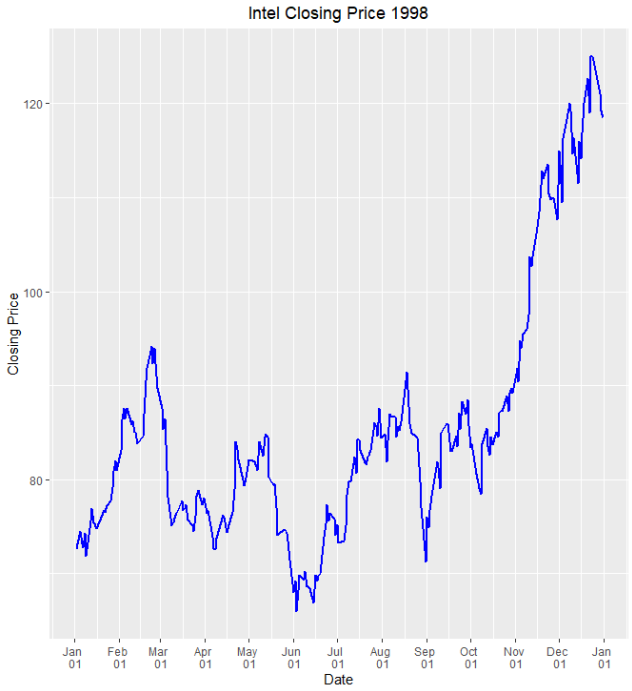
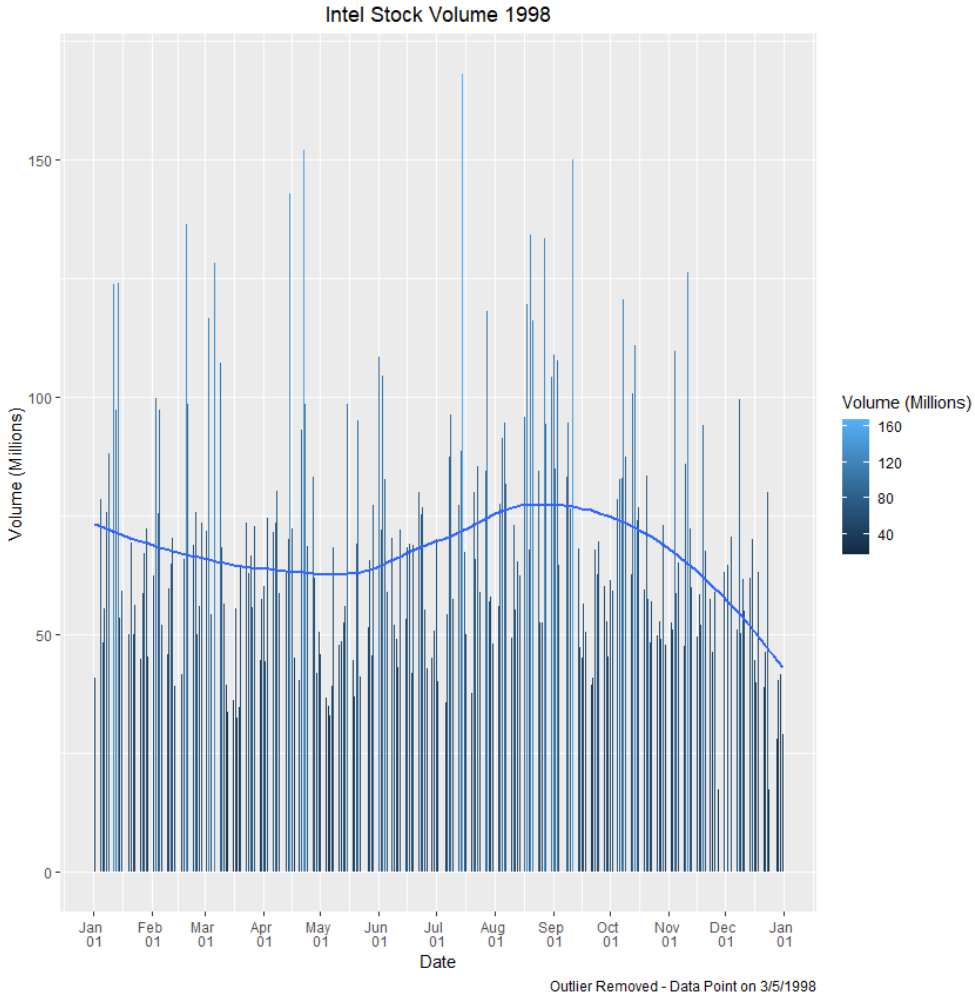
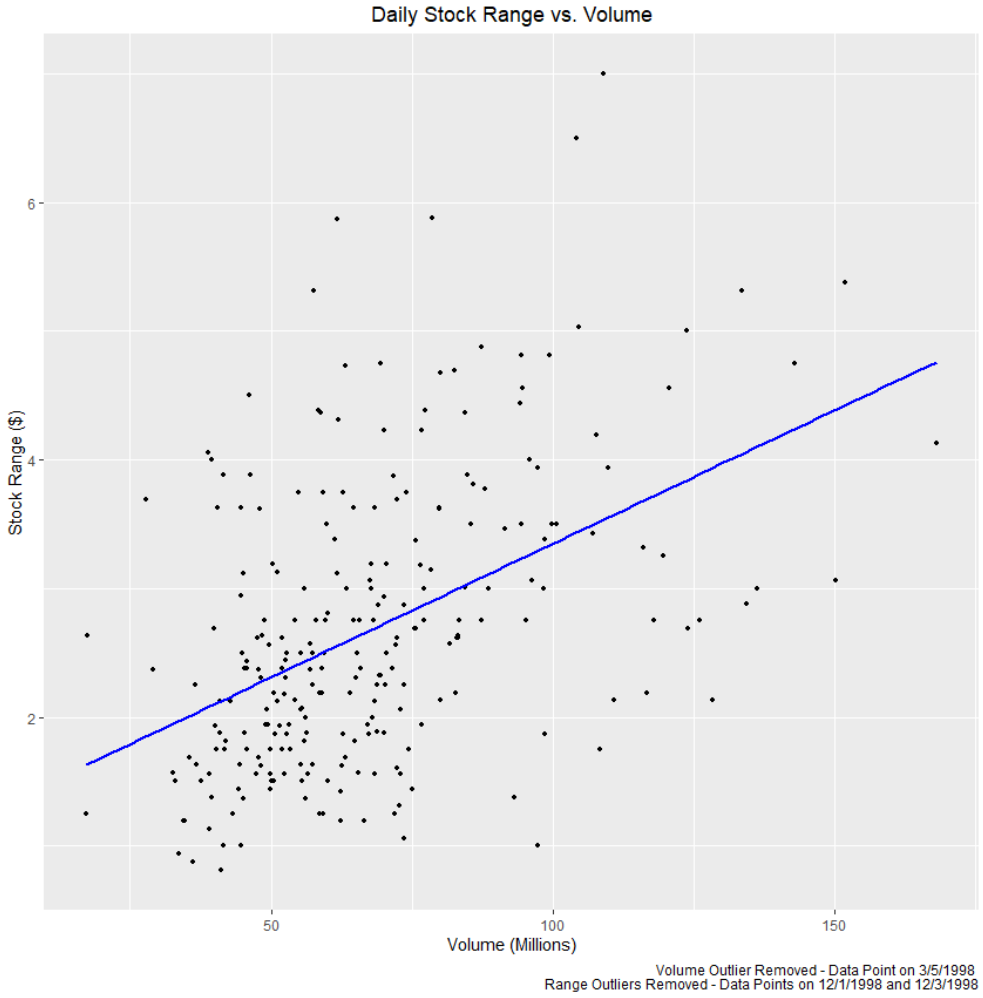
```
Intel=read.csv(“Intel-1998.csv”)
library(dplyr)
library(mosaic)
library(lubridate)
library(ggplot2)
ggplot(Intel, aes(x = Date, y = Close)) +  geom_line(color = "Blue",size = 1) + ggtitle("Intel Closing Price 1998") + xlab("Date") + ylab("Closing Price") + theme(plot.title = element_text(hjust = 0.5)) + scale_x_date(date_labels = "%b \n %d", date_breaks = "1 months")
```



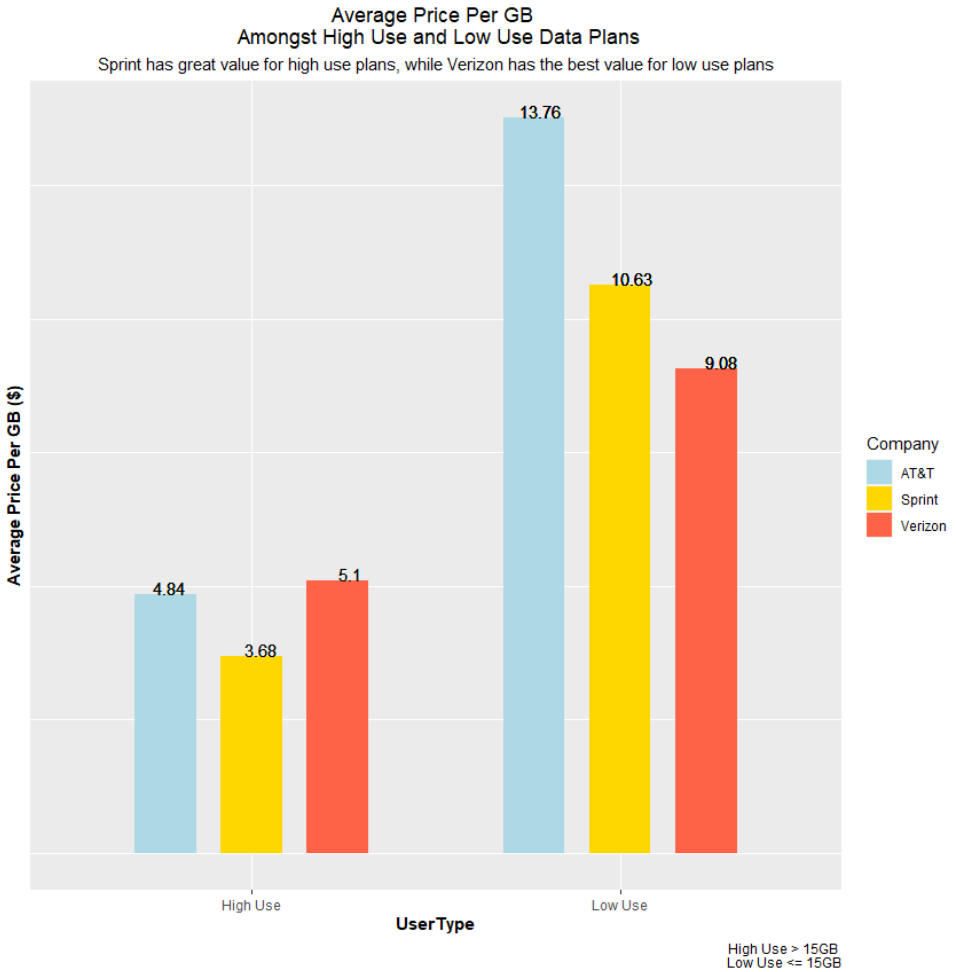
```
ggplot(Intel, aes(x = Date,y = Volume/1000000))+labs(x = "Date", y = "Volume (Millions)",color = "Volume (Millions)")+geom_segment(aes(xend = Date, yend = 0, color = Volume/1000000)) +geom_smooth(method = "loess",se = FALSE)+ggtitle("Intel Stock Volume 1998")+labs(caption = "Outlier Removed - Data Point on 3/5/1998")+ theme(plot.title = element_text(hjust = 0.5),plot.subtitle = element_text(hjust = 0.5))+scale_x_date(date_labels = "%b \n %d", date_breaks = "1 months")
```



```
ggplot(Intel, aes(x = Volume/1000000,y = Range))+geom_point(size=2, shape=20)+ geom_smooth(method = "lm",color = "blue",se = FALSE)+ggtitle("Daily Stock Range vs. Volume")+labs( caption = "Volume Outlier Removed - Data Point on 3/5/1998 \n Range Outliers Removed - Data Points on 12/1/1998 and 12/3/1998")+theme(plot.title = element_text(hjust = .5),plot.subtitle = element_text(hjust = .5))+labs(x = "Volume (Millions)", y = "Stock Range ($)")
```



```
cellPlans = data.frame(
  c("AT&T", "Sprint", "Verizon", "AT&T", "Sprint",
    "Verizon", "AT&T", "Sprint", "Verizon", "AT&T",
    "Verizon", "Sprint", "Verizon", "AT&T",
    "Verizon", "Sprint", "AT&T", "AT&T", "Sprint"),
  c(1, 1, 2, 3, 3, 4, 6, 6, 8, 10, 12, 12, 16, 16,
    24, 24, 25, 30, 40),
  c(30, 20, 35, 40, 30, 50, 60, 45, 70, 80, 80, 60,
    90, 90, 110, 80, 110, 135, 100))

names(cellPlans) = c("Company","DataGB","Price")

ggplot(cellPlans2,aes(fill = Company, x = UserType, y=AvgPricePerGB))+geom_bar(width =
.5,position=position_dodge(width = .7),stat ="identity")+theme_grey()+scale_fill_manual(values = c("AT&T" = "Light Blue",
"Sprint" = "Gold", "Verizon" = "Tomato"))+geom_text(aes(label = AvgPricePerGB,hjust=.2),vjust =
.0002,color="black",position = position_dodge(.75), size=4.0)+ggtitle("Average Price Per GB \n Amongst High Use and Low
Use Data Plans","Sprint has great value for high use plans, while Verizon has the best value for low use
plans")+theme(plot.title = element_text(hjust =.5),plot.subtitle = element_text(hjust =.5),axis.text.y =
element_blank(),axis.ticks.y = element_blank(), axis.title.x = element_text(face = "bold"),axis.title.y = element_text(face =
"bold"))+labs(y="Average Price Per GB ($)",caption = "High Use > 15GB \n Low Use <= 15GB")
```



An interesting metric that I wanted to explore for this question is the 10-year growth rate for food service in a given state. As a former analyst at food distribution firm, I found that it may be useful to show this information to companies the industry.

First, I wanted to aggregate the data by state. The result is below, for new edited data frame.

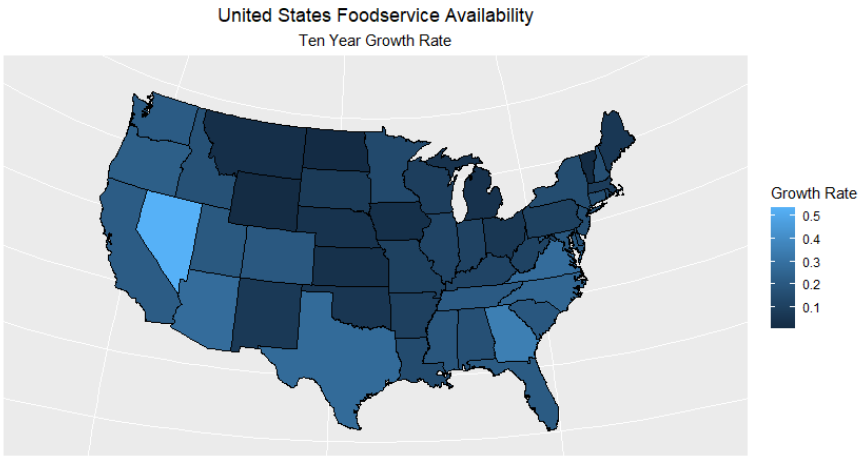
Then after running str(), I realized that “State” needed to be transformed to a character. After this, “State” needed to be spelled out in full and lower case match our state map brought in from the “maps” package (as used in the tutorial), an unneeded space needed to be taken out, and the first row containing all of the US, needed to be taken out (because we are looking at state by state). Finally, we needed to add in our 10-year growth rate variable and then left_join our information, so we could map.

```
states_map <- map_data("state")
food <- read.delim("FoodSrvcByCounty.txt")
library(tidyverse)
food_97<-aggregate(food$FoodServices.97,by = list(State=food$State),FUN = sum)
food_02<-aggregate(food$FoodServices.2002,by = list(State=food$State),FUN = sum)
food_07<-aggregate(food$FoodServices.2007,by = list(State=food$State),FUN = sum)
food_over_years<- aggregate(cbind(food$FoodServices.97,food$FoodServices.2002,food$FoodServices.2007),by =
list(State=food$State),FUN = sum)
head(food_growth)
```

```
> head(food_growth)
  State       V1       V2       V3
1       1091836 1132979 1270870
2    AK    1763     1849     1996
3    AL    6955     7075     8093
4    AR    4663     4659     5112
5    AZ    9094     9944    11610
6    CA   62629    66568    75989
```

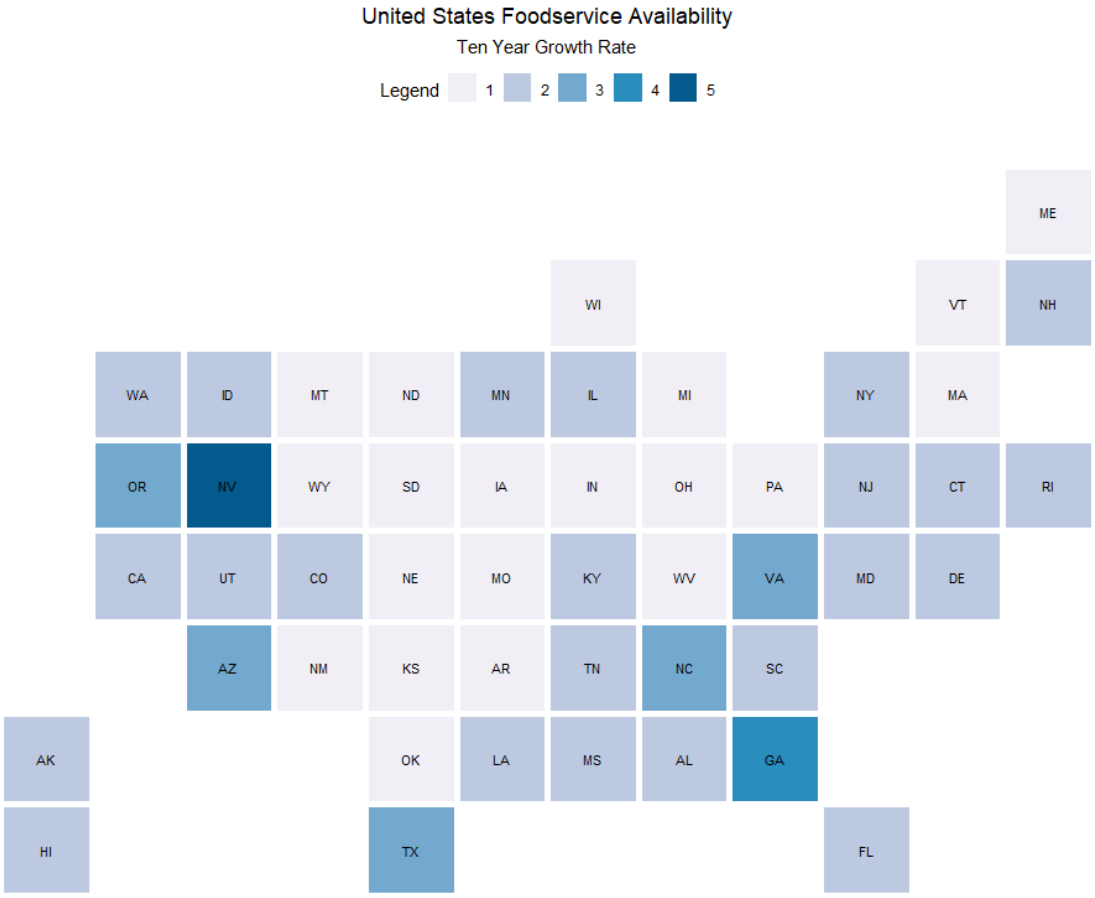
```
food_growth<-as.data.frame(food_over_years)
grwth<-left_join(states_map,food_growth,by= c("region"="State_Full"))
> head(grwth)
       long      lat group order  region subregion State   V1   V2   V3
1 -87.46201 30.38968     1     1 alabama      <NA>    AL 6955 7075 8093
2 -87.48493 30.37249     1     2 alabama      <NA>    AL 6955 7075 8093
3 -87.52503 30.37249     1     3 alabama      <NA>    AL 6955 7075 8093
4 -87.53076 30.33239     1     4 alabama      <NA>    AL 6955 7075 8093
5 -87.57087 30.32665     1     5 alabama      <NA>    AL 6955 7075 8093
6 -87.58806 30.32665     1     6 alabama      <NA>    AL 6955 7075 8093
  ten_year_Growth
1       0.1636233
2       0.1636233
3       0.1636233
4       0.1636233
5       0.1636233
6       0.1636233
```

```
grwth_map<-ggplot(grwth,
      aes(x = long, y = lat, group = group, fill = ten_year_Growth)) +
  geom_polygon(colour = "black") +coord_map("polyconic")+ labs(fill="Growth Rate")
grwth_map+ggtitle("United States Foodservice Availability","Ten Year Growth Rate") + xlab("") + ylab("") +
theme(axis.text.x = element_blank(),axis.text.y = element_blank(),axis.ticks = element_blank(),plot.title = element_text(hjust
= 0.5),plot.subtitle = element_text(hjust = .5))
```



Another View

```
grwth_bin<-statebins(food_growth,state_col = "State",value_col ="ten_year_Growth")
grwth_bin+ggtitle("United States Foodservice Availability","Ten Year Growth Rate") + xlab("") + ylab("") + theme(axis.text.x = element_blank(),axis.text.
```



Looking at the growth rate over ten years, we can clearly see that Nevada has had the largest growth, in food service availability. From this information, we potentially interpret the information is showing that maybe Las Vegas has expanded at a large rate. I was initially surprised that the growth rate wasn't higher in northeast states, but it's not too surprising, since there was probably plenty of food service established beforehand. We can see that Georgia, Virginia, and Texas have a growing rate of food service availability as well. States that seem to not have large growth rate are a large amount of the mountain west, such as Montana and Wyoming.

By County

Next, we want to see the growth by county. Similar to the steps we had in 1a.), we had to clean the data. We will start by using aggregation by county:

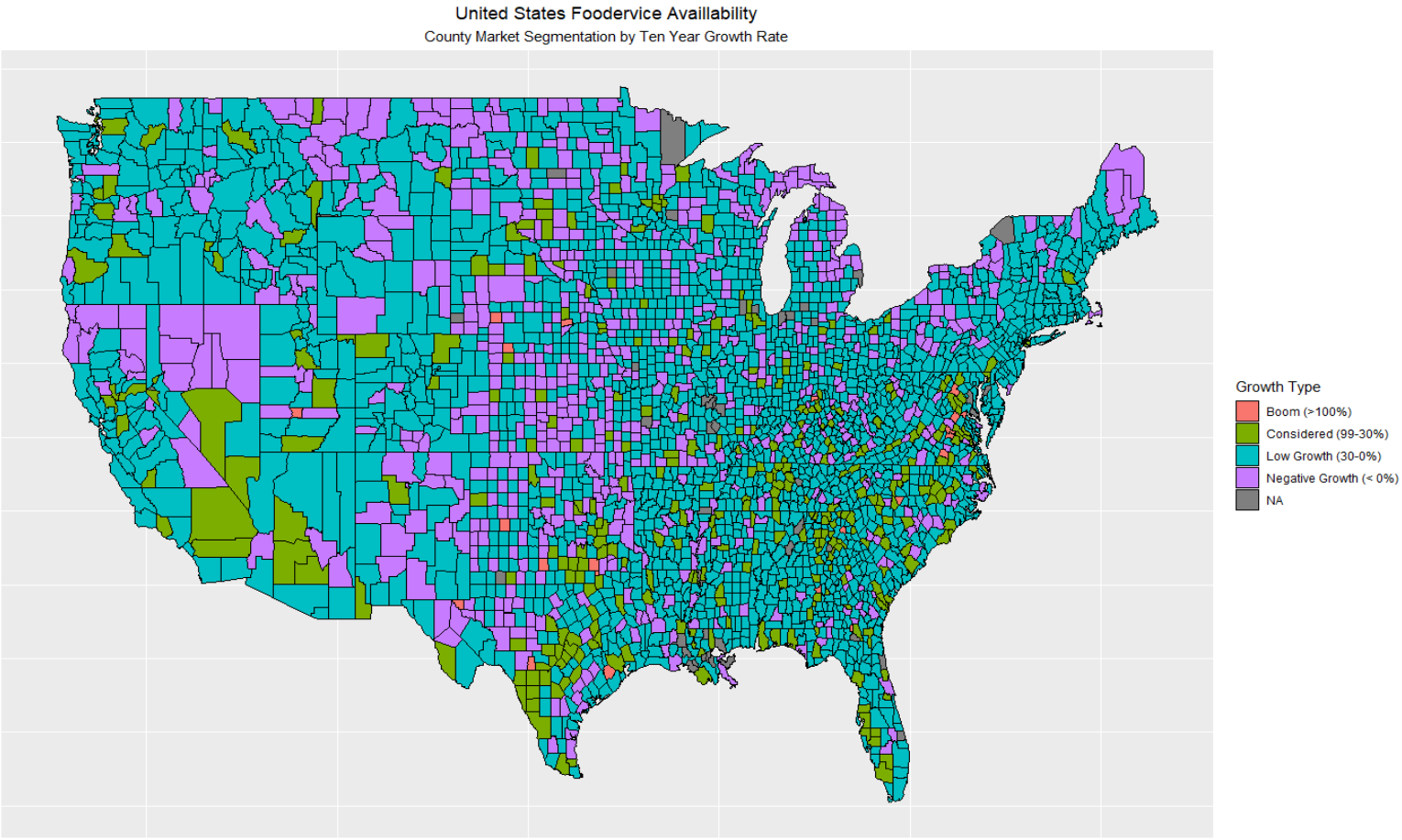
```
> food_97_C<-aggregate(food$FoodServices.97,by = list(County=food$County),FUN = sum)
> food_02_C<-aggregate(food$FoodServices.2002,by = list(County=food$County),FUN = sum)
> food_07_C<-aggregate(food$FoodServices.2007,by = list(County=food$County),FUN = sum)
> food_over_years_C<- aggregate(cbind(food$FoodServices.97,food$FoodServices.2002,food
dServices.2007),by = list(County=food$County),FUN = sum)
> food_growth_C<-as.data.frame(food_over_years_C)
> head(food_growth_C)
     County   V1   V2   V3
1 Abbeville   26   29   32
2    Acadia   58   55   67
3  Accomack  100   94  105
4       Ada  649  745  900
5     Adair  112  109  109
6     Adams 1220 1256 1367
```

Then, I wanted to remove the state names from the County information. This was achieved by creating a subset, as shown below:

```
> food_growth_C2<-subset(food_growth_C,!(food_growth_C$County%in%toupper(state.name)))
> head(food_growth_C2)
     County   V1   V2   V3
1 Abbeville   26   29   32
2    Acadia   58   55   67
3  Accomack  100   94  105
4       Ada  649  745  900
5     Adair  112  109  109
6     Adams 1220 1256 1367
```

Next, we brought in and matched the county data, with the maps package by county. Our growth variable, was created during this time as well, as shown below:

```
> food_growth_C2$ten_year_growth<-(food_growth_C2$V3-food_growth_C2$V1)/(food_growth
1)
> food_growth_C2$County<-tolower(food_growth_C2$County)
> counties<-map_data("county")
> grwth_C<-left_join(counties,food_growth_C2,by= c("subregion"="County"))
> head(grwth_C)
      long      lat group order  region subregion V1 V2 V3 ten_year_growth
1 -86.50517 32.34920     1     1 alabama   autauga 50 67 92            0.84
2 -86.53382 32.35493     1     2 alabama   autauga 50 67 92            0.84
3 -86.54527 32.36639     1     3 alabama   autauga 50 67 92            0.84
4 -86.55673 32.37785     1     4 alabama   autauga 50 67 92            0.84
5 -86.57966 32.38357     1     5 alabama   autauga 50 67 92            0.84
6 -86.59111 32.37785     1     6 alabama   autauga 50 67 92            0.84
```



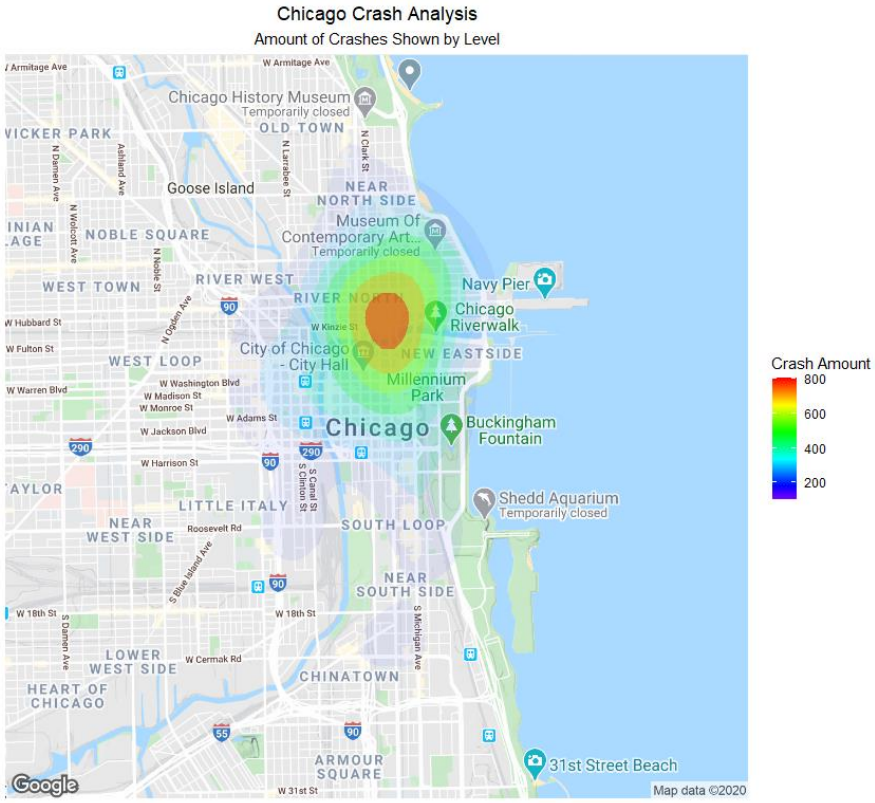
The categories distributions are dependent on the end user, but these are just a base level. Given more time, I would have liked to maybe more evenly distributed the categories, but I did not have the chance to that in this case. The goal of this was to not over mark the “considered” and “boom” markets, because those should be fewer if they are going to be potential investments for a food distributor in the continental United States (my fictional end user in this scenario). From this information, if I were given the power to do so, I would recommend placing distribution centers outside of Dallas or central Texas, DMV (DC, Maryland, Virginia) region, and in between Vegas and LA to meet the demand of “boom” and large “considered” markets.

In terms of color, played around with using intensity of a single color, but I felt that the different colors helped in terms of clustering market segments. For example, it’s easy to see that Vegas and the LA county region have a large “considered” market. Let me know if you believe another color scheme would have worked better.

GGMAP

After getting an api key for ggmap, I was able to get a google map image of Chicago. After that, we use the combine the Chicago crash data with ggmap, shown below:

```
chi<-get_map(location="Chicago",zoom =13)
> chi_crash<-ggmap(chi)+stat_density2d(aes(x = LONGITUDE, y = LATITUDE, fill = ..level..,alpha = ..level..),data
  = crash,geom ="Polygon")+scale_fill_gradientn(colours = rev(rainbow(100, start=0, end=0.75)))+scale_alpha_contin
uous(guide="none",range=c(0,.4))
> chi_crash
```

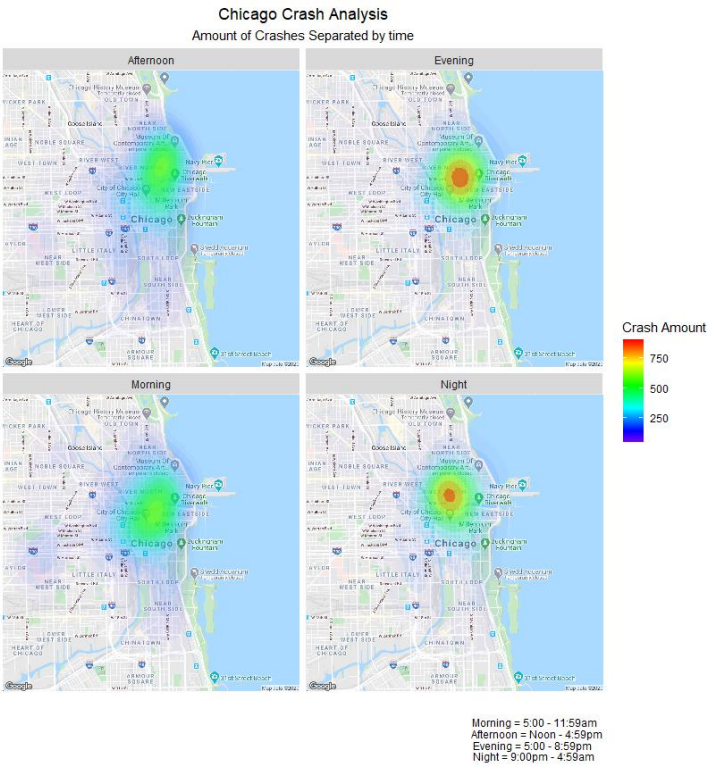


The above shows where all of the crashes occur in Chicago. I chose this visualization because it does a good job of showing where a majority of the crashes occur. As shown above, it displays that a majority of accidents occur in the loop area, and that as you get further away from the loop, the accidents go down. I'm still unsure of what the level / the legend indicates on the right. Any insight on how to work with the level, would be helpful.

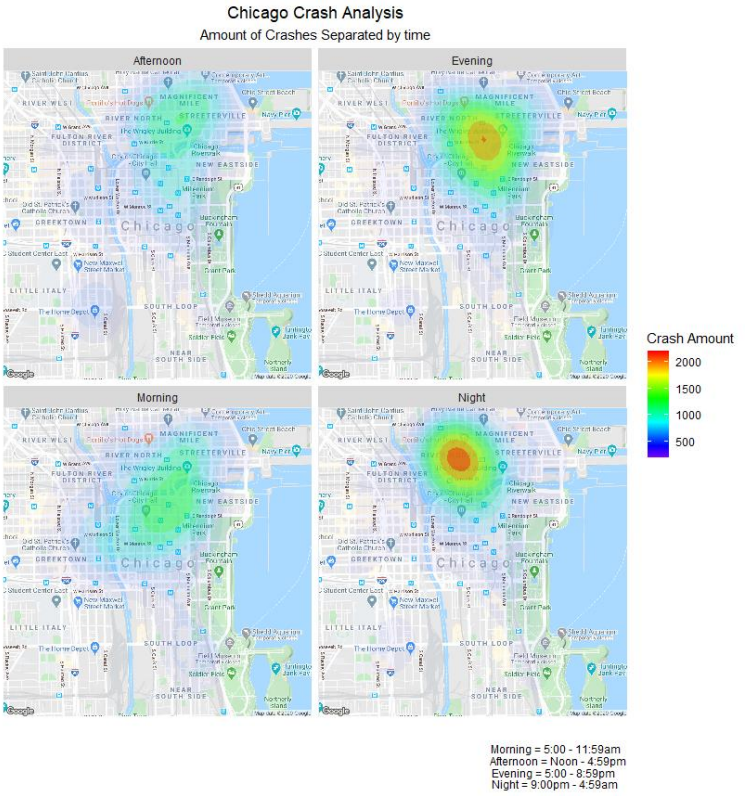
To show where crashes occur in the city during different times of the day, I created a conditional / if statement in excel, that showed whether it was morning, afternoon, evening, or night. Then I applied a ggmap+ a facet wrap (to show the windows). This is shown below:

```
> chi_crash<-ggmap(chi)+stat_density2d(aes(x = LONGITUDE, y = LATITUDE, fill = ..level..,alpha = ..level..),data = crash,geom ="Polygon")+scale_fill_
(colours = rev(rainbow(100, start=0, end=0.75)))+scale_alpha_continuous(guide="none",range=c(0,.4))
> chi_crash+facet_wrap(.~TIME_OF_DAY) + ggtitle("Chicago Crash Analysis","Amount of Crashes Separated by time")+xlab("") + ylab("") + theme(axis.text
ent_blank(),axis.text.y = element_blank(),axis.ticks = element_blank(),plot.title = element_text(hjust = 0.5),plot.subtitle = element_text(hjust = .5
ill = "Crash Amount", caption = "Morning = 5:00 - 11:59am  \n Afternoon = Noon - 4:59pm\nEvening = 5:00 - 8:59pm    \nNight = 9:00pm - 4:59am    ")
```

Zoom = 13



Zoom = 14



The above is shows a window approach of car accidents in Chicago during different periods of time throughout the day. The times of the day were morning (5:00-11:59am), Afternoon (Noon-4:59pm), Evening (5:00 – 8:59pm), and Night (9:00pm – 4:59am). I had difficultly selecting the zoom on this application because “zoom = 13” seemed a little too zoomed out and “zoom = 14” seemed too zoomed in. I tried shifting the map to River North, but then it seemed to take other data out of focus. Any advice on how to fit data with the zoom, would be appreciated.

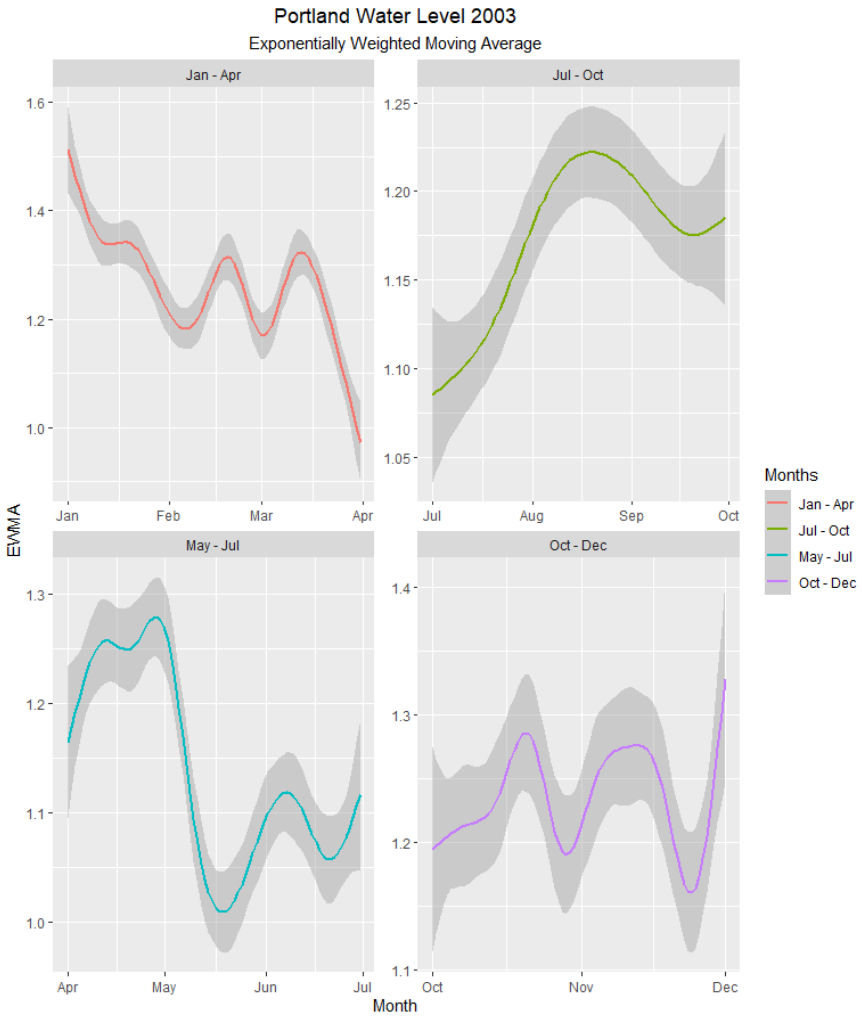
Moving Averages

For the Portland water level data set, I chose wrote a condition / if statement in excel that showed the abbreviated month name of the time period of the calculation. For example, “Jan – Apr”, would be 1/1/2013-3/31/2013 in conditional statement because the EMA function in R needed future data from the next month, and included April. I still need clarification on how this function works, but that is how I interpreted it.

I chose an exponentially weighted moving average (EWMA) because doesn’t smooth as much of the volatility which I wanted to display with the water level. Also, the EWMA helps de-weight the past, and picks up on the nearest trend. It is worth noting there is still a bit of lag though.

```
water_trend$EMA_W<-EMA(water_trend$WL)
water_level<-ggplot(water_trend,aes(x = Date, y = EMA_W,color = Months))
water_level+facet_wrap(.~Months,scales = "free")+geom_smooth()+scale_x_date(date_labels = "%b",breaks = "1 month")
```

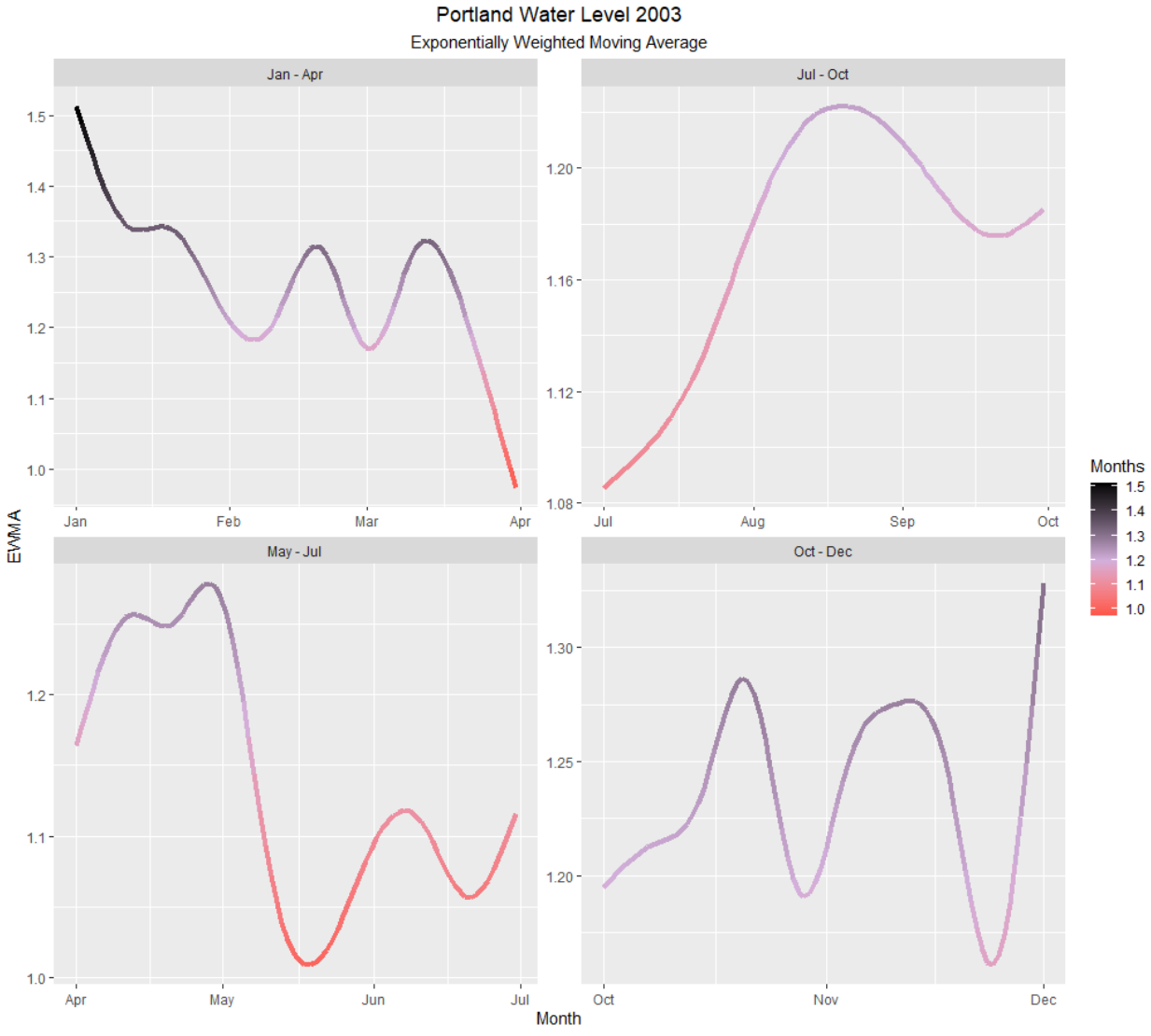
Notice here the monthly break down, and the scale changes between the sections. For example, you can see that “Jan – Apr” ends near 1.0, then “May – Jul” picks back up around 1.15. I chose this way to format the data because it shows the differences in the water level, during different periods of the time of the year. In other words, using the EWMA helped smooth out the data. We still se some of the noise with the outline of the smooth, but nothing to the point where it takes away from the overall trend.



This is a different view of the same information.

```
#4
water_level+facet_wrap(.~Months,scales = "free")+geom_smooth(aes(color=..y..),size = 1.5, se = FALSE)+scale_x_date(date_labels = "%b",breaks = "1 month")+ggti
+scale_colour_gradient2(low = hsv(1,1,1), mid = hsv(.8,.2,.86) , high = hsv(0,.3,0), midpoint= avg)
```

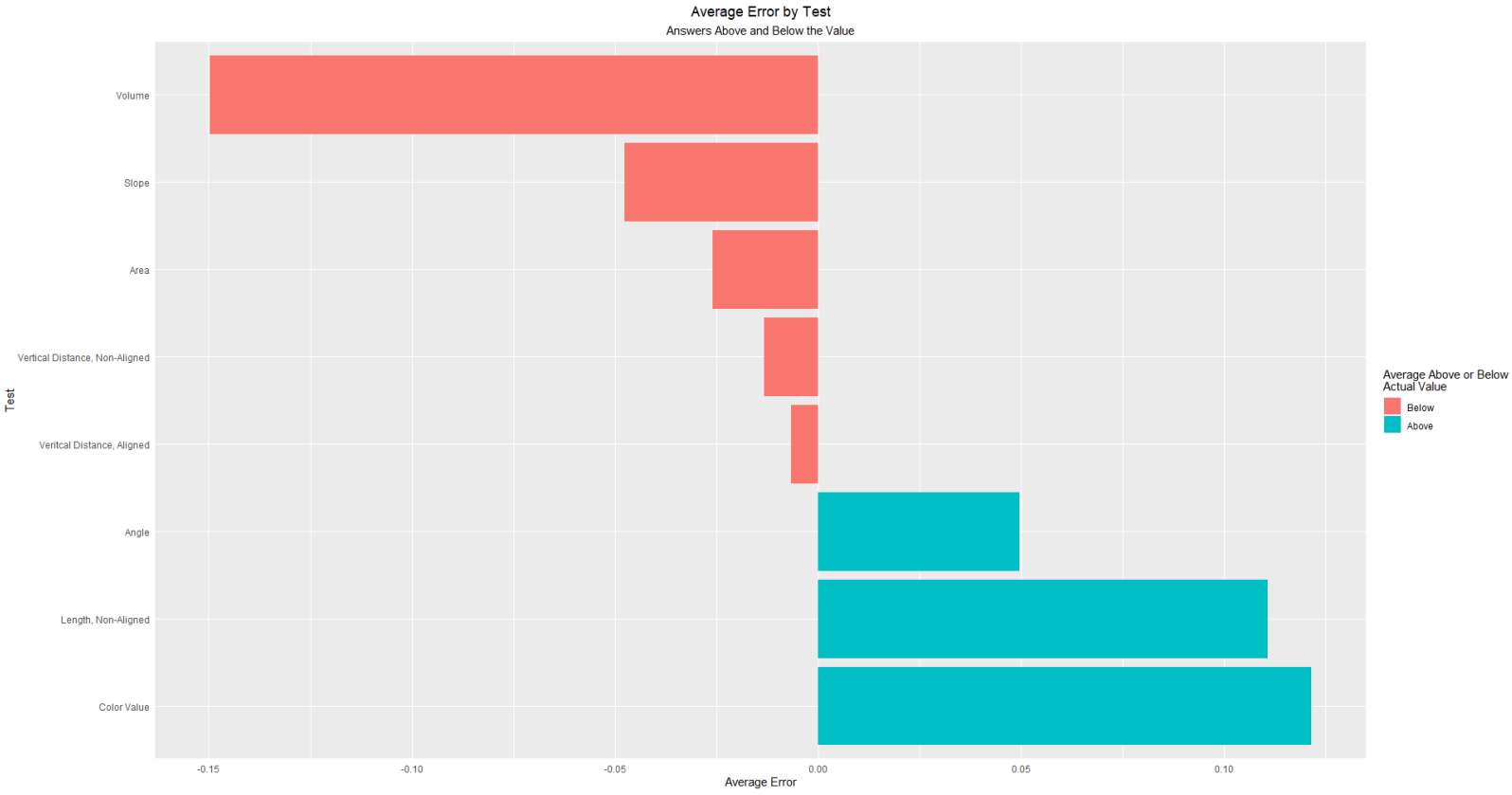
```
> avg
[1] 1.198726
```



The mid-point is set at the average, along with red at the bottom, blue in the middle, and black at the top interval.

Working with Survey Response Errors of Data Visualizations

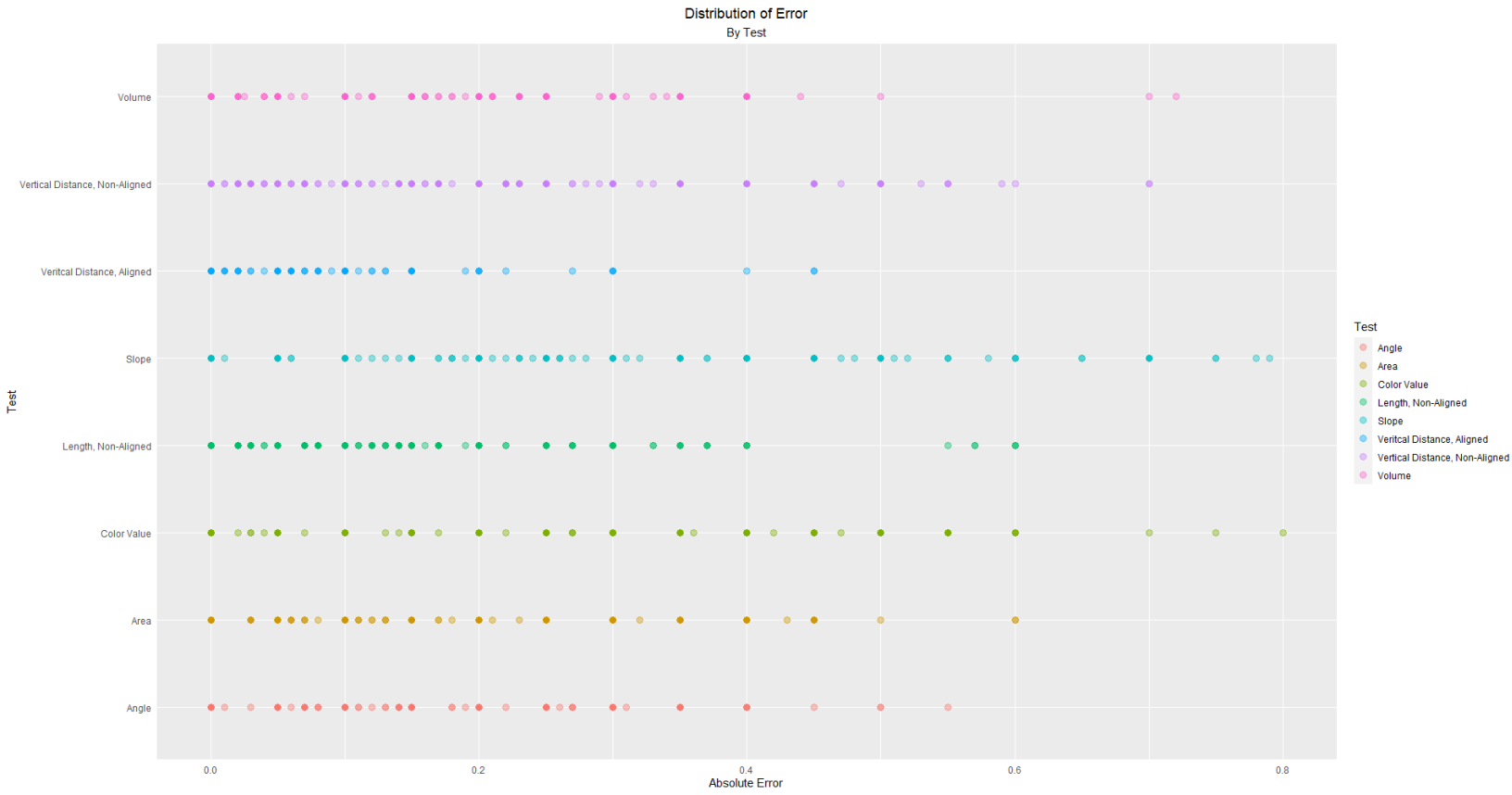
```
> percep$Error<-percep$Response-percep$TrueValue
> percep$AbsoluteError<-abs(percep$Error)
> p<-percep %>%group_by(Test) %>%summarise(average=mean(Error))%>%arrange(average)
> head(p)
# A tibble: 6 x 2
  Test                          average
  <fct>                           <dbl>
1 Color Value                  -0.150
2 Length, Non-Aligned          -0.0475
3 Angle                        -0.0260
4 Veritcal Distance, Aligned   -0.0132
5 Vertical Distance, Non-Aligned -0.00663
6 Area                          0.0496
> p$Test<-as.character(p$Test)
> p$Test <- factor(p$Test, levels=unique(p$Test))
> #^to order
```



It is often frowned upon to order a bar chart, but I believe this helps display the message of clearly showing what values have underestimated and overestimated the error. We can see from the average error, that a lot of people underestimated “Volume”, with an average error of -.15. We can also see that a low people overestimated “Length,Non-Aligned” and “Color Value” with both having average errors above .1.

Distribution of Error

```
ggplot(percep,aes(AbsoluteError,Test,color = Test))+geom_point(alpha=.4,size = 3)+scale_fill_viridis(discrete = TRUE)+ggtitle("Distribution of Erro
```

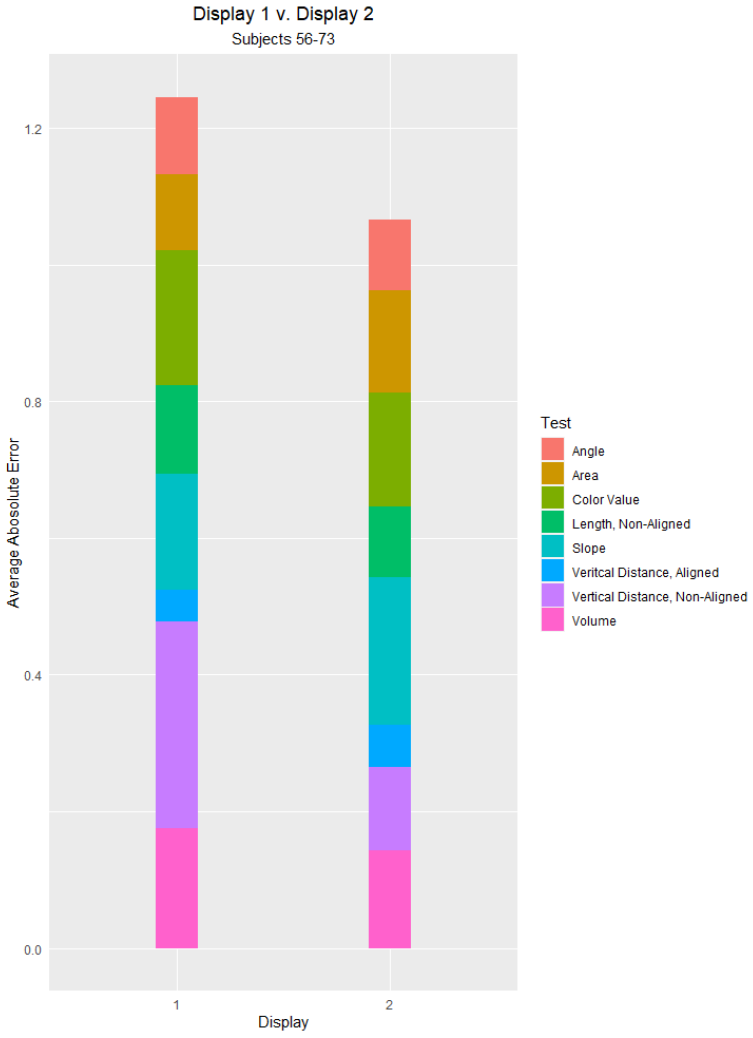


I was debating on using color or not, but decided on color since it helps somewhat in the difference of test, and we used it in a similar example within the tutorial.

In regards to the distribution, we can see that “Vertical Distance, Aligned” and “Angle” are the most compact, with little amount of outliers. We can see that “Slope” is the most spread out. We can say for nearly every variable though, that the majority of absolute error is within the 0 and .4.

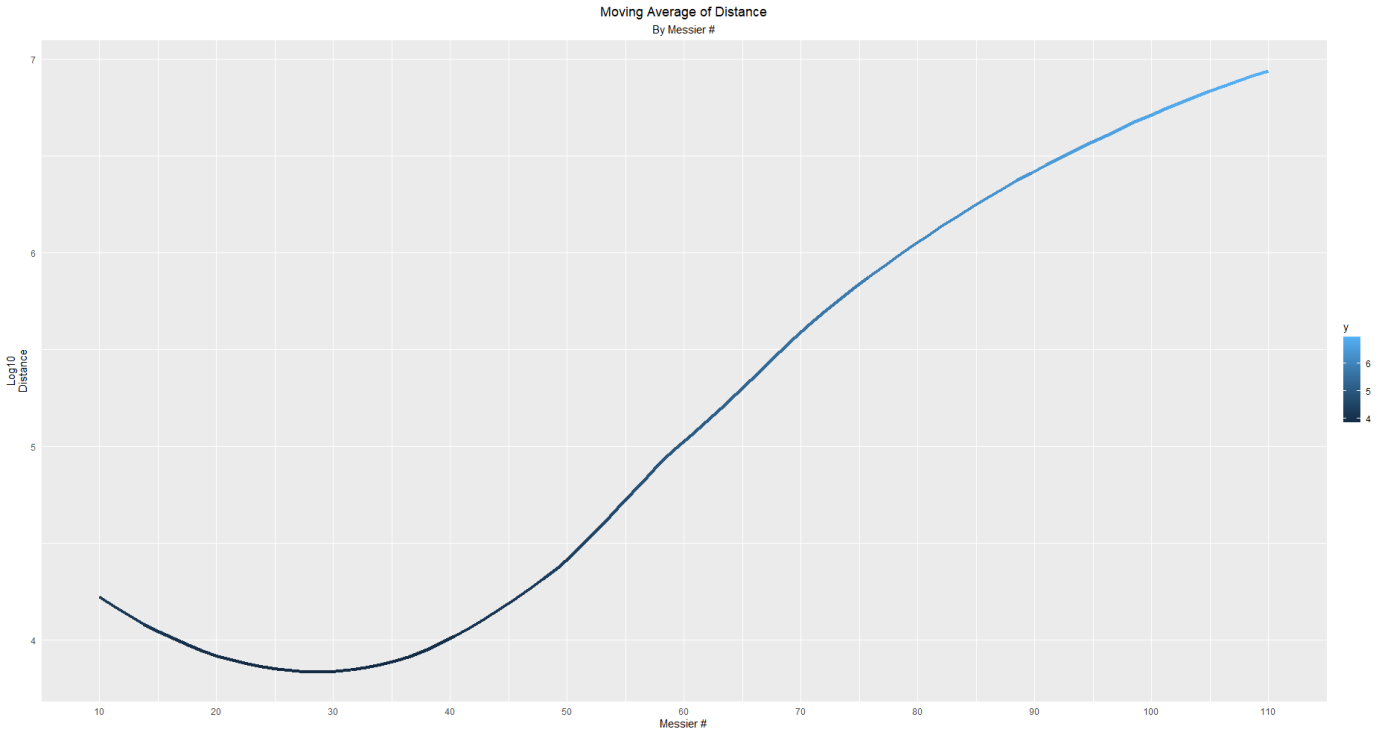
Looking at Specific Displays of Certain Subjects

```
#c
install.packages("plotly")
library(plotly)
#Displays 1 and 2,subject 56-73
#56-73
head(percep)
disp<-c("1","2")
c<-filter(percep,between(Display,1,2),between(Subject,56,73))
head(c)
View(c)
c_filter<-c%>%group_by(.dots=c("Test","Display")) %>%summarise(average=mean(AbsoluteError))%>%arrange(average)
head(c_filter)
c_filter$Display<-as.character(c_filter$Display)
c_plot<-ggplot(data = c_filter, aes(x=Display, y = average, fill = Test))
c_plot+geom_bar(stat = 'identity',width = .2)+ggtitle("Display 1 v. Display 2","Subjects 56-73") + xlab("Display")+ylab("Average Abosolute Error") +
```



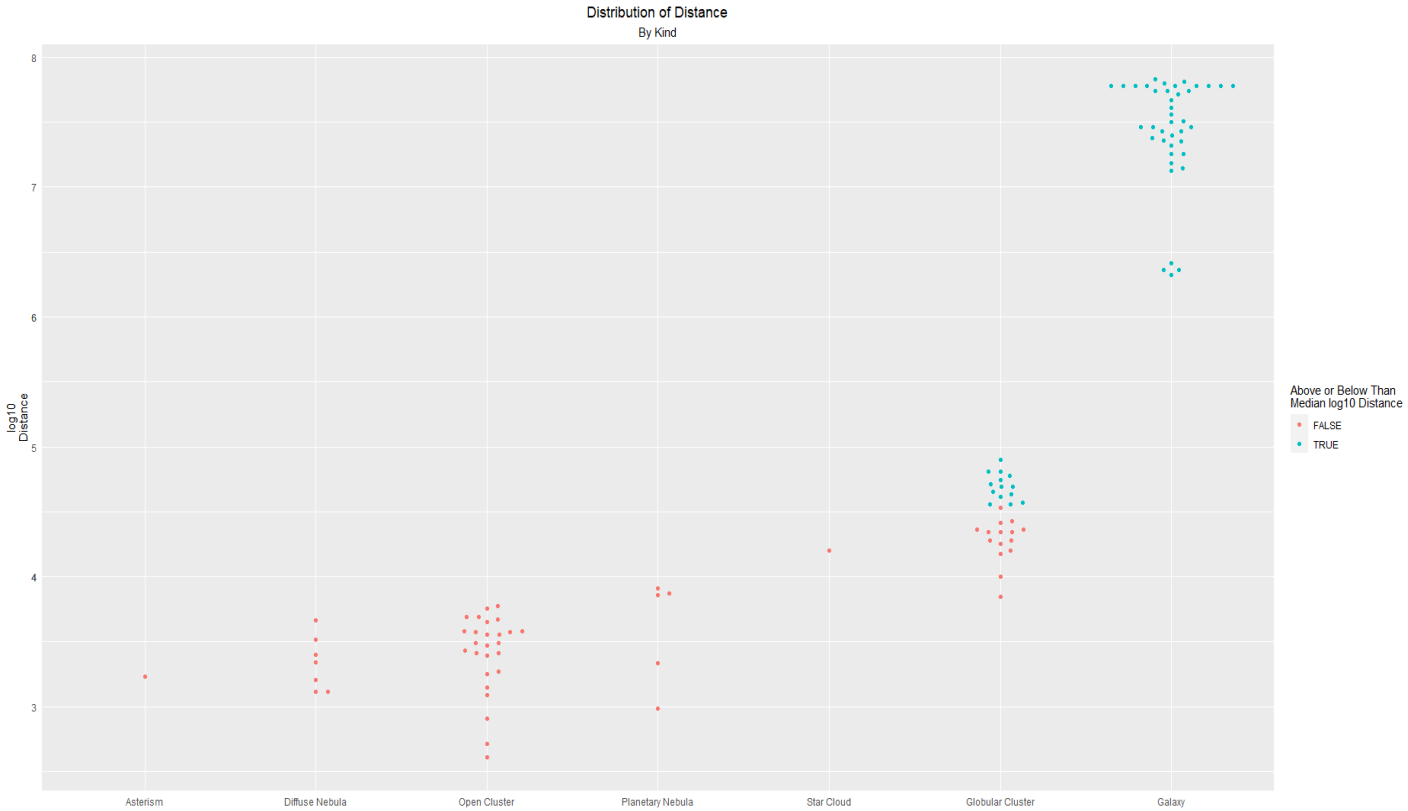
You could say overall, the subject got better at responding on Display 2, after responding to Display 1. As shown above, our chart shows that overall, they responded nearly .2 better. A large part in this came from “Vertical Distance, Non-Aligned” though. Besides the aforementioned variable, it is tough to say whether there was overall improvement made from Display 1 to Display 2. Specifically, we could look at “Slope”, and say that the error has gotten a little worse.

Working with Messier Distance



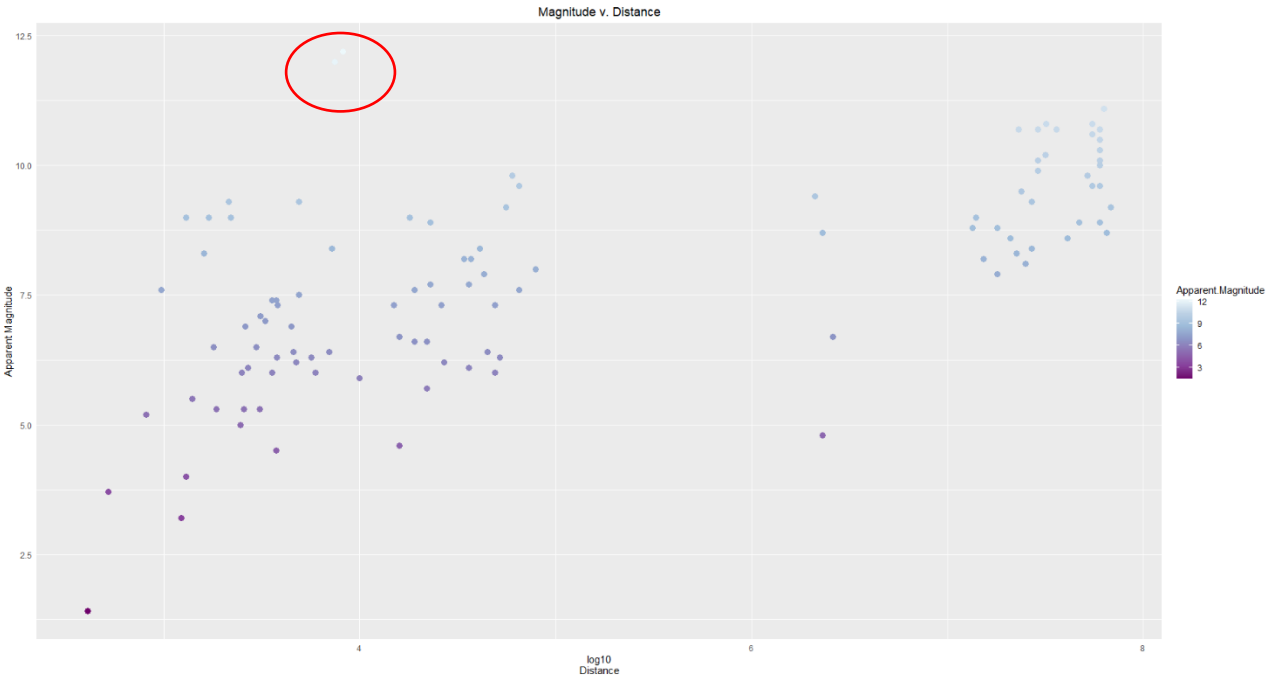
I took the log10 distance based on the messier number, then converted this to a moving average. It's recommended use log10 in astronomy, physics, etc. . The trend is shown here, is that the average distance increases as the messier # go further along (it is noted that the messier # is more of just a list).

```
messier<-read.csv('MessierData.csv')
messier<-messier[-c(40),]#na row for dist
messier$log10Distance<-log10( messier$Distance..LY.)
messier$log10Distance_EMA_W<-EMA(messier$log10Distance)
```



In the beeswarm plot above, we can see the log10 distribution of distance based on Kind. The Kind is ordered in terms of the top distance. We can gather from this plot that Galaxy has most of its points higher than the other Kinds. I also wanted to see what Kind had a distance greater or less than the median log10 distance. Obviously, galaxy did, but did any others? We can see from the plot above that Globular Cluster had some distribution above the median. I chose the median because the mean was skewed from Galaxy.

```
install.packages("shades")
library(shades)
m_mag<-ggplot(messier, aes(x=log10Distance, y=Apparent.Magnitude,colour = Apparent.Magnitude)) +geom_point(size=3)+scale_colour_distiller(palette="B
m_mag+ggtitle("Magnitude v. Distance") + xlab("log10\nDistance")+ylab("Apparent Magnitude") + theme(axis.ticks = element_blank(),plot.title = elemen
```

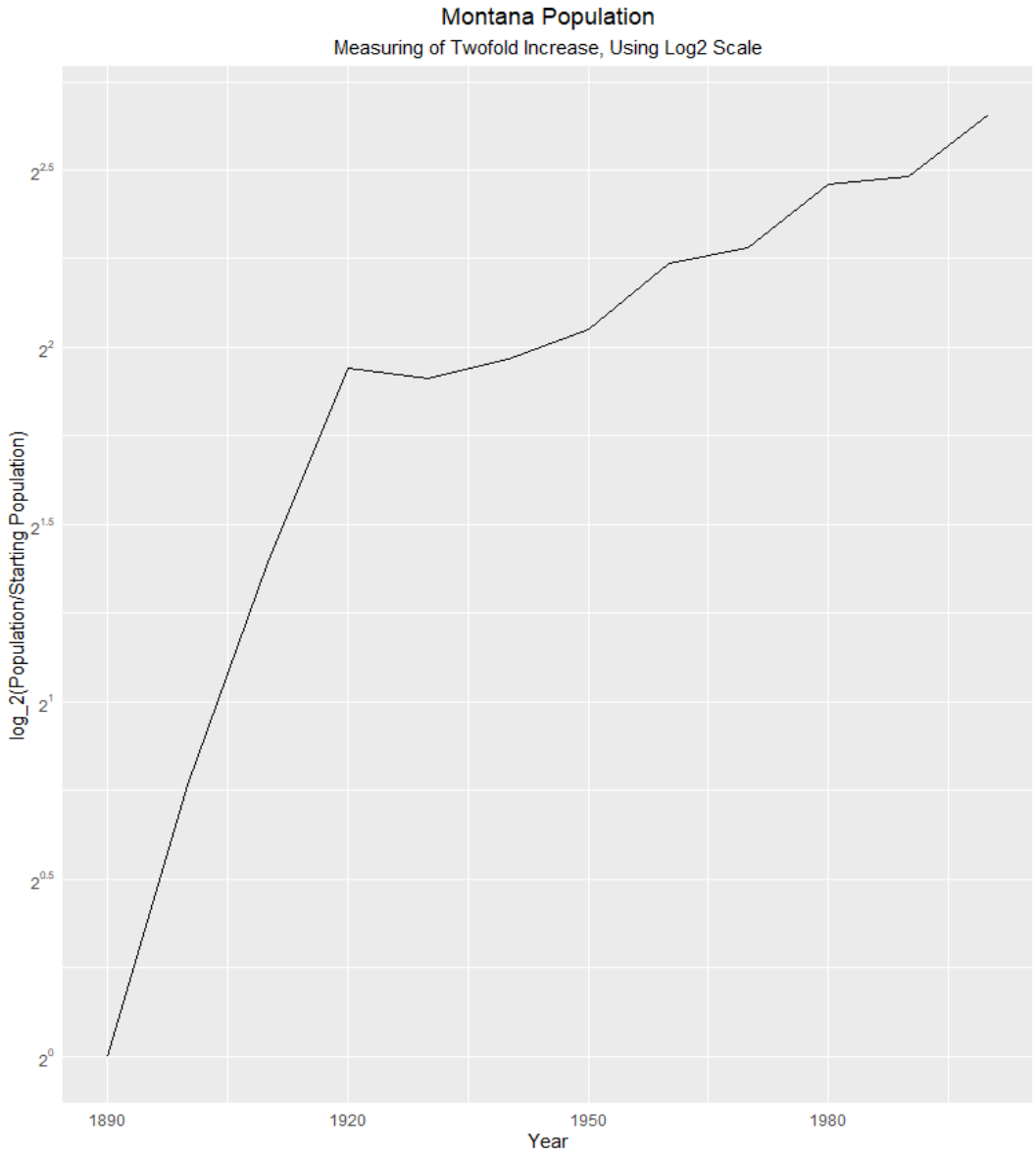


We can see from the above, that with the higher magnitude, the points are harder to see. For example, I highlighted in red the points near 12.5 that are barely noticeable. The visualization shows this with the higher magnitude being lighter.

Working with Montana Reservation Population Dataset

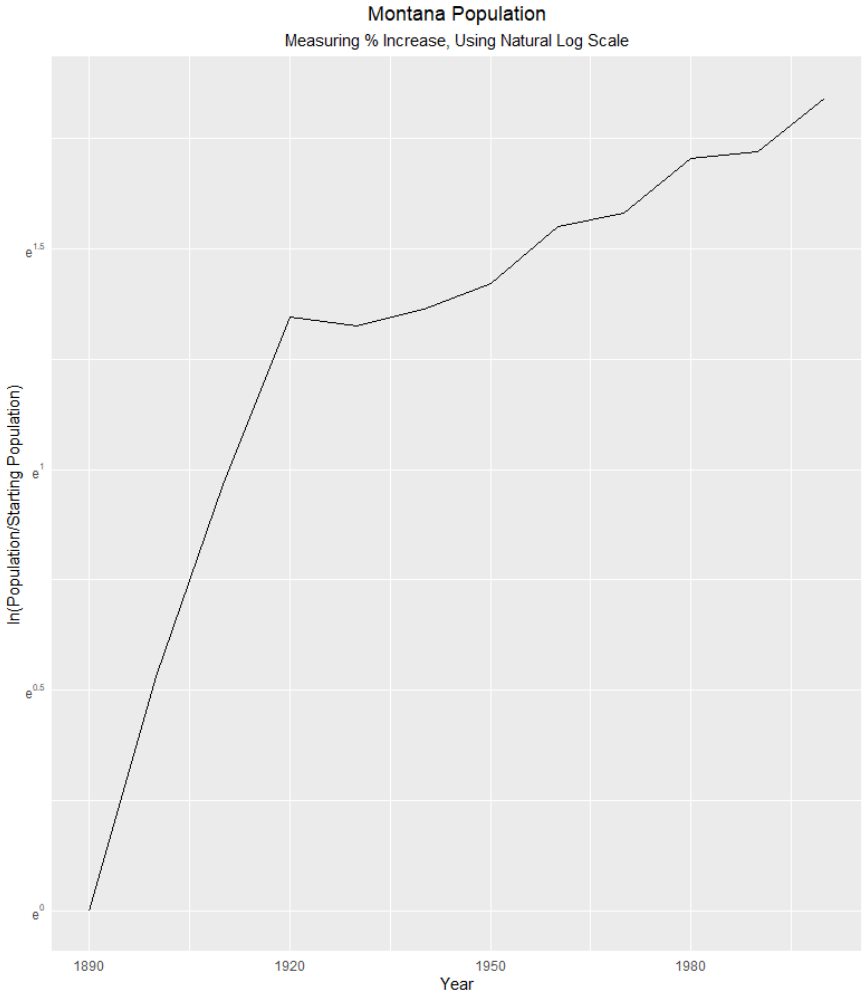
How much does the population double?

```
> head(mpd)
  Year Population PopLog logGrowth.Rate GRTRUE
1 1890     142924   1.00            NA     NA
2 1900     243329   1.70     0.70000000   TRUE
3 1910     376053   2.63     0.54705882   TRUE
4 1920     548889   3.84     0.46007605   TRUE
5 1930     537606   3.76    -0.02083333  FALSE
6 1940     559456   3.91     0.03989362  FALSE
> mpd_plot<-ggplot(mpd, aes(x = Year,y = PopLog))+geom_line()
> mpd_plot+scale_y_continuous(trans = "log2",breaks = trans_breaks("log2", function(x) 2^x),labels = trans_format("log2", ma
bs(y="log_2(Population/Starting Population)",x = "Year")+ggtitle("Montana Population","Measuring of Twofold Increase, Using
(axis.ticks = element_blank(),plot.title = element_text(hjust = 0.5),plot.subtitle = element_text(hjust = .5))
```



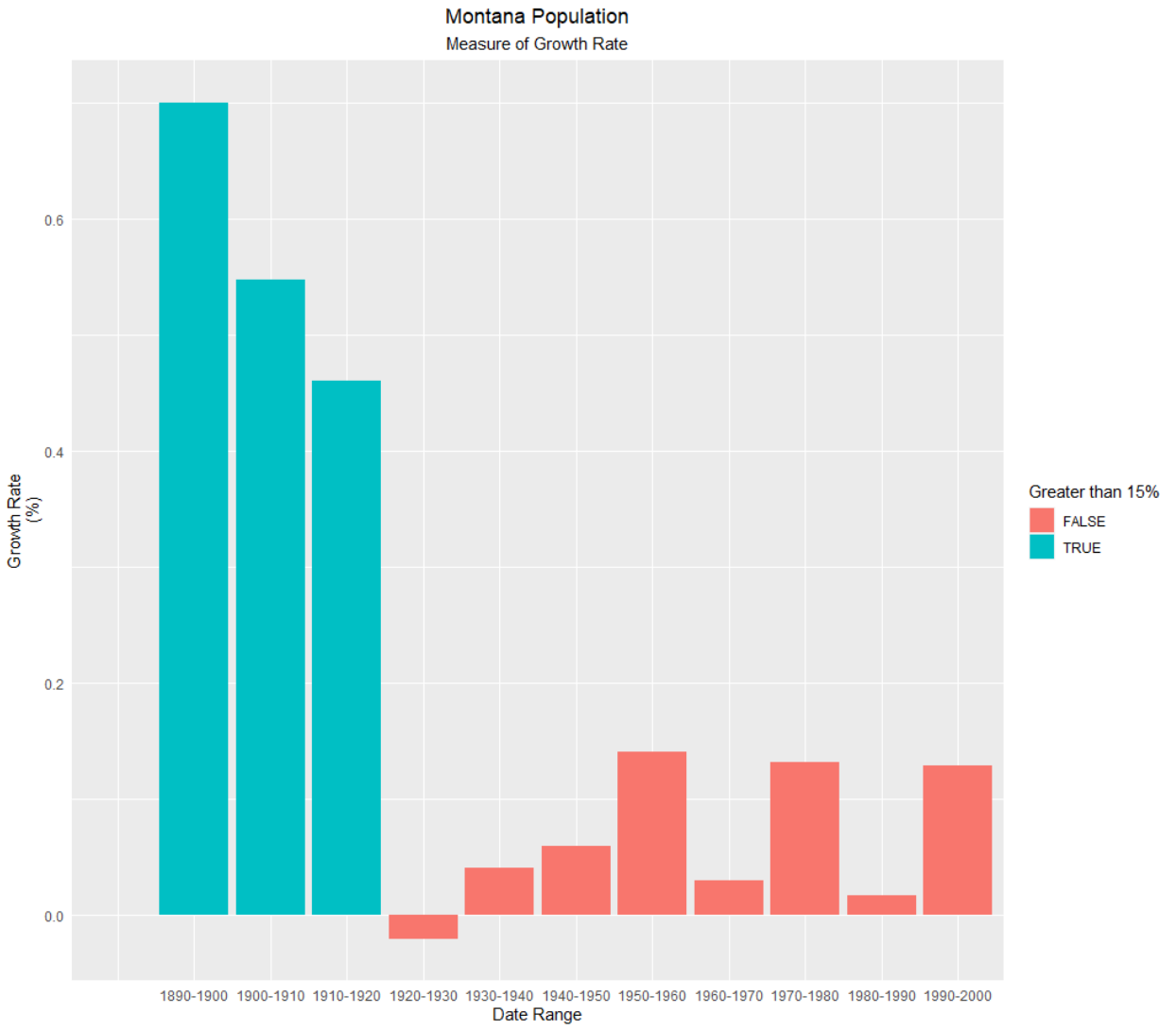
In our line plot above, we can see that the population doubles just short of 3 times and is on trend to at some to double again by the early 2000's or 2010.

```
mpd_plot<-ggplot(mpd, aes(x = Year,y =PopLog))+geom_line()
mpd_plot+scale_y_continuous(trans = log_trans(),breaks = trans_breaks("log", function(x) exp(x)),labels = trans_format("log", math_fo
```



Looking at our natural log, line graph to the right, we can see that % rate of change in the population has increased over the years. As displayed on the right, the biggest rate of increase is from 1890 to 1920.

```
> mpd_plot<-ggplot(mpd, aes(x =DateRange,y = logGrowth.Rate,fill = logGrowth.Rate>.15))+geom_bar(stat='identity')
> mpd_plot+labs(y="Growth Rate\n     (%)",x = "Date Range",fill = "Greater than 15%")+ggtitle("Montana Population","Measure of Growth Rate")+ t
eme(axis.ticks = element_blank(),plot.title = element_text(hjust = 0.5),plot.subtitle = element_text(hjust = .5))
```



We can see from the above, that the % growth rate was above 15% from 1890 to 1920. It is important to note though, that from 1920 to 1930 it was negative growth. It was not necessary to log scale for this part, but one could calculate % change from the part b if they wanted to. For example, getting close to the number with a calculator, in b – 1920 = e^1.3455 = 3.84. Reference below for the math :

3.84-e^0/e^0 = 2.84

These matches:

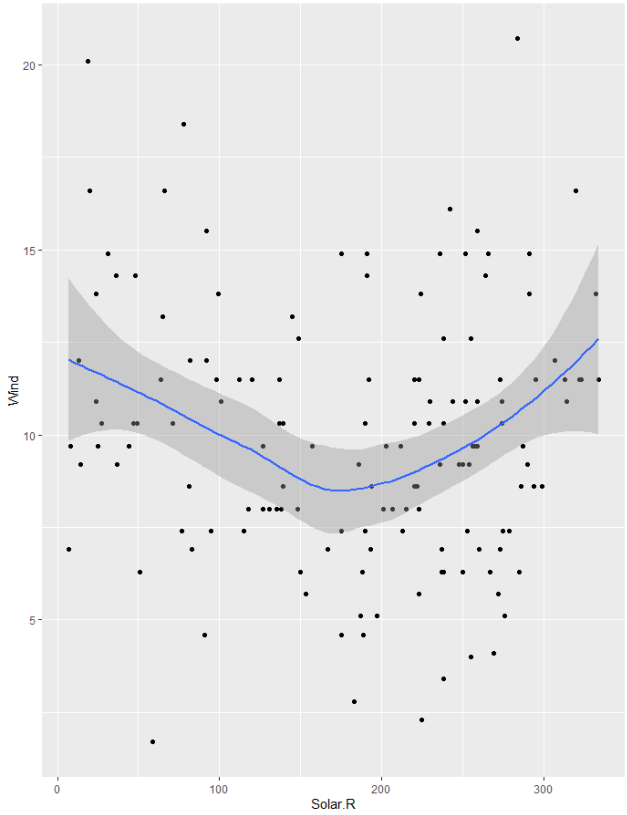
548889-142924/142924 = 2.84

Both of these are the growth rate from 1890-1920, but just we just got there in different ways.

Working with Air Quality Dataset

```
> air<-read.csv('AirQuality.csv')
> air_plot<-ggplot(air, aes(x = Solar.R,y = Wind))+geom_point()+geom_smooth()
> air_plot
```

This is mostly uncorrelated as seen to the right. The loess trend line tries to show a little bit of a trend, but it does not add that much because there is so little correlation.



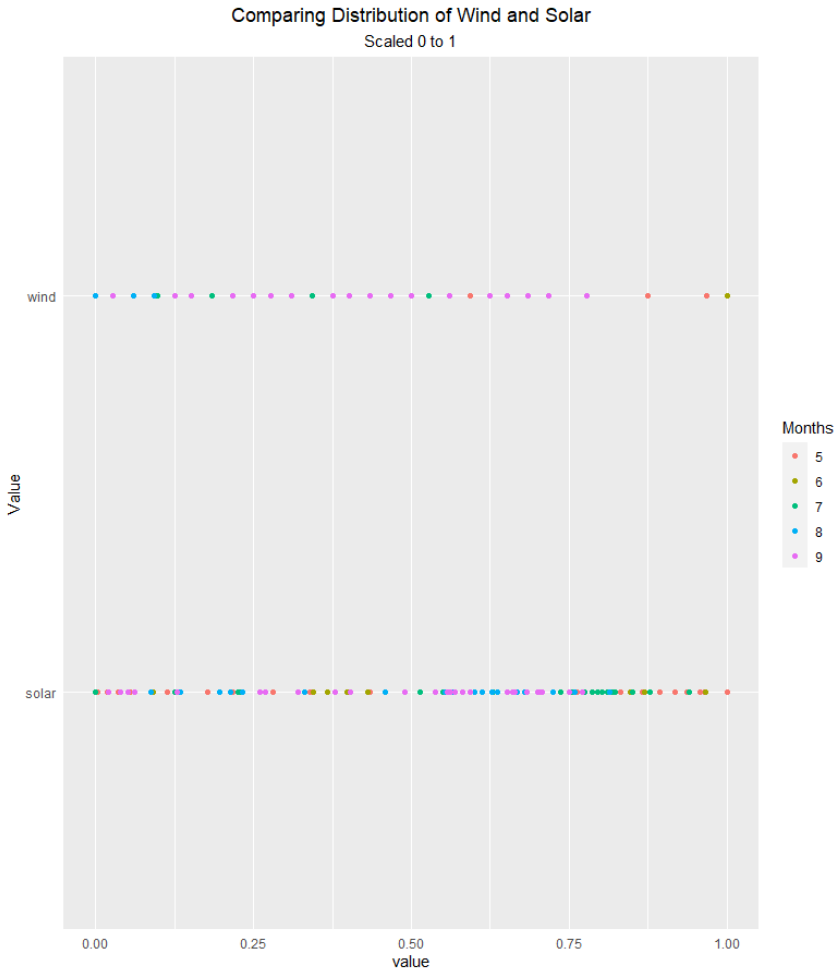
```
> aq.edit <- aq %$%
+   data.frame(wind=Wind,
+              ozone=Ozone,
+              solar=Solar.R,
+              month=Month,
+              day=Day,
+              temp=rescale(Temp, to=c(0,1)))
```

Scale wind and solar

```
> aq.edit_b<-aq.edit%>%select("month","solar","wind")%>%mutate(wind=(wind-min(wind))/(max(wind)-min(wind))) %>%mutate(solar=rescale(solar, to=c
(0,1)))
> aq_b<-aq.edit_b%>%pivot_longer(-month,names_to="Measurement",values_to="value")
```

^Select only solar and wind

```
> aq_b%>%ggplot(aes(Measurement, value,color = factor(month))) +
+   geom_point()+coord_flip()+ggtitle("Comparing Distribution of Wind and Solar","Scaled 0 to 1")+ theme(axis.ticks = ele
 = element_text(hjust = 0.5),plot.subtitle = element_text(hjust = .5))+labs(x = "Value",color = "Months")
```

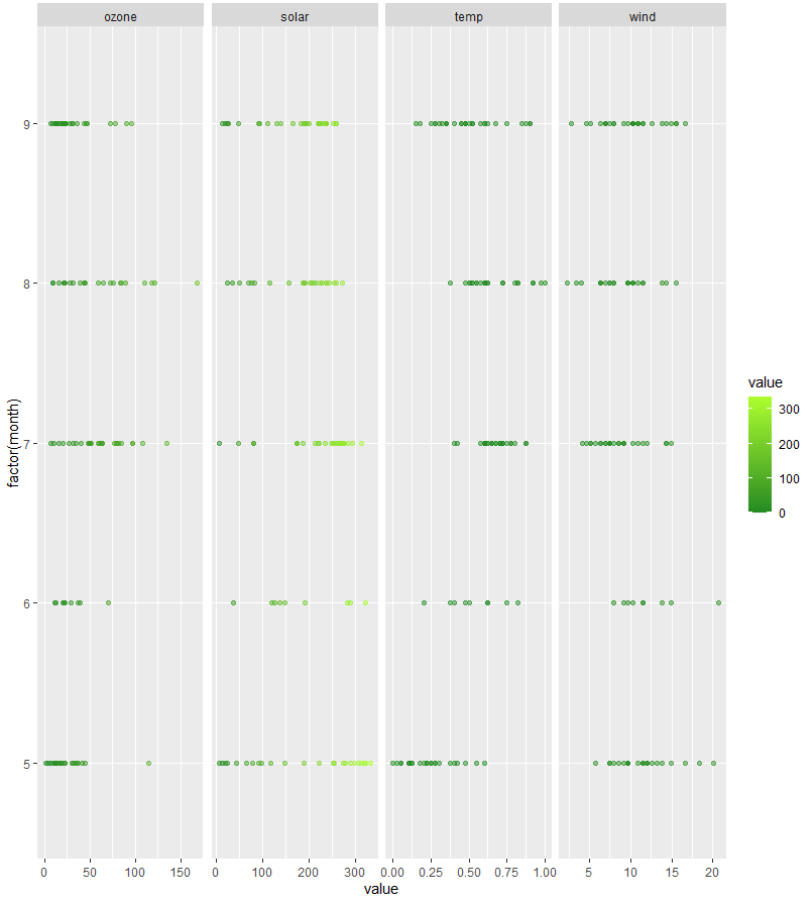


We can gather from this plot, that the distribution of wind is more evenly spread, while the distribution of solar is more clustered.

```
> aq_c2_plot+geom_point(alpha=.5)+coord_flip()+facet_grid(~Measurement,scales = "free")+  scale_color_gradient(low = 'forestgreen',high = 'green
yellow', )
```

```
> aq_c2<-aq_c[,c(1,4,2,3)]
> aq_c2
# A tibble: 444 x 4
   month   day Measurement   value
   <int> <int> <chr>         <dbl>
 1     5     1 temp          0.25
 2     5     2 ozone        41
 3     5     3 solar       190
 4     5     4 wind          7.4
 5     5     7 temp          0.375
 6     5     8 ozone        36
 7     5     9 solar       118
 8     5    12 wind          8
 9     5    13 temp          0.425
10     5    14 ozone        12
# ... with 434 more rows
> aq_c2<-as.data.frame(aq_c2)
```

```
> head(aq_c2)
  month day Measurement   value
1     5   1        temp   0.250
2     5   2       ozone  41.000
3     5   3       solar 190.000
4     5   4        wind   7.400
5     5   7        temp   0.375
6     5   8       ozone  36.000
>
```

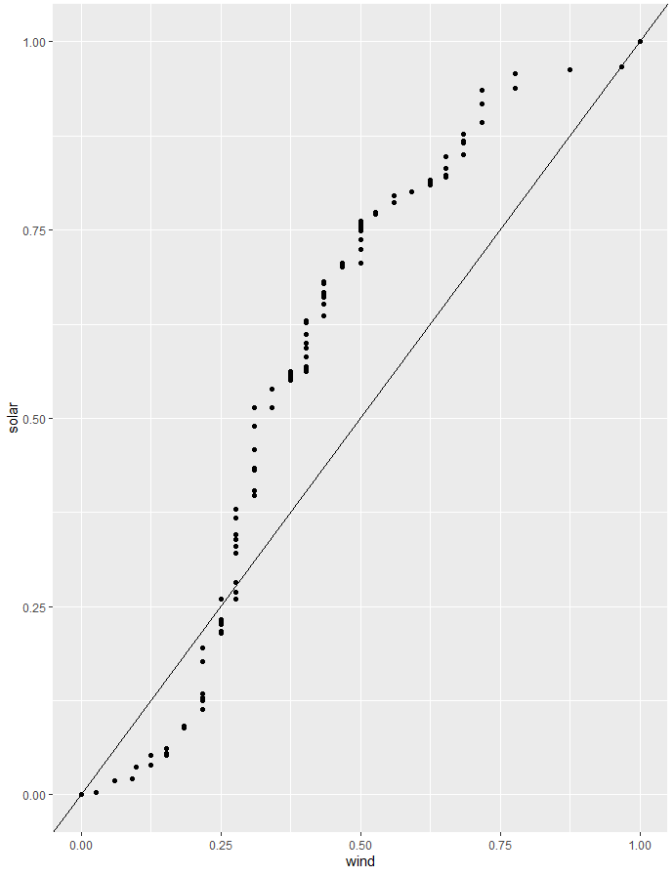


We can see from the plot that the windiest day is likeliest to happen in either June or May and there is also a some clustered distribution of low solar radiation in June and May as well.

```
> aq.qq <- aq %$%
+   data.frame(wind=sort(Wind),
+              ozone=sort(Ozone),
+              solar=sort(Solar.R),
+              temp=rescale(Temp, to=c(0,1)) %>% sort)
> aq.qq %>%
+   mutate(wind=(wind-min(wind))/(max(wind)-min(wind))) %>%
+   mutate(solar=rescale(solar, to=c(0,1))) %>%
+   ggplot(aes(wind,solar)) +
+   geom_point() +
+   geom_abline(slope=1, intercept=0)
```

The QQ plot helps compare the distribution between Wind and Solar.R in fine detail. It plots the points in order, so the point n the graph are not the actual data points, but rather go in order from 1st, 2nd, etc….

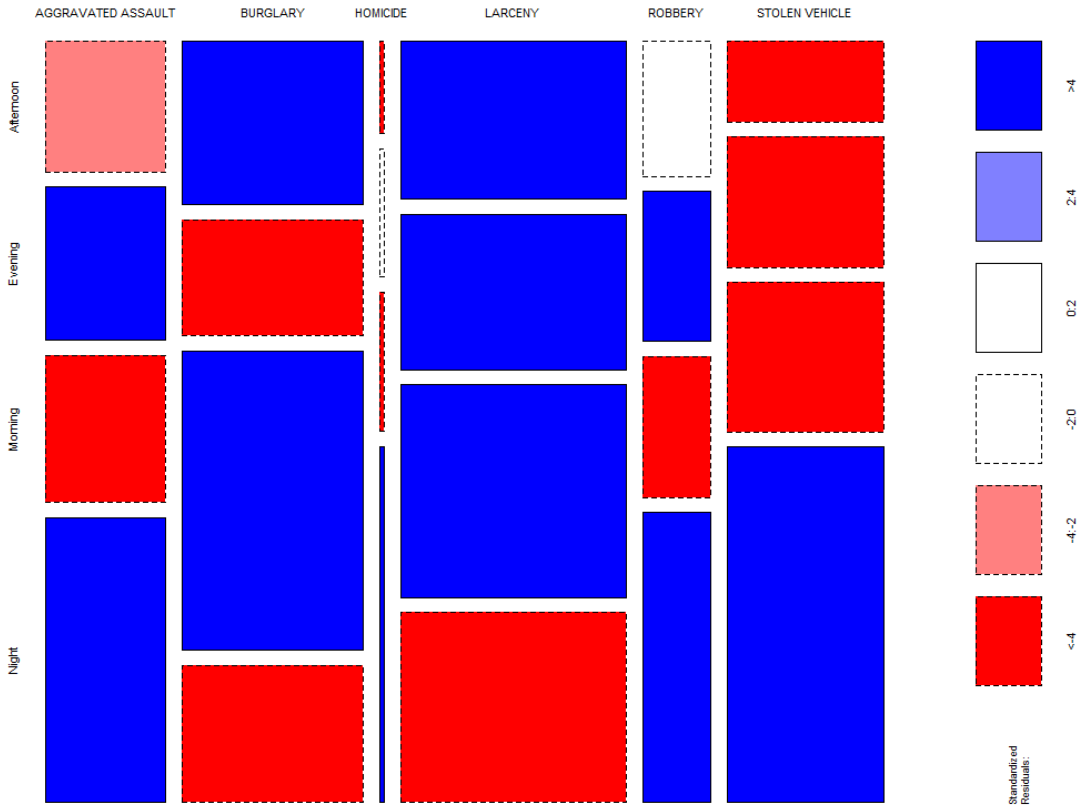
This helps us get a fine detail of between the two variables. Since they don't perfectly line up on our 45-degree angle line, we know that they do not have the same distribution.



Working with Detroit Crime Dataset

This mosaic plot allows us to see the frequency of occurrences we are looking for. One example we can see is that a majority of stolen vehicle incidents occur at night.

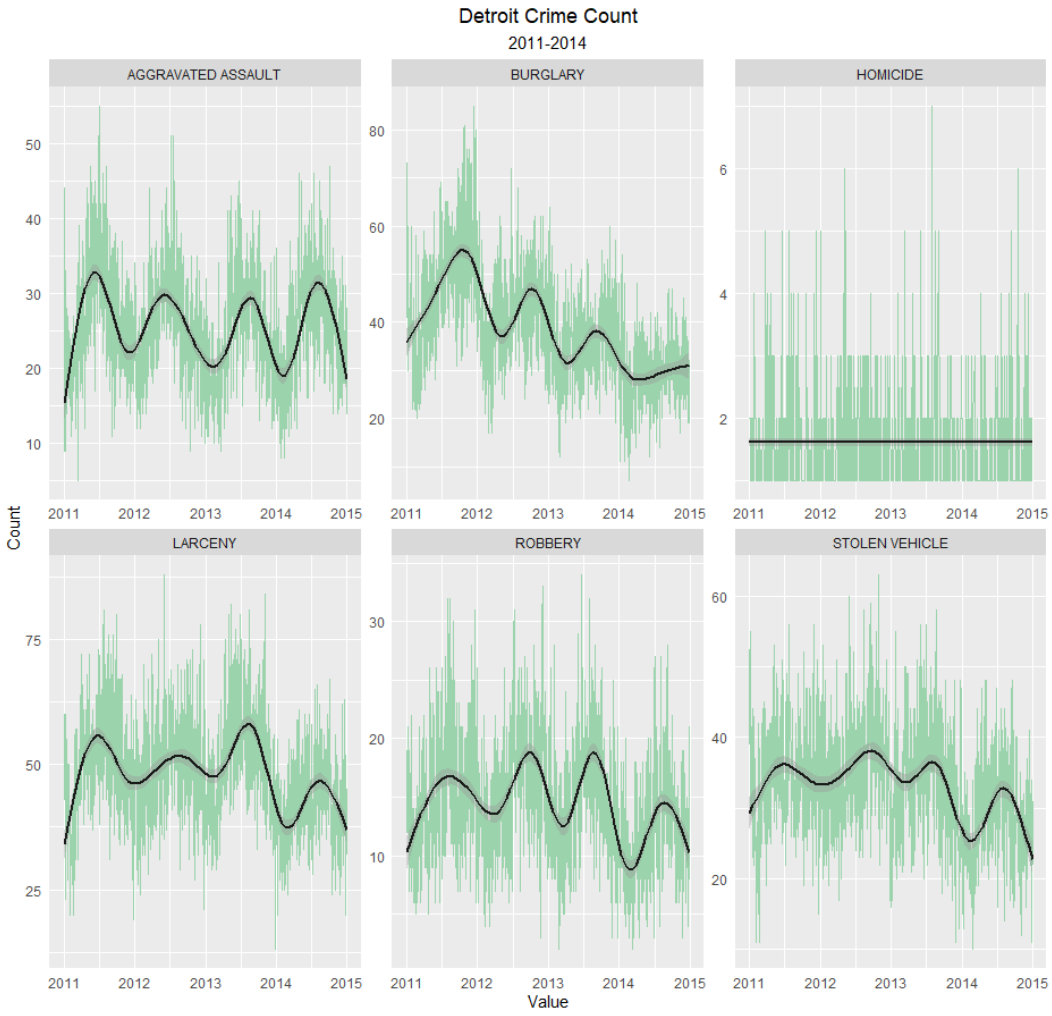
```
> todtbl<-table(crimeLOC$CATEGORY,crimeLOC$TIMEOFDAY)
> mosaicplot(todtbl,shade = T, main="")
```

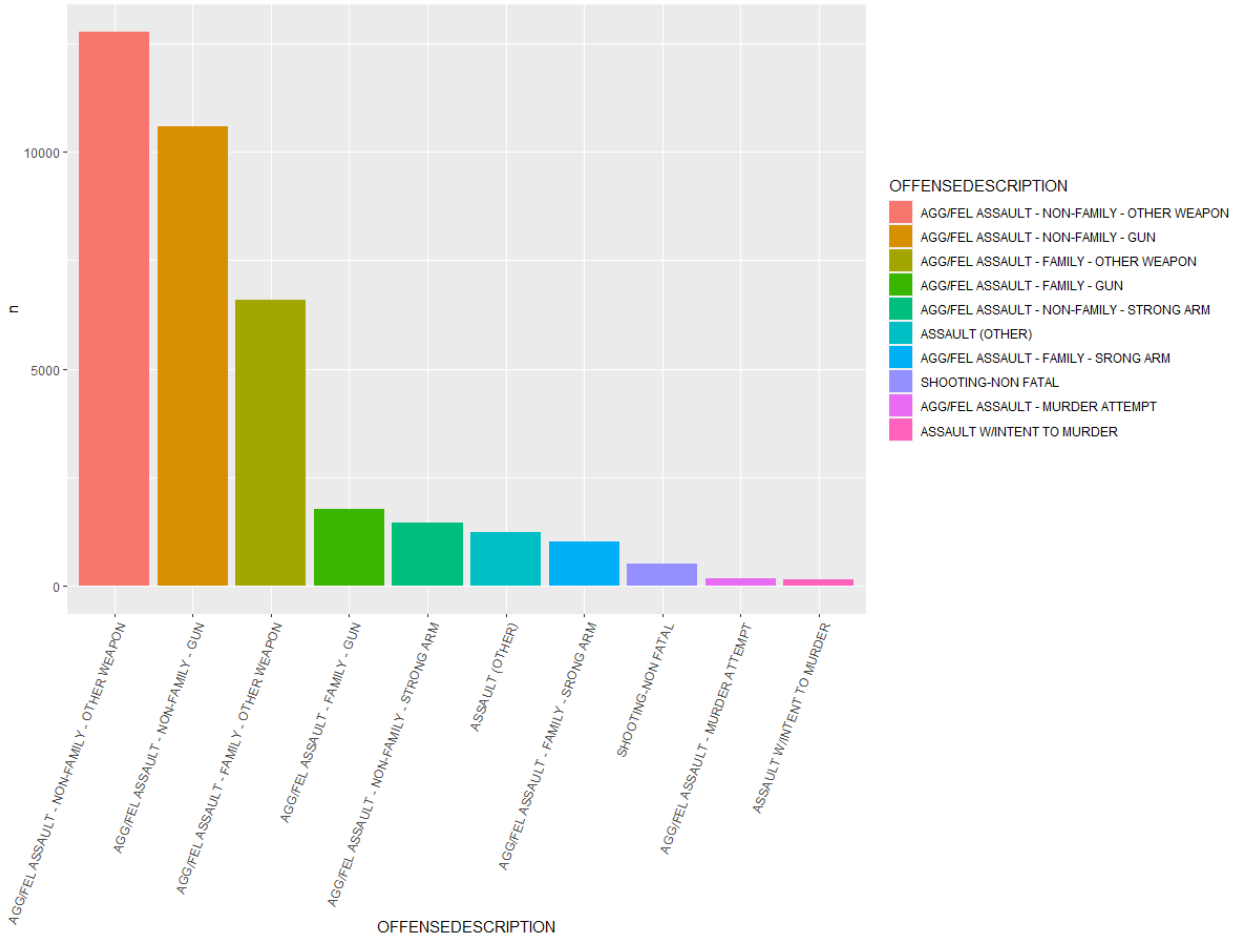


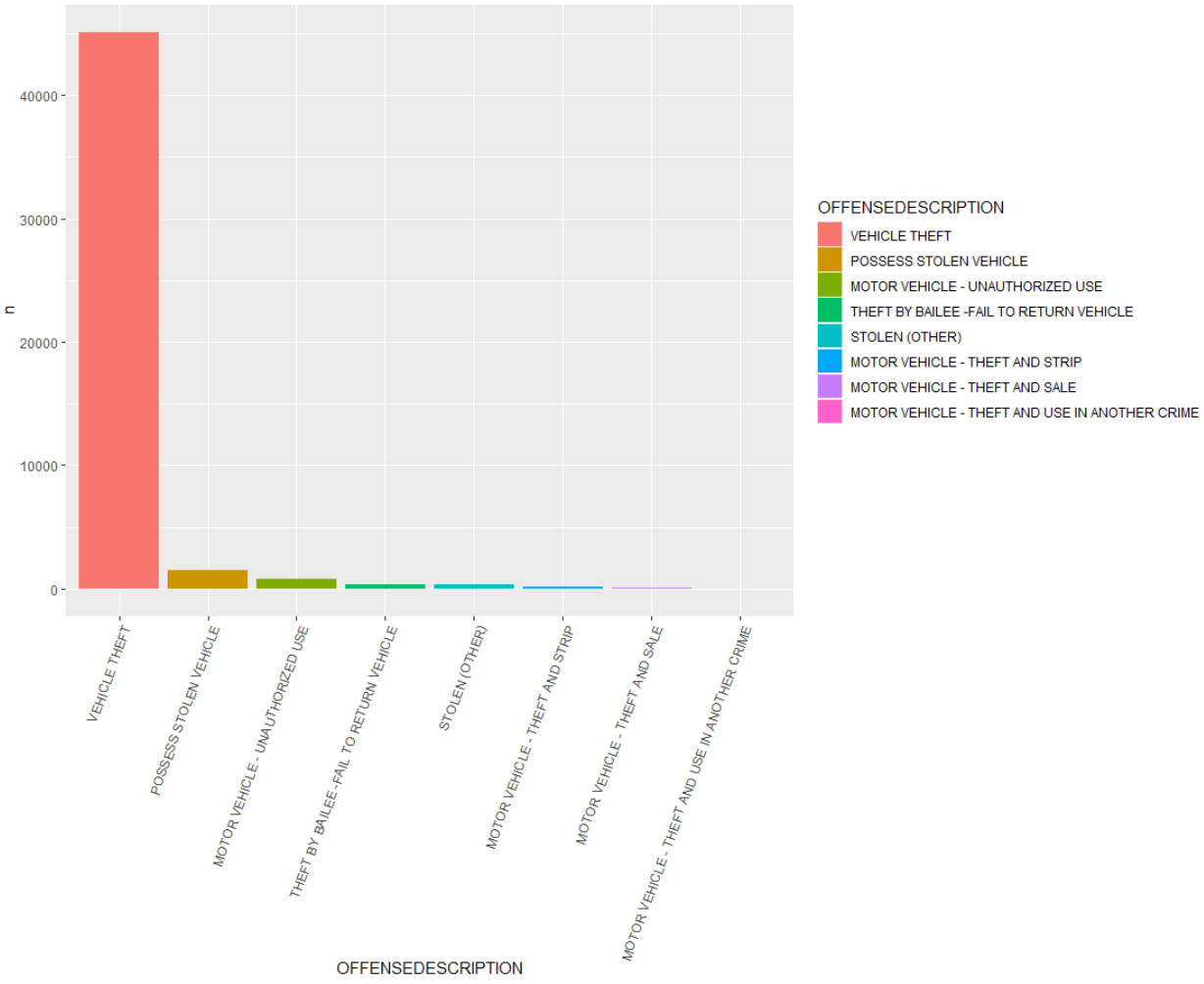
```
det_crime<-ggmap(det)+stat_density2d(aes(x = LONG, y = LAT, fill = ..level..,alpha = ..level..),data = crimeLOC,geom ="Polygon")+scale_fill_gradientn(colours =
det_crime+ggtitle("Detroit Crime Analyisis","Amount of Crime by Time of Day")+xlab("") + ylab("") + theme(axis.text.x = element_blank(),axis.text.y = element_t
```

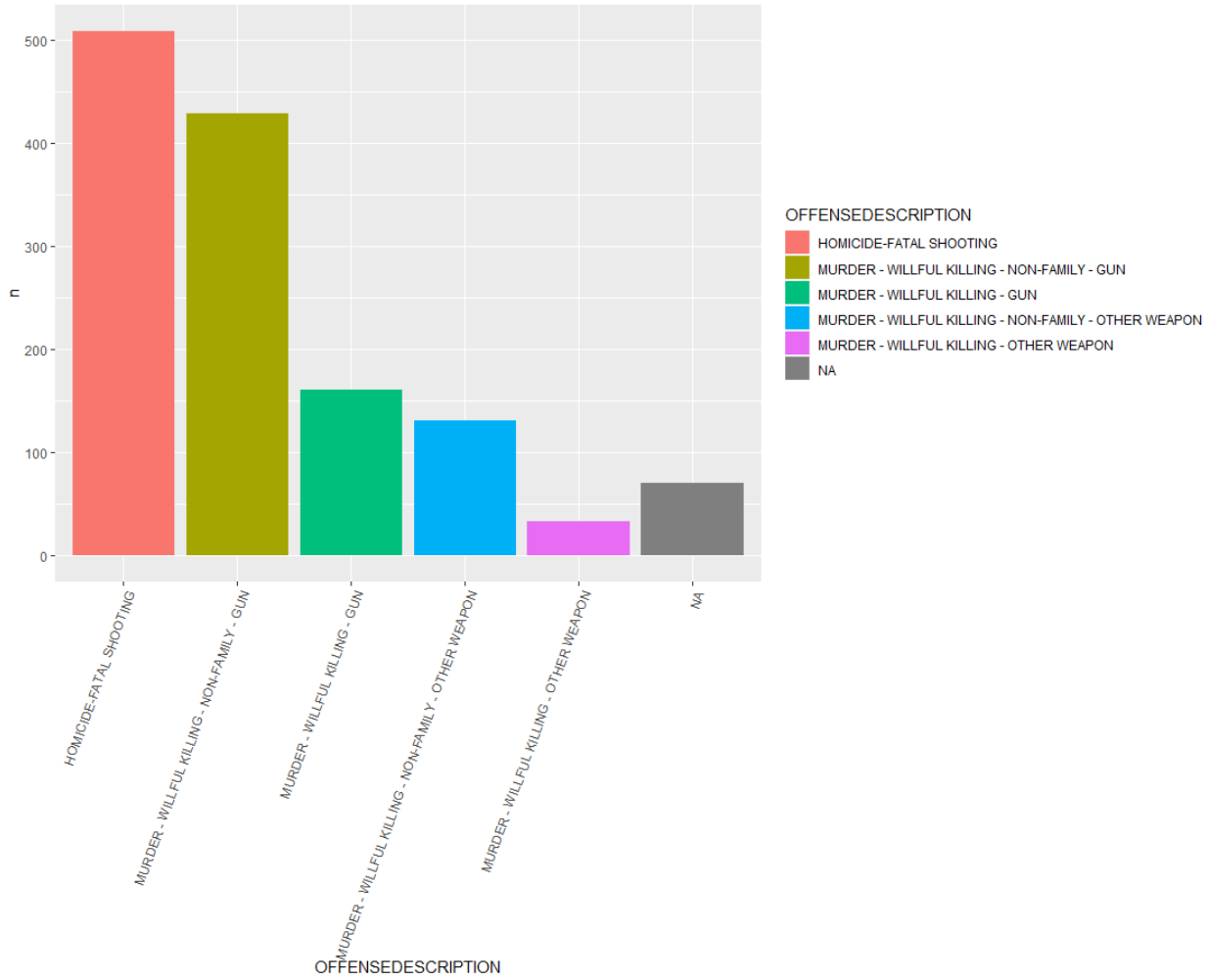
As one may suspect, the amount of crime is more prevalent at evening (5:00-9:00pm) and night (9:00pm-4:00am). This is shown in our facet wrap, that the intensity goes up. In next couple of weeks, I will hope to show the density with category of crimes.

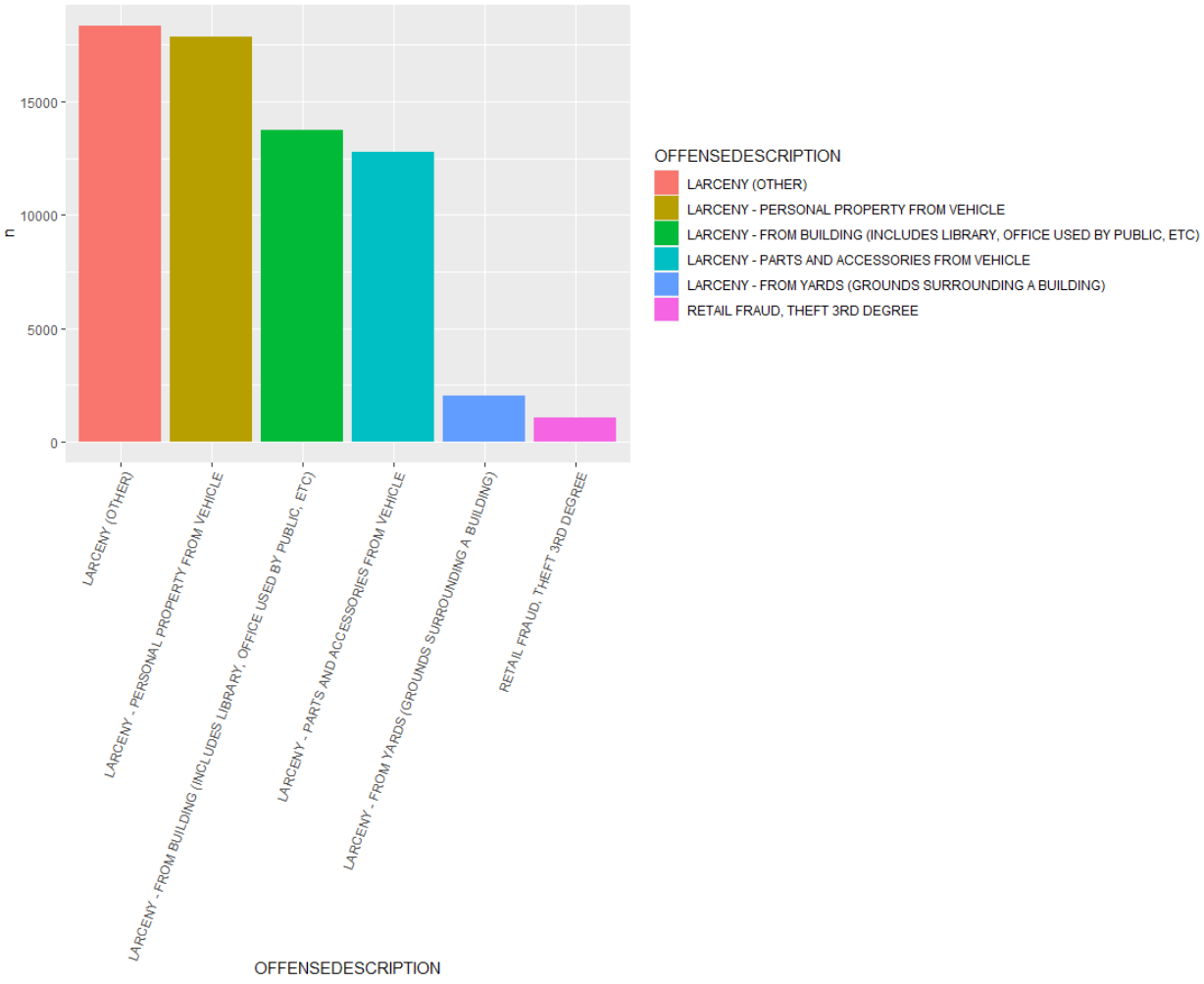
```
> table(crimeLOC$CATEGORY,crimeLOC$YEAR)

                     2011  2012  2013  2014
AGGRAVATED ASSAULT   9677  9389  8759  9144
BURGLARY            17504 14772 12840 10523
HOMICIDE              347   390   330   304
LARCENY             17889 17549 18478 15198
ROBBERY              5503  5642  5464  4304
STOLEN VEHICLE      12397 13115 12261 10358
```

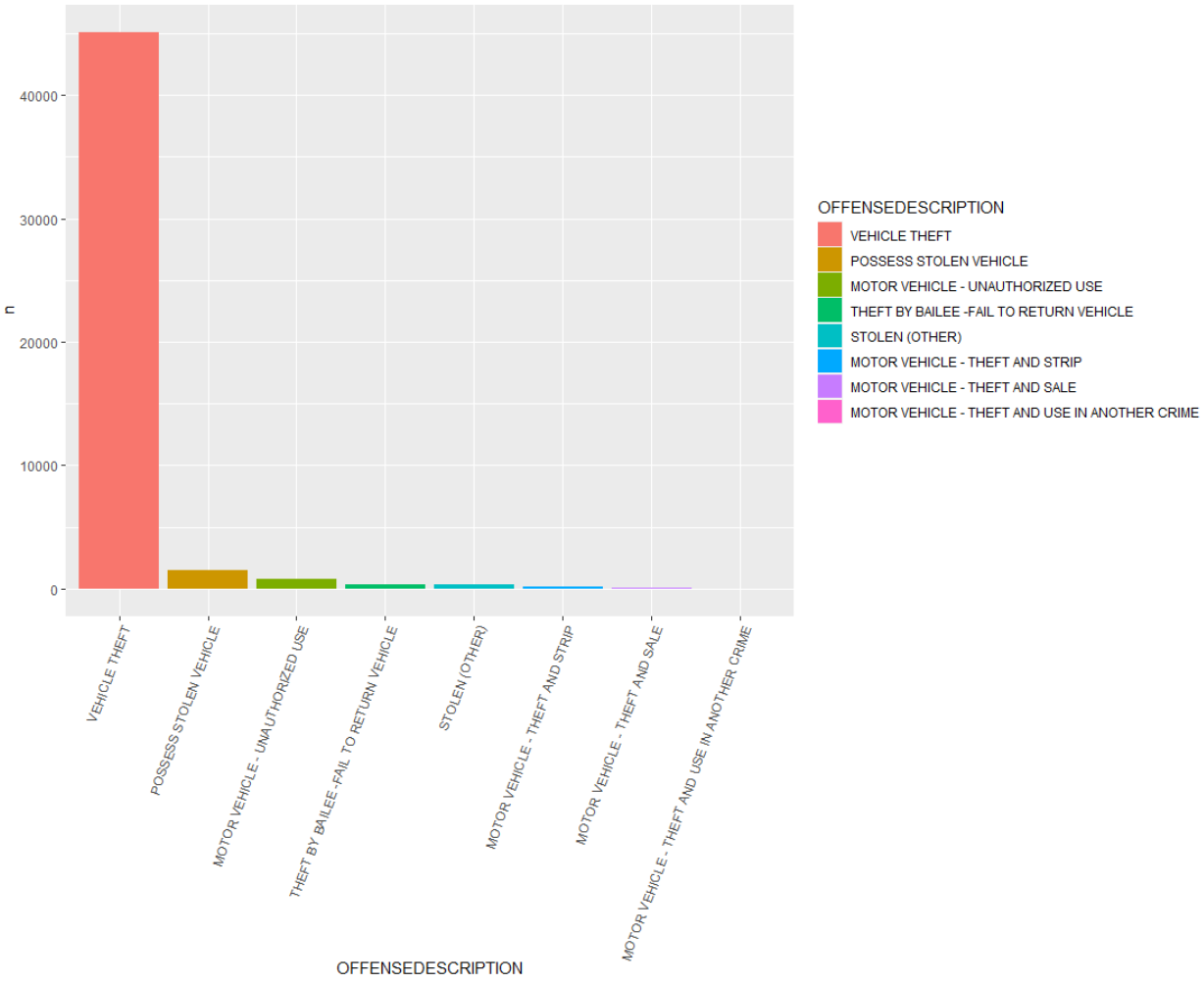



10000
5000
0
n
AGG/FEL ASSAULT - NON-FAMILY - OTHER WEAPON
AGG/FEL ASSAULT - NON-FAMILY - GUN
AGG/FEL ASSAULT - FAMILY - OTHER WEAPON
AGG/FEL ASSAULT - FAMILY - GUN
AGG/FEL ASSAULT - NON-FAMILY - STRONG ARM
ASSAULT (OTHER)
AGG/FEL ASSAULT - FAMILY - SRONG ARM
SHOOTING-NON FATAL
AGG/FEL ASSAULT - MURDER ATTEMPT
ASSAULT W/INTENT TO MURDER
OFFENSEDESCRIPTION
OFFENSEDESCRIPTION
AGG/FEL ASSAULT - NON-FAMILY - OTHER WEAPON
AGG/FEL ASSAULT - NON-FAMILY - GUN
AGG/FEL ASSAULT - FAMILY - OTHER WEAPON
AGG/FEL ASSAULT - FAMILY - GUN
AGG/FEL ASSAULT - NON-FAMILY - STRONG ARM
ASSAULT (OTHER)
AGG/FEL ASSAULT - FAMILY - SRONG ARM
SHOOTING-NON FATAL
AGG/FEL ASSAULT - MURDER ATTEMPT
ASSAULT W/INTENT TO MURDER

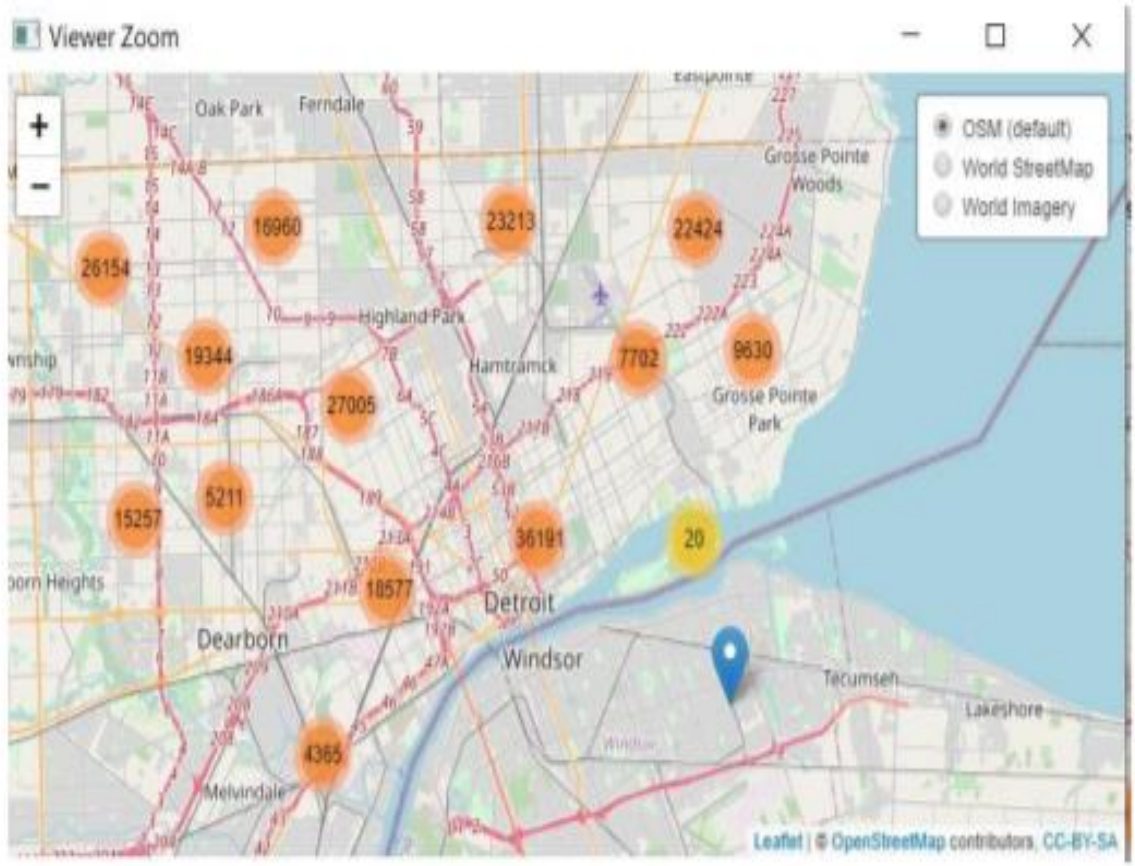

40000
30000
20000
10000
0
n
VEHICLE THEFT
POSSESS STOLEN VEHICLE
MOTOR VEHICLE - UNAUTHORIZED USE
THEFT BY BAILEE -FAIL TO RETURN VEHICLE
STOLEN (OTHER)
MOTOR VEHICLE - THEFT AND STRIP
MOTOR VEHICLE - THEFT AND SALE
MOTOR VEHICLE - THEFT AND USE IN ANOTHER CRIME
OFFENSEDESCRIPTION
OFFENSEDESCRIPTION
VEHICLE THEFT
POSSESS STOLEN VEHICLE
MOTOR VEHICLE - UNAUTHORIZED USE
THEFT BY BAILEE -FAIL TO RETURN VEHICLE
STOLEN (OTHER)
MOTOR VEHICLE - THEFT AND STRIP
MOTOR VEHICLE - THEFT AND SALE
MOTOR VEHICLE - THEFT AND USE IN ANOTHER CRIME

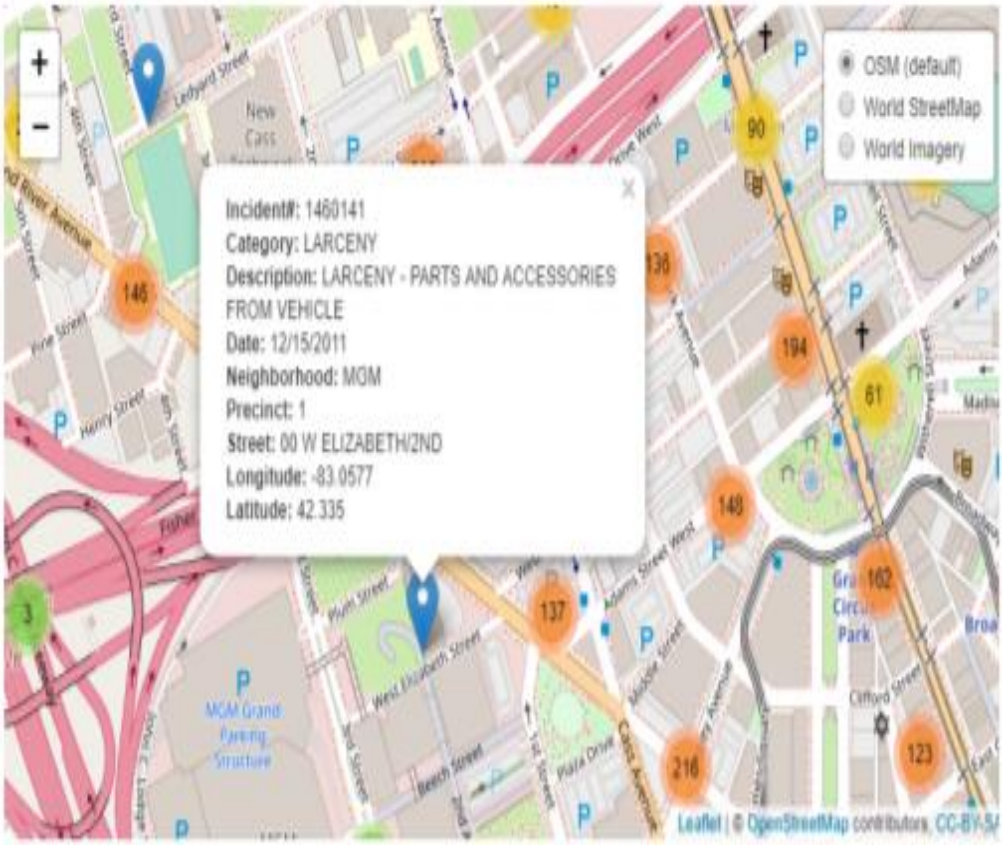

500
400
300
200
100
0
n
HOMICIDE-FATAL SHOOTING
MURDER - WILLFUL KILLING - NON-FAMILY - GUN
MURDER - WILLFUL KILLING - GUN
MURDER - WILLFUL KILLING - NON-FAMILY - OTHER WEAPON
MURDER - WILLFUL KILLING - OTHER WEAPON
NA
OFFENSEDESCRIPTION
OFFENSEDESCRIPTION
HOMICIDE-FATAL SHOOTING
MURDER - WILLFUL KILLING - NON-FAMILY - GUN
MURDER - WILLFUL KILLING - GUN
MURDER - WILLFUL KILLING - NON-FAMILY - OTHER WEAPON
MURDER - WILLFUL KILLING - OTHER WEAPON
NA


15000
10000
5000
0
n
LARCENY (OTHER)
LARCENY - PERSONAL PROPERTY FROM VEHICLE
LARCENY - FROM BUILDING (INCLUDES LIBRARY, OFFICE USED BY PUBLIC, ETC)
LARCENY - PARTS AND ACCESSORIES FROM VEHICLE
LARCENY - FROM YARDS (GROUNDS SURROUNDING A BUILDING)
RETAIL FRAUD, THEFT 3RD DEGREE
OFFENSEDESCRIPTION
OFFENSEDESCRIPTION
LARCENY (OTHER)
LARCENY - PERSONAL PROPERTY FROM VEHICLE
LARCENY - FROM BUILDING (INCLUDES LIBRARY, OFFICE USED BY PUBLIC, ETC)
LARCENY - PARTS AND ACCESSORIES FROM VEHICLE
LARCENY - FROM YARDS (GROUNDS SURROUNDING A BUILDING)
RETAIL FRAUD, THEFT 3RD DEGREE


40000
30000
20000
10000
0
n
VEHICLE THEFT
POSSESS STOLEN VEHICLE
MOTOR VEHICLE - UNAUTHORIZED USE
THEFT BY BAILEE -FAIL TO RETURN VEHICLE
STOLEN (OTHER)
MOTOR VEHICLE - THEFT AND STRIP
MOTOR VEHICLE - THEFT AND SALE
MOTOR VEHICLE - THEFT AND USE IN ANOTHER CRIME
OFFENSEDESCRIPTION
OFFENSEDESCRIPTION
VEHICLE THEFT
POSSESS STOLEN VEHICLE
MOTOR VEHICLE - UNAUTHORIZED USE
THEFT BY BAILEE -FAIL TO RETURN VEHICLE
STOLEN (OTHER)
MOTOR VEHICLE - THEFT AND STRIP
MOTOR VEHICLE - THEFT AND SALE
MOTOR VEHICLE - THEFT AND USE IN ANOTHER CRIME

Leaflet


Viewer Zoom
OSM (default)
World StreetMap
World Imagery
Oak Park
Ferndale
Grosse Pointe Woods
Highland Park
Hamtramck
Grosse Pointe Park
Detroit
Windsor
Dearborn
Tecumseh
Lakeshore
Melvindale
16960
23213
22424
26154
19344
27005
7702
9630
5211
15257
36191
20
18577
4365
Leaflet | © OpenStreetMap contributors, CC-BY-SA


OSM (default)
World StreetMap
World Imagery
Incident#: 1460141
Category: LARCENY
Description: LARCENY - PARTS AND ACCESSORIES FROM VEHICLE
Date: 12/15/2011
Neighborhood: MGM
Precinct: 1
Street: 00 W ELIZABETH/2ND
Longitude: -83.0577
Latitude: 42.335
MGM Grand Parking Structure
Leaflet | © OpenStreetMap contributors, CC-BY-SA