NYC Shooting Data Report

Mason Scheer

2024-11-04

The data for this report is NYPD data listing every shooting incident that occurred in NYC going back to 2006 through the end of the previous calendar year.

As summarized on the catalog.data.gov, "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."

```
#When exploring what kind of analysis I wanted to do, these are the libraries I imported. Not all ended
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(readr)
library(dplyr)
library(ggplot2)
library(hms)
```

```
##
## Attaching package: 'hms'
##
## The following object is masked from 'package:lubridate':
##
##     hms
```

```
library(lubridate)
library(ggmap)
```

```
## i Google's Terms of Service: <https://mapsplatform.google.com>
##   Stadia Maps' Terms of Service: <https://stadiamaps.com/terms-of-service/>
##   OpenStreetMap's Tile Usage Policy: <https://operations.osmfoundation.org/policies/tiles/>
## i Please cite ggmap if you use it! Use `citation("ggmap")` for details.
```

```
library(maps)
```

```
##
## Attaching package: 'maps'
##
## The following object is masked from 'package:purrr':
##
##     map
```

Loading the Data

To start the process, I loaded the CSV into R via the linked address to the data on the City of New York website.

```
#load the CSV file
original_data <- read_csv(
  "https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD")
```

```
## Rows: 28562 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (12): OCCUR_DATE, BORO, LOC_OF_OCCUR_DESC, LOC_CLASSFCTN_DESC, LOCATION...
## 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.
```

```
#Take a look at the first few lines of data
head(original_data)
```

```
## # A tibble: 6 x 21
##   INCIDENT_KEY OCCUR_DATE OCCUR_TIME BORO      LOC_OF_OCCUR_DESC PRECINCT
##          <dbl> <chr>      <time>     <chr>     <chr>                <dbl>
## 1    244608249 05/05/2022 00:10      MANHATTAN INSIDE                  14
## 2    247542571 07/04/2022 22:20      BRONX     OUTSIDE                 48
## 3     84967535 05/27/2012 19:35      QUEENS    <NA>                   103
## 4    202853370 09/24/2019 21:00      BRONX     <NA>                    42
## 5     27078636 02/25/2007 21:00      BROOKLYN  <NA>                    83
## 6    230311078 07/01/2021 23:07      MANHATTAN <NA>                    23
## # i 15 more variables: JURISDICTION_CODE <dbl>, LOC_CLASSFCTN_DESC <chr>,
## #   LOCATION_DESC <chr>, STATISTICAL_MURDER_FLAG <lgl>, PERP_AGE_GROUP <chr>,
## #   PERP_SEX <chr>, PERP_RACE <chr>, VIC_AGE_GROUP <chr>, VIC_SEX <chr>,
## #   VIC_RACE <chr>, X_COORD_CD <dbl>, Y_COORD_CD <dbl>, Latitude <dbl>,
## #   Longitude <dbl>, Lon_Lat <chr>
```

Cleaning the Data

After loading the NYC Shooting Data, I explored what the data looked like. I wanted to drop columns I knew I wouldn't be using for my exploration and analysis. I also needed to clean up the date and time fields

available as they were initially just character types. I ended up creating one datetime field and dropping the individual date and time columns. The last piece of clean up I did to begin my analysis was factoring the Boroughs field.

```
#This chunk will be used to clean up the data
#First, I want to drop columns that I won't be using
cleaned_data <- original_data %>%
  select(-c("LOC_OF_OCCUR_DESC", "LOC_CLASSFCTN_DESC", "LOCATION_DESC",
            "PRECINCT", "JURISDICTION_CODE", "X_COORD_CD", "Y_COORD_CD",
            "Lon_Lat")) %>%
  mutate(
    OCCUR_DATE = as.Date(OCCUR_DATE, format = "%m/%d/%Y"),
    OCCUR_TIME = hms::as_hms(OCCUR_TIME),
    OCCUR_DATETIME = as.POSIXct(paste(OCCUR_DATE, OCCUR_TIME),
                                format = "%Y-%m-%d %H:%M:%S"),
    BORO = factor(BORO, levels = c("MANHATTAN", "BRONX", "BROOKLYN", "QUEENS",
                                   "STATEN ISLAND"))
  ) %>%
  select(-c("OCCUR_DATE", "OCCUR_TIME")) %>%
  relocate(OCCUR_DATETIME, .after = 1)
```

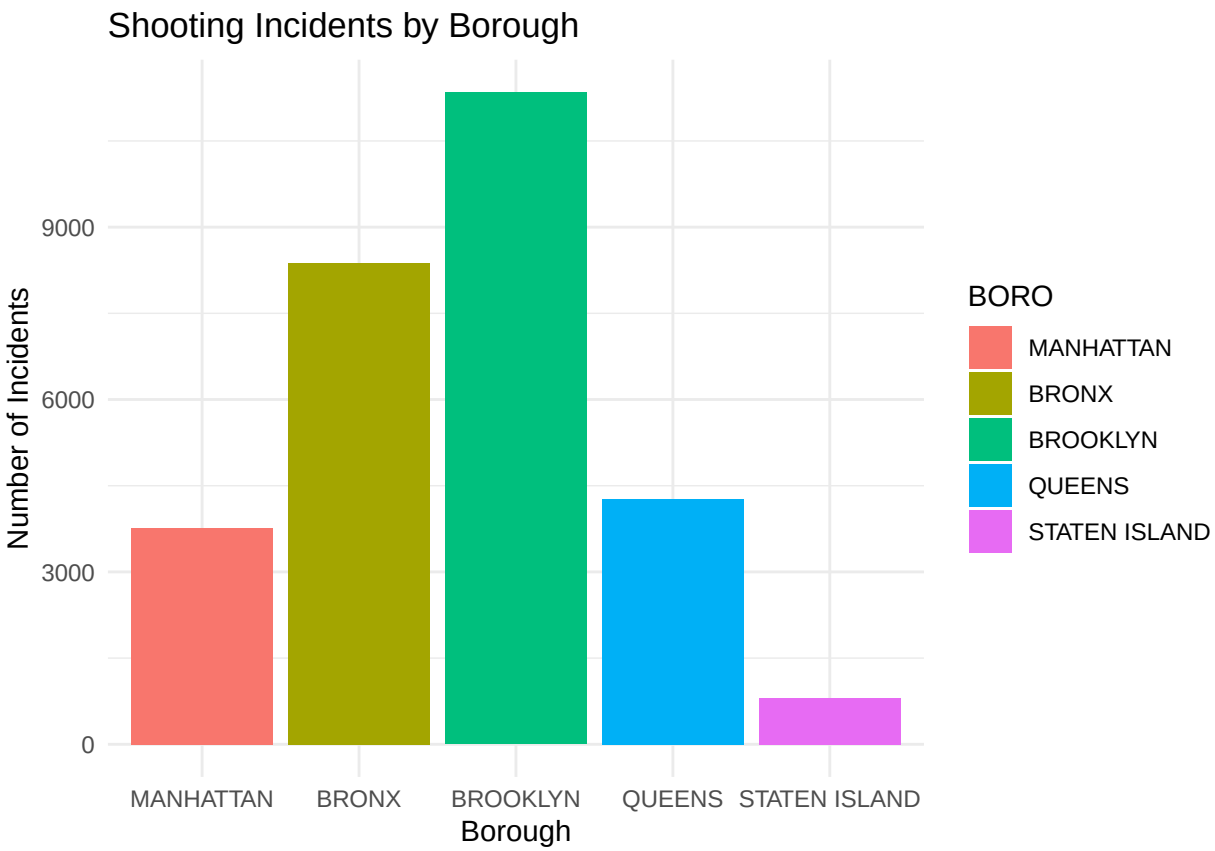
Shooting Summary By Neighborhood

With my data clean, the first visual I wanted to see in my analysis was a summary of shootings by neighborhood. First I grouped the boro field in the data to summarize, and then created a bar chart to create an easy way to see how the neighborhoods in this dataset compare to each other as far as shootings go. I am generally unfamiliar with NYC, so I relied on both research and testimonials from friends who live there to make sense of the summary. I was most interested in the low of Staten Island and high of Brooklyn. Per my research on Staten Island, it is considered one of the safer neighborhoods of the NYC area. My sister lives in Brooklyn, which has the highest shooting count. While she lives in a safe pocket of Brooklyn, she agreed that this data makes sense after exploring most of the region. She specified that Brooklyn "gets a lot sketchier" the further east you go away from her home in Williamsburg.

In my exploration of this data, I used the latitude and longitude points for each shooting and integrated them with the Google maps API to plot each point on the graph and show the density of areas with high shootings. I chose not to include this visual due to the inability to interactively zoom into more specific areas without creating either a Shiny app or webpage using javascript/html. The bar chart was sufficient to portray the information the map would have been able to at a zoomed out view of the region. In a future enhancement of this project, I would love to create a Shiny App version of this analysis.

```
#creating a variable that explores the shootings summary by neighborhood in NY
summary_by_hood <- cleaned_data %>%
  group_by(BORO)%>%
  summarise(Count = n())


# Create bar plot for summary by neighborhood.
ggplot(summary_by_hood, aes(x = BORO, y = Count, fill = BORO)) +
  geom_bar(stat = "identity") +
  labs(title = "Shooting Incidents by Borough", x = "Borough",
       y = "Number of Incidents") +
  theme_minimal()
```



Shooting Incidents Over Time

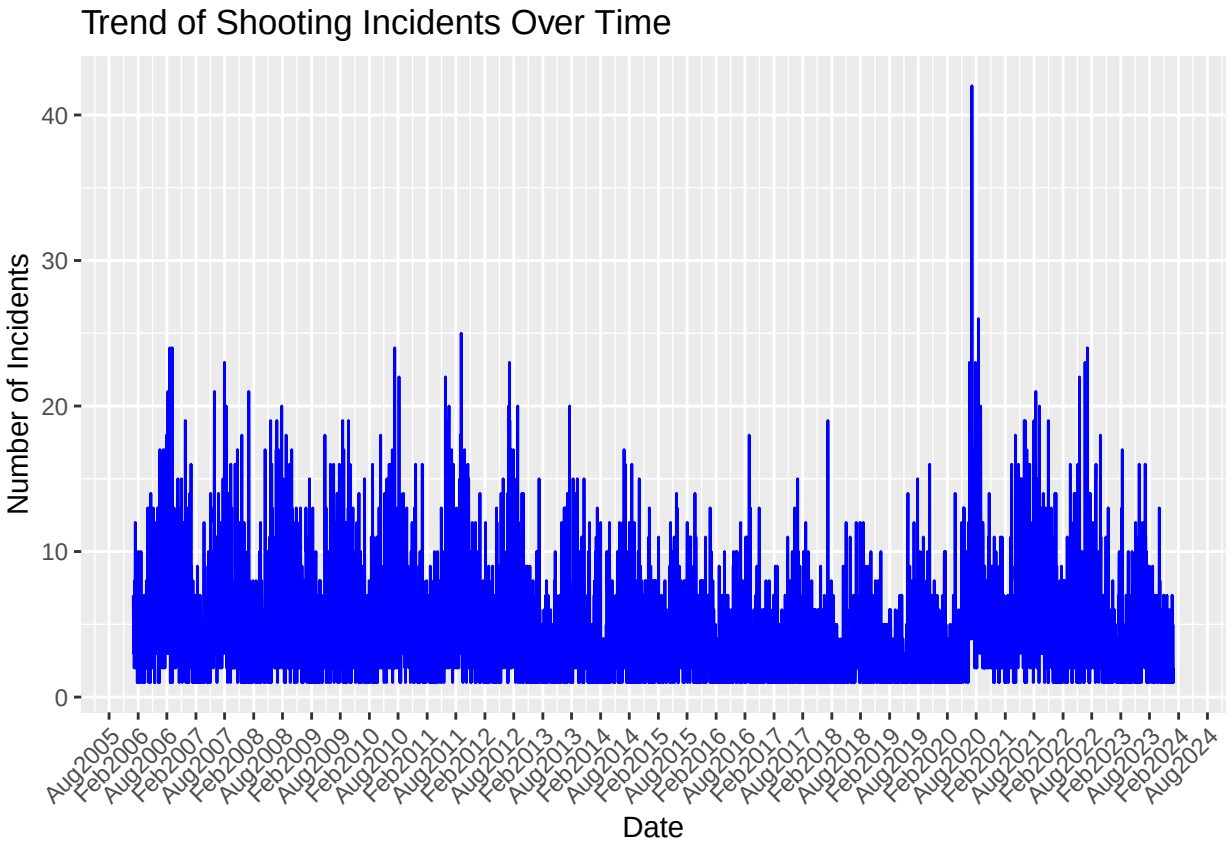
Another visual I wanted to create using this data was a line graph showing shooting frequencies over time. I was interested to see if there would be any sort of general trend that would indicate specific time periods where shootings were more common. The X axis is set in a series of every six months and the Y axis is the number of shootings per day. As you can see, this creates a dense line, however it still allows us to guage significant times where there were highs and lows. This graph shows us that since 2006, shooting incidents per day have been pretty steady. From around 2012-2019, there appeared to even be a slight decline in daily incidents. Then, as you will see, there is one major interesting point right around Summer 2020 where shooting incidents skyrocket. My immediate hypothesis for this sharp rise was the BLM movement that was prominent in NYC and other major cities across the US. After doing some research, the timeline of the movement would line up with this rise. However to confirm causation rather than just correlating it, I would want to find ways to bring in additional data around the movement such as specific rallies and any acts of violence that occurred directly during them.

```
# Extract date from OCCUR_DATETIME
cleaned_data$Date <- as.Date(cleaned_data$OCCUR_DATETIME)

# Count incidents by date
daily_counts <- cleaned_data %>%
  group_by(Date) %>%
  summarise(Incident_Count = n())

# Create time series plot
ggplot(daily_counts, aes(x = Date, y = Incident_Count)) +
```

```
  geom_line(color = "blue") +
  labs(title = "Trend of Shooting Incidents Over Time", x = "Date",
       y = "Number of Incidents") +
  scale_x_date(date_breaks = "6 month", date_labels = "%b%Y") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Shooting Incidents by Time of Day

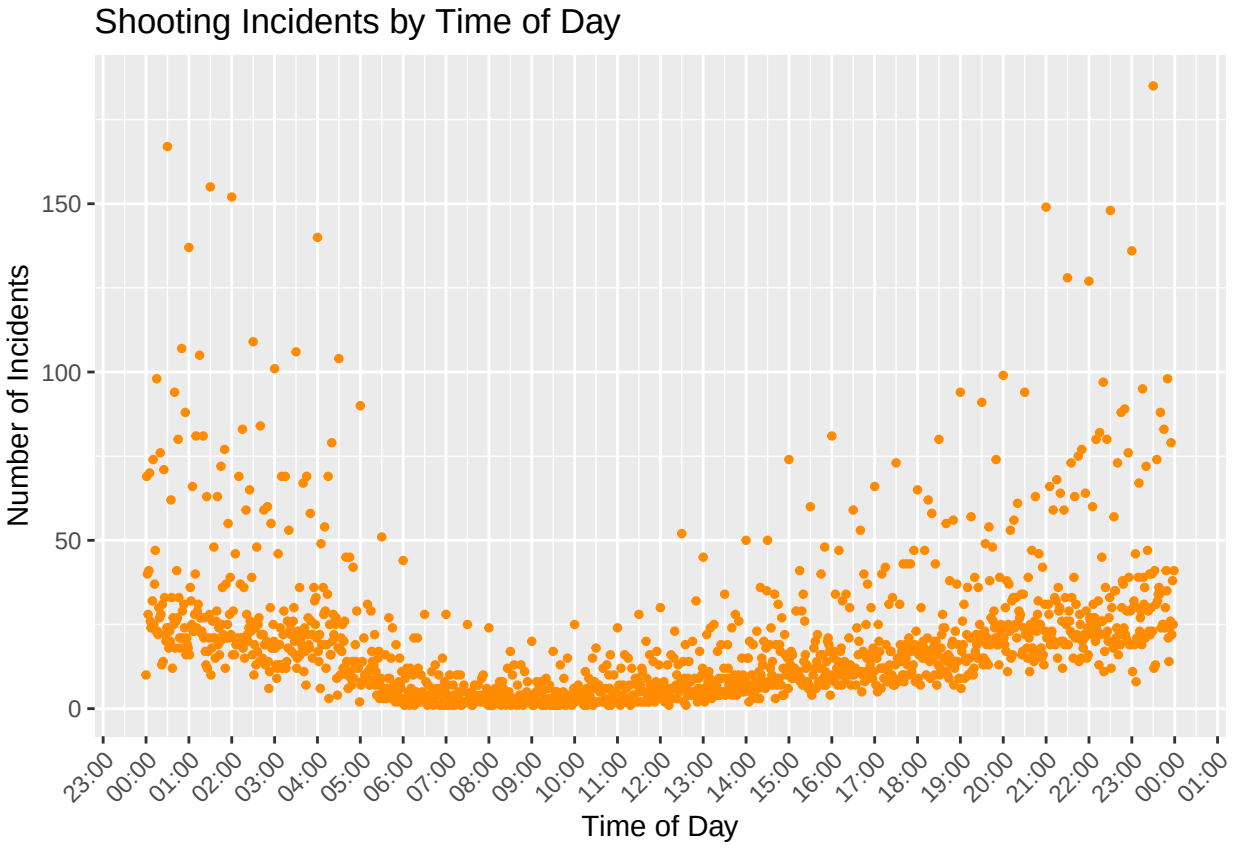
After exploring the shootings over time, I decided I wanted to focus the rest of this project specifically around time. I wanted to create a visual plotting the time of day where shooting incidents occured. My hypothesis was that the plots would create a U shape with the X axis series of the 24 hours of a day. I infered that shooting incidents were higher during the dark hours of each day, and lower during the light hours. As you can see, the general pattern with the plots follows exactly that prediction. From 12am-7am, the occurences started high and declined as it got later into the early morning. During the morning until the afternoon, the incident rate stayed relatively low. As afternoon progressed into evening and then night, the incidents began to rise again.

```
#create field for time
cleaned_data$Time <- format(cleaned_data$OCCUR_DATETIME, "%H:%M:%S")

#group the incidents by time of day
time_of_day <- cleaned_data %>%
  group_by(Time) %>%
  summarise(Incident_Count = n())
```

```r
# Ensure Time is in the correct format
time_of_day$Time <- as.POSIXct(time_of_day$Time, format = "%H:%M:%S")

#plot shooting incidents by time of day
ggplot(time_of_day, aes(x = Time, y = Incident_Count)) +
  geom_point(color = "darkorange", size = 1) +
  labs(title = "Shooting Incidents by Time of Day",
       x = "Time of Day",
       y = "Number of Incidents") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_x_datetime(date_labels = "%H:%M",
                   breaks = scales::breaks_width("1 hour"))
```



Model on Relationship Between Incidents by Time of Day and Borough

Using all aspects of my analysis so far, I wanted to explore the relationship between the time of day and neighborhood for shooting incidents that occurred. I also wanted to model out a prediction, based on the data, of how many crimes may occur going forward based on the neighborhood and time of day. I researched different ways I could achieve this, and chose a Poisson regression model. As a result, the model suggest there are several significant predictors indicating that time of day and borough are important factors when predicting shooting counts that may occur. This could be used in many ways. For the public, it is useful for taking extra safetly measures when visiting certain areas during specific times of the day. For law enforcement, this is good information for policing strategies and allocation of officers in specific neighborhoods at different times of the day.

```r
#Extract hour from OCCUR_DATETIME
cleaned_data <- cleaned_data %>%
  mutate(HOUR = format(OCCUR_DATETIME, "%H"))

#Aggregate data by hour and borough
shooting_summary <- cleaned_data %>%
  group_by(BORO, HOUR) %>%
  summarise(SHOOTING_COUNT = n(), .groups = 'drop')


#Based on my research, I wanted to use a Poisson regression model to predict
#crime counts versus actual
model <- glm(SHOOTING_COUNT ~ HOUR + BORO, data = shooting_summary,
             family = "poisson")
summary(model)
```

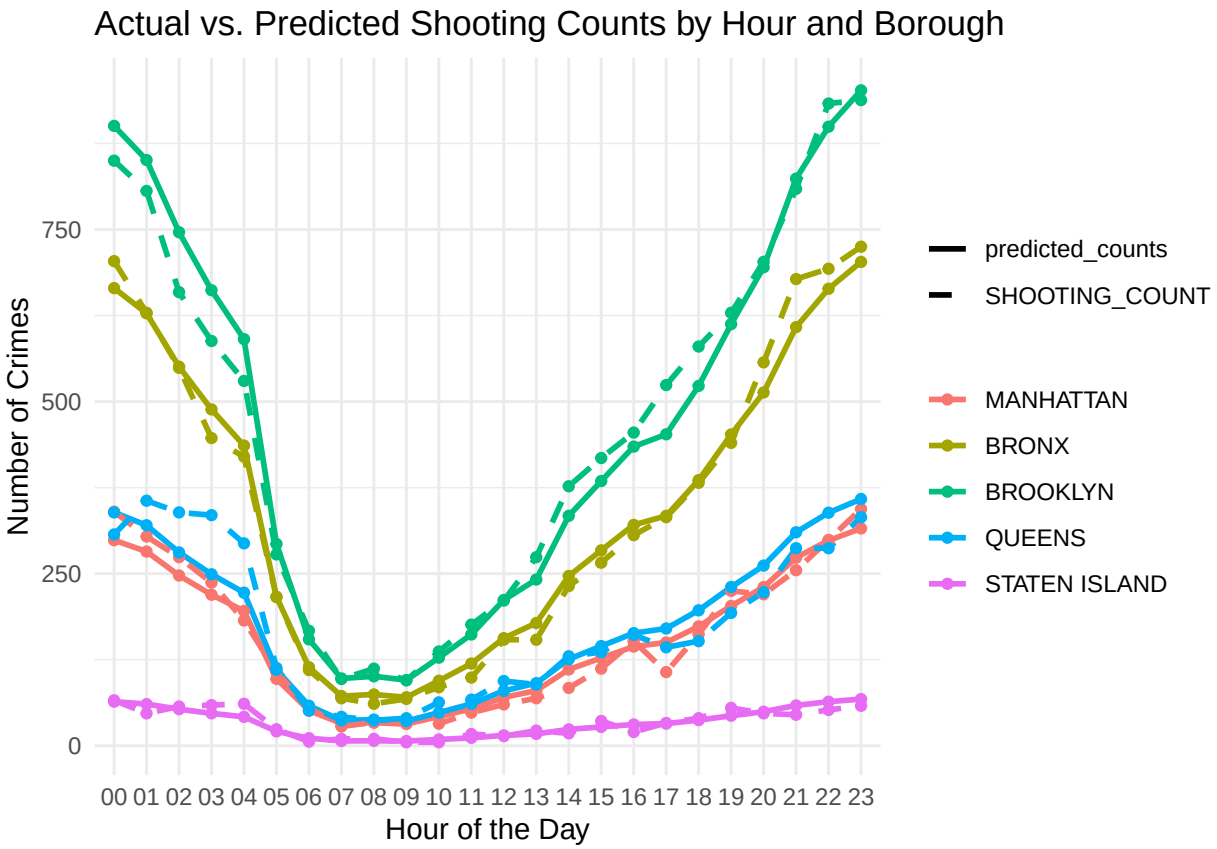
```
##
## Call:
## glm(formula = SHOOTING_COUNT ~ HOUR + BORO, family = "poisson",
##     data = shooting_summary)
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)        5.699086   0.025921 219.861  < 2e-16 ***
## HOUR01            -0.056717   0.030132  -1.882  0.05980 .
## HOUR02            -0.188250   0.031202  -6.033 1.61e-09 ***
## HOUR03            -0.308032   0.032270  -9.545  < 2e-16 ***
## HOUR04            -0.421697   0.033371 -12.637  < 2e-16 ***
## HOUR05            -1.122269   0.042381 -26.481  < 2e-16 ***
## HOUR06            -1.762633   0.054880 -32.118  < 2e-16 ***
## HOUR07            -2.220881   0.067128 -33.084  < 2e-16 ***
## HOUR08            -2.188878   0.066167 -33.081  < 2e-16 ***
## HOUR09            -2.245574   0.067881 -33.081  < 2e-16 ***
## HOUR10            -1.951661   0.059554 -32.771  < 2e-16 ***
## HOUR11            -1.717399   0.053834 -31.902  < 2e-16 ***
## HOUR12            -1.447691   0.048138 -30.074  < 2e-16 ***
## HOUR13            -1.316038   0.045671 -28.816  < 2e-16 ***
## HOUR14            -0.991621   0.040375 -24.560  < 2e-16 ***
## HOUR15            -0.850981   0.038395 -22.164  < 2e-16 ***
## HOUR16            -0.728617   0.036813 -19.792  < 2e-16 ***
## HOUR17            -0.688307   0.036319 -18.952  < 2e-16 ***
## HOUR18            -0.543861   0.034655 -15.693  < 2e-16 ***
## HOUR19            -0.385377   0.033009 -11.675  < 2e-16 ***
## HOUR20            -0.258842   0.031820  -8.134 4.14e-16 ***
## HOUR21            -0.088978   0.030385  -2.928  0.00341 **
## HOUR22            -0.001324   0.029712  -0.045  0.96445
## HOUR23             0.055761   0.029297   1.903  0.05700 .
## BOROBRONX          0.800420   0.019627  40.782  < 2e-16 ***
## BOROBROOKLYN       1.103915   0.018814  58.676  < 2e-16 ***
## BOROQUEENS         0.126897   0.022360   5.675 1.38e-08 ***
## BOROSTATEN ISLAND -1.539382   0.038794 -39.681  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 24874.17  on 119  degrees of freedom
## Residual deviance:   313.94  on  92  degrees of freedom
## AIC: 1177.8
##
## Number of Fisher Scoring iterations: 4
```

```
shooting_summary$predicted_counts <- predict(model, type = "response")

#Create a graph to show the actual versus predicted shootings by time of day by
#neighborhood
shooting_graph <- shooting_summary %>%
  pivot_longer(cols = c(SHOOTING_COUNT, predicted_counts),
               names_to = "Type",
               values_to = "Count")
ggplot(shooting_graph, aes(x = HOUR, y = Count, color = BORO,
                           group = interaction(BORO, Type))) +
  geom_line(aes(linetype = Type), size = 1) +
  geom_point() +
  labs(title = "Actual vs. Predicted Shooting Counts by Hour and Borough",
       x = "Hour of the Day",
       y = "Number of Crimes") +
  theme_minimal() +
  scale_linetype_manual(values = c("solid", "dashed")) +
  theme(legend.title = element_blank())
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```



Conclusion

Overall, there were many interesting ways I could have explored this dataset. During my exploration, I was initially most interested in the significance of time and location of where these shootings were occuring. I had limited bias when exploring this dataset since I am generally unfamiliar with New York City, but there was some. I expected to see higher crime in Manhattan especially because that is where shootings and other major crimes occur in New York City that often make national headlines as I recollect. I also had a bias regarding Brooklyn since my sister lives there, creating a personal connection. To mitigate this bias, I took the approach that all areas have their safe and not as safe areas that create hot spots for overall statistics which is why a deeper analysis could be done using the coordinate data. In the future, I may build a Shiny app and do a spatial analysis to dig deeper into specific areas of specific neighborhoods. For the time of day, I was happy to see both the actual and predictive lines on the graph line up with what I was expecting. This analysis has likely been done by data scientists for the city of New York, but is definitely important information for both the public and law enforcement. It would be very interesting to explore other correlations outside this dataset such as other crime statistics such as illegal drug use to explore any significance.