Analysis of Historic Shootings in New York

2022-05-31

The data set used in this project is from Data.Gov. The following description is taken from the source website. “This is a breakdown of every shooting incident that occurred in NYC going back to 2006 through the end of the previous calendar year. This data is manually extracted every quarter and reviewed by the Office of Management Analysis and Planning before being posted on the NYPD website. Each record represents a shooting incident in NYC and includes information about the event, the location and time of occurrence. In addition, information related to suspect and victim demographics is also included. This data can be used by the public to explore the nature of shooting/criminal activity.”

Importing the Data

```
data_raw <- read_csv("https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD")
```

```
## Rows: 25596 Columns: 19
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (10): OCCUR_DATE, BORO, LOCATION_DESC, PERP_AGE_GROUP, PERP_SEX, PERP_R...
## dbl   (7): INCIDENT_KEY, PRECINCT, JURISDICTION_CODE, X_COORD_CD, Y_COORD_CD...
## lgl   (1): STATISTICAL_MURDER_FLAG
## time  (1): OCCUR_TIME
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

Tidy and Transform the Data

Summarize the raw data.

```
summary(data_raw)
```

```
##   INCIDENT_KEY        OCCUR_DATE          OCCUR_TIME            BORO
##  Min.   :  9953245   Length:25596       Length:25596      Length:25596
##  1st Qu.: 61593633   Class :character   Class1:hms        Class :character
##  Median : 86437258   Mode  :character   Class2:difftime   Mode  :character
##  Mean   :112382648                      Mode  :numeric
##  3rd Qu.:166660833
##  Max.   :238490103
##
##     PRECINCT      JURISDICTION_CODE LOCATION_DESC      STATISTICAL_MURDER_FLAG
##  Min.   :  1.00   Min.   :0.0000    Length:25596       Mode :logical
##  1st Qu.: 44.00   1st Qu.:0.0000    Class :character   FALSE:20668
##  Median : 69.00   Median :0.0000    Mode  :character   TRUE :4928
##  Mean   : 65.87   Mean   :0.3316
##  3rd Qu.: 81.00   3rd Qu.:0.0000
##  Max.   :123.00   Max.   :2.0000
##                   NA's   :2
##  PERP_AGE_GROUP       PERP_SEX          PERP_RACE         VIC_AGE_GROUP
##  Length:25596       Length:25596       Length:25596       Length:25596
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    VIC_SEX            VIC_RACE           X_COORD_CD        Y_COORD_CD
##  Length:25596       Length:25596       Min.   : 914928   Min.   :125757
##  Class :character   Class :character   1st Qu.:1000011   1st Qu.:182782
##  Mode  :character   Mode  :character   Median :1007715   Median :194038
##                                         Mean   :1009455   Mean   :207894
##                                         3rd Qu.:1016838   3rd Qu.:239429
##                                         Max.   :1066815   Max.   :271128
##
##     Latitude       Longitude        Lon_Lat
##  Min.   :40.51   Min.   :-74.25   Length:25596
##  1st Qu.:40.67   1st Qu.:-73.94   Class :character
##  Median :40.70   Median :-73.92   Mode  :character
##  Mean   :40.74   Mean   :-73.91
##  3rd Qu.:40.82   3rd Qu.:-73.88
##  Max.   :40.91   Max.   :-73.70
##
```

The following columns will be dropped as they will not be used in our analysis.

- INCIDENT_KEY
- JURISDICTION_CODE
- STATISTICAL_MURDER_FLAG
- PERP_AGE_GROUP
- PERP_SEX
- PERP_RACE
- VIC_AGE_GROUP
- VIC_SEX
- VIC_RACE
- X_COORD_CD
- Y_COORD_CD
- Latitude
- Longitude
- Lon_Lat

The following columns are converted to the appropriate type (also provided).

Column	Original Data Type	Converted Data Type
OCCUR_DATE	chr	date
OCCUR_TIME	chr	factor

Column	Original Data Type	Converted Data Type
LOCATION_DESC	chr	factor
BORO	chr	factor
PRECINCT	chr	factor

```
data <- data_raw %>%
  select(OCCUR_DATE:LOCATION_DESC) %>%
  select(-JURISDICTION_CODE) %>%
  mutate(OCCUR_DATE = mdy(OCCUR_DATE),
         OCCUR_TIME = as.factor(OCCUR_TIME),
         LOCATION_DESC = as.factor(LOCATION_DESC),
         BORO = as.factor(BORO),
         PRECINCT = as.factor(PRECINCT))
```

Summarize the cleaned data.

```
summary(data)
```

```
##    OCCUR_DATE            OCCUR_TIME               BORO          PRECINCT
##  Min.   :2006-01-01   23:30:00:  171   BRONX        : 7402   75     : 1470
##  1st Qu.:2009-05-10   00:30:00:  151   BROOKLYN     :10365   73     : 1372
##  Median :2012-08-26   01:30:00:  147   MANHATTAN    : 3265   67     : 1160
##  Mean   :2013-06-13   02:00:00:  142   QUEENS       : 3828   79     :  982
##  3rd Qu.:2017-07-01   21:00:00:  138   STATEN ISLAND:  736   44     :  949
##  Max.   :2021-12-31   22:30:00:  132                         47     :  903
##                       (Other) :24715                         (Other):18760
##                   LOCATION_DESC
##  MULTI DWELL - PUBLIC HOUS: 4559
##  MULTI DWELL - APT BUILD  : 2664
##  PVT HOUSE                :  893
##  GROCERY/BODEGA           :  622
##  BAR/NIGHT CLUB           :  588
##  (Other)                  : 1293
##  NA's                     :14977
```

There are still missing values in LOCATION_DESC. We will not drop it yet as it may provide some interesting analysis, but we will not use it where missing data could affect results.

Visualize and Analyze the Data

Where do people get shot?

There are too many distinct location descriptions to easily see how shootings are distributed. Therefore we will group the locations into buckets. We will group residential, business, and service locations.

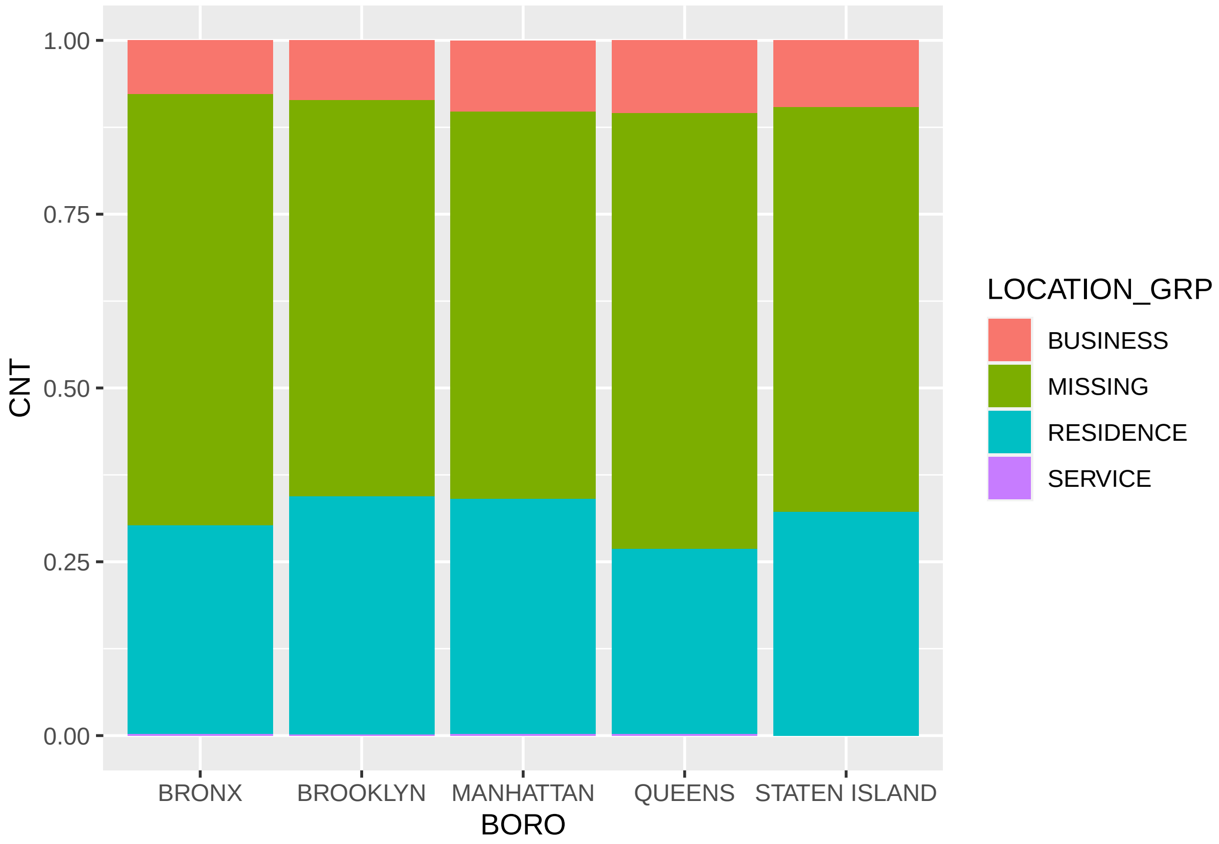
```
data <- data %>%
  mutate(LOCATION_GRP = ifelse((LOCATION_DESC == "NONE"
                                 |is.na(LOCATION_DESC))
                                , 'MISSING', as.character(LOCATION_DESC))) %>%
  mutate(LOCATION_GRP = ifelse((LOCATION_DESC %like% "DWELL"
                                 |LOCATION_DESC %like% "HOUSE")
                                , 'RESIDENCE', as.character(LOCATION_GRP))) %>%
  mutate(LOCATION_GRP = ifelse(LOCATION_DESC %like% "STORE"
                                |LOCATION_DESC %like% "COMPANY"
                                |LOCATION_DESC %like% "MERCHANT"
                                |LOCATION_DESC %like% "GROCERY"
                                |LOCATION_DESC %like% "FAST FOOD"
                                |LOCATION_DESC %like% "SALON"
                                |LOCATION_DESC %like% "CLUB"
                                |LOCATION_DESC %like% "COMMERCIAL"
                                |LOCATION_DESC %like% "CLOTHING"
                                |LOCATION_DESC %like% "SUPERMARKET"
                                |LOCATION_DESC %like% "STORAGE"
                                |LOCATION_DESC %like% "HOTEL"
                                |LOCATION_DESC %like% "GYM"
                                |LOCATION_DESC %like% "GAS"
                                |LOCATION_DESC %like% "LAUNDRY"
                                |LOCATION_DESC %like% "RESTAURANT"
                                , 'BUSINESS', as.character(LOCATION_GRP))) %>%
  mutate(LOCATION_GRP = ifelse(LOCATION_DESC %like% "ATM"
                                |LOCATION_DESC %like% "HOSPITAL"
                                |LOCATION_DESC %like% "CASH"
                                |LOCATION_DESC %like% "SCHOOL"
                                |LOCATION_DESC %like% "BANK"
                                |LOCATION_DESC %like% "DOCTOR"
                                , 'SERVICE', as.character(LOCATION_GRP)))
```

Now that we have locations in groups, we will visualize the percent of shootings in each location type by boro.

```
loc_boro_grp <- data %>%
  group_by(LOCATION_GRP, BORO) %>%
  summarise(CNT = n())
```

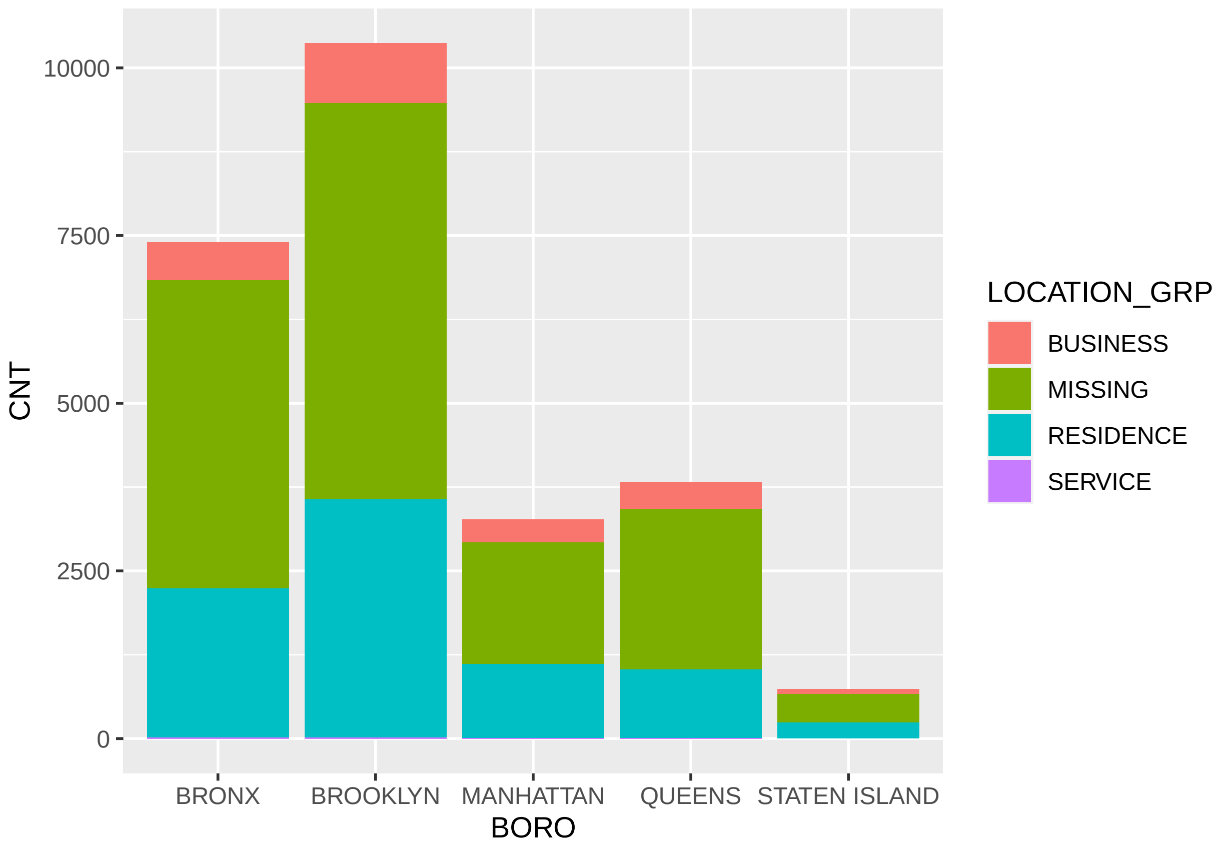
```
## `summarise()` has grouped output by 'LOCATION_GRP'. You can override using the
## `.groups` argument.
```

```
loc_boro_grp %>%
  ggplot(aes(fill=LOCATION_GRP, y=CNT, x=BORO)) +
  geom_bar(position="fill", stat="identity")
```



We can see from the plot above that the distribution across location types is simial across each boro. However this doesn't show us the relative number of shootings in each boro, so we will recreate the plot using incident counts rather than percents.

```
loc_boro_grp %>%
  ggplot(aes(fill=LOCATION_GRP, y=CNT, x=BORO)) +
  geom_col(position="stack")
```



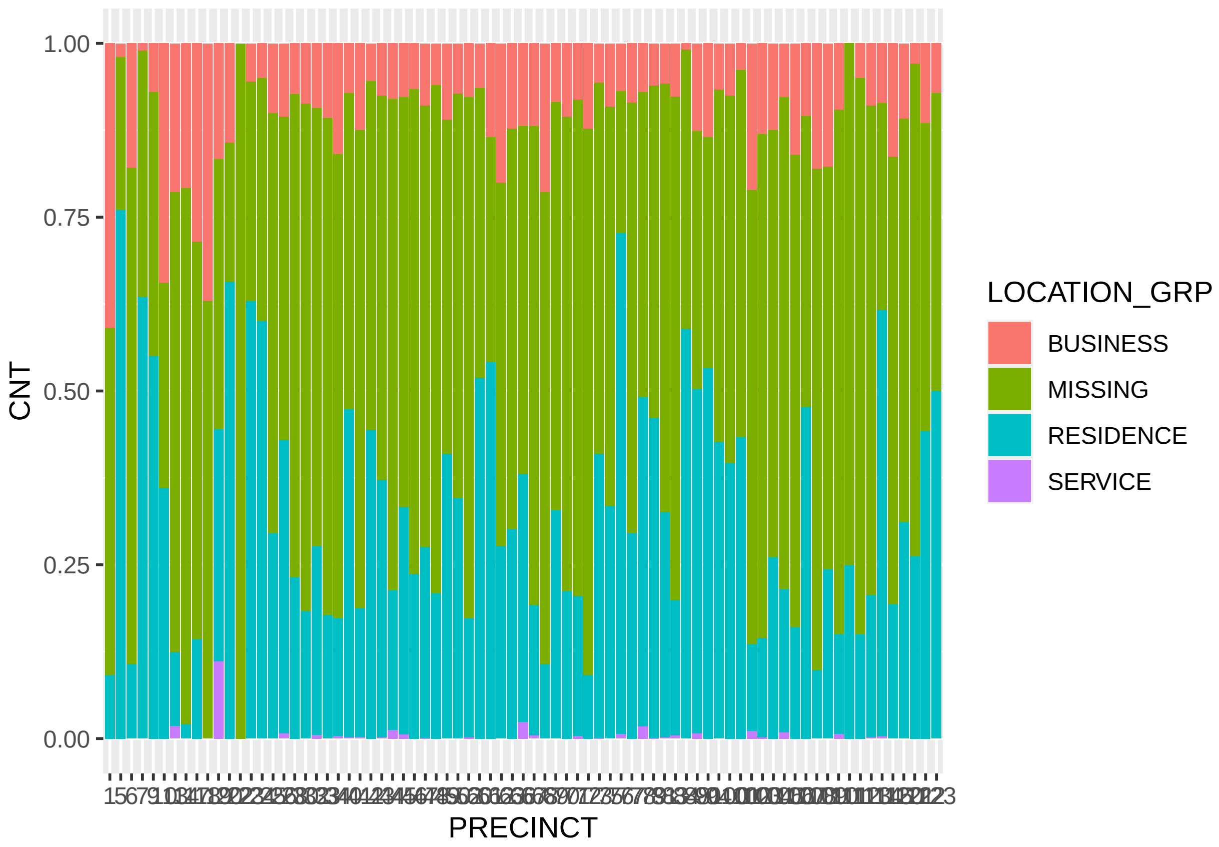
We can see from this plot that Brooklyn has significantly more shootings than the other boros and Staten Island has significantly less. Intuition tells us that Brooklyn is probably the largest boro and Staten Island is probably the smallest. A quick internet search confirms this fact with Brooklyn having ~2.7 MM people, Staten Island having ~0.5 MM in the 2020 census. Surprisingly, the assumption that incidents correlate strongly to populations falls apart when you see that Manhattan and The Bronx have similar populations of ~1.5 MM with Manhattan having ~0.2 MM more people than The Bronx, but there are significantly more incidents in Manhattan. Additionally, Queens has the second largest population with ~2.4 MM people, but is closer to Manhattan in number of incidents. This is particularly surprising since Manhattan has the highers population density of all the Boros. (source: https://en.wikipedia.org/wiki/Boroughs_of_New_York_City)

Assuming that precincts are more homogeneous in the area and/or population that they serve, we can redo the visualizations above grouping by precinct to see if the distribution of crimes by location type are similar across precincts.

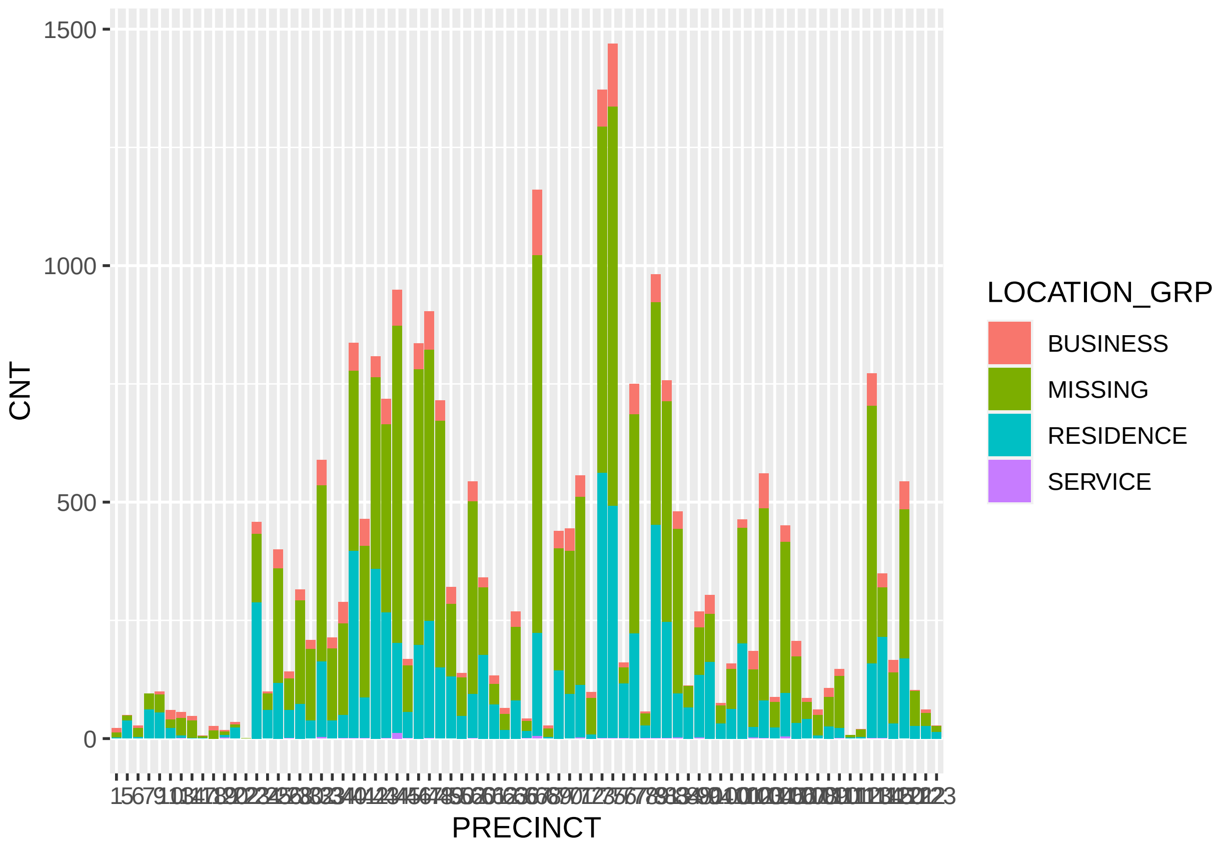
```
loc_precinct_grp <- data %>%
  group_by(LOCATION_GRP, PRECINCT) %>%
  summarise(CNT = n())
```

```
## ‘summarise()‘ has grouped output by ’LOCATION_GRP’. You can override using the
## ‘.groups‘ argument.
```

```
loc_precinct_grp %>%
  ggplot(aes(fill=LOCATION_GRP, y=CNT, x=PRECINCT)) +
  geom_bar(position="fill", stat="identity")
```



```
loc_precinct_grp %>%
  ggplot(aes(fill=LOCATION_GRP, y=CNT, x=PRECINCT)) +
  geom_col(position="stack")
```



Grouping the data by precincts, we do see more diversity in the breakdown by location. While most precincts still have a majority of incidents occurring in residential areas, there are some precincts that are primarily business and more precincts with more than 5% occurring in service areas.

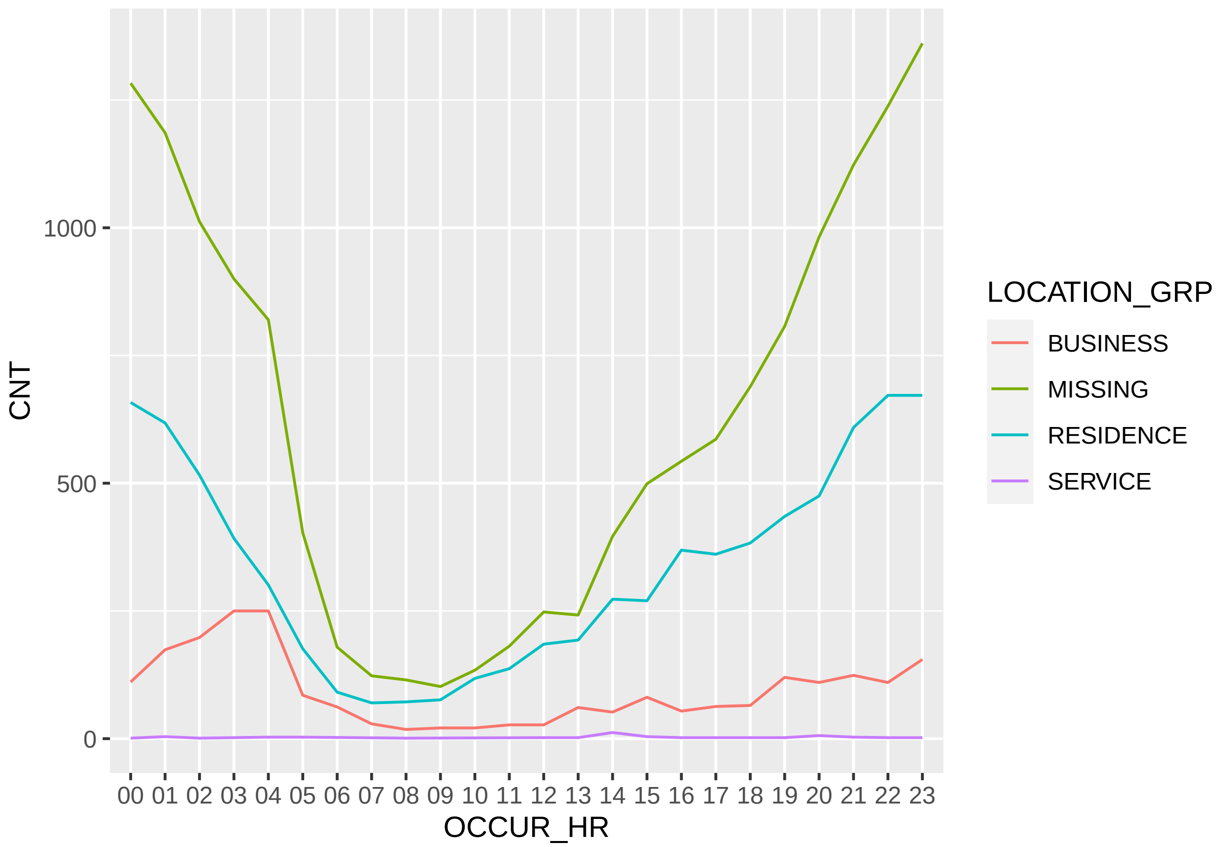
When do people get shot?

Next, we will look at the breakdown of incidents based on the hour of the day. First, we must extract the hour from the OCCURR_TIME. Then, we will plot the number of incidents by time and by location type.

```
data <- data %>%
  mutate(OCCUR_HR = substr(OCCUR_TIME, 0, 2))

time_boro_grp <- data %>%
  group_by(OCCUR_HR, LOCATION_GRP) %>%
  summarise(CNT = n())
```

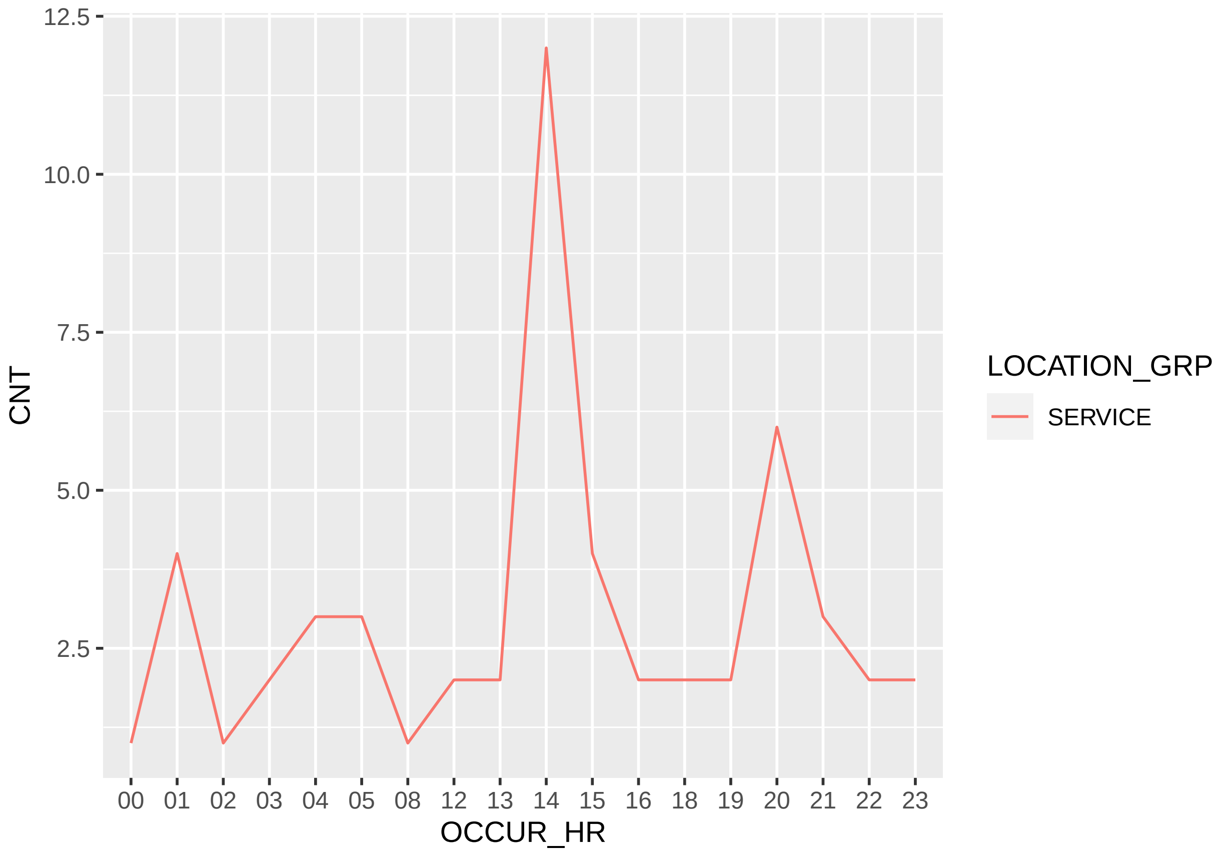
```
## ‘summarise()‘ has grouped output by ’OCCUR_HR’. You can override using the
## ‘.groups‘ argument.
```

```
time_boro_grp %>%
  ggplot(aes(x=OCCUR_HR, y=CNT, group=LOCATION_GRP, color=LOCATION_GRP)) +
  geom_line()
```



We can see a trend in incidents at businesses and residential areas where the majority occur in the evening and night hours. We can not see this trend in service locations. This could be due to the scale of the plot for this location type. Therefore we will plot service incidents by time separately so the data is not dwarfed by the other location types.

```
time_boro_grp %>%
  filter(LOCATION_GRP == 'SERVICE') %>%
  ggplot(aes(x=OCCUR_HR, y=CNT, group=LOCATION_GRP, color=LOCATION_GRP)) +
  geom_line()
```



Now that we can see the trend more clearly, we see that is does not in fact follow the same trend as business and residential location types. There is a very interesting peak occurring at 2pm and smaller peaks at 1am and 8pm. The grouping of service locations may be obscuring what is happening here, so for future analysis it would be interesting to plot the locations that make up this location type.

Modeling

Lets try clustering precincts based on different features in the shooting incident data.

First, we will group by precinct and location description and create a pivot table to store the count by precinct and location.

```
precinct_feats_loc <- data %>%
  group_by(PRECINCT, LOCATION_DESC) %>%
  summarise(LOC_CNT = n()) %>%
  mutate(LOC_FREQ = LOC_CNT/sum(LOC_CNT)) %>%
  select(-LOC_CNT) %>%
  pivot_wider(names_from = LOCATION_DESC, values_from = LOC_FREQ, values_fill = 0)
```

```
## `summarise()` has grouped output by 'PRECINCT'. You can override using the
## `.groups` argument.
```

Next, we want to create features based on the occur hour. Instead of using the count at each hour, we will split the day into four chunks each representing 6 hours.

```
data <- data %>%
  mutate(OCCUR_HR = as.integer(OCCUR_HR)) %>%
  mutate(OCCUR_TIME_GRP = ifelse((OCCUR_HR >= 0
                                 & OCCUR_HR < 6), 'DAWN', 'None')) %>%
  mutate(OCCUR_TIME_GRP = ifelse((OCCUR_HR >= 6
                                 & OCCUR_HR < 12), 'MORNING', OCCUR_TIME_GRP)) %>%
  mutate(OCCUR_TIME_GRP = ifelse((OCCUR_HR >= 12
                                 & OCCUR_HR < 18), 'AFTERNOON', OCCUR_TIME_GRP)) %>%
  mutate(OCCUR_TIME_GRP = ifelse((OCCUR_HR >= 18
                                 & OCCUR_HR < 24), 'NIGHT', OCCUR_TIME_GRP))

precinct_feats_time <- data %>%
  group_by(PRECINCT, OCCUR_TIME_GRP) %>%
  summarise(OCCUR_GRP_CNT = n()) %>%
  mutate(OCCUR_GRP_FREQ = OCCUR_GRP_CNT/sum(OCCUR_GRP_CNT)) %>%
  select(-OCCUR_GRP_CNT) %>%
  pivot_wider(names_from = OCCUR_TIME_GRP, values_from = OCCUR_GRP_FREQ, values_fill = 0)
```

```
## ‘summarise()‘ has grouped output by ’PRECINCT’. You can override using the
## ‘.groups‘ argument.
```

Finally, we will calculate the number of incidents for each precinct and combine all of the features.

```
precinct_feats <- data %>%
  group_by(PRECINCT) %>%
  summarise(CNT = n()) %>%
  mutate(PCT_TOT = CNT/sum(CNT)) %>%
  select(-CNT)


precinct_feats <- merge(precinct_feats, precinct_feats_loc, by = 'PRECINCT')
precinct_feats <- merge(precinct_feats, precinct_feats_time, by = 'PRECINCT')
```

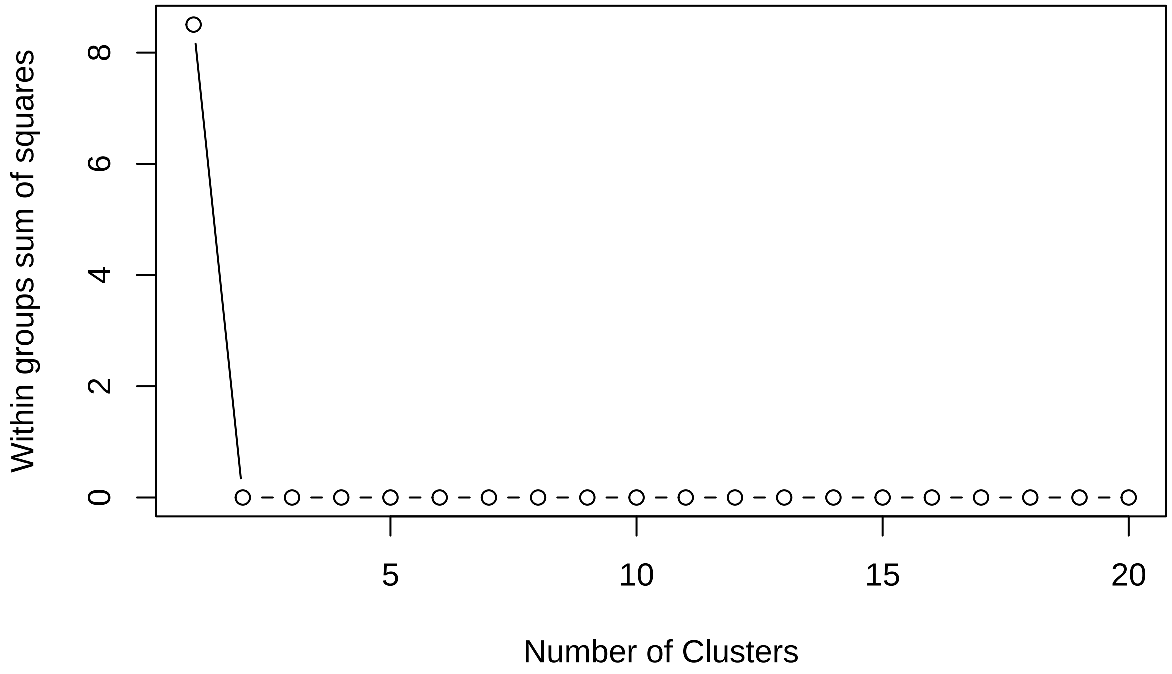
If we were to print the data summary we could see that there are a lot of location descriptions that have very few incidents. We will drop locations where the median location frequency across precincts is less than 0%.

```
 precinct_feats <- precinct_feats %>%
  mutate_all(~as.numeric(as.character(.))) %>%
  select_if(~median(., na.rm = TRUE) > 0)

summary(precinct_feats)
```

```
##     PRECINCT         PCT_TOT           BAR/NIGHT CLUB    COMMERCIAL BLDG
##  Min.   :  1.00   Min.   :3.907e-05   Min.   :0.00000   Min.   :0.000000
##  1st Qu.: 32.00   1st Qu.:2.539e-03   1st Qu.:0.00655   1st Qu.:0.000000
##  Median : 66.00   Median :7.228e-03   Median :0.02083   Median :0.008639
##  Mean   : 63.32   Mean   :1.299e-02   Mean   :0.03594   Mean   :0.016638
##  3rd Qu.:100.00   3rd Qu.:1.879e-02   3rd Qu.:0.04152   3rd Qu.:0.017857
##  Max.   :123.00   Max.   :5.743e-02   Max.   :0.27869   Max.   :0.142857
##  MULTI DWELL - APT BUILD       NA         MULTI DWELL - PUBLIC HOUS
##  Min.   :0.00000         Min.   :0.2000   Min.   :0.0000
##  1st Qu.:0.04833         1st Qu.:0.4492   1st Qu.:0.0000  
##  Median :0.09036         Median :0.5827   Median :0.1190  
##  Mean   :0.09137         Mean   :0.5630   Mean   :0.1822  
##  3rd Qu.:0.12925         3rd Qu.:0.6916   3rd Qu.:0.3279  
##  Max.   :0.28571         Max.   :1.0000   Max.   :0.7200  
##  RESTAURANT/DINER   GROCERY/BODEGA          NONE            PVT HOUSE      
##  Min.   :0.000000   Min.   :0.000000   Min.   :0.000000   Min.   :0.00000  
##  1st Qu.:0.000000   1st Qu.:0.005952   1st Qu.:0.000000   1st Qu.:0.00000  
##  Median :0.005594   Median :0.020335   Median :0.003589   Median :0.01361  
##  Mean   :0.011465   Mean   :0.019233   Mean   :0.005440   Mean   :0.04227  
##  3rd Qu.:0.014019   3rd Qu.:0.028000   3rd Qu.:0.009191   3rd Qu.:0.05616  
##  Max.   :0.107143   Max.   :0.071429   Max.   :0.040816   Max.   :0.39286  
##    AFTERNOON            DAWN           MORNING            NIGHT       
##  Min.   :0.0000   Min.   :0.1667   Min.   :0.00000   Min.   :0.0000  
##  1st Qu.:0.1215   1st Qu.:0.3304   1st Qu.:0.04762   1st Qu.:0.3310  
##  Median :0.1633   Median :0.3832   Median :0.06015   Median :0.3857  
##  Mean   :0.1606   Mean   :0.4035   Mean   :0.07272   Mean   :0.3632  
##  3rd Qu.:0.1947   3rd Qu.:0.4323   3rd Qu.:0.07641   3rd Qu.:0.4299  
##  Max.   :0.4286   Max.   :1.0000   Max.   :0.42857   Max.   :0.5455  
```

Now we are ready to use kmeans clustering to group our precincts. We will first cluster with k from 1 to 20 and see how the within group sum of squares.

```
points <- precinct_feats %>%
  select(-PRECINCT)

wss <- (nrow(points)-1) * sum(apply(points, 2, var))
for (i in 2:20) wss[i] <- sum(kmeans(points, centers=i)$withiness)
plot(1:20, wss, type='b', xlab='Number of Clusters', ylab='Within groups sum of squares')
```



From the plot above, we can see that there is a really strong elbow at k=2, so we will not need more than 2 clusters. Now, lets cluster the data with k=2 and look at the characteristic of each cluster.

```
kc <- kmeans(points, 2)
kc
```

```
## K-means clustering with 2 clusters of sizes 50, 27
##
## Cluster means:
##      PCT_TOT BAR/NIGHT CLUB COMMERCIAL BLDG MULTI DWELL - APT BUILD        NA
## 1 0.01327551     0.04167774      0.01900507              0.10026491 0.6542924
## 2 0.01245276     0.02532524      0.01225330              0.07489013 0.3940590
##   MULTI DWELL - PUBLIC HOUS RESTAURANT/DINER GROCERY/BODEGA        NONE
## 1                0.05370813      0.013331579     0.02048358 0.006395000
## 2                0.42004082      0.008007261     0.01691727 0.003670969
##    PVT HOUSE AFTERNOON      DAWN    MORNING     NIGHT
## 1 0.05851979 0.1468792 0.4216865 0.07393351 0.3575008
## 2 0.01218288 0.1859246 0.3699575 0.07047435 0.3736436
##
## Clustering vector:
##  [1] 1 2 2 2 1 1 1 1 1 2 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 2 2 1 2 2 1 2 1 1 1 1 1
## [39] 2 1 2 2 1 1 1 1 1 2 2 1 1 1 2 2 1 1 1 1 1 1 2 1 1 1 2 1 2 1 2 2 1 1 2 2 2 2
## [77] 2
##
## Within cluster sum of squares by cluster:
## [1] 3.557685 1.271943
##  (between_SS / total_SS =  43.2 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

Conclusion

When we look at the number of shooting incidents by location type and boro, we see similar distributions across boros. However, when we break them out by precinct we see less homogeneity. When we looked at the breakdown of incidents by hour of the day and location type, we saw that Business and Residential type location had a similar trend where incidents went up in the evening hours and dropped drastically during the daylight hours. However, incidents in service locations did not follow this trend, but instead had unexplained peaks at several hours with the largest peak occurring in the middle of the afternoon.

Our clustering analysis showed 2 clusters that were very similar in most dimensions with the main difference being the percent of incidents in MULTI DWELL - PUBLIC HOUSE.

This initial analysis brings up more questions than it answers. Future work would benefit from bringing in census data to see how the population and population density correlate to the number of incidents in a particular boro or precinct.

Bias Identification

The author's personal bias is that some of the individuals responsible for accurately recording shooting incidents are likely to exclude or misreport information based on demographics of the parties involved. That is, the author believes that shootings with a white, affluent suspect or minority, impoverished victim are less likely to be reported or reported accurately. To prevent this bias from directing the analysis, demographic features about the suspect and victim were removed prior to the analysis. However, features related to where the shooting happened are likely to have strong correlations to these demographics and should therefore be used cautiously in any further analysis.