,
                   yend =  end_lat,
                   xend = end_long), alpha = 0.1)+
  theme(legend.position="none")+
  labs(title = "BABS as a network.")+
  geom_nodeset(aes(x=longitude,
                   y=latitude),
                bikeGraph,
               size=1,
               alpha =0.9,
               col="red")+
  theme(axis.line=element_blank(),
        axis.text.x=element_blank(),
          axis.text.y=element_blank(),
        axis.ticks=element_blank(),
          axis.title.x=element_blank(),
          axis.title.y=element_blank(),
          legend.position="none",
          panel.background=element_blank(),
          panel.border=element_blank(),
          panel.grid.major=element_blank(),
          panel.grid.minor=element_blank(),
          plot.background=element_blank())
```

BABS as a network.

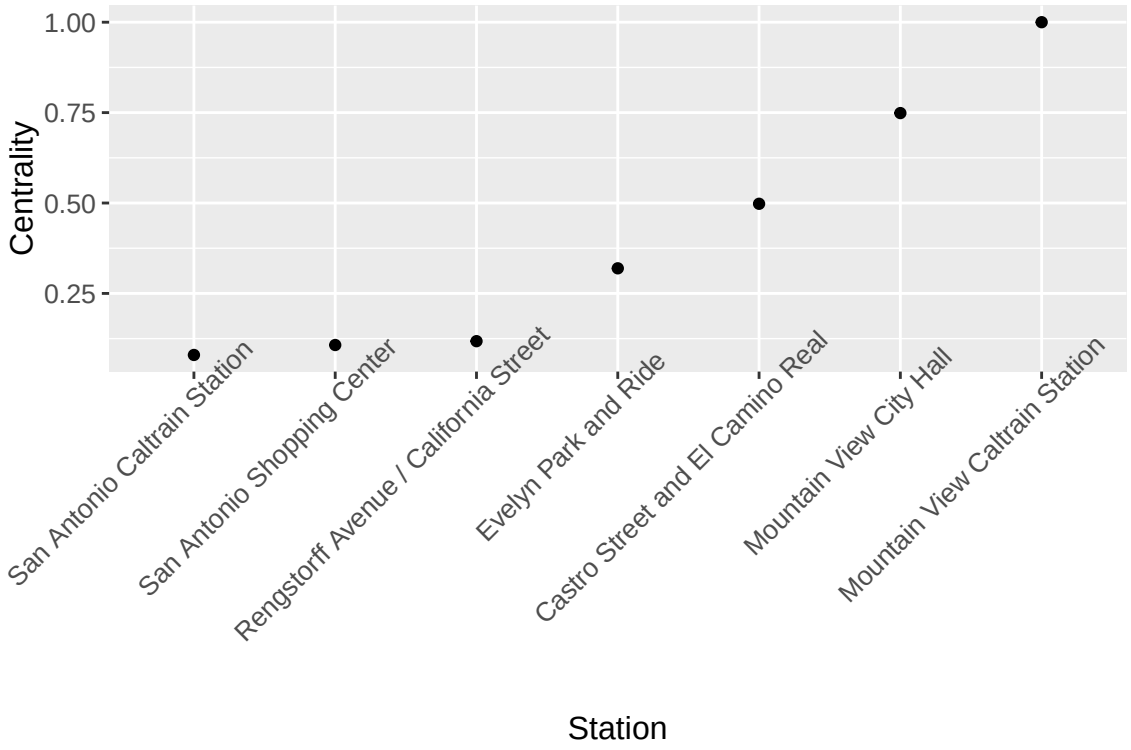


The graph ranking algorithm we use is *eigenvalue centrality*. This looks at each node and considers how many links it makes. In addition it takes into account the quality of the links made. That is if a node is connected to lots of highly connected nodes it is ranked higher than one that is connected to lots of low connected nodes. See here for more info. The higher the eigenvalue centrality the more important the station. We calculate it on a city by city basis.

3.1.1 Mountain View

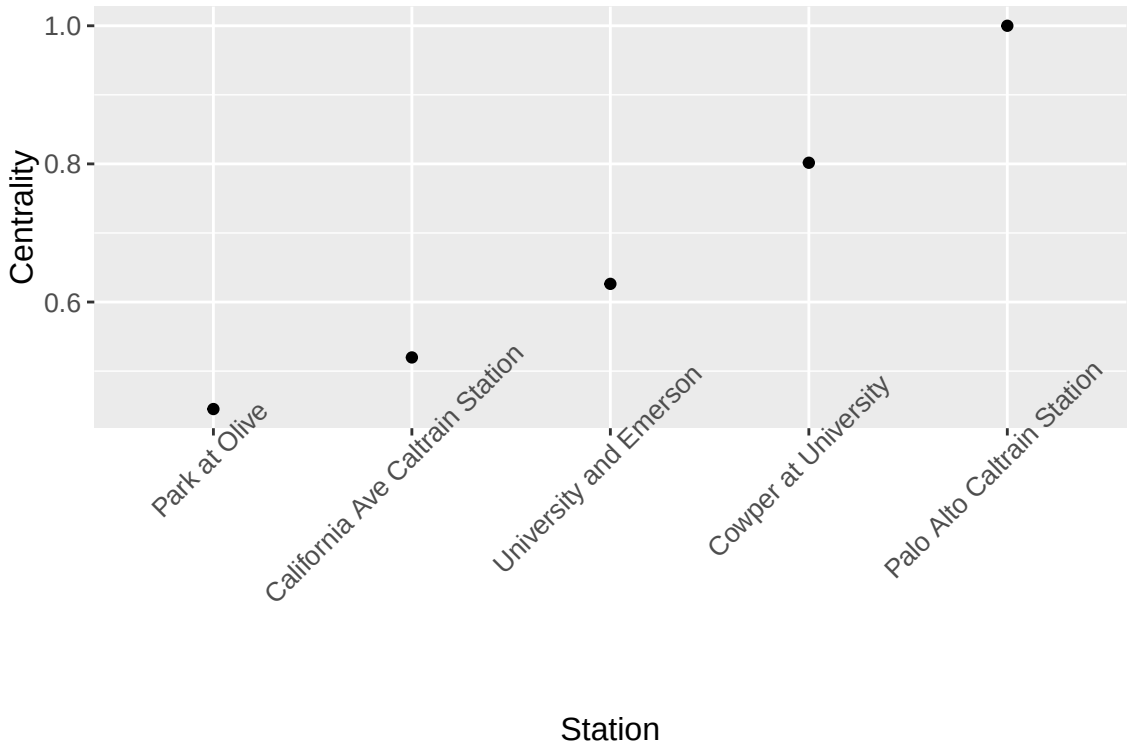
```
tripNumbers <- trip %>%
  filter(startLandmark == "Mountain View", endLandmark == "Mountain View") %>%
  group_by(Start.Station,End.Station) %>%
  summarise(weight = n())  %>%
  ungroup() %>%
  mutate(weight = (weight - min(weight))/(max(weight)-min(weight)))

bikeGraph <- graph_from_data_frame(tripNumbers, directed=TRUE)
bikeE <- eigen_centrality(bikeGraph, directed = TRUE)
bikeEigen <- data.frame(station = names(bikeE$vector), eig = bikeE$vector)
bikeEigen <- dplyr::tbl_df(bikeEigen)
bikeEigen <- mutate(bikeEigen, station = as.character(station))
bikeEigen <- arrange(bikeEigen, eig)
ggplot(bikeEigen, aes(x=station, y = eig))+
  geom_point()+
  theme(text = element_text(size=12),
        axis.text.x = element_text(angle=45, vjust=1, hjust = 0.8))+
```

```
  labs(x="Station", y= "Centrality")+
  scale_x_discrete(limits = bikeEigen$station)
```



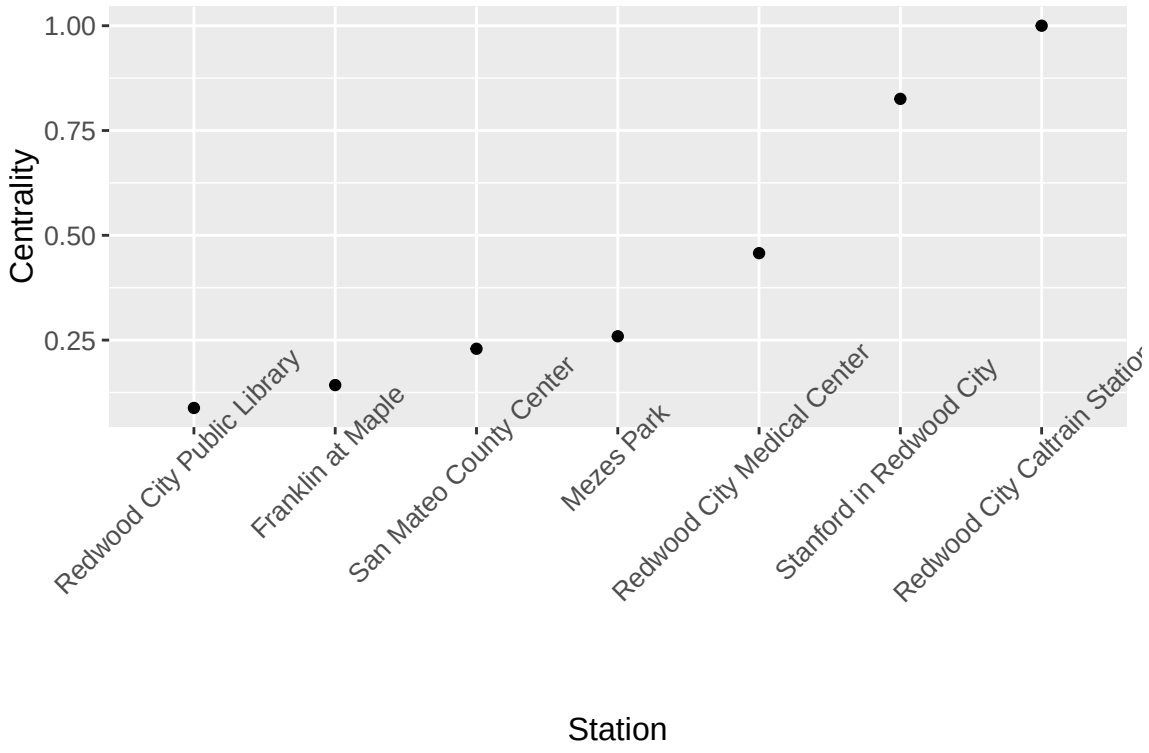
3.1.2 Palo Alto

```
tripNumbers <- trip %>%
  filter(startLandmark == "Palo Alto", endLandmark == "Palo Alto") %>%
  group_by(Start.Station,End.Station) %>%
  summarise(weight = n())  %>%
  ungroup() %>%
  mutate(weight = (weight - min(weight))/(max(weight)-min(weight)))

bikeGraph <- graph_from_data_frame(tripNumbers, directed=TRUE)
bikeE <- eigen_centrality(bikeGraph, directed = TRUE)
bikeEigen <- data.frame(station = names(bikeE$vector), eig = bikeE$vector)
bikeEigen <- dplyr::tbl_df(bikeEigen)
bikeEigen <- mutate(bikeEigen, station = as.character(station))
bikeEigen <- arrange(bikeEigen, eig)
ggplot(bikeEigen, aes(x=station, y = eig))+
  geom_point()+
  theme(text = element_text(size=12),
        axis.text.x = element_text(angle=45, vjust=1, hjust = 0.6))+
  labs(x="Station", y= "Centrality")+
  scale_x_discrete(limits = bikeEigen$station)
```



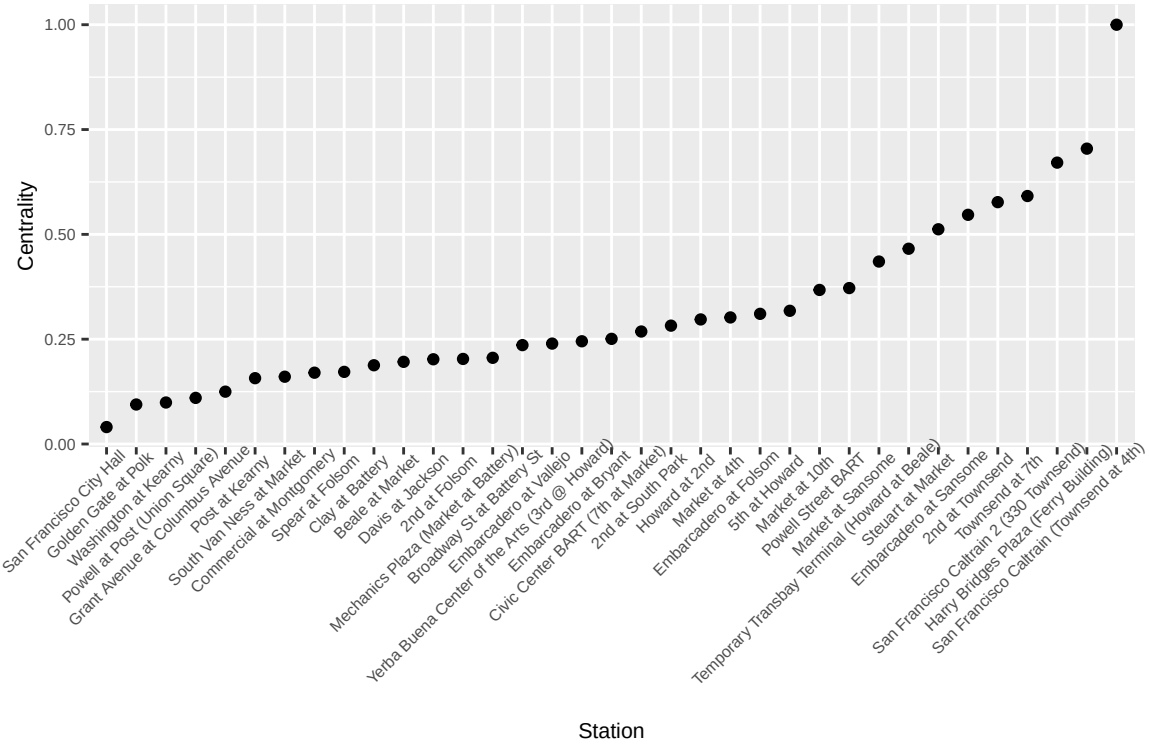
3.1.3 Redwood City

```
tripNumbers <- trip %>%
  filter(startLandmark == "Redwood City", endLandmark == "Redwood City") %>%
  group_by(Start.Station,End.Station) %>%
  summarise(weight = n())  %>%
  ungroup() %>%
  mutate(weight = (weight - min(weight))/(max(weight)-min(weight)))

bikeGraph <- graph_from_data_frame(tripNumbers, directed=TRUE)
bikeE <- eigen_centrality(bikeGraph, directed = TRUE)
bikeEigen <- data.frame(station = names(bikeE$vector), eig = bikeE$vector)
bikeEigen <- dplyr::tbl_df(bikeEigen)
bikeEigen <- mutate(bikeEigen, station = as.character(station))
bikeEigen <- arrange(bikeEigen, eig)
ggplot(bikeEigen, aes(x=station, y = eig))+
  geom_point()+
  theme(text = element_text(size=12),
        axis.text.x = element_text(angle=45, vjust=1,hjust = 0.6))+
  labs(x="Station", y= "Centrality")+
  scale_x_discrete(limits = bikeEigen$station)
```



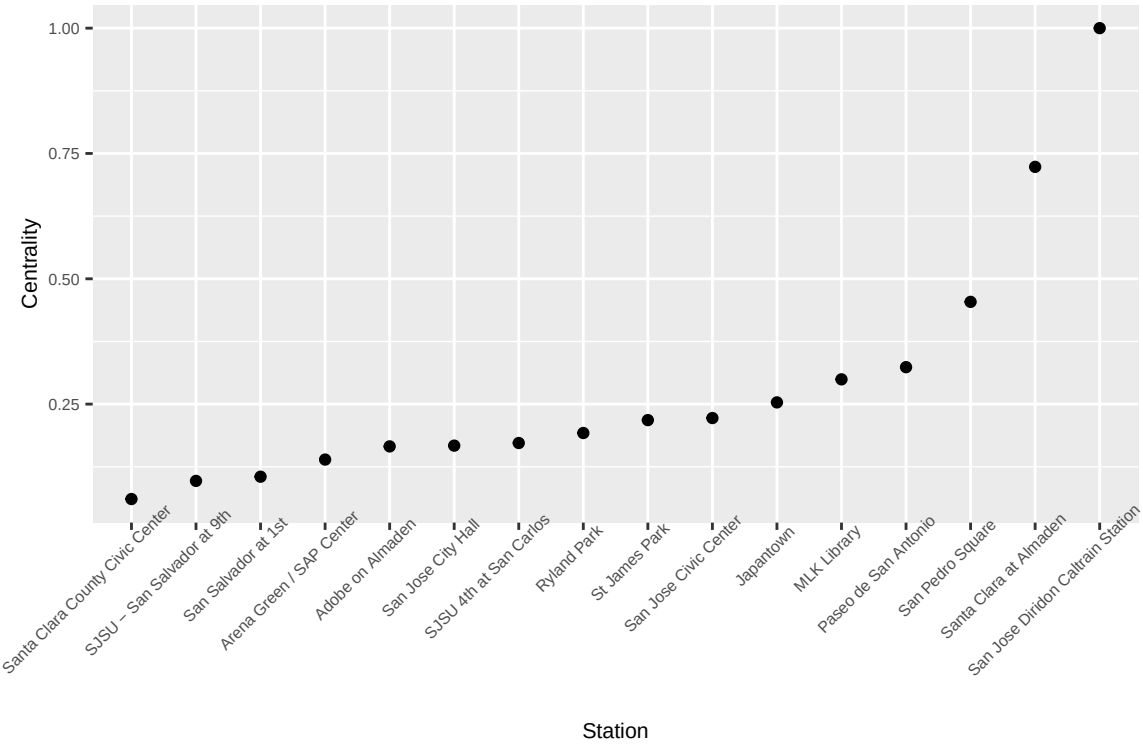
3.1.4 San Francisco

```
tripNumbers <- trip %>%
  filter(startLandmark == "San Francisco", endLandmark == "San Francisco") %>%
  group_by(Start.Station,End.Station) %>%
  summarise(weight = n())  %>%
  ungroup() %>%
  mutate(weight = (weight - min(weight))/(max(weight)-min(weight)))

bikeGraph <- graph_from_data_frame(tripNumbers, directed=TRUE)
bikeE <- eigen_centrality(bikeGraph, directed = TRUE)
bikeEigen <- data.frame(station = names(bikeE$vector), eig = bikeE$vector)
bikeEigen <- dplyr::tbl_df(bikeEigen)
bikeEigen <- mutate(bikeEigen, station = as.character(station))
bikeEigen <- arrange(bikeEigen, eig)
ggplot(bikeEigen, aes(x=station, y = eig))+
  geom_point()+
  theme(text = element_text(size=8),
        axis.text.x = element_text(angle=45, vjust=1, hjust = 0.9))+
  labs(x="Station", y= "Centrality")+
  scale_x_discrete(limits = bikeEigen$station)
```



3.1.5 San Jose

```
tripNumbers <- trip %>%
  filter(startLandmark == "San Jose", endLandmark == "San Jose") %>%
  group_by(Start.Station,End.Station) %>%
  summarise(weight = n())  %>%
  ungroup() %>%
  mutate(weight = (weight - min(weight))/(max(weight)-min(weight)))

bikeGraph <- graph_from_data_frame(tripNumbers, directed=TRUE)
bikeE <- eigen_centrality(bikeGraph, directed = TRUE)
bikeEigen <- data.frame(station = names(bikeE$vector), eig = bikeE$vector)
bikeEigen <- dplyr::tbl_df(bikeEigen)
bikeEigen <- mutate(bikeEigen, station = as.character(station))
bikeEigen <- arrange(bikeEigen, eig)
ggplot(bikeEigen, aes(x=station, y = eig))+
  geom_point()+
  theme(text = element_text(size=8),
        axis.text.x = element_text(angle=45, vjust=1, hjust = 0.8))+
  labs(x="Station", y= "Centrality")+
  scale_x_discrete(limits = bikeEigen$station)
```



Note the number one ranked stations in each city:

- **Mountain View**: *Mountain View Caltrain station.*
- **Palo Alto**: *Palo Alto Caltrain station.*
- **Redwood City**: *Redwood City Caltrain station.*
- **San Francisco**: *San Francisco Caltrain (Townsend at 4th).*
- **San Jose**: *San Jose Diridon Caltrain station.*

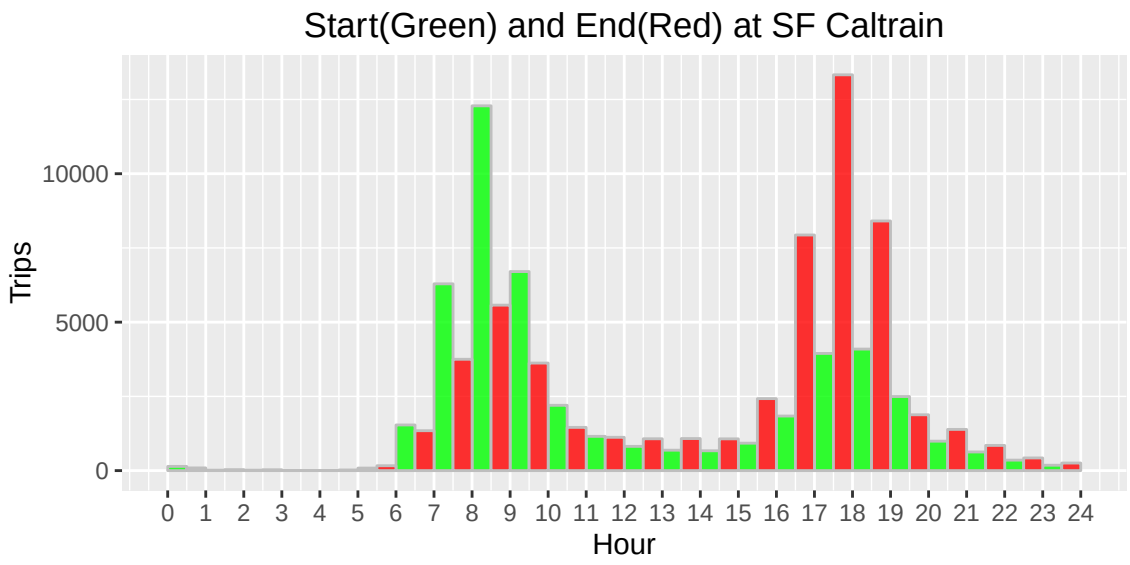
We see that they are all at Caltrain stations. This indicates that commuters are using the bikes on their trips to work. Also note that the least busy station in San Francisco is at the CIty Hall, perhaps the workers there could lead by example...

3.2 Commutes to San Francisco Caltrain

```
SFCT <- trip %>%
  filter(Start.Station ==
           "San Francisco Caltrain 2 (330 Townsend)" |
           Start.Station ==
           "San Francisco Caltrain (Townsend at 4th)") %>%
  group_by(Subscriber.Type) %>%
  summarise(count = n())
```

There are two stations located at the San Francisco Caltrain station, 95 per cent of usage here is from subscribers. We see that bikes are generally picked up in the morning, and dropped off in the evening, presumably by commuters.

```
trip_per_hour_start <- trip %>%
  filter(Start.Station ==
           "San Francisco Caltrain 2 (330 Townsend)" |
           Start.Station ==
           "San Francisco Caltrain (Townsend at 4th)") %>%
  mutate(sHour = hour(Start_trip))

trip_per_hour_end <- trip %>%
  filter(End.Station ==
           "San Francisco Caltrain 2 (330 Townsend)" |
           End.Station ==
           "San Francisco Caltrain (Townsend at 4th)") %>%
  mutate(eHour = hour(End_trip))

ggplot(trip_per_hour_start, aes(x = sHour))+
geom_bar(col = 'grey', fill = "green", alpha = 0.8,
         width = 0.5, position = position_nudge(x = 0.25))+
geom_bar(data = trip_per_hour_end, aes(x=eHour),
         col = 'grey', fill = "red", alpha = 0.8, width = 0.5,
         position = position_nudge(x = 0.75))+
labs(x= "Hour",
     y="Trips",
     title = "Start(Green) and End(Red) at SF Caltrain")+
  scale_x_continuous(breaks =seq(0,24,1))
```



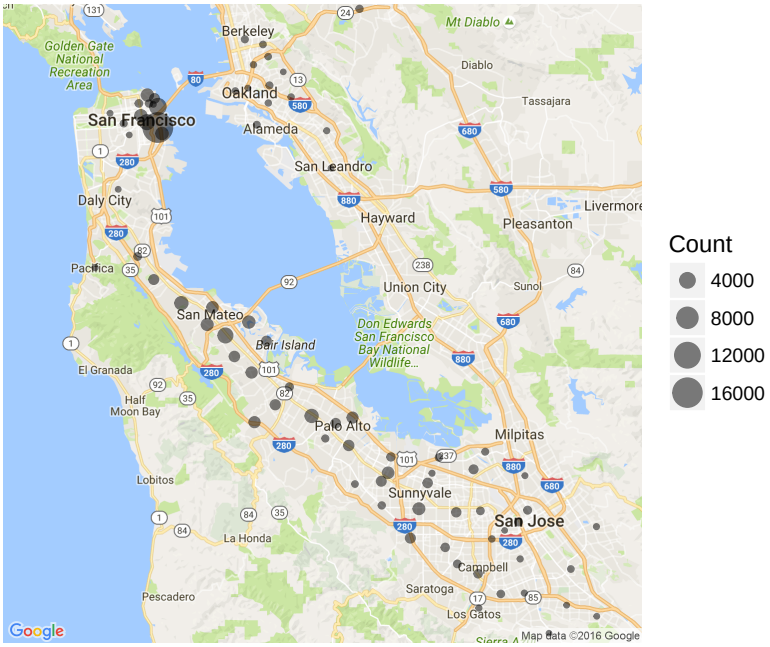

```
data("zipcode")
zips <- trip %>%
  filter(Start.Station ==
           "San Francisco Caltrain 2 (330 Townsend)" |
           Start.Station == "San Francisco Caltrain (Townsend at 4th)"|
           End.Station == "San Francisco Caltrain 2 (330 Townsend)" |
           End.Station == "San Francisco Caltrain (Townsend at 4th)") %>%
  group_by(Zip.Code) %>%
  summarise(count = n()) %>%
```

```
  ungroup() %>%
  mutate(zip = clean.zipcodes(Zip.Code)) %>%
  merge(zipcode, by.x='zip', by.y='zip') %>%
  arrange(desc(count))
tp <- 80
zipsShort <- slice(zips, 1:tp)
```

Looking at the zip codes of these users we see that they are commuting from 780 zip codes. Note that the 80 most common zip codes account for 92 per cent of usage. Visualising these we see that lots of people who live along the caltrain line commute in to San Francisco and use BABS as part of their trip to work:

```
location <- c( mean(zipsShort$longitude), mean(zipsShort$latitude))
BABSmap <- get_map(location,maptype="roadmap", zoom=10)
ggmap(BABSmap)+
  geom_point(data = zipsShort, aes(x=longitude, y=latitude, size = count), alpha = 0.5)+
  labs(x = "longitude",
       y = "latitude",
       size = "Count",
       title = "Zip codes of users at SF Caltrain station.")+
  scale_size_area()+
  theme(axis.line=element_blank(),
        axis.text.x=element_blank(),
          axis.text.y=element_blank(),
          axis.ticks=element_blank(),
          axis.title.x=element_blank(),
          axis.title.y=element_blank(),
          panel.background=element_blank(),
          panel.border=element_blank(),
          panel.grid.major=element_blank(),
          panel.grid.minor=element_blank(),
          plot.background=element_blank())
```

Zip codes of users at SF Caltrain station.



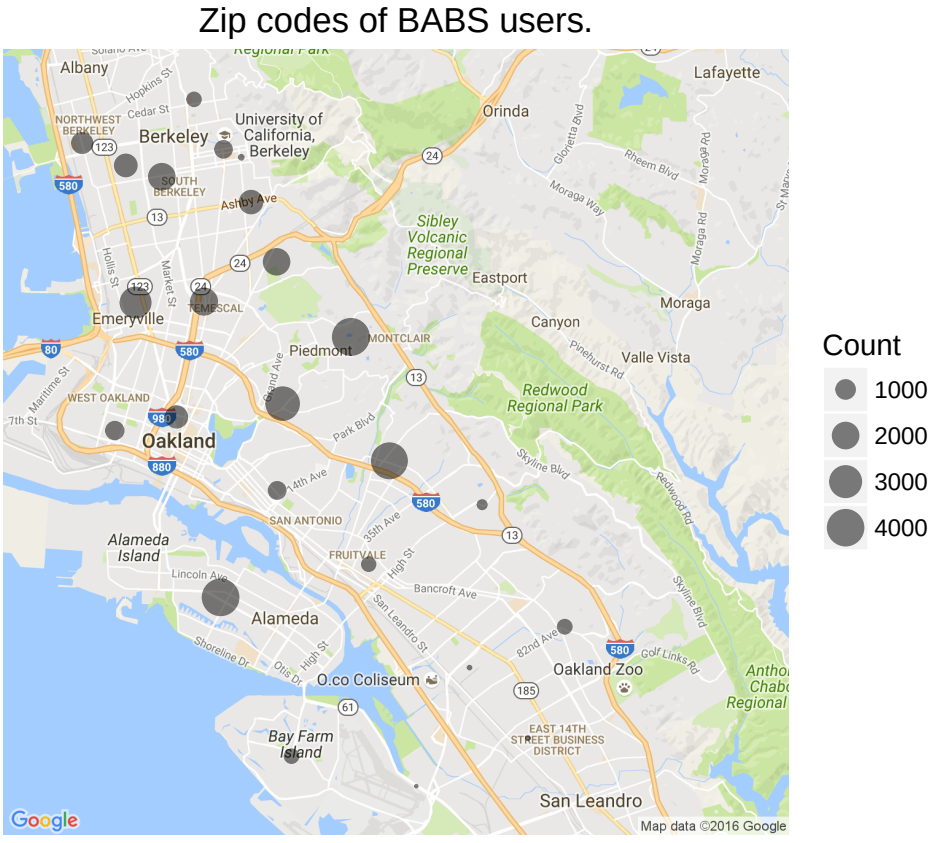
3.3 Expansion to the north

```
zipsNorth <- trip %>%
  group_by(Zip.Code) %>%
  summarise(count = n()) %>%
  ungroup() %>%
  mutate(zip = clean.zipcodes(Zip.Code)) %>%
  merge(zipcode, by.x='zip', by.y='zip') %>%
  filter(state == "CA") %>%
  filter(city ==
           "Oakland"|
           city == "Berkeley"|
           city == "Emeryville"|
           city =="Alameda") %>%
  arrange(desc(count))
```

Part of the planned expansion of the system is to build stations to the north in the cities of Oakland, Emeryville and Berkeley. From the usage pattern we have seen of commuters a good idea would be to Integrate these new stations with the Bay area rapid transit stations and other public transport that exists in the area. This will be popular with commuters to San Francisco and will encourage people to use the bikes. There were already 35910 trips made from users who live in this area.

```
location <- c( mean(zipsNorth$longitude), mean(zipsNorth$latitude))
BABSmap <- get_map(location,maptype="roadmap", zoom=12)
ggmap(BABSmap)+
  geom_point(data = zipsNorth,
             aes(x=longitude,
                 y=latitude,
```

```
                size = count),
            alpha = 0.5)+
labs(x = "longitude",
     y = "latitude",
     size = "Count",
     title = "Zip codes of BABS users.")+
        scale_size_area()+
        theme(axis.line=element_blank(),
              axis.text.x=element_blank(),
              axis.text.y=element_blank(),
              axis.ticks=element_blank(),
              axis.title.x=element_blank(),
              axis.title.y=element_blank(),
              panel.background=element_blank(),
              panel.border=element_blank(),
              panel.grid.major=element_blank(),
              panel.grid.minor=element_blank(),
              plot.background=element_blank())
```



4 San Francisco open data

In order to get an idea of where cycling is popular in general we will analyse an external data source not associated with the BABS.

4.1 Bike usage statistics

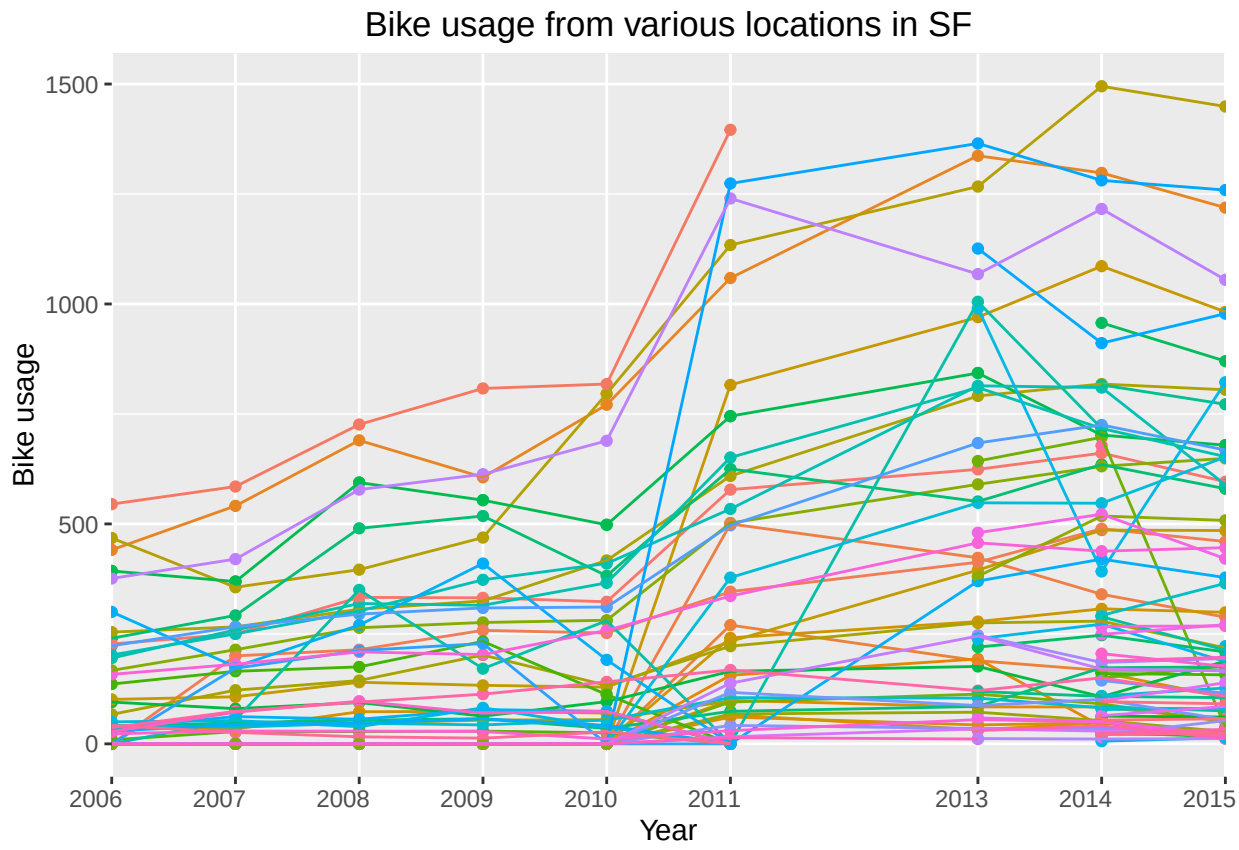
The City and County of San Francisco has measured bicycle usage by performing manual counts at various locations around the city. They have records going back to 2006. The data is freely available here. We can use it to get an idea of where people are using bikes (not necessarily BABS bikes) throughout the city.

4.2 Bike usage increasing

We first see that bike usage has generally increased throughout the city which is good news for the expansion. Each line in the figure represents a location throughout the city.

```
bikeCounts <-read_csv("Bike_Volume_Manual_Counts.csv")
bikeCount15 <-read.csv("2015bikecount.txt", sep = " ")
bikeCounts <- dplyr::tbl_df(bikeCounts)
bikeCount15 <- dplyr::tbl_df(bikeCount15)

names(bikeCounts) <-
  make.names(names(bikeCounts), unique=TRUE)
names(bikeCount15) <-
  c("ID","Locations","X2014","Bike.Count.2015.Afternoon"  )

bikeCounts <- arrange(bikeCounts, Location.ID)
bikeCount15 <- arrange(bikeCount15, ID)
bikeCounts <-
  left_join(bikeCounts, bikeCount15,
            by = c("Location.ID" = "ID"))

bikeCounts <- bikeCounts %>%
  separate(Geom, c("latitude", "longitude"), sep = ", ") %>%
  mutate(latitude =
           as.numeric(gsub("\\(|\\)", "",latitude)),
         longitude =
           as.numeric(gsub("\\(|\\)", "",longitude)))
names(bikeCounts) <- make.names(names(bikeCounts), unique=TRUE)
bikeCounts <- select(bikeCounts,c(2,3,5,6,7,8,9,10,11,12,22,18,19))
bikeCounts <- bikeCounts %>%
  mutate(Bike.Count.2006.Afternoon =
           ifelse(Bike.Count.2006.Afternoon<0, NA,Bike.Count.2006.Afternoon)) %>%
  mutate(Bike.Count.2007.Afternoon =
           ifelse(Bike.Count.2007.Afternoon<0, NA,Bike.Count.2007.Afternoon)) %>%
  mutate(Bike.Count.2008.Afternoon =
           ifelse(Bike.Count.2008.Afternoon<0, NA,Bike.Count.2008.Afternoon)) %>%
  mutate(Bike.Count.2009.Afternoon =
           ifelse(Bike.Count.2009.Afternoon<0, NA,Bike.Count.2009.Afternoon)) %>%
  mutate(Bike.Count.2010.Afternoon =
           ifelse(Bike.Count.2010.Afternoon<0, NA,Bike.Count.2010.Afternoon)) %>%
  mutate(Bike.Count.2011.Afternoon =
           ifelse(Bike.Count.2011.Afternoon<0, NA,Bike.Count.2011.Afternoon)) %>%
  mutate(Bike.Count.2013.Afternoon =
           ifelse(Bike.Count.2013.Afternoon<0, NA,Bike.Count.2013.Afternoon)) %>%
  mutate(Bike.Count.2014.Afternoon =
           ifelse(Bike.Count.2014.Afternoon<0, NA,Bike.Count.2014.Afternoon))
```

```r
bikeCountsNew <- bikeCounts %>%
  gather(Year, value, 3:11 ) %>%
  mutate(Year = extract_numeric(gsub("\\.", "",Year)))

ggplot(bikeCountsNew, aes(x=Year, y = value, col = Location))+
  geom_point()+
  geom_line()+
  theme(legend.position="none")+
  labs(x="Year",
       y= "Bike usage",
       title = "Bike usage from various locations in SF") +
  scale_x_discrete(limits = unique(bikeCountsNew$Year))+
  theme(axis.text.x = element_text(hjust = 0.8))
```

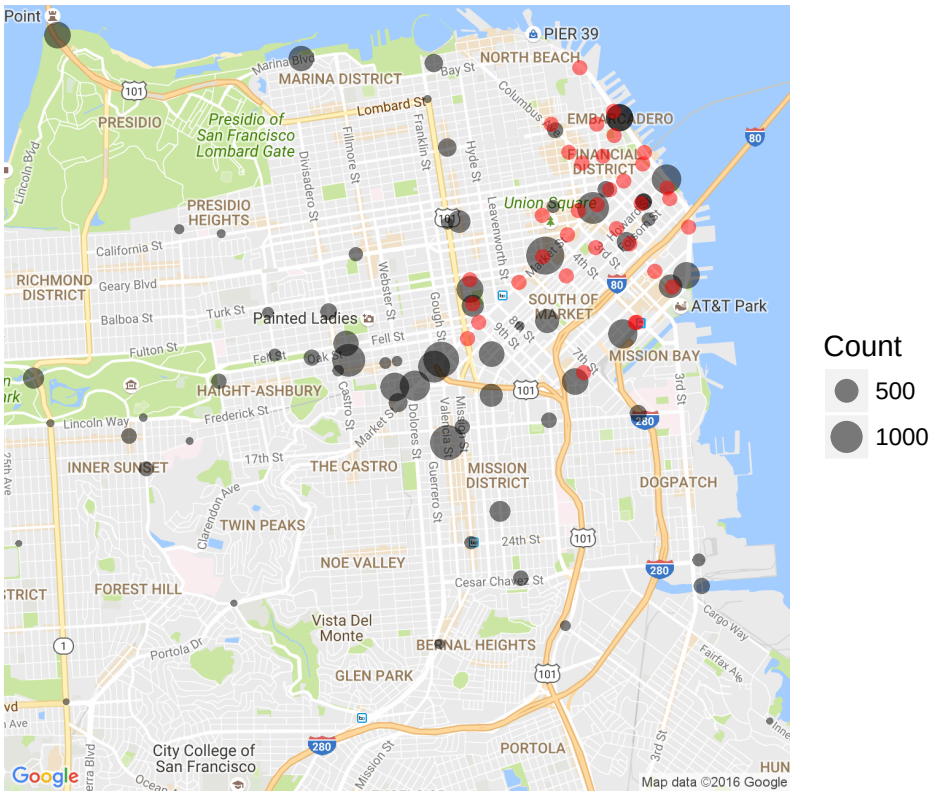


4.3 Where are people cycling?

We can also look at the particular locations to see where is popular.

```
location <- c( mean(bikeCountsNew$longitude, na.rm = TRUE),
               mean(bikeCountsNew$latitude,na.rm = TRUE))
BABSmap <- get_map(location,maptype ="roadmap", zoom=13)

ggmap(BABSmap)+
  geom_point(data = filter(bikeCountsNew, Year == 2015),
             aes(x=longitude, y=latitude, size = value), alpha = 0.5)+
  labs(size ="Count",
       title = "Manual bike count stats 2015 (BABS stations in red).")+
  scale_size_area()+
  geom_point(data = station, aes(x=long, y=lat),
             size =2,col = "red", alpha = 0.5)+
          theme(axis.line=element_blank(),
                axis.text.x=element_blank(),
          axis.text.y=element_blank(),
          axis.ticks=element_blank(),
          axis.title.x=element_blank(),
          axis.title.y=element_blank(),
          panel.background=element_blank(),
          panel.border=element_blank(),
          panel.grid.major=element_blank(),
          panel.grid.minor=element_blank(),
          plot.background=element_blank())
```

Manual bike count stats 2015 (BABS stations in red).



We see that cycling is common in Marina District, Lower Haight and Mission District. Cycling over the Golden gate bridge seems to be popular too. In fact according to San Francisco wikitravel:

A very popular ride for visitors to San Francisco is the ride across the Golden Gate Bridge to Sausalito.

Bike sations located at either ends of the bridge would seemingly be popular. We see that other recreational areas like the park are also popular with cyclists. We can look at TripAdvisor to see the locations of popular tourist hotspots. In Figure 2 we see these locations and we note that many of the popular tourist hotspots are not catered for by BABS. Expansion into these areas may encourage more leisure, weekend use.

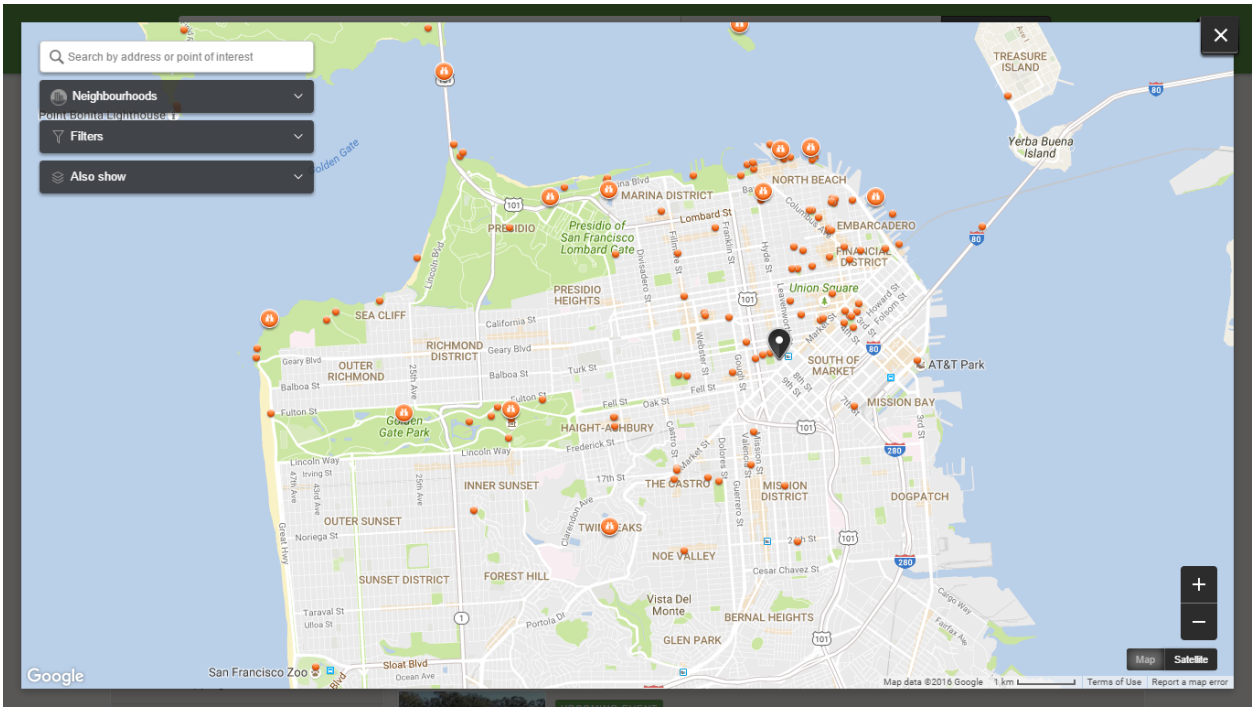


Figure 3: Tourist hotspots in San Francisco from TripAdvisor.

5 Imbalanced stations

5.1 Problem

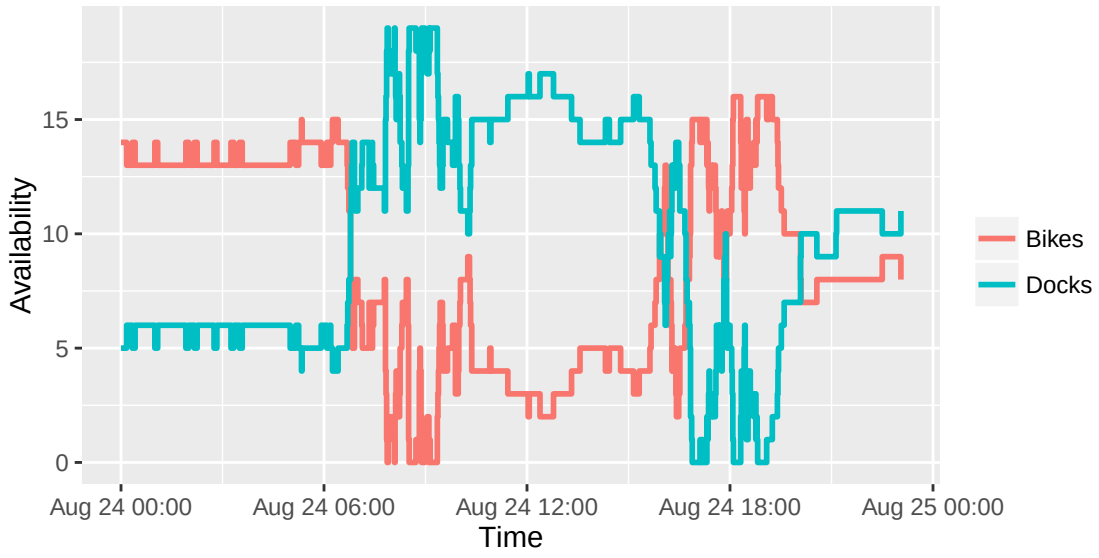
There are two main problems that users of a bike sharing system face: Firstly, there may be no bikes available when they arrive at a station. This could result in being late for work or an appointment. Secondly the station may be full meaning there is no space to deposit a bike. Similarly this can result in frustration for the user. In general if more cycles are hired than deposited from a station then the station will require manual re-stocking. If less cycles are hired than deposited the station fills up and some bikes will need to be removed.

5.1.1 Availability at San Francisco Caltrain (Townsend at 4th)

We can look at the San Francisco Caltrain station. We see that during peak hours it can be a problem that there are no bikes when they're needed in the morning and no space to deposit when it's needed in the evening.

```
hires <- trip  %>%
  group_by(Start.Station) %>%
  summarise(meanHires = round(n()/365))
deposits <- trip %>%
  group_by(End.Station) %>%
  summarise(meanDeposits = round(n()/365))

inOut <- left_join(hires, deposits,
                   by = c("Start.Station" = "End.Station"))
inOut <- inOut %>%
  mutate(diff = meanDeposits-meanHires) %>%
```

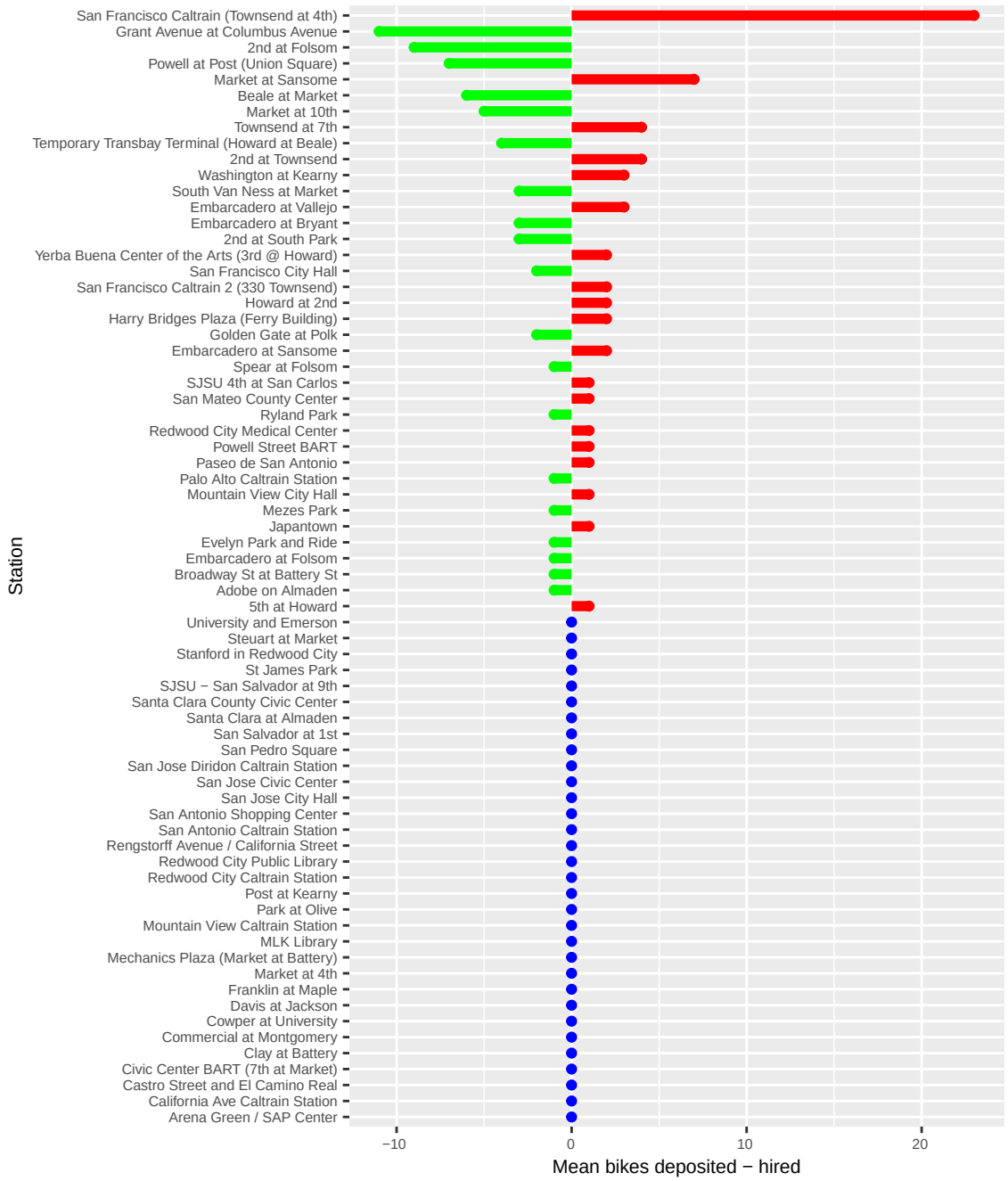
```
  mutate(absDiff = abs(diff),
         loss =
           ifelse(diff > 0 , "+",
           ifelse(diff< 0, "-", 0))) %>%
  arrange(desc(absDiff))

names(inOut) <-
  c("Station",
    "meanHires",
    "meanDeposits",
    "diff",
    "absDiff",
    "difference")

inOut <- left_join(inOut, station, by = c("Station" = "name"))

inOutSF <- filter(inOut, landmark == "San Francisco")

skewedStations <- slice(inOut,1:1)
status_sf_select <- status %>%
  filter(name  %in% skewedStations$Station) %>%
  filter(
    between(Date,
            as.Date("2015-08-24"),
            as.Date("2015-08-24"))) %>%
  select(time, bikes_available, docks_available)

names(status_sf_select) <-
  c("Time", "Bikes", "Docks")

stat_long <-  melt(status_sf_select, id="Time")
ggplot(stat_long, aes(x=Time, y=value, colour=variable))+
  geom_step(size=1)+
  labs(x= "Time", y= "Availability", colour ="")
```



5.2 Daily imbalance

To quantify this we calculate the daily average imbalance at each station throughout the system. This is the daily average deposits minus hires at each station.

```
inOut <- arrange(inOut, absDiff)

ggplot(inOut, aes(x = Station,
                  y = diff,
                  col = as.factor(difference)))+
  geom_point()+
  geom_segment(aes(x=Station,
                   xend = Station,
                   y=0,
                   yend = diff),
               size = 2)+
  theme(text = element_text(size=8),
        legend.position="none",
        axis.text.x = element_text(angle=90,
                                   vjust=0,hjust = 0.6))+
  labs(x="Station", y= "Mean bikes deposited - hired")+
  scale_x_discrete(limits = inOut$Station)+
  coord_flip()+
  theme(axis.text.x = element_text(angle=0))+
  scale_colour_manual(
    values = c("green", "red", "blue"))
```

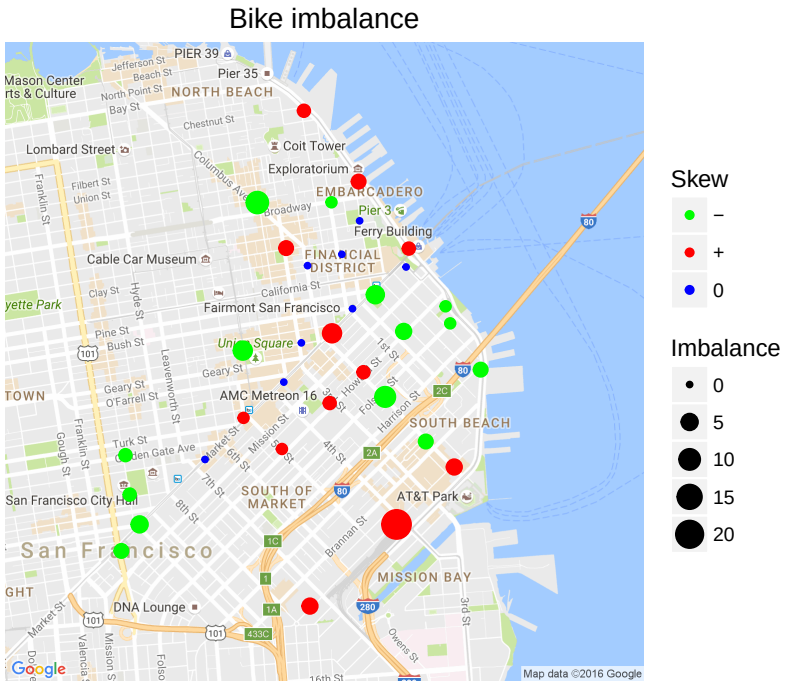


5.3 Imbalance in San Francisco

Wes see that the imbalanced stations are mainly located in the busy San Francisco.

```
location <- c( mean(inOutSF$long), mean(inOutSF$lat))
BABSmap <- get_map(location,maptype ="roadmap", zoom=14)

ggmap(BABSmap)+
  geom_point(data = inOutSF,
             aes(x=long,
```

```
                y=lat,
                size = absDiff,
                col = as.factor(difference)))+
labs(x = "longitude",
     y = "latitude",
     size ="Imbalance",
     col ="Skew",
     title = "Bike imbalance")+
        theme(axis.line=element_blank(),
              axis.text.x=element_blank(),
        axis.text.y=element_blank(),
        axis.ticks=element_blank(),
        axis.title.x=element_blank(),
        axis.title.y=element_blank(),
        panel.background=element_blank(),
        panel.border=element_blank(),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        plot.background=element_blank())+
scale_colour_manual(values =
                      c("green", "red", "blue"))
```



One option to alleviate the problems caused by the imbalance in the stations is to simply increase the size of the station in terms of the number of docking stations.

6 Conclusion and recommendations

- The BABS system is popular with commuters.
- Peak usage is during typical commuting hours.
- Usage drops on rainy days and holidays.
- The busiest stations are located at transport hubs like Caltrain stations.
- Expansion to new cities should be along existing public transport infrastructure.
- Within San Francisco cycling is already popular in Marina District, along the 101, in Lower Haight and Mission District. Bike stations here will be welcome.
- Increasing the size of the most imbalanced stations will help mitigate any disruption to users caused by lack of space or bikes.
- Recreational areas like the parks are currently popular with cyclists but not catered for by BABS. Bike stations in these locations will promote the BABS as a leisure activity.
- Promotion to tourists as a travel option would help too: There is currently no mention of BABS on San Francisco wikitravel or Bay area wikitravel.
- There is some confusion about how the system works, some users don't seem to understand the 30 minute rule (or don't care!).