NYPD Shooting Incident Report

Robert Forrest

April 11th 2024

```
knitr::opts_chunk$set(echo = TRUE)
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   3.5.0     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(lubridate)
```

Intro

In this report I will import and clean the NYPD shooting incident data and then do some analysis. Afterwards I will create a model that will predict the number of shootings expected.

Import Data

Reads in the data from the government dataset catalog website.

```
data <- read_csv("https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv",
                 show_col_types = FALSE)
summary(data)
```

```
##   INCIDENT_KEY        OCCUR_DATE         OCCUR_TIME           BORO
##  Min.   :  9953245   Length:27312       Length:27312       Length:27312
##  1st Qu.: 63860880   Class :character   Class1:hms         Class :character
##  Median : 90372218   Mode  :character   Class2:difftime    Mode  :character
##  Mean   :120860536                      Mode  :numeric
##  3rd Qu.:188810230
##  Max.   :261190187
##
##  LOC_OF_OCCUR_DESC     PRECINCT      JURISDICTION_CODE LOC_CLASSFCTN_DESC
##  Length:27312       Min.   :  1.00   Min.   :0.0000    Length:27312
##  Class :character   1st Qu.: 44.00   1st Qu.:0.0000    Class :character
##  Mode  :character   Median : 68.00   Median :0.0000    Mode  :character
##                     Mean   : 65.64   Mean   :0.3269
##                     3rd Qu.: 81.00   3rd Qu.:0.0000
##                     Max.   :123.00   Max.   :2.0000
```

```
##                                     NA's   :2
##  LOCATION_DESC      STATISTICAL_MURDER_FLAG PERP_AGE_GROUP
##  Length:27312       Mode :logical           Length:27312
##  Class :character   FALSE:22046             Class :character
##  Mode  :character   TRUE :5266              Mode  :character
##
##
##
##
##    PERP_SEX           PERP_RACE          VIC_AGE_GROUP        VIC_SEX
##  Length:27312       Length:27312       Length:27312       Length:27312
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    VIC_RACE           X_COORD_CD        Y_COORD_CD        Latitude
##  Length:27312       Min.   : 914928   Min.   :125757   Min.   :40.51
##  Class :character   1st Qu.:1000028   1st Qu.:182834   1st Qu.:40.67
##  Mode  :character   Median :1007731   Median :194487   Median :40.70
##                     Mean   :1009449   Mean   :208127   Mean   :40.74
##                     3rd Qu.:1016838   3rd Qu.:239518   3rd Qu.:40.82
##                     Max.   :1066815   Max.   :271128   Max.   :40.91
##                                                        NA's   :10
##    Longitude         Lon_Lat
##  Min.   :-74.25   Length:27312
##  1st Qu.:-73.94   Class :character
##  Median :-73.92   Mode  :character
##  Mean   :-73.91
##  3rd Qu.:-73.88
##  Max.   :-73.70
##  NA's   :10
```

Tidy and Transform

To clean the data, I will drop columns that I'm not interested in and will fix the occur_date column by transforming it into a real date column.

```
data <- data %>%
  mutate(OCCUR_DATE = mdy(OCCUR_DATE)) %>%
  select(-c(INCIDENT_KEY, JURISDICTION_CODE, LOC_OF_OCCUR_DESC, LOC_CLASSFCTN_DESC,
            X_COORD_CD, Y_COORD_CD, Latitude, Longitude, Lon_Lat))
print(names(data))
```

```
##  [1] "OCCUR_DATE"              "OCCUR_TIME"
##  [3] "BORO"                    "PRECINCT"
##  [5] "LOCATION_DESC"           "STATISTICAL_MURDER_FLAG"
##  [7] "PERP_AGE_GROUP"          "PERP_SEX"
##  [9] "PERP_RACE"               "VIC_AGE_GROUP"
## [11] "VIC_SEX"                 "VIC_RACE"
```

Look at unique values in order to do some potential replacements.

```
# These are the only columns I want to look at for unique values.
columns_to_check <- setdiff(names(data), c("OCCUR_DATE", "OCCUR_TIME", "PRECINCT"))
```

```r
# Custom function to display counts of unique values in each column
print_unique_counts <- function(df) {
  for (col in names(df)) {
    cat("Column:", col, "\n")
    cat("------------\n")
    counts <- table(df[[col]])
    for (val in names(counts)) {
      cat(val, ": ", counts[val], "\n")
    }
    cat("\n")
  }
}

print_unique_counts(data[columns_to_check])
```

```
## Column: BORO
## ------------
## BRONX :  7937
## BROOKLYN :  10933
## MANHATTAN :  3572
## QUEENS :  4094
## STATEN ISLAND :  776
##
## Column: LOCATION_DESC
## ------------
## (null) :  977
## ATM :  1
## BANK :  3
## BAR/NIGHT CLUB :  628
## BEAUTY/NAIL SALON :  112
## CANDY STORE :  7
## CHAIN STORE :  5
## CHECK CASH :  1
## CLOTHING BOUTIQUE :  14
## COMMERCIAL BLDG :  292
## DEPT STORE :  9
## DOCTOR/DENTIST :  1
## DRUG STORE :  14
## DRY CLEANER/LAUNDRY :  31
## FACTORY/WAREHOUSE :  8
## FAST FOOD :  104
## GAS STATION :  71
## GROCERY/BODEGA :  694
## GYM/FITNESS FACILITY :  3
## HOSPITAL :  65
## HOTEL/MOTEL :  35
## JEWELRY STORE :  12
## LIQUOR STORE :  41
## LOAN COMPANY :  1
## MULTI DWELL - APT BUILD :  2835
## MULTI DWELL - PUBLIC HOUS :  4832
## NONE :  175
## PHOTO/COPY STORE :  1
```

```
## PVT HOUSE :  951
## RESTAURANT/DINER :  204
## SCHOOL :  1
## SHOE STORE :  10
## SMALL MERCHANT :  37
## SOCIAL CLUB/POLICY LOCATI :  72
## STORAGE FACILITY :  1
## STORE UNCLASSIFIED :  36
## SUPERMARKET :  21
## TELECOMM. STORE :  11
## VARIETY STORE :  11
## VIDEO STORE :  8
##
## Column: STATISTICAL_MURDER_FLAG
## ------------
## FALSE :  22046
## TRUE :  5266
##
## Column: PERP_AGE_GROUP
## ------------
## (null) :  640
## <18 :  1591
## 1020 :  1
## 18-24 :  6222
## 224 :  1
## 25-44 :  5687
## 45-64 :  617
## 65+ :  60
## 940 :  1
## UNKNOWN :  3148
##
## Column: PERP_SEX
## ------------
## (null) :  640
## F :  424
## M :  15439
## U :  1499
##
## Column: PERP_RACE
## ------------
## (null) :  640
## AMERICAN INDIAN/ALASKAN NATIVE :  2
## ASIAN / PACIFIC ISLANDER :  154
## BLACK :  11432
## BLACK HISPANIC :  1314
## UNKNOWN :  1836
## WHITE :  283
## WHITE HISPANIC :  2341
##
## Column: VIC_AGE_GROUP
## ------------
## <18 :  2839
## 1022 :  1
## 18-24 :  10086
```

```
## 25-44 :  12281
## 45-64 :  1863
## 65+ :  181
## UNKNOWN :  61
##
## Column: VIC_SEX
## ------------
## F :  2615
## M :  24686
## U :  11
##
## Column: VIC_RACE
## ------------
## AMERICAN INDIAN/ALASKAN NATIVE :  10
## ASIAN / PACIFIC ISLANDER :  404
## BLACK :  19439
## BLACK HISPANIC :  2646
## UNKNOWN :  66
## WHITE :  698
## WHITE HISPANIC :  4049
```

Next I will replace empty string data or unknown-like/messy data with "UNKNOWN".

```
# Define function to handle replacement unknown-like values
replace_with_unknown <- function(x) {
  ifelse(x == "" | x == "(null)" | is.na(x) | x == "U" | x == "1022" | x == "1020"
         | x == "940" | x == "224", "UNKNOWN", x)
}

data <- data %>%
  mutate(across(all_of(columns_to_check), ~ replace_with_unknown(.)))

print_unique_counts(data[columns_to_check])
```

```
## Column: BORO
## ------------
## BRONX :  7937
## BROOKLYN :  10933
## MANHATTAN :  3572
## QUEENS :  4094
## STATEN ISLAND :  776
##
## Column: LOCATION_DESC
## ------------
## ATM :  1
## BANK :  3
## BAR/NIGHT CLUB :  628
## BEAUTY/NAIL SALON :  112
## CANDY STORE :  7
## CHAIN STORE :  5
## CHECK CASH :  1
## CLOTHING BOUTIQUE :  14
## COMMERCIAL BLDG :  292
## DEPT STORE :  9
## DOCTOR/DENTIST :  1
```

```
## DRUG STORE :  14
## DRY CLEANER/LAUNDRY :  31
## FACTORY/WAREHOUSE :  8
## FAST FOOD :  104
## GAS STATION :  71
## GROCERY/BODEGA :  694
## GYM/FITNESS FACILITY :  3
## HOSPITAL :  65
## HOTEL/MOTEL :  35
## JEWELRY STORE :  12
## LIQUOR STORE :  41
## LOAN COMPANY :  1
## MULTI DWELL - APT BUILD :  2835
## MULTI DWELL - PUBLIC HOUS :  4832
## NONE :  175
## PHOTO/COPY STORE :  1
## PVT HOUSE :  951
## RESTAURANT/DINER :  204
## SCHOOL :  1
## SHOE STORE :  10
## SMALL MERCHANT :  37
## SOCIAL CLUB/POLICY LOCATI :  72
## STORAGE FACILITY :  1
## STORE UNCLASSIFIED :  36
## SUPERMARKET :  21
## TELECOMM. STORE :  11
## UNKNOWN :  15954
## VARIETY STORE :  11
## VIDEO STORE :  8
##
## Column: STATISTICAL_MURDER_FLAG
## ------------
## FALSE :  22046
## TRUE :  5266
##
## Column: PERP_AGE_GROUP
## ------------
## <18 :  1591
## 18-24 :  6222
## 25-44 :  5687
## 45-64 :  617
## 65+ :  60
## UNKNOWN :  13135
##
## Column: PERP_SEX
## ------------
## F :  424
## M :  15439
## UNKNOWN :  11449
##
## Column: PERP_RACE
## ------------
## AMERICAN INDIAN/ALASKAN NATIVE :  2
## ASIAN / PACIFIC ISLANDER :  154
```

```
## BLACK :  11432
## BLACK HISPANIC :  1314
## UNKNOWN :  11786
## WHITE :  283
## WHITE HISPANIC :  2341
##
## Column: VIC_AGE_GROUP
## ------------
## <18 :  2839
## 18-24 :  10086
## 25-44 :  12281
## 45-64 :  1863
## 65+ :  181
## UNKNOWN :  62
##
## Column: VIC_SEX
## ------------
## F :  2615
## M :  24686
## UNKNOWN :  11
##
## Column: VIC_RACE
## ------------
## AMERICAN INDIAN/ALASKAN NATIVE :  10
## ASIAN / PACIFIC ISLANDER :  404
## BLACK :  19439
## BLACK HISPANIC :  2646
## UNKNOWN :  66
## WHITE :  698
## WHITE HISPANIC :  4049
```

Next I'll add some more columns of interest.

```
data <- data %>%
  mutate(DAY_OF_WEEK = wday(OCCUR_DATE, label = TRUE, abbr = FALSE)) %>%
  mutate(DAYS_SINCE_DATA_START = as.numeric(OCCUR_DATE - min(data$OCCUR_DATE)))

data_start_date = format(min(data$OCCUR_DATE), "%Y-%m-%d")

cat("Data start date=", data_start_date)
```

```
## Data start date= 2006-01-01
```

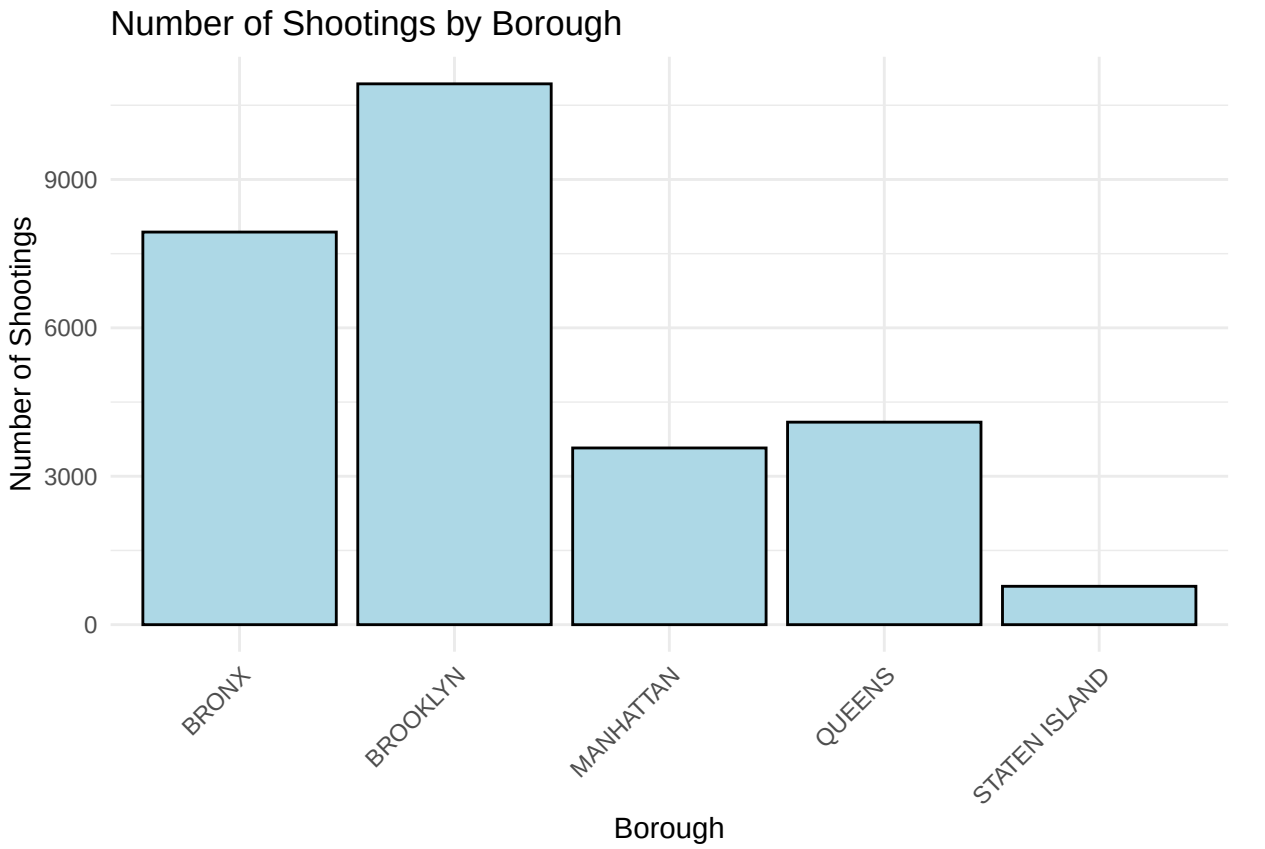
Visualization and Analysis

This is a plot of the number of shootings by Borough.

```
event_counts_boro <- data %>%
  group_by(BORO) %>%
  summarise(NUM_SHOOTINGS = n())

ggplot(event_counts_boro, aes(x = BORO, y = NUM_SHOOTINGS)) +
  geom_bar(stat = "identity", fill = "lightblue", color = "black") +
  labs(title = "Number of Shootings by Borough",
       x = "Borough",
       y = "Number of Shootings") +
```

```
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

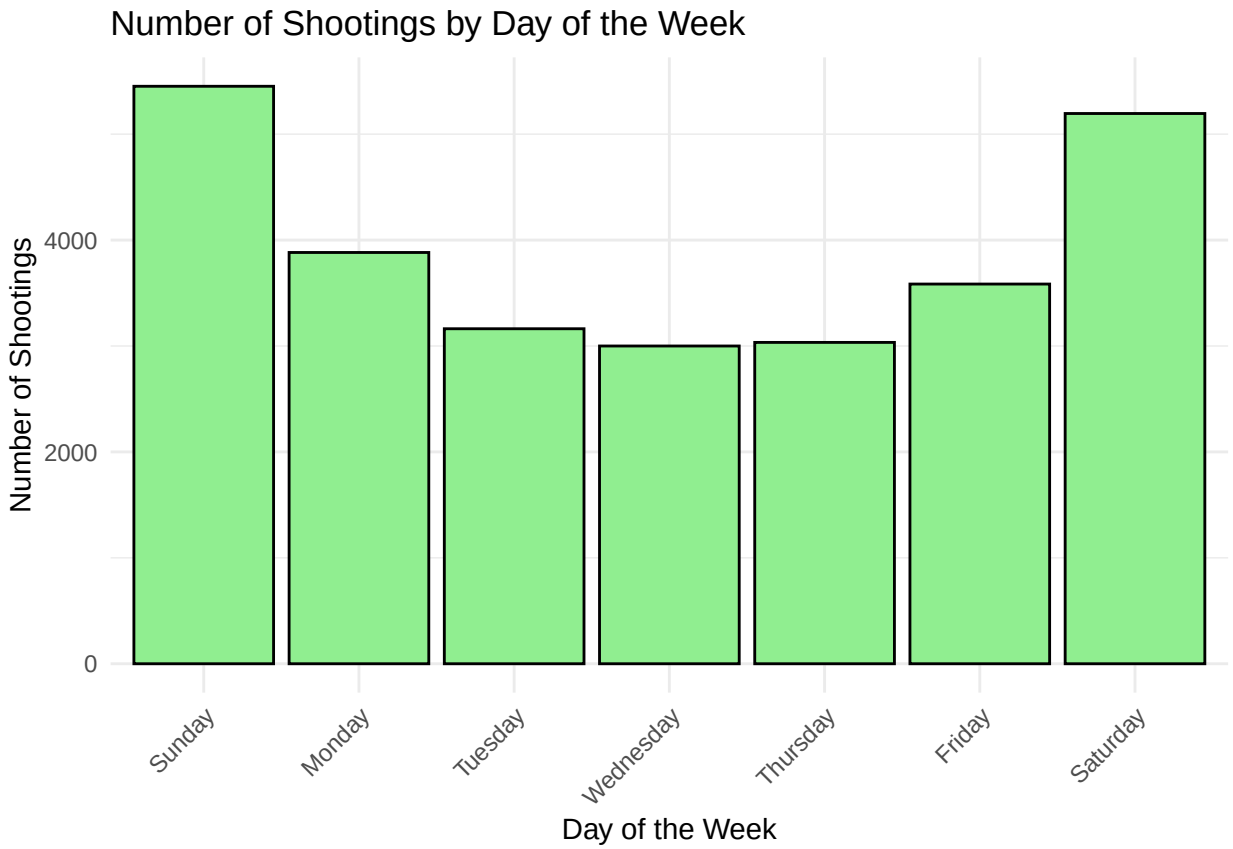


So here we see that Brooklyn has the highest number of shootings and Staten Island has the lowest. Its possible that the differences are related to the population, so further analysis would be required in order to rule that possibility out.

This is a plot of the number of shootings by day of the week.

```
event_counts_day <- data %>%
  group_by(DAY_OF_WEEK) %>%
  summarise(NUM_SHOOTINGS = n())

ggplot(event_counts_day, aes(x = DAY_OF_WEEK, y = NUM_SHOOTINGS)) +
  geom_bar(stat = "identity", fill = "lightgreen", color = "black") +
  labs(title = "Number of Shootings by Day of the Week",
       x = "Day of the Week",
       y = "Number of Shootings") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

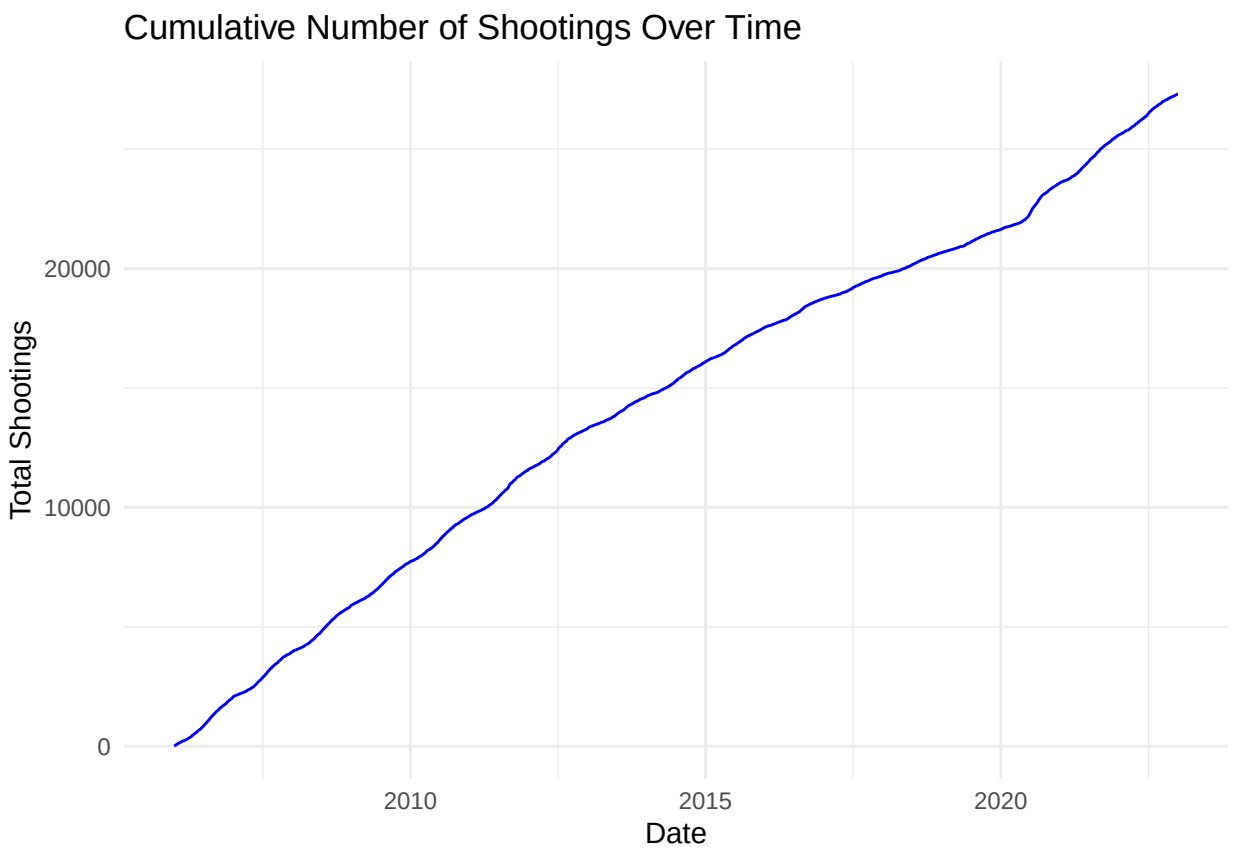


It looks like the weekends are when shootings happen the most. I suspect that its due to people being more "free" during the weekends typically. Or if these are happening at night, people are more likely to me active during weekend nights vs weekday nights.

Plot the cumulative number of events over time.

```
event_counts_cumu <- data %>%
  group_by(OCCUR_DATE) %>%
  summarise(NUM_SHOOTINGS = n()) %>%
  arrange(OCCUR_DATE) %>%
  mutate(CUMULATIVE_SHOOTINGS = cumsum(NUM_SHOOTINGS))

ggplot(event_counts_cumu, aes(x = OCCUR_DATE, y = CUMULATIVE_SHOOTINGS)) +
  geom_line(color = "blue") +
  labs(title = "Cumulative Number of Shootings Over Time",
       x = "Date",
       y = "Total Shootings") +
  theme_minimal()
```



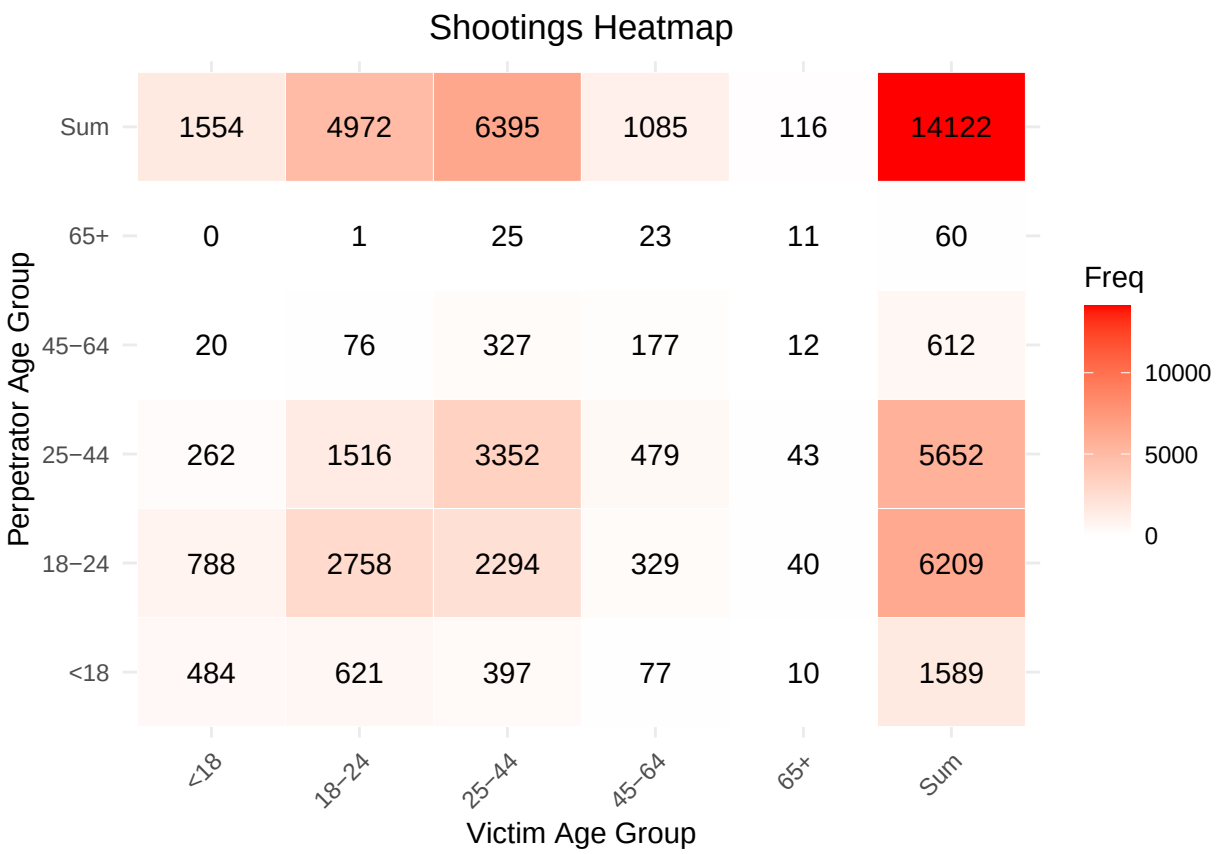
Based on the plot above, it looks like the shootings are going up in a straight line. This makes me wonder if there are certain age groups that are more at risk. So here I will plot a heatmap to understand which groups are involved with the most shootings.

```
# Filter out rows where either PERP_AGE_GROUP or VIC_AGE_GROUP is "UNKNOWN"
filtered_data <- data %>%
  filter(PERP_AGE_GROUP != "UNKNOWN" & VIC_AGE_GROUP != "UNKNOWN")

# Create a table of counts for each combination of PERP_AGE_GROUP and VIC_AGE_GROUP
heatmap_data <- table(filtered_data$PERP_AGE_GROUP, filtered_data$VIC_AGE_GROUP)

# Add row and column sums for cumulative counts
heatmap_data <- addmargins(heatmap_data)

# Plot the heatmap with counts within each cell
ggplot(as.data.frame(heatmap_data), aes(x = Var2, y = Var1, fill = Freq, label = Freq)) +
  geom_tile(color = "white") +
  scale_fill_gradient(low = "white", high = "red") +
  geom_text(color = "black") +
  labs(title = "Shootings Heatmap",
       x = "Victim Age Group",
       y = "Perpetrator Age Group") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        plot.title = element_text(hjust = 0.5))
```



These results tell us that the perpetrator group causing the most shootings are the 18-24 year olds. They mostly go after other 18-24 year olds. The group most victimized are the 25-44 year olds. They are mostly victimized by other 25-44 year olds. Contrary to what one might think, the older people (45+) are not victims as often as younger people.

Model

Create a model to predict the total number of shootings, based on the data start date.

```
event_counts_for_model <- data %>%
  group_by(DAYS_SINCE_DATA_START, BORO, PRECINCT, VIC_AGE_GROUP, PERP_AGE_GROUP) %>%
  summarise(NUM_SHOOTINGS = n(), .groups = "drop") %>%
  arrange(DAYS_SINCE_DATA_START) %>%
  mutate(CUMULATIVE_SHOOTINGS = cumsum(NUM_SHOOTINGS))


mod <- lm(CUMULATIVE_SHOOTINGS ~ DAYS_SINCE_DATA_START, data= event_counts_for_model %>%
 filter(PERP_AGE_GROUP != "UNKNOWN" & VIC_AGE_GROUP != "UNKNOWN"))

summary(mod)
```

```
##
## Call:
## lm(formula = CUMULATIVE_SHOOTINGS ~ DAYS_SINCE_DATA_START, data = event_counts_for_model %>%
##     filter(PERP_AGE_GROUP != "UNKNOWN" & VIC_AGE_GROUP != "UNKNOWN"))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -1410.30  -629.52   -75.33   717.14  1291.99
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.360e+03  1.282e+01     106   <2e-16 ***
## DAYS_SINCE_DATA_START 4.195e+00  3.684e-03    1139   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 766.7 on 12020 degrees of freedom
## Multiple R-squared:  0.9908, Adjusted R-squared:  0.9908
## F-statistic: 1.297e+06 on 1 and 12020 DF,  p-value: < 2.2e-16
```

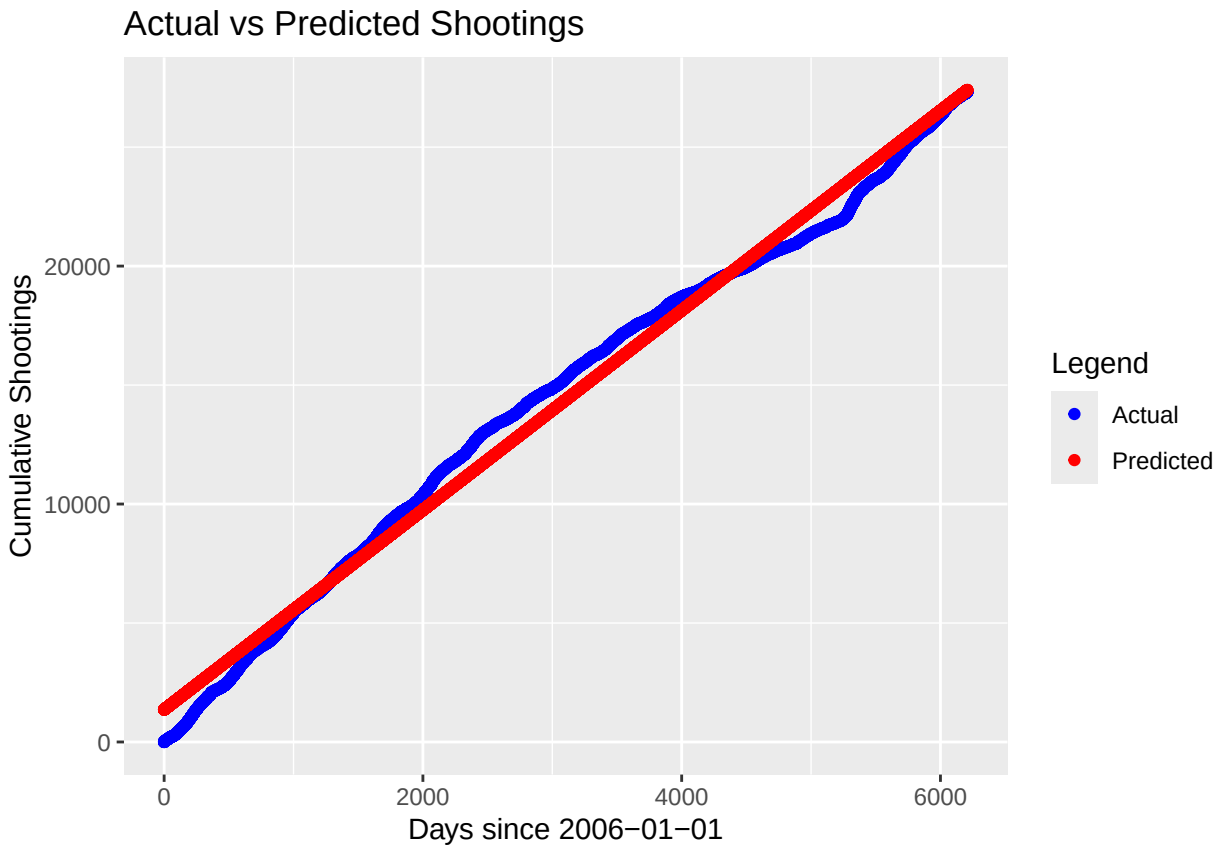
The results from the linear model look good. The p-value is < 0.05 and the R-squared is close to 1.

Visualize the predicted shootings vs actual shootings.

```r
# Create a data frame for predictions with the same structure as event_counts_for_model
new_data <- data.frame(DAYS_SINCE_DATA_START = event_counts_for_model$DAYS_SINCE_DATA_START)

# Add a column for the predicted values
event_counts_for_model$predicted <- predict(mod, newdata = new_data)

# Now can plot and see actual vs predicted
event_counts_for_model %>%
  ggplot() +
  geom_point(aes(x = DAYS_SINCE_DATA_START, y = CUMULATIVE_SHOOTINGS, color = "Actual")) +
  geom_point(aes(x = DAYS_SINCE_DATA_START, y = predicted, color = "Predicted")) +
  labs(title = "Actual vs Predicted Shootings",
       x = paste("Days since", data_start_date),
       y = "Cumulative Shootings") +
  scale_color_manual(values = c("Actual" = "blue", "Predicted" = "red"),
                     labels = c("Actual", "Predicted")) +
  guides(color = guide_legend(title = "Legend"))
```



Conclusions and Bias Identification

In conclusion, there are areas in New York that are more dangerous than others. The primary victims and perpetrators are from 2 different age groups, but are still rather close in age. There unfortunately doesn't appear to be a noticeable decrease in shootings over time.

A source of bias that I have when analyzing this data is in how I handled missing or unknown values. I decided to make all missing and unknown values equal to "UNKNOWN". This means that my analysis and model could greatly under represent reality if it turns out that the most important shootings are originating from the "UNKNOWN" category.

A source of bias from this data, would be due to the shootings only originating from New York. There is almost a universal mindset that New York can be particularly dangerous. Therefore, its important to keep in mind that the data collected here and the associated model, should not be used for other cities. Also, this only has data reported from NYPD. There could be many shootings happening that go unreported or that are in a different police district.

Due to these biases, its important to mitigate them by having the audience understand that they exist and to gather data from other sources before making further conclusions.

Show Session Info

```
sessionInfo()
```

```
## R version 4.3.3 (2024-02-29)
## Platform: aarch64-apple-darwin20 (64-bit)
## Running under: macOS Sonoma 14.2.1
##
```

```
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRlapack.dylib;  LAPACK v
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## time zone: America/New_York
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] lubridate_1.9.3 forcats_1.0.0   stringr_1.5.1   dplyr_1.1.4
##  [5] purrr_1.0.2     readr_2.1.5     tidyr_1.3.1     tibble_3.2.1
##  [9] ggplot2_3.5.0   tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
##  [1] bit_4.0.5         gtable_0.3.4      highr_0.10        crayon_1.5.2
##  [5] compiler_4.3.3    tidyselect_1.2.1  parallel_4.3.3    scales_1.3.0
##  [9] yaml_2.3.8        fastmap_1.1.1     R6_2.5.1          labeling_0.4.3
## [13] generics_0.1.3    curl_5.2.1        knitr_1.45        munsell_0.5.0
## [17] pillar_1.9.0      tzdb_0.4.0        rlang_1.1.3       utf8_1.2.4
## [21] stringi_1.8.3     xfun_0.43         bit64_4.0.5       timechange_0.3.0
## [25] cli_3.6.2         withr_3.0.0       magrittr_2.0.3    digest_0.6.35
## [29] grid_4.3.3        vroom_1.6.5       rstudioapi_0.16.0 hms_1.1.3
## [33] lifecycle_1.0.4   vctrs_0.6.5       evaluate_0.23     glue_1.7.0
## [37] farver_2.1.1      fansi_1.0.6       colorspace_2.1-0  rmarkdown_2.26
## [41] tools_4.3.3       pkgconfig_2.0.3   htmltools_0.5.8
```