Chicago_Crimes_2015

Rawan Hammad

1/10/2022

In this R Markdown file, I'll be going over my methodology of producing the visualizations and results presented in my report.

First, I'll start by loading each library I may need.

```
library(plotly)
library(dplyr)
library(tidyverse)
library(gridExtra)
library(ggrepel)
library(GGally)
library(ggmap)
library(readxl)
```

Next, I downloaded and read-in the 2015 crimes data from this website: https://data.cityofchicago.org/Public-Safety/Crimes-2015/vwwp-7yr9

```
Crimes_2015 <- read.csv("C:/Users/rawan/OneDrive/Desktop/DePaul/Job
Assessments/Civis Analytics/Crimes_-_2015.csv")
summary(Crimes_2015)
```

```
##        ID           Case.Number           Date              Block
##  Min.   :   21714   Length:264613      Length:264613      Length:264613
##  1st Qu.:10033741   Class :character   Class :character   Class :character
##  Median :10148588   Mode  :character   Mode  :character   Mode  :character
##  Mean   :10136841
##  3rd Qu.:10262962
##  Max.   :12585297
##
##      IUCR           Primary.Type       Description
Location.Description
##  Length:264613      Length:264613      Length:264613      Length:264613
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##     Arrest            Domestic             Beat         District
##  Length:264613      Length:264613      Min.   : 111   Min.   : 1.00
##  Class :character   Class :character   1st Qu.: 612   1st Qu.: 6.00
##  Mode  :character   Mode  :character   Median :1023   Median :10.00
```

```
##                                        Mean   :1144   Mean   :11.21
##                                        3rd Qu.:1654   3rd Qu.:16.00
##                                        Max.   :2535   Max.   :31.00
##
##       Ward       Community.Area    FBI.Code          X.Coordinate
##  Min.   : 1.00   Min.   : 1.00   Length:264613      Min.   :1094231
##  1st Qu.:10.00   1st Qu.:23.00   Class :character   1st Qu.:1152410
##  Median :23.00   Median :32.00   Mode  :character   Median :1166064
##  Mean   :22.81   Mean   :37.58                      Mean   :1164457
##  3rd Qu.:34.00   3rd Qu.:57.00                      3rd Qu.:1176389
##  Max.   :50.00   Max.   :77.00                      Max.   :1205111
##  NA's   :2                                          NA's   :6702
##   Y.Coordinate          Year       Updated.On          Latitude
##  Min.   :1813897   Min.   :2015   Length:264613      Min.   :41.65
##  1st Qu.:1858595   1st Qu.:2015   Class :character   1st Qu.:41.77
##  Median :1891472   Median :2015   Mode  :character   Median :41.86
##  Mean   :1885560   Mean   :2015                      Mean   :41.84
##  3rd Qu.:1908452   3rd Qu.:2015                      3rd Qu.:41.91
##  Max.   :1951523   Max.   :2015                      Max.   :42.02
##  NA's   :6702                                        NA's   :6702
##    Longitude        Location
##  Min.   :-87.93   Length:264613
##  1st Qu.:-87.72   Class :character
##  Median :-87.67   Mode  :character
##  Mean   :-87.67
##  3rd Qu.:-87.63
##  Max.   :-87.53
##  NA's   :6702
```

I'll now start pre-processing this data. First, I'll save the file in a new dataframe called chicago2015_df. Next, I renamed the primary.type to offense for clarity, removed unneeded columns such as ID, case number, description, year, and the UTM coordinates. The ID and case number are unique and not needed in such an analysis. The descriptions aren't needed either since I'll be focusing more on the offense itself. We do not need the year column since all the data provided is of the year 2015. The UTM X and Y coordinates aren't needed since we're already provided with the latituded and logitude of each crime.

```
#saving the file in a new dataframe called chicago2015_df
chicago2015_df <- Crimes_2015

#rename primary type to offense for clarity
chicago2015_df <- chicago2015_df %>% rename(offense = Primary.Type)

#removed ID, case number, description, year, X and Y coordinates
chicago2015_df <- chicago2015_df %>% select(!c(ID, Case.Number, Description,
Year, X.Coordinate, Y.Coordinate))

#converted blank values to NA
chicago2015_df <- chicago2015_df %>% mutate_all(list(~na_if(.,"")))
```

We can clean it up a bit more by removing any data with missing values or NA. This dataset is quite large so, as long as we're not compromising the quality of the dataset, we should be good to go!

```
#check for any missing values
chicago2015_df %>% summarise_all(~sum(is.na(.)))
```

```
##   Date Block IUCR offense Location.Description Arrest Domestic Beat
District
## 1    0     0    0       0                  573      0        0    0
0
##   Ward Community.Area FBI.Code Updated.On Latitude Longitude Location
## 1    2              0        0          0     6702      6702     6702
```

```
#total number of entries with missing values
rows_before <- nrow(chicago2015_df)
sprintf("Total Rows: %d", rows_before)
```

```
## [1] "Total Rows: 264613"
```

We have missing fields in location.description, wards, longitude, and latitude, and we have 264613 crimes total. Thus, we can safely remove rows with missing data without compromising the quality of our dataset.

```
#removed all NA values
chicago2015_df <- chicago2015_df[complete.cases(chicago2015_df),]

#confirm that they're actually removed
chicago2015_df %>% summarise_all(~sum(is.na(.)))
```

```
##   Date Block IUCR offense Location.Description Arrest Domestic Beat
District
## 1    0     0    0       0                    0      0        0    0
0
##   Ward Community.Area FBI.Code Updated.On Latitude Longitude Location
## 1    0              0        0          0        0         0        0
```

```
rows_before <- nrow(chicago2015_df)
sprintf("Total Rows: %d", rows_before)
```

```
## [1] "Total Rows: 257768"
```

We're now down to 257768 crimes total and have successfully remove all rows with missing data.

As for the last step, I'll create a separate column for the months to help calculate the frequency of crimes per month.

```
#getting the first two numbers from the Date column and inputting them into a
new column
chicago2015_df$months <- substr(chicago2015_df$Date, 1, 2)
head(chicago2015_df)
```

```
##                    Date                 Block IUCR   offense
## 1 09/05/2015 01:30:00 PM      043XX S WOOD ST 0486   BATTERY
## 2 09/04/2015 11:30:00 AM  008XX N CENTRAL AVE 0870     THEFT
## 3 09/05/2015 12:45:00 PM    035XX W BARRY AVE 2023 NARCOTICS
## 4 09/05/2015 01:00:00 PM  0000X N LARAMIE AVE 0560   ASSAULT
## 5 09/05/2015 10:55:00 AM  082XX S LOOMIS BLVD 0610  BURGLARY
## 6 09/04/2015 06:00:00 PM 021XX W CHURCHILL ST 0620  BURGLARY
##   Location.Description Arrest Domestic Beat District Ward Community.Area
## 1            RESIDENCE  false     true  924        9   12             61
## 2              CTA BUS  false    false 1511       15   29             25
## 3             SIDEWALK   true    false 1412       14   35             21
## 4            APARTMENT  false     true 1522       15   28             25
## 5            RESIDENCE  false    false  614        6   21             71
## 6     RESIDENCE-GARAGE  false    false 1434       14   32             24
##   FBI.Code             Updated.On Latitude Longitude
## 1      08B 02/10/2018 03:50:01 PM 41.81512 -87.67000
## 2       06 02/10/2018 03:50:01 PM 41.89508 -87.76540
## 3       18 02/10/2018 03:50:01 PM 41.93741 -87.71665
## 4      08A 02/10/2018 03:50:01 PM 41.88190 -87.75512
## 5       05 02/10/2018 03:50:01 PM 41.74438 -87.65843
## 6       05 02/10/2018 03:50:01 PM 41.91464 -87.68163
##                       Location months
## 1 (41.815117282, -87.669999562)     09
## 2 (41.895080471, -87.765400451)     09
## 3 (41.937405765, -87.716649687)     09
## 4 (41.881903443, -87.755121152)     09
## 5 (41.744378879, -87.658430635)     09
## 6 (41.914635603, -87.681630909)     09
```

I think I'm done preprocessing this data and I'm ready to analyze it. First, I'll create a visual to see the 20 most frequent crimes in Chicago by offense.

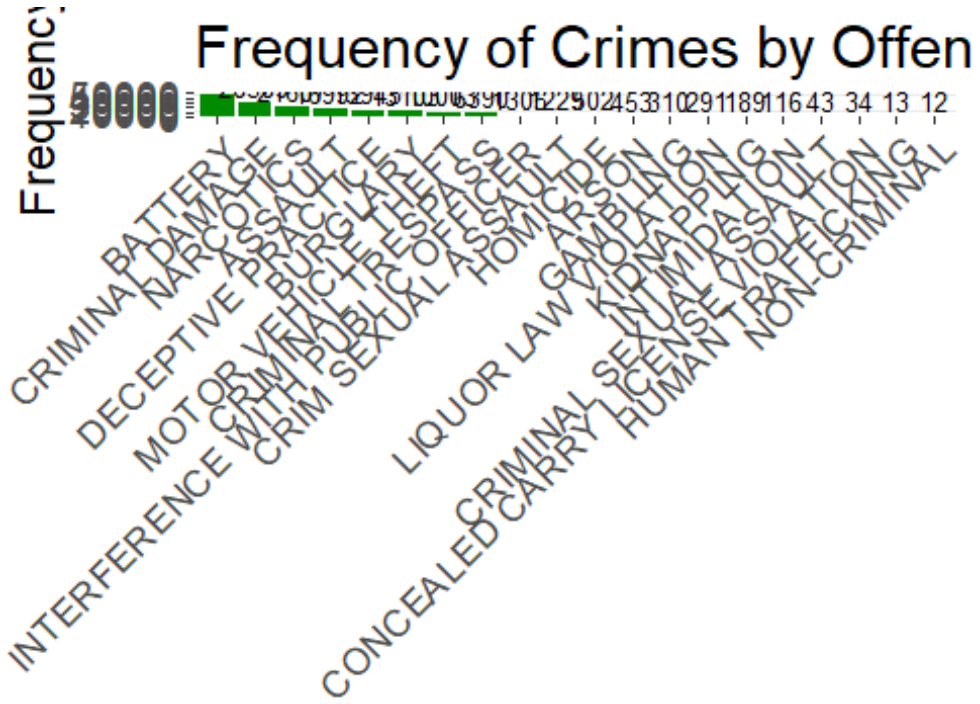
```
#create a new table for crime frequencies
crime_freq <- as.data.frame(table(chicago2015_df$offense))
crime_freq <- crime_freq %>% filter(Freq>0)

#get the top 20 crimes
crime_freq <- crime_freq[1:20,]

#set plot size
options(repr.plot.width = 14, repr.plot.height = 8)

#create the bar plot
freq_bar <- ggplot(crime_freq, aes(x=reorder(Var1,-Freq), y=Freq)) +
                    geom_bar(stat = "identity", fill="green4") +
                    labs(title="Frequency of Crimes by Offense (top 20)",
x="", y="Frequency") +
                    geom_text(aes(label=Freq), vjust=-0.3, size=3.5)+
                    theme(axis.text.x = element_text(angle = 45, hjust = 1),
```

```
                          text = element_text(size = 18))
freq_bar
```



We can also get these in a numerical format for all offenses:

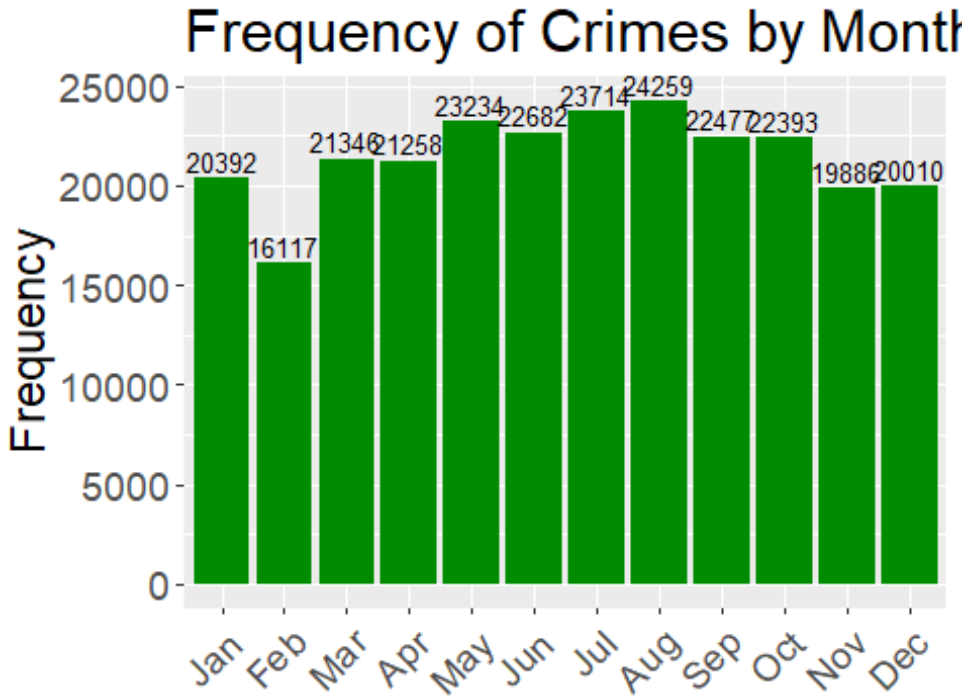
```
chicago2015_df %>% group_by(offense) %>% summarise("count"=n())
```

```
## # A tibble: 33 x 2
##    offense                             count
##    <chr>                               <int>
##  1 ARSON                                 453
##  2 ASSAULT                             16992
##  3 BATTERY                             48821
##  4 BURGLARY                            13103
##  5 CONCEALED CARRY LICENSE VIOLATION      34
##  6 CRIM SEXUAL ASSAULT                  1229
##  7 CRIMINAL DAMAGE                     28589
##  8 CRIMINAL SEXUAL ASSAULT                43
##  9 CRIMINAL TRESPASS                    6390
## 10 DECEPTIVE PRACTICE                  13945
## # ... with 23 more rows
```

Let's visualize what the frequency of crimes looks like each month.

```
#create a new table
crime_freq_month <- as.data.frame(table("month" = chicago2015_df$months))
monthnames <- c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug",
"Sep", "Oct", "Nov", "Dec")
```

```
#plot the frequency per month
month_bar <- ggplot(crime_freq_month, aes(y=Freq, x=month)) +
    geom_bar(stat="identity", fill="green4") +
    scale_x_discrete(label=monthnames) +
    labs(title="Frequency of Crimes by Month", x="", y="Frequency") +
  geom_text(aes(label=Freq), vjust=-0.3, size=3.5)+
    theme(axis.text.x = element_text(angle = 45, hjust = 1),
          text = element_text(size = 18), legend.position = "none")
month_bar
```



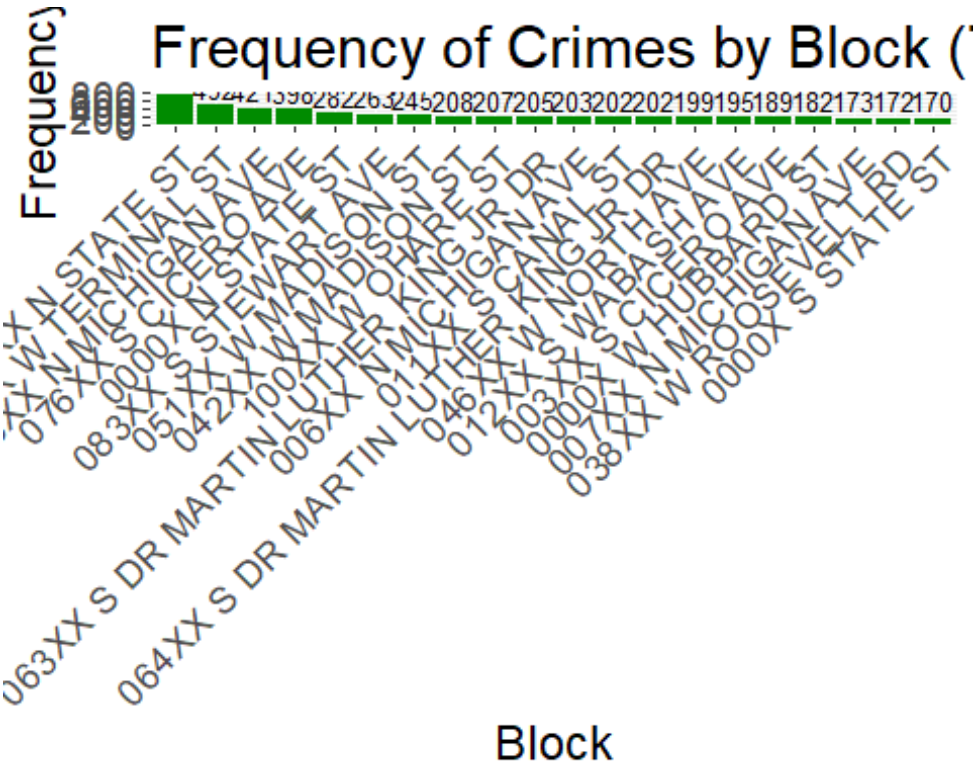
It might be interesting to look into crimes per block.

```
#Now let's look into the frequency of crimes by block.

crime_by_block <- as.data.frame(table("Block" = chicago2015_df$Block))
crime_by_block <- crime_by_block %>%
                    filter(Freq>0) %>%
                    arrange(desc(Freq))
crime_by_block <- crime_by_block[1:20,]

# plot
block_bar <- ggplot(crime_by_block, aes(y=Freq, x=reorder(Block,-Freq))) +
                    geom_bar(stat="identity", fill="green4") +
                    labs(title="Frequency of Crimes by Block (Top 20)",
                         x="Block", y="Frequency") +
                    geom_text(aes(label=Freq), vjust=-0.3, size=3.5)+
```

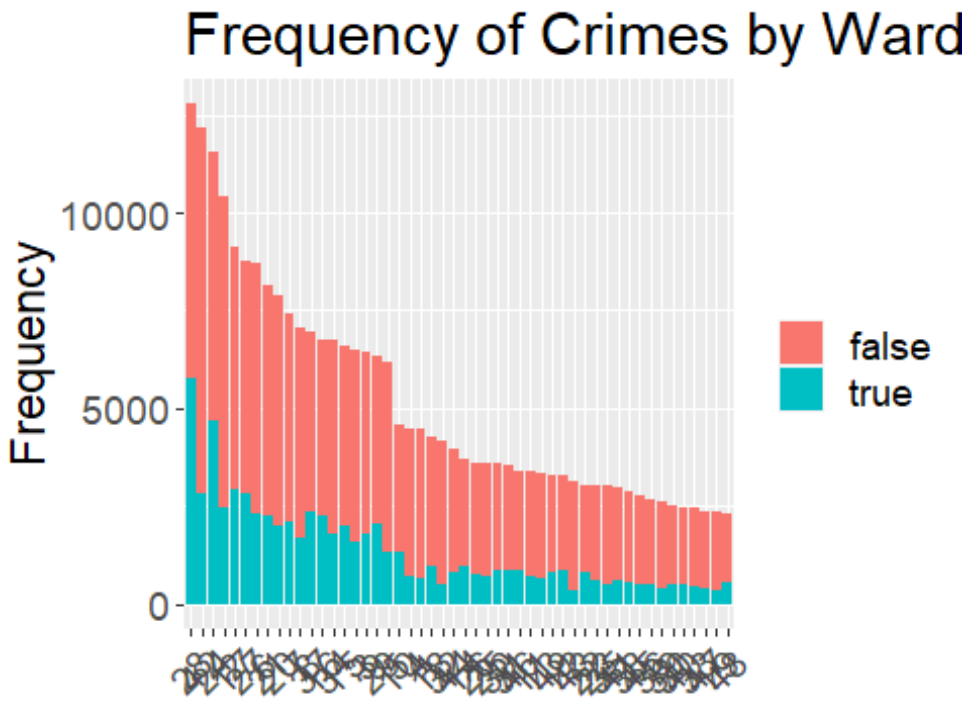
```
                    theme(axis.text.x = element_text(angle = 45, hjust = 1),
                          text = element_text(size = 18))
block_bar
```



I thought it might be interesting to visually compare the number of arrests made in each ward versus the actual frequency of crimes across that ward.

```
crime_freq_ward <- as.data.frame(table("ward" = chicago2015_df$Ward, "arrest"
= chicago2015_df$Arrest))
crime_freq_ward <- crime_freq_ward %>% filter(Freq>0)

ward_bar <- ggplot(crime_freq_ward, aes(fill=arrest, y=Freq, x=reorder(ward,-
Freq))) +
                    geom_bar(position="stack", stat="identity") +
                    labs(title="Frequency of Crimes by Ward & Arrests Made",
x="", y="Frequency") +
                    theme(axis.text.x = element_text(angle = 45, hjust = 1),
                          text = element_text(size = 18),
legend.title=element_blank())
ward_bar
```



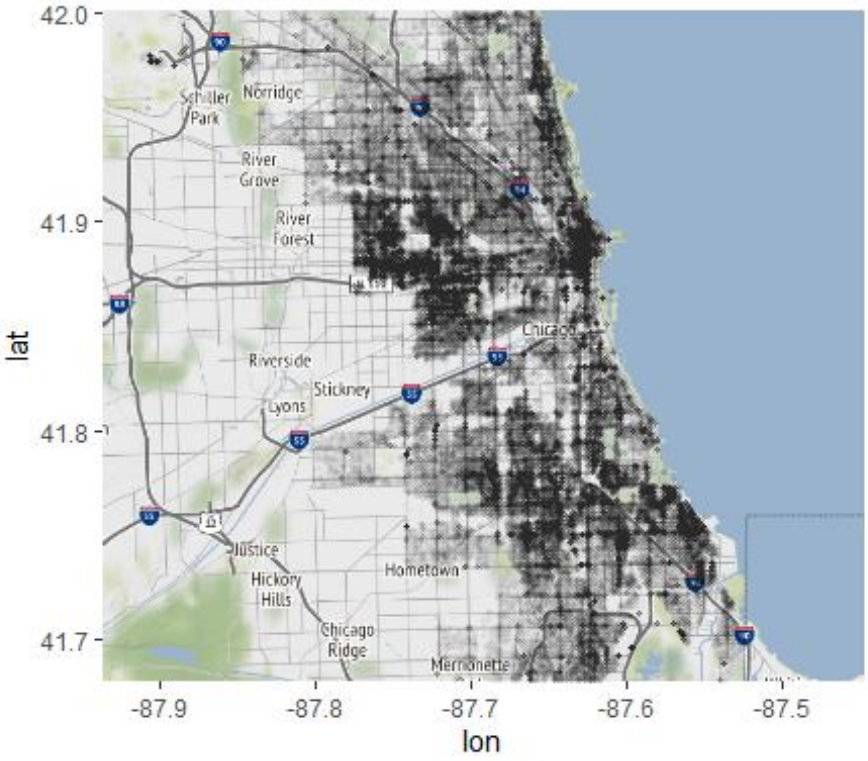
Finally, I'll visualize the density of crimes across Chicago using ggmap.

```
#store bounding box coordinates
chi_bb <- c(
    left = -87.936287,
    bottom = 41.679835,
    right = -87.447052,
    top = 42.000835
)

chicago_map <- get_stamenmap(
    bbox = chi_bb,
    zoom = 11
)

chicago <- chicago_map

ggmap(chicago) +
    geom_point(
        data = chicago2015_df,
        aes(x = Longitude, y = Latitude),
        size = 0.8,
        alpha = .01
    )
```



This sums up what I'm doing in R for now. Next, I'll go into researching a bit more about Chicago to write up the report.