homework iii

Vivek Panchal,Rohit Kunjilikattil

2019-09-24

Introduction

In this assignment we perform exploratory analysis on our nyc311 dataset.We have perfomed analysis by taking two appropriate columns from our dataset which gave us answers to what we tried to visualize. For example, we found relationship between complaints and their statuses in different boroughs. For this we made use of two columns namely,`Borough` and `Complaint Type`. Our aim was to visualize and obtain as many answers as we could from the relation between two columns from the nyc311 dataset. ‘

Initialization

Here we load the tidyverse packages and the `data.table` package and load the nyc311 data set. Then we fix the column names of the nyc311 data so that they have no spaces.

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

In order to visualize maps, we make use of “ggmap”. with the help of this package, it is easy to retrieve map tiles from sources such as Google Maps, Open Street Maps and many more. They have the necessary tools which helps in smooth functioning for our routing. The two columns which are very essential to visualize any maps are `Latitude` and `Longitude`. Using these columns we expanded our imagination to find suitable relations using them.

For the purpose of our visualization we need to enable google static map service. If this service was to be used before July 2017, we didn’t need to signup for any google map services. But now it requires setting up an API Key and enabling billing. While signing up for the service you need to copy your API key which will be added to your chunk.

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------
```

```
## v tibble  2.1.3     v purrr   0.3.2
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts ---------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
install.packages('tidyverse', repos = "http://cran.us.r-project.org")
```

```
##
## The downloaded binary packages are in
##  /var/folders/b8/hxw52n9d3wz9th1p6y6sb8zh0000gn/T//Rtmp9Bkp0Z/downloaded_packages
```

```
library(data.table)
```

```
##
## Attaching package: 'data.table'

## The following object is masked from 'package:purrr':
##
##     transpose

## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
nyc311<-fread("311_Service_Requests_from_2010_to_Present.csv")
options("scipen" = 100,"digits" = 4)
```

EXPLORATION 1

```
nyc311_map<-nyc311[sample(nrow(nyc311),10000),]
nyc_map_data <- nyc311_map %>% select(24,50,51,20)

nyc_map_data<- nyc_map_data %>% filter(!str_detect(Borough, "Unspecified"))
nyc_map_data<-na.omit(nyc_map_data)
xtab<-dplyr::filter(nyc_map_data,Status == "Open")

#library(devtools)
#devtools::install_github("fresques/ggmap")
# devtools::install_github("dkahle/ggmap")

library(ggmap)
```

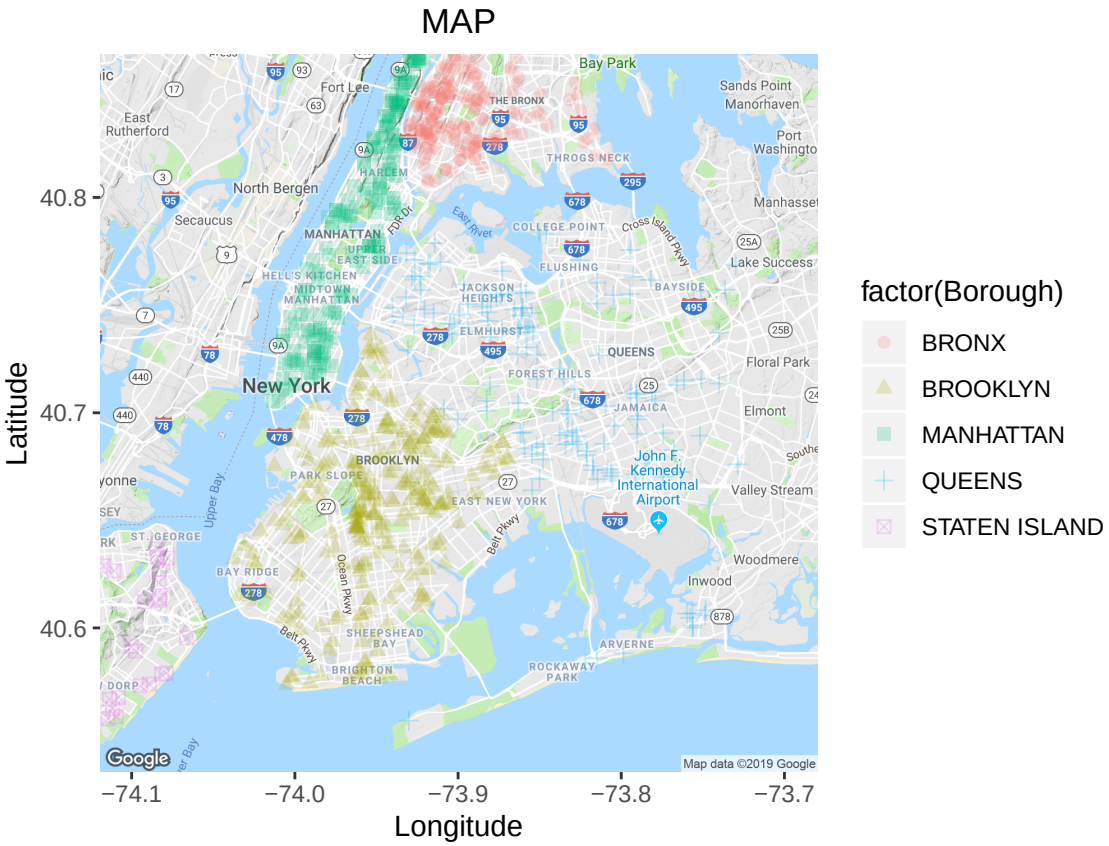
```
## Google's Terms of Service: https://cloud.google.com/maps-platform/terms/.

## Please cite ggmap if you use it! See citation("ggmap") for details.
```

```
key<-'AIzaSyD8Zu8HBooWB2MP9yHqbXOA5ERj5oTQ-mA'
register_google(key ="AIzaSyD8Zu8HBooWB2MP9yHqbXOA5ERj5oTQ-mA")
nyc_map = get_map(location = c(lon = -73.9, lat = 40.7),
                  zoom = 11, maptype="terrain")
```

```
## Source : https://maps.googleapis.com/maps/api/staticmap?center=40.7,-73.9&zoom=11&size=640x640&scale=
```

```
map<-ggmap(nyc_map) + geom_point(data=xtab,aes(x=xtab$Longitude,y=xtab$Latitude,shape =factor(Borough),
map
```

```
## Warning: Removed 109 rows containing missing values (geom_point).
```

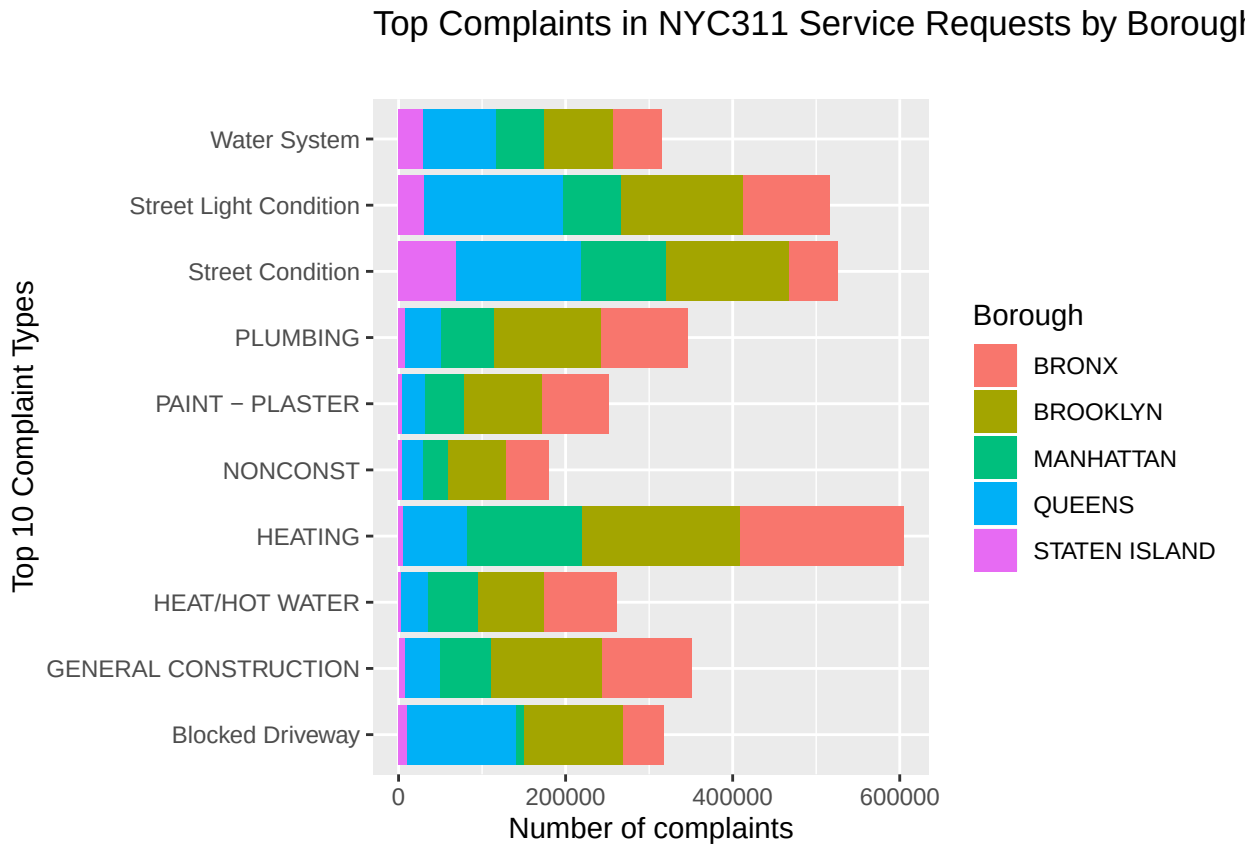


Location data is very helpful in visualization. We use the ggmap library in order **to find out which locations have complaints with status as "OPEN".** This can help the official identify target areas which need more attention. Staff management can also be done based on this. For example, if a particular location has more open complaints, more staff should be assigned to that area.

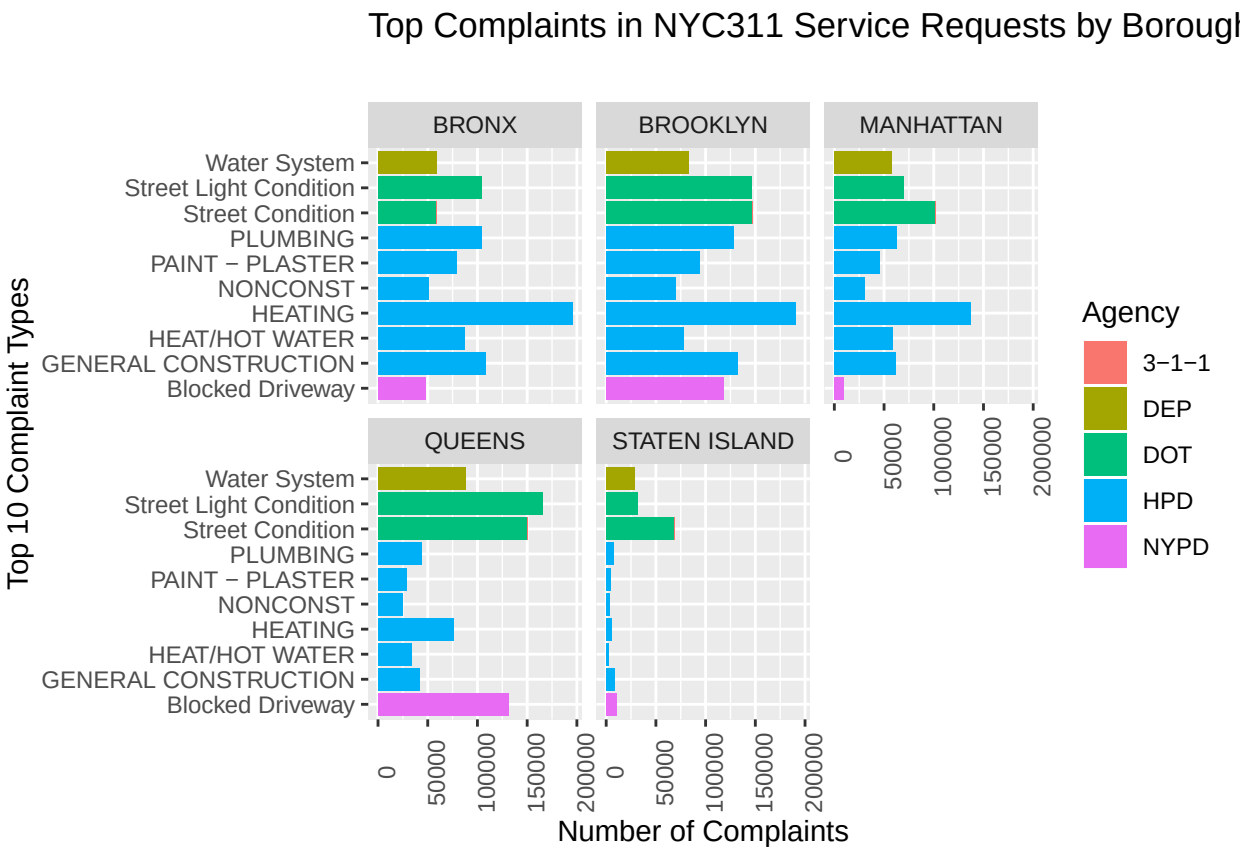
In the above Map we have selected four columns `Borough`,`Latitude`,`Longitude`,`Status`.For our visualization we have used map type "terrain".In our dataset there are few rows where the name of Borough is "unspecified" so we have omitted it. This will answer the question when someone wants to visualize or extract information regarding the complaints that have status as "Open" by looking at the Boroughs.We have distinguised boroughs using different colours.

EXPLORATION 2

```
nyc311_subset <- subset(nyc311,`Complaint Type` %in% count(nyc311,`Complaint Type`,sort = T)[1:10,]$`Co
nyc311_subset <- nyc311_subset %>% select(`Complaint Type`,Borough,Status,Agency)  %>% group_by(Borough
nyc311_subset <- nyc311_subset %>% filter(!str_detect(Borough, "Unspecified"))
ggplot(nyc311_subset, aes(`Complaint Type`)) + geom_bar(stat = "count")  + labs(x = "Top 10 Complaint Ty
                              y = "Number of complaints",
                              title = "Top Complaints in NYC311 Service Requests by Borough",
                              subtitle = "")  +coord_flip()  + aes(fill=Borough)
```




```
ggplot(nyc311_subset, aes(`Complaint Type`)) +
  geom_bar(stat = "count") +
  facet_wrap(~Borough) + labs(x = "Top 10 Complaint Types",
                              y = "Number of Complaints",
                              title = "Top Complaints in NYC311 Service Requests by Borough ",
                              subtitle = "") + coord_flip()  + aes(fill=Agency) + theme(axis.text.x = el
```

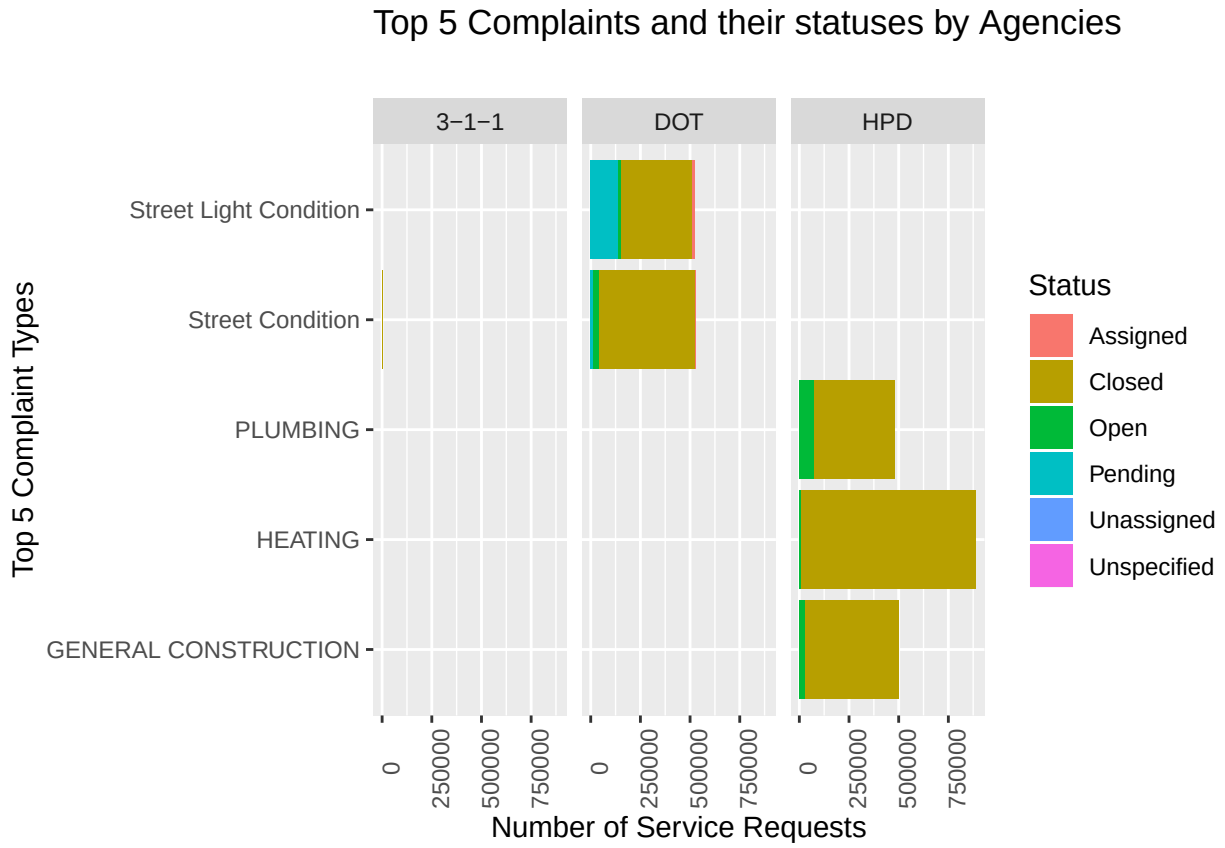


Getting the numbers for the top complaints for all the borough can be very educational in terms of prevention. By studying this data the government can take respective preventive measures in each borough based on the top complaints registered in that borough. This exploration helps in **figuring out what preventive measures should be taken in which borough** . The first figure we look at an overview of distribution of complaints across different boroughs, while the second graph is a closer look at the situation in each borough. For the second graph, we have also factored the color scheme based on agencies. This can help in showing which agency is responsible for handling which type of complaint.

For this exploration we have made use of package ggplot. We have made use of facets that creates trellis graphs with x,y labs showing horizontal and vertical axis labels. The graph is a bar plot of the top 10 complaints against their count, color-factored by borough, while the second graph shows relation between the Number of complaints, Complaints and their agencies .We have picked top 10 complaint types and have shown their status in each borough We have made use of aes() function of R for our visualization. It helps in constructing aesthethic mappings which are set in ggplot.The above visualization shows relationship between boroughs and complaints.

EXPLORATION 3

```
nyc311_subset2 <- subset(nyc311, `Complaint Type` %in% count(nyc311,`Complaint Type`,sort = T)[1:5,]$`C
nyc311_subset2 <- nyc311_subset2 %>% select(`Complaint Type`,`Agency`,`Status`)
#nyc311_subset2 <- nyc311_subset2 %>% filter(!str_detect(Borough, "Unspecified"))
ggplot(nyc311_subset2, aes(`Complaint Type`)) + geom_bar(stat = "count") + coord_flip()    + theme_get()
```



One of the most important factors for government agencies is their efficiency. The gov ernment can use the following exploration **to keep a check on the efficiency of the different agencies** . This exploration shows the different types of complaints handled by the agencies along with their statuses. Comparing the number of open and closed complaints, the government can get an idea about the efficiency or inefficiency of a particular agency.

In this visualization we answer which agency has received the maximum number of top 5 complaint types. We intended to show what relation can be shown between an Agency and Complaint Type.

EXPLORATION 4

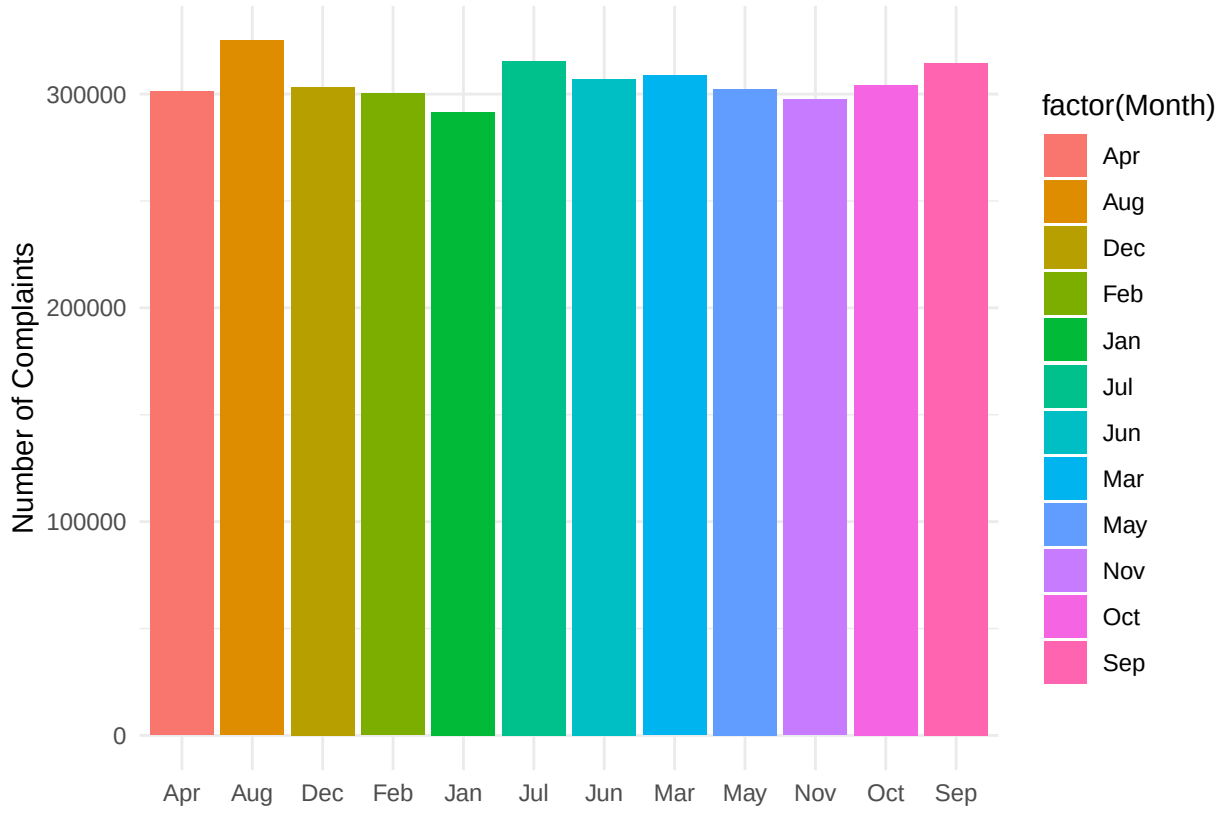
```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:data.table':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday,
##     week, yday, year
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
nyc311_subset3 <- nyc311 %>% select(`Created Date`,`Complaint Type`,Borough)
nyc311_subset3$Date <- as.POSIXct(nyc311$`Created Date`, format = "%d/%m/%Y %I:%M:%S %p")
nyc311_subset3$Month <- month(as.Date(nyc311_subset3$Date))
mymonths <- c("Jan","Feb","Mar",
              "Apr","May","Jun",
              "Jul","Aug","Sep",
              "Oct","Nov","Dec")

nyc311_subset3$Month <- mymonths[nyc311_subset3$Month]
nyc311_subset3$Year <- year(as.POSIXct(nyc311$`Created Date`, format = "%d/%m/%Y %I:%M:%S %p"))
nyc311_subset3$day <- weekdays(as.Date(nyc311_subset3$Date))
nyc311_subset3<- na.omit(nyc311_subset3)

nyc311_subset3_by_month <- nyc311_subset3 %>%
  group_by(Month) %>%
  summarise(count=n())
ggplot(nyc311_subset3_by_month) + geom_bar(aes(x=nyc311_subset3_by_month$Month , y = count
),
stat = "identity") + theme_minimal() +
  xlab("") + ylab("Number of Complaints")  + aes(fill=factor(Month))
```

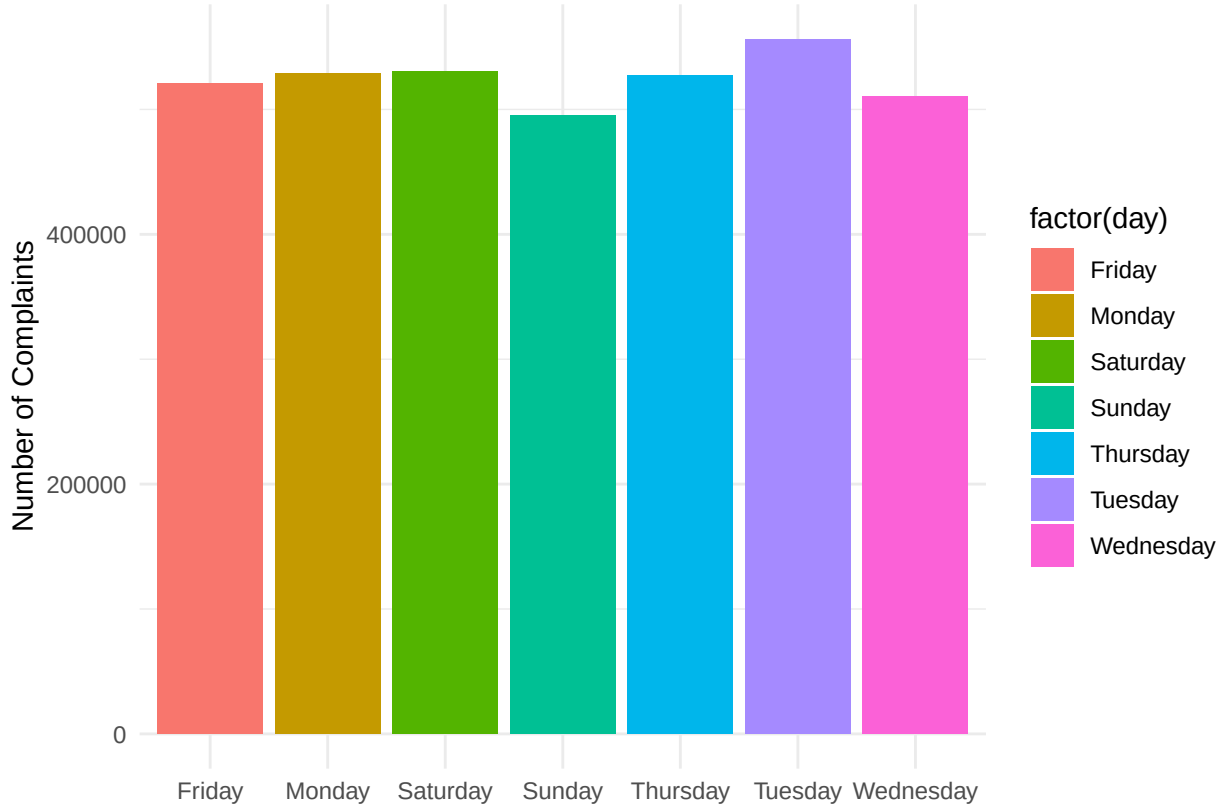


This exploration is completely based on the registration time of a complaint. With this exploration, we hope to acheive some insights about the time when most complaints are registered. **Here, we take a look**

at the data month wise to understand which month has the most complaints registered. Then authorities can use this figure to better equip themselves during that particular month. This can be a very productive preventive measure.

EXPLORATION 5

```
nyc311_subset3_by_weekday <- nyc311_subset3 %>%
  group_by(day) %>%
  summarise(count=n())
ggplot(nyc311_subset3_by_weekday) + geom_bar(aes(x=nyc311_subset3_by_weekday$day , y = count
),
stat = "identity") + theme_minimal() +
  xlab("") + ylab("Number of Complaints")  + aes(fill=factor(day))
```



This exploration is also based on the registration time of a complaint. With this exploration, we hope to dwell deeper and look at the data at a more minute level. **Here, we take a look at the data day wise to understand which day has the most complaints registered.**. In this way, the authorities can be prepared for the day when most complaints are registered. One way they can better prepare is that they can plan their weekly holiday around this day to make sure that there is enough staff to respond to the complaints.

CONCLUSION

In this phase of data exploration, we dive deeper into aur datasets to find meaningful relationships among the columns in our dataset which can be used for answering some empirical questions about the data. We also explored the library ggmap and learnt how to use the google maps api's for plotting maps for the location related data in our dataset. One of the other things we explored in details was the library ggplot, which was used for plotting all the graphs. Our first exploration shows us the location concentration of open complaints in NYC. Moving on, in the next exploration both the graphs help in painting a picture of the complaint types, their total count in each borough as well for each agency. The third exploration shows us how efficient an agency is in handling and closing complaints. Finally, the fourth and fifth exploration perform analysis on the registration time of the complaint and reveal a particular month(in exploration 4), and a particular day(in exploration 5) when the highest amount of complaints were registered.