# NYPD Shooting Data Report based on Age, Gender, Race, and other factors

Tarun Sudhams

2024-11-23

## Introduction and source of dataset

I used the NYPD data set "NYPD Gun Violence Historical" available on New York City's website.

The NYPD dataset has multiple factors that can be analyzed and visualized to get a better understanding of the data that we are looking at. This includes factors like victim's age, sex and race. This visualization aims at understanding the patterns and trends of gun violence in New York City.

## Import required libraries

These are the two main libraries that we would be using for visualization and data wrangling.

```r
options(repos='http://cran.rstudio.com/')
install.packages("ggplot2")
```

```
##
## The downloaded binary packages are in
##  /var/folders/90/8dfd1w4x72l5rzmf906_854c0000gn/T//Rtmp9ReKr0/downloaded_packages
```

```r
install.packages("tidyverse")
```

```
##
## The downloaded binary packages are in
##  /var/folders/90/8dfd1w4x72l5rzmf906_854c0000gn/T//Rtmp9ReKr0/downloaded_packages
```

```r
install.packages("dplyr")
```

```
##
## The downloaded binary packages are in
##  /var/folders/90/8dfd1w4x72l5rzmf906_854c0000gn/T//Rtmp9ReKr0/downloaded_packages
```

```r
install.packages("mgcv")
```

```
##
## The downloaded binary packages are in
##  /var/folders/90/8dfd1w4x72l5rzmf906_854c0000gn/T//Rtmp9ReKr0/downloaded_packages
```

```r
library(ggplot2)
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 lubridate 1.9.3     v tibble    3.2.1
## v purrr     1.0.2     v tidyr     1.3.1
## -- 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
```

```r
library(dplyr)
library(mgcv)
```

```
## Loading required package: nlme
##
## Attaching package: 'nlme'
##
## The following object is masked from 'package:dplyr':
##
##     collapse
##
## This is mgcv 1.9-1. For overview type 'help("mgcv-package")'.
```

## Import the dataset

```r
dataset_url <- "https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv"
dataset <- read_csv(dataset_url)
```

```
## 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.
```

## Explore the dataset

Before we start tidying and transforming the dataset, let's take a look at how the dataset looks like and what each column looks like and the sample data points in each of the columns.

```r
glimpse(dataset)
```

```
## Rows: 28,562
## Columns: 21
## $ INCIDENT_KEY            <dbl> 244608249, 247542571, 84967535, 202853370, 270~
## $ OCCUR_DATE              <chr> "05/05/2022", "07/04/2022", "05/27/2012", "09/~
## $ OCCUR_TIME              <time> 00:10:00, 22:20:00, 19:35:00, 21:00:00, 21:00~
## $ BORO                    <chr> "MANHATTAN", "BRONX", "QUEENS", "BRONX", "BROO~
## $ LOC_OF_OCCUR_DESC       <chr> "INSIDE", "OUTSIDE", NA, NA, NA, NA, NA, NA, N~
## $ PRECINCT                <dbl> 14, 48, 103, 42, 83, 23, 113, 77, 48, 49, 73, ~
## $ JURISDICTION_CODE       <dbl> 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
## $ LOC_CLASSFCTN_DESC      <chr> "COMMERCIAL", "STREET", NA, NA, NA, NA, NA, NA~
## $ LOCATION_DESC           <chr> "VIDEO STORE", "(null)", NA, NA, NA, "MULTI DW~
## $ STATISTICAL_MURDER_FLAG <lgl> TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, TRUE, ~
## $ PERP_AGE_GROUP          <chr> "25-44", "(null)", NA, "25-44", "25-44", NA, N~
## $ PERP_SEX                <chr> "M", "(null)", NA, "M", "M", NA, NA, NA, NA, "~
## $ PERP_RACE               <chr> "BLACK", "(null)", NA, "UNKNOWN", "BLACK", NA,~
## $ VIC_AGE_GROUP           <chr> "25-44", "18-24", "18-24", "25-44", "25-44", "~
## $ VIC_SEX                 <chr> "M", "M", "M", "M", "M", "M", "M", "M", "M", "~
## $ VIC_RACE                <chr> "BLACK", "BLACK", "BLACK", "BLACK", "BLACK", "~
## $ X_COORD_CD              <dbl> 986050, 1016802, 1048632, 1014493, 1009149, 99~
## $ Y_COORD_CD              <dbl> 214231.0, 250581.0, 198262.0, 242565.0, 190104~
## $ Latitude                <dbl> 40.75469, 40.85440, 40.71063, 40.83242, 40.688~
## $ Longitude               <dbl> -73.99350, -73.88233, -73.76777, -73.89071, -7~
## $ Lon_Lat                 <chr> "POINT (-73.9935 40.754692)", "POINT (-73.8823~
```

This gives up a glimpse at the kind of datatypes each column contains. Now with this information, we can try and look at following factors in the subsequent sections. This is in an attempt to model our problem to see possible indicators to help us answer the trends that we could see in the dataset.

- Geographical Distribution
- Monthly Shooting Trends
- Victims/Perpetrators Demographics by Sex
- Victims/Perpetrators Demographics by Race

But before we move on to that, we should focus on data cleaning, processing to get it ready for analysis and visualizations.

# Data Preprocessing

1. Handle missing values in our critical variables [age, sex, race, outcome]:

```r
dataset <- dataset %>%filter(!is.na(VIC_AGE_GROUP), !is.na(VIC_SEX), !is.na(VIC_RACE))
```

2. Convert the datatypes of columns

```r
dataset$VIC_AGE_GROUP <- as.factor(dataset$VIC_AGE_GROUP)
dataset$VIC_SEX <- as.factor(dataset$VIC_SEX)
dataset$VIC_RACE <- as.factor(dataset$VIC_RACE)
dataset$OCCUR_DATE <- as.Date(dataset$OCCUR_DATE, "%m/%d/%Y")
```

3. Create a new variable for time of day based on incident time

```
dataset <- dataset %>%
  filter(!is.na(VIC_AGE_GROUP), !is.na(VIC_SEX), !is.na(VIC_RACE)) %>%
  mutate(
    VIC_AGE_GROUP = as.factor(VIC_AGE_GROUP),
    VIC_SEX = as.factor(VIC_SEX),
    VIC_RACE = as.factor(VIC_RACE),
    OCCUR_DATE = as.Date(OCCUR_DATE),
    MONTH = format(OCCUR_DATE, '%m'),
    time_of_day = cut(as.POSIXlt(OCCUR_TIME, format='%H:%M:%S')$hour,
                      breaks=c(-1, 6, 12, 18, 24),
                      labels=c('Night', 'Morning', 'Afternoon', 'Evening'))
  )
```

# Data Visualization

## Geographic Distribution (Heatmap)

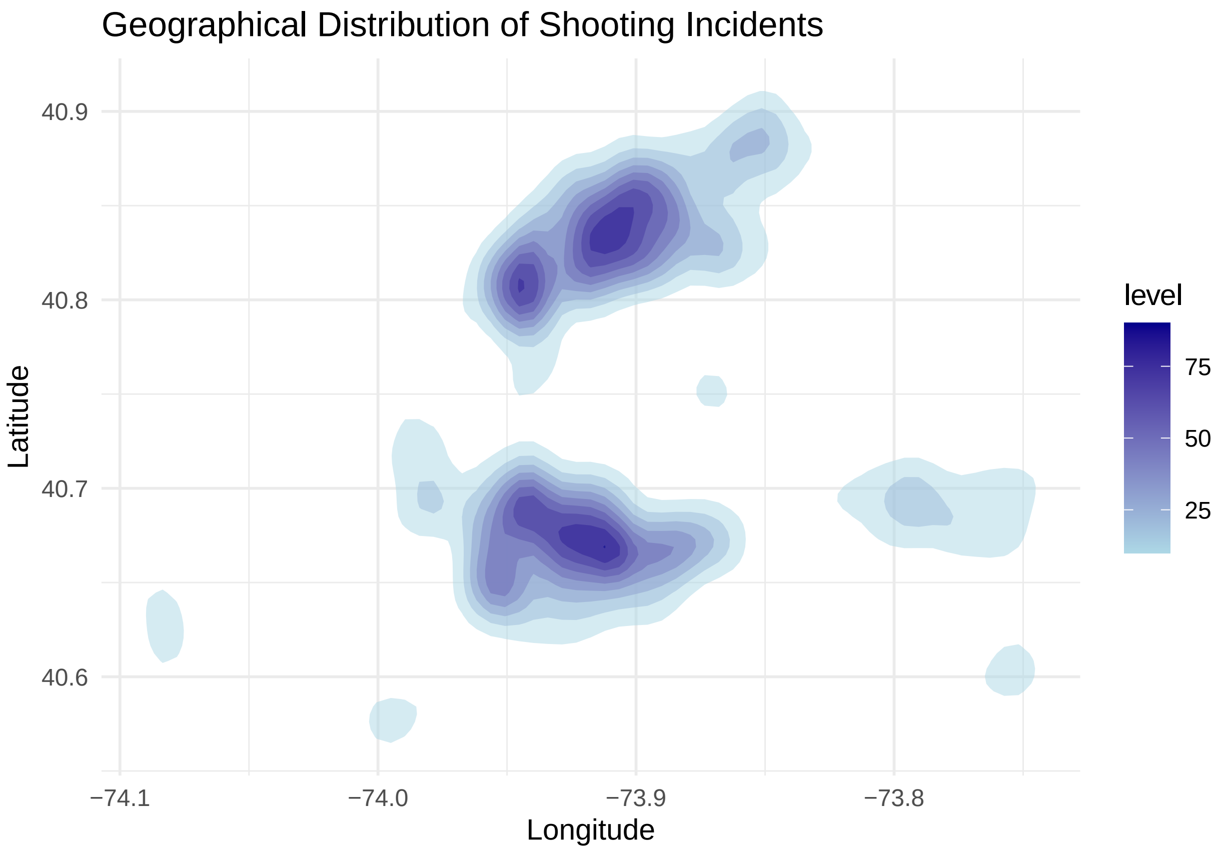
We already have some location data that we could use to generate a heatmap to see which areas are more prone to gun violence

### Create a heatmap

```
ggplot(dataset, aes(x = Longitude, y = Latitude)) +
  stat_density_2d(aes(fill = ..level..), geom = "polygon", alpha = 0.5) +
  scale_fill_gradient(low = "lightblue", high = "darkblue") +
  labs(title = "Geographical Distribution of Shooting Incidents",
       x = "Longitude", y = "Latitude") +
  theme_minimal()
```

```
## Warning: The dot-dot notation (‘..level..‘) was deprecated in ggplot2 3.4.0.
## i Please use ‘after_stat(level)‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

```
## Warning: Removed 59 rows containing non-finite outside the scale range
## (‘stat_density2d()‘).
```




Now that we have a heatmap, let's find the top 5 locations based on this heatmap data.

#### Count the number of incidents per location

```
top_locations <- dataset %>%
  group_by(BORO) %>%
  summarise(count = n()) %>%
  arrange(desc(count))
```

#### Display the top locations

```
print(top_locations)
```

```
## # A tibble: 5 x 2
##   BORO          count
##   <chr>         <int>
## 1 BROOKLYN      11346
## 2 BRONX          8376
## 3 QUEENS         4271
## 4 MANHATTAN      3762
## 5 STATEN ISLAND   807
```

This makes it clear that Brooklyn is probably the most dangerous area in all of New York City in terms of gun violence followed by Bronx, Queens, Manhattan and Staten Island.

## Monthly Shooting Trends (Line Chart)

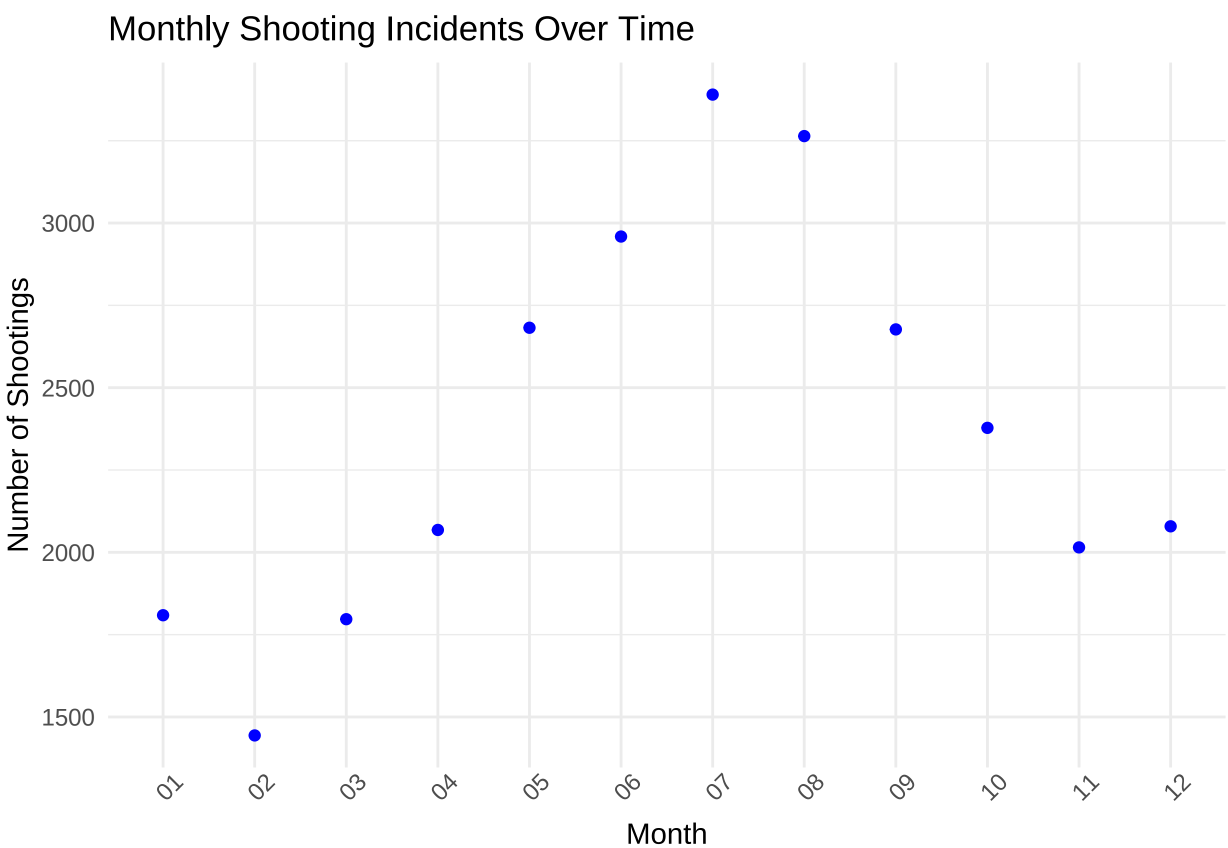
Now, let's take a look at the monthly shooting numbers by aggregating it on a monthly basis:

```
### Aggregate data by month
monthly_shootings <- dataset %>%
  group_by(MONTH) %>%
  summarise(count = n())
```

### Create a line chart for monthly shootings

```
ggplot(monthly_shootings, aes(x = MONTH, y = count)) +
  geom_line(color="blue") +
  geom_point(color="blue") +
  labs(title="Monthly Shooting Incidents Over Time",
       x="Month", y="Number of Shootings") +
  theme_minimal() +
  theme(axis.text.x=element_text(angle=45))
```

```
## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
```




So the chart actually shows that there is steady rise in shootings in the middle of the year and it shootings reduce in number towards the later part of the year. That's an interesting observation although this might not actually mean anything.

## Victim/Perpetrator Demographics by Sex (Bar Chart)

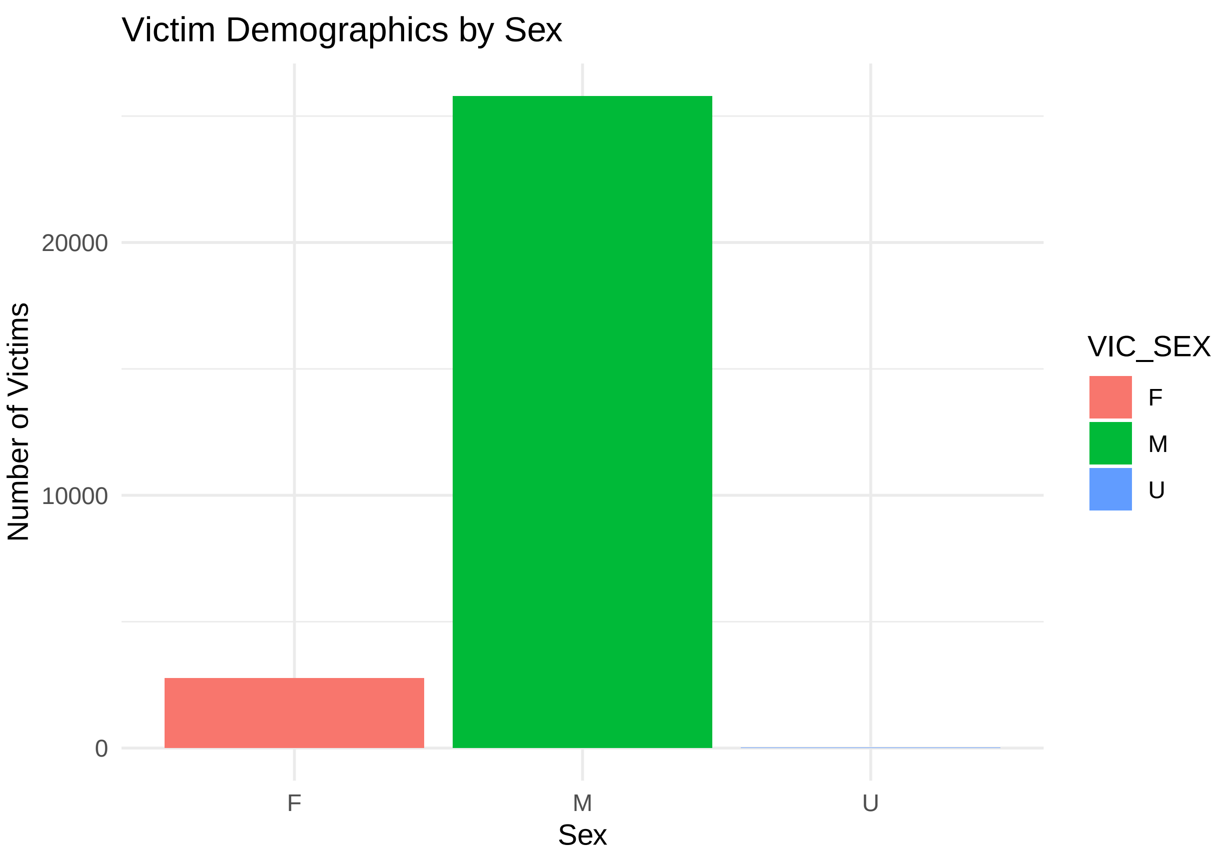
Let's begin with preparing the data for victim's sex and perperator's sex to analyse which sex causes more violence and is subjected to gun violence in the city.

### Prep data based on victom and perpetrator's sex

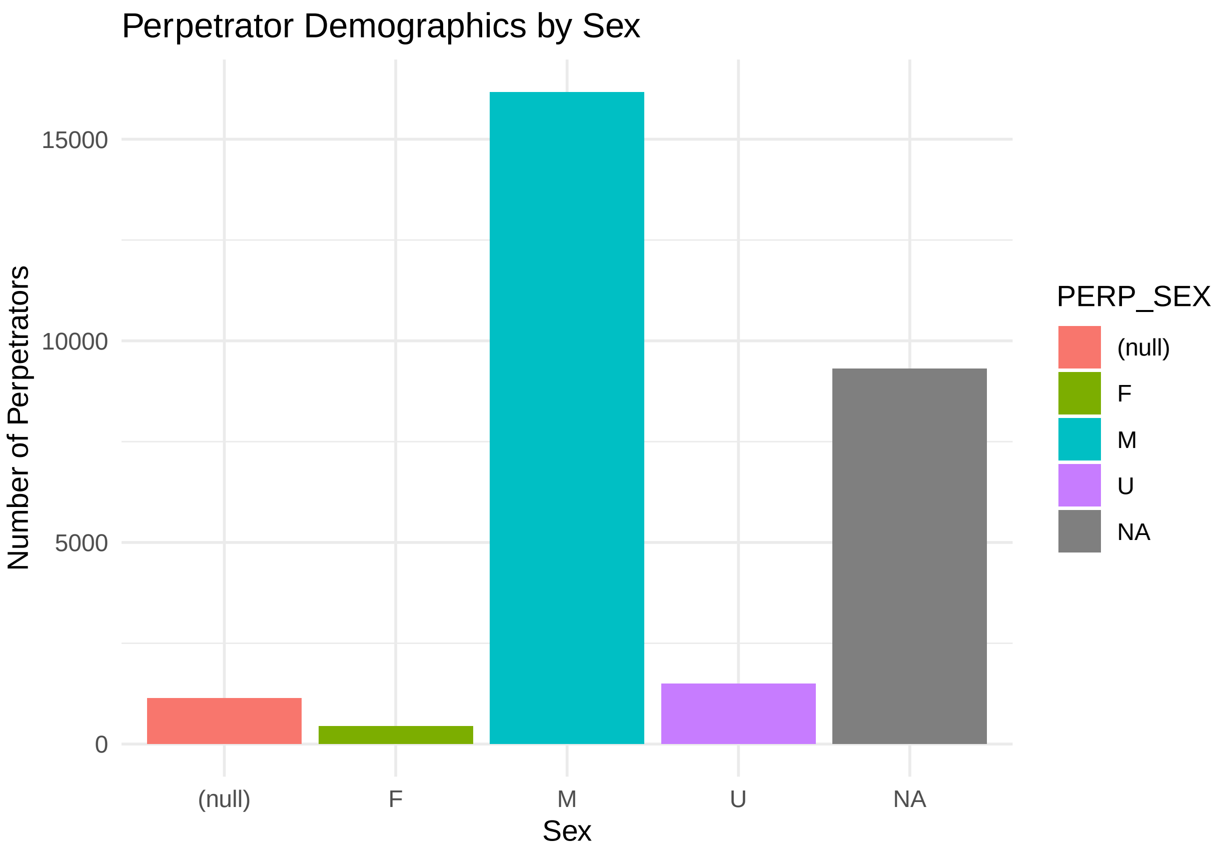
```
victim_sex <- dataset %>%
  group_by(VIC_SEX) %>%
  summarise(count=n())

perpetrator_sex <- dataset %>%
  group_by(PERP_SEX) %>%
  summarise(count=n())
```

### Create a bar chart

```
ggplot(victim_sex, aes(x=VIC_SEX, y=count, fill=VIC_SEX)) +
  geom_bar(stat="identity") +
  labs(title="Victim Demographics by Sex",
       x="Sex", y="Number of Victims") +
  theme_minimal()
```



```r
ggplot(perpetrator_sex, aes(x=PERP_SEX, y=count, fill=PERP_SEX)) +
  geom_bar(stat="identity") +
  labs(title="Perpetrator Demographics by Sex",
       x="Sex", y="Number of Perpetrators") +
  theme_minimal()
```




It's quite clear that victims are mostly males howvever we can't draw the same conclusion about Prepetrator's sex since there are a lot of null and NA values also in the data we makes it difficult to come to a conclusion.

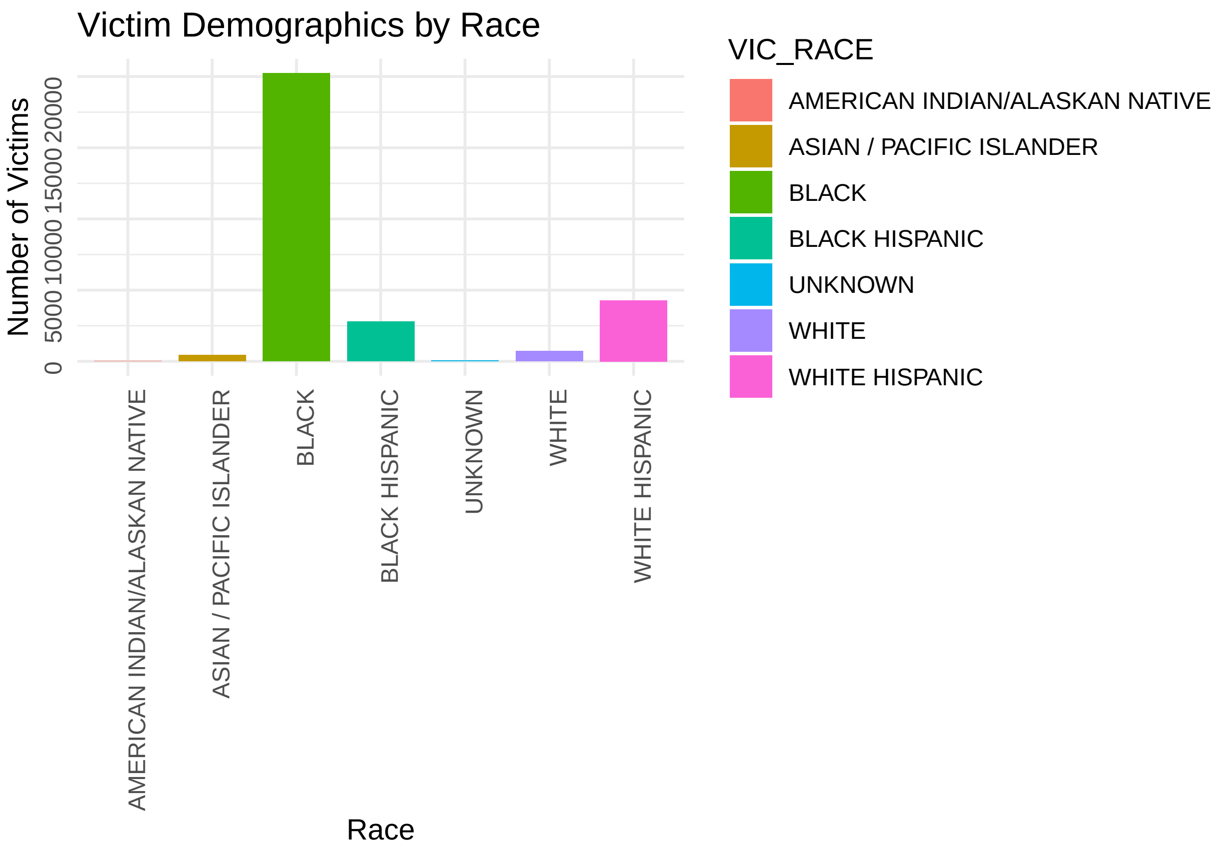
## Victim/Perpetrator's Demography by Race

We also have some information on the kind of weapons used which can be useful to deduce and understand what kinds of weapons caused the most harm.

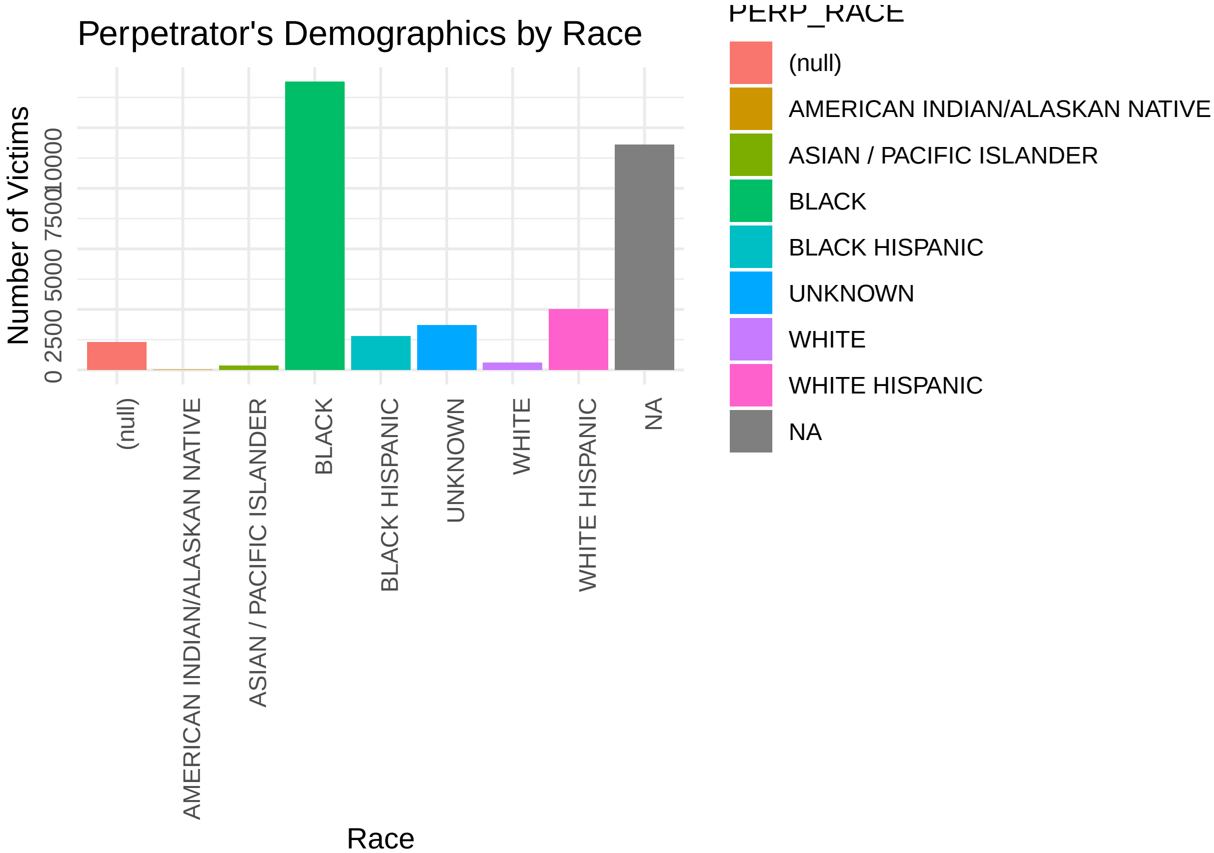
### Group dataset by victim's and perpetrator's race

```r
victim_race <- dataset %>%
  group_by(VIC_RACE) %>%
  summarise(count=n())

perpetrators_race <- dataset %>%
  group_by(PERP_RACE) %>%
  summarise(count=n())
```

```r
### Create a bar chart for victim's race
ggplot(victim_race, aes(x=VIC_RACE, y=count, fill=VIC_RACE)) +
  geom_bar(stat="identity", width=0.8) +  # Adjust the width as needed
  labs(title="Victim Demographics by Race",
       x="Race", y="Number of Victims") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1),  # Vertical x-axis labels
        axis.text.y = element_text(angle = 90, vjust = 0.5)) # Vertical y-axis labels
```




```r
### Create a bar chart for perpetrator's race
ggplot(perpetrators_race, aes(x=PERP_RACE, y=count, fill=PERP_RACE)) +
  geom_bar(stat="identity") +
  labs(title="Perpetrator's Demographics by Race",
       x="Race", y="Number of Victims") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1),  # Vertical x-axis labels
        axis.text.y = element_text(angle = 90, vjust = 0.5)) # Vertical y-axis labels
```




Similar conclusions can be drawn here about the race of the victim/perpetrator. We can see that there is a lot of cases where the data about the race is missing for perpetrator's which makes it difficult to say which race is the most violent. More importantly, socioeconomic factors also play an important role in determining whether the race of the victim/perpetrator should actually matter. In our case, it makes more sense to attribute it to socioeconomic factors rather than simply making a conclusion based on the race of the perpetrator or the victim.

### An attempt to predict shootings in the year 2024

In order to predict the shootings, we must first analyze the crimes happening each year to check whether there is linear trend in number of shootings per year or if it is monotonic in nature.

First we start with aligning some of variables that we want to get prediction

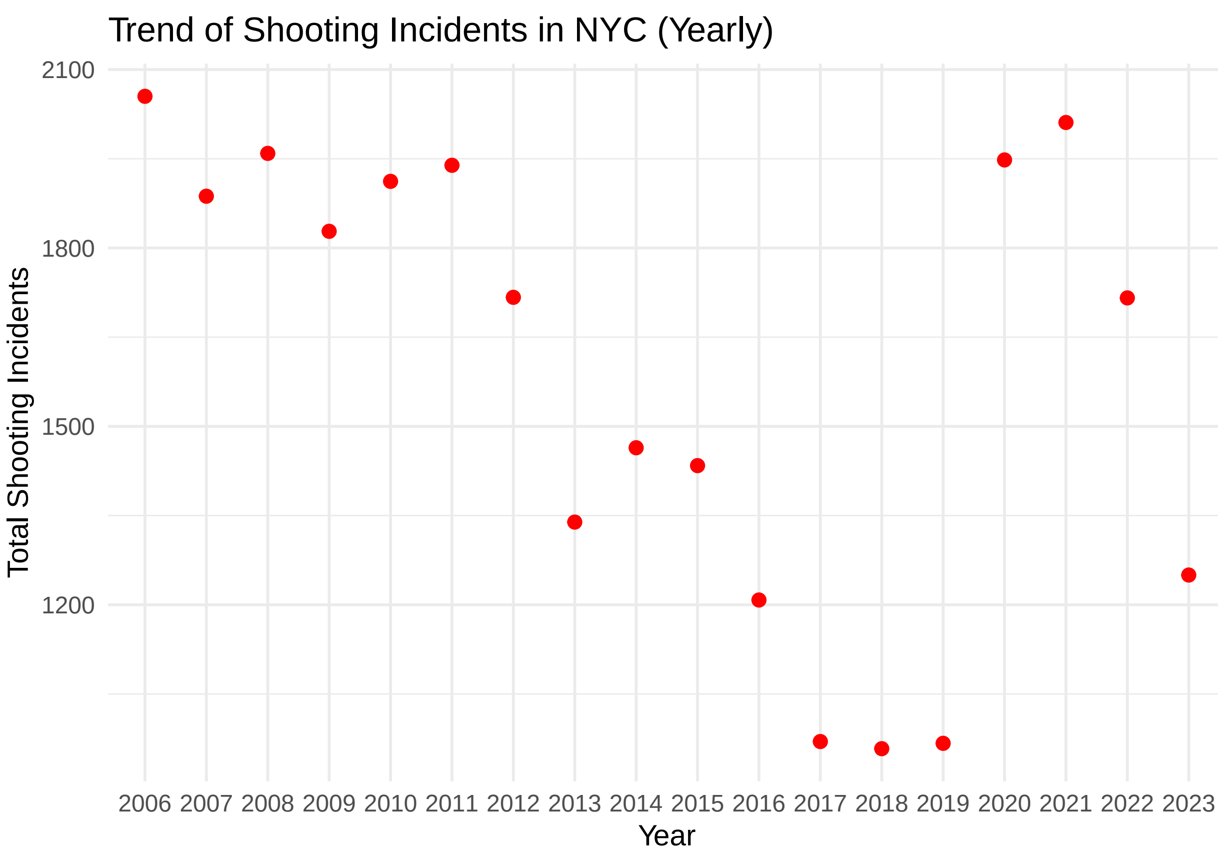
```
model_shooting <- dataset$OCCUR_DATE
shooting_by_year <- format(model_shooting, "%Y")
count_by_year <- table(shooting_by_year)
dataset_group_by_year <- as.data.frame(count_by_year)
names(dataset_group_by_year) <- c("Year", "Count")
```

Now, we can plot the shootings per year to check for the pattern that is visible.

```
ggplot(dataset_group_by_year, aes(x = Year, y = Count)) +
  geom_line(color = "blue", size = 1) +
  geom_point(color = "red", size = 2) +
  labs(title = "Trend of Shooting Incidents in NYC (Yearly)",
```

```
      x = "Year",
      y = "Total Shooting Incidents") +
  theme_minimal()
```

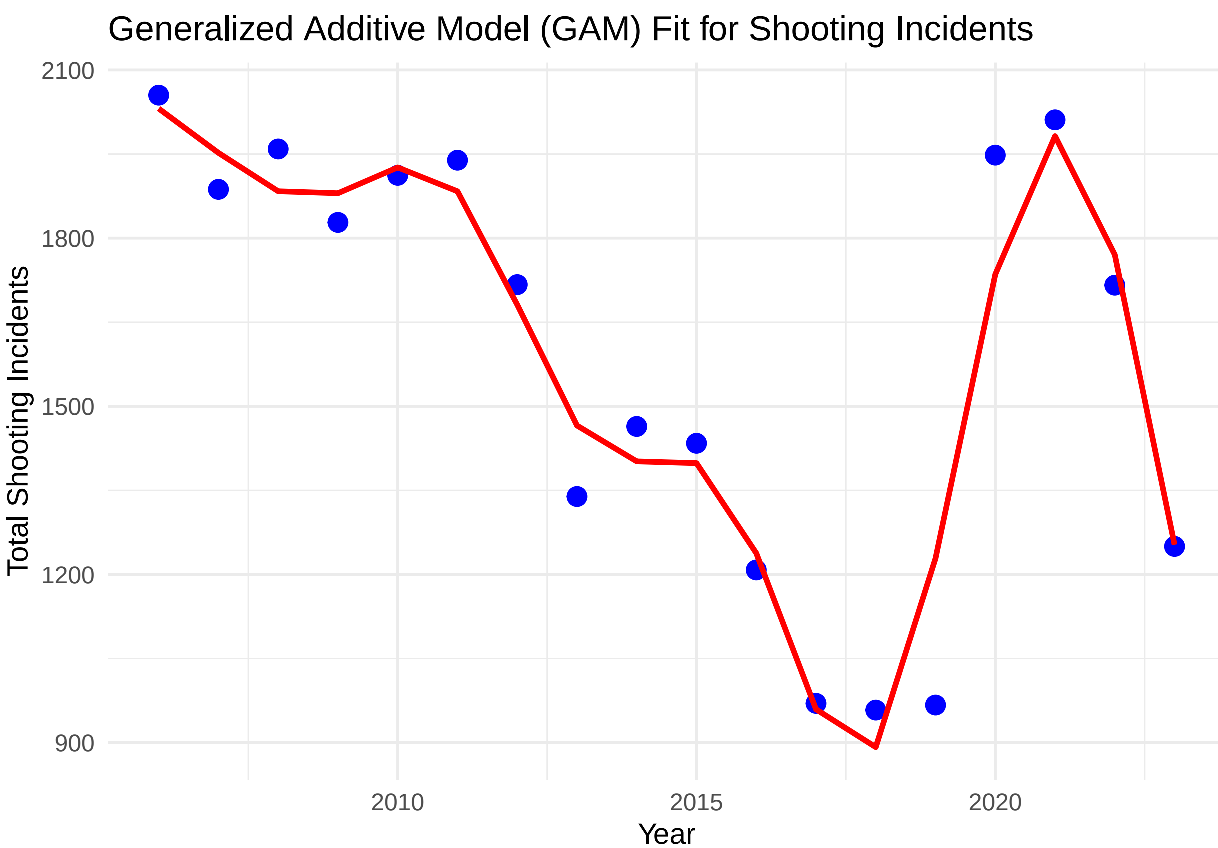
```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.

## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
```



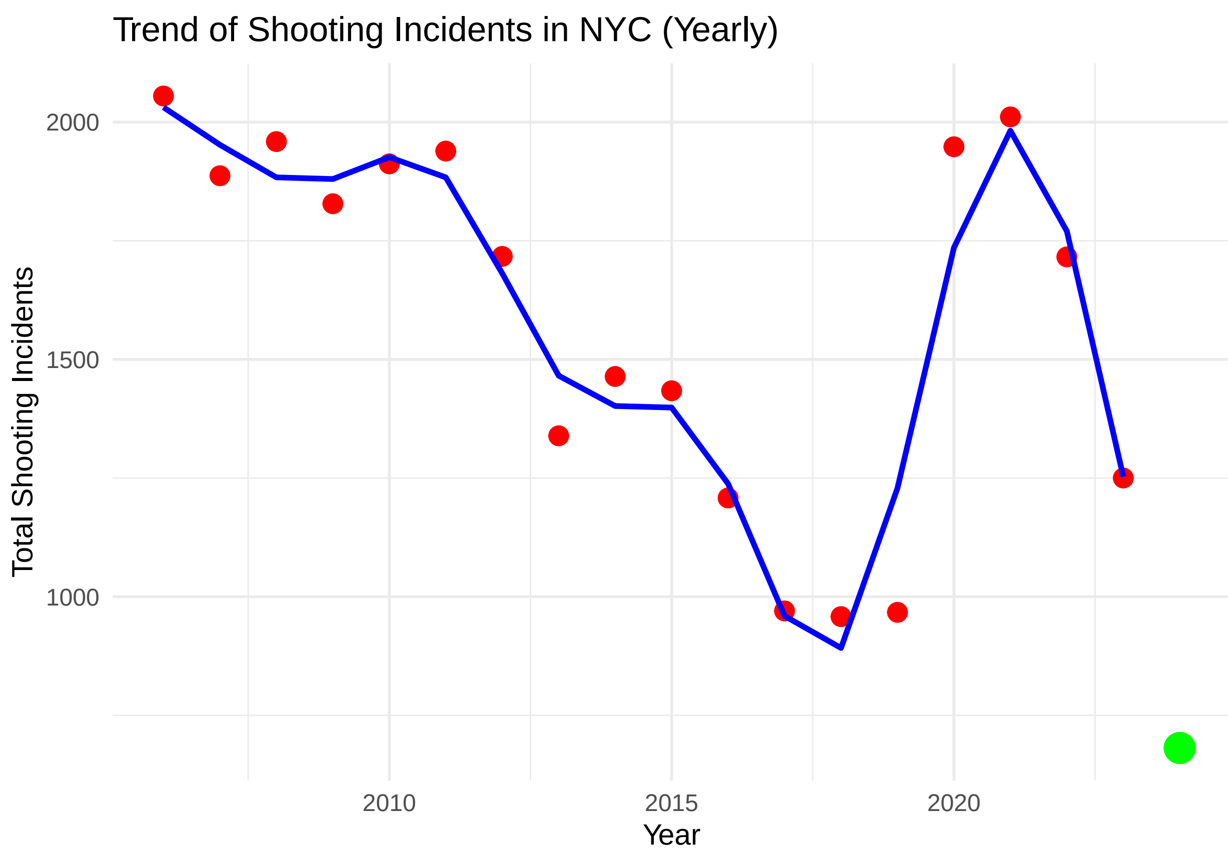

Looking at the general trend of shootings over the years, we can notice that the data is non monotonic in nature. Due to the nature of the data, we can use Generalized Addive Model instead to predict the shootings that could take place for 2024. GAM models are highly accurate for predicting future trends within the range of observed data due to their flexibility and ability to capture non-linear relationships effectively.

```
dataset_group_by_year$Year <- as.numeric(as.character(dataset_group_by_year$Year))
gam_model <- gam(Count ~ s(Year), data = dataset_group_by_year)
# Create predictions for visualization
dataset_group_by_year$predicted_incidents <- predict(gam_model)

# Plot original data and GAM fit
ggplot(dataset_group_by_year, aes(x = Year)) +
  geom_point(aes(y = Count), color = "blue", size = 3) + # Original data points
```

```
  geom_line(aes(y = predicted_incidents), color = "red", size = 1) + # GAM fit
  labs(title = "Generalized Additive Model (GAM) Fit for Shooting Incidents",
       x = "Year",
       y = "Total Shooting Incidents") +
  theme_minimal()
```




```
prediction <- predict(gam_model, newdata = data.frame(Year = 2024))

# Add the prediction to the dataset for plotting
annual_shootings_with_prediction <- rbind(
  dataset_group_by_year,
  data.frame(Year = 2024, Count = NA, predicted_incidents = prediction)
)

# Create predictions for all years in the dataset
dataset_group_by_year$predicted_incidents <- predict(gam_model)

# Plot original data, fitted curve, and prediction for 2024
ggplot(dataset_group_by_year, aes(x = Year)) +
  geom_point(aes(y = Count), color = "red", size = 3) + # Original data points
  geom_line(aes(y = predicted_incidents), color = "blue", size = 1) + # Fitted curve
  geom_point(data = data.frame(Year = 2024), aes(x = Year, y = prediction), color = "green", size = 5)
  labs(title = "Trend of Shooting Incidents in NYC (Yearly)",
       x = "Year",
       y = "Total Shooting Incidents") +
  theme_minimal()
```




```r
#Display it on a table as well
print(annual_shootings_with_prediction)
```

```
##    Year Count predicted_incidents
## 1  2006  2055           2031.1357
## 2  2007  1887           1952.1059
## 3  2008  1959           1883.5584
## 4  2009  1828           1880.0367
## 5  2010  1912           1926.1174
## 6  2011  1939           1883.5244
## 7  2012  1717           1681.2103
## 8  2013  1339           1465.6308
## 9  2014  1464           1401.7248
## 10 2015  1434           1398.5517
## 11 2016  1208           1237.9556
## 12 2017   970            959.4039
## 13 2018   958            891.8791
## 14 2019   967           1228.7555
## 15 2020  1948           1735.3574
## 16 2021  2011           1981.9584
## 17 2022  1716           1770.2132
## 18 2023  1250           1252.8809
## 19 2024    NA            681.1709
```

## Bias and Conclusion

In terms of bias, the data quite easily would push us to conclude that race plays a role in deciding which groups have the highest shooting incidents. However, this also requires us to consider other datasets as well to better understand the socioeconomic situation the regions where there is a lot of gun violence.

We can only make a proper conclusion once we have that data and hence it makes it important to mitigate our bias by just looking at the data and drawing a conclusion when we should clearly be asking more questions.