NYPD Assignment

Cameron S.

2025-07-28

Libraries

```r
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   3.5.2     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.1.0
## -- 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(lubridate)
```

NYPD Shooting Incident Report Data

```r
##################
# NYPD data load

nypd_data <- read_csv("NYPD_Shooting_Incident_Data__Historic_.csv")
```

```
## Rows: 29744 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (12): OCCUR_DATE, BORO, LOC_OF_OCCUR_DESC, LOC_CLASSFCTN_DESC, LOCATION...
## dbl   (5): INCIDENT_KEY, PRECINCT, JURISDICTION_CODE, Latitude, Longitude
## num   (2): 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.
```

```r
#nypd_data <- read_csv("Masters Coursework/Data Science as a Field/NYPD_Shooting_Incident_Data__Histori
```

Data Cleaning and Manipulation

```r
# Drop columns that we don't need
# Initial Thoughts: can compare perp to vic data, can visualize frequencies of perp and vic ages, can v
#                   can analyze BORO/PRECINCT statistics, can analyze date statistics, and look more in
nypd_data <- nypd_data %>% select(OCCUR_DATE,BORO,LOC_OF_OCCUR_DESC,PRECINCT,PERP_AGE_GROUP,PERP_SEX,PE
                           mutate(OCCUR_DATE = mdy(OCCUR_DATE))

# Rename column names
nypd_data <- nypd_data %>% rename(date = "OCCUR_DATE", boro = "BORO", location_desc = "LOC_OF_OCCUR_DES
                                  perp_sex = "PERP_SEX", perp_race = "PERP_RACE", vic_age_group = "VIC_

# Get summary statistics
summary(nypd_data)
```

```
##       date                boro           location_desc         precinct
##  Min.   :2006-01-01   Length:29744       Length:29744       Min.   :  1.00
##  1st Qu.:2009-10-29   Class :character   Class :character   1st Qu.: 44.00
##  Median :2014-03-25   Mode  :character   Mode  :character   Median : 67.00
##  Mean   :2014-10-31                                         Mean   : 65.23
##  3rd Qu.:2020-06-29                                         3rd Qu.: 81.00
##  Max.   :2024-12-31                                         Max.   :123.00
##
##  perp_age_group       perp_sex          perp_race         vic_age_group
##  Length:29744       Length:29744       Length:29744       Length:29744
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    vic_sex            vic_race            Latitude       Longitude
##  Length:29744       Length:29744       Min.   :40.51   Min.   :-74.25
##  Class :character   Class :character   1st Qu.:40.67   1st Qu.:-73.94
##  Mode  :character   Mode  :character   Median :40.70   Median :-73.91
##                                        Mean   :40.74   Mean   :-73.91
##                                        3rd Qu.:40.83   3rd Qu.:-73.88
##                                        Max.   :40.91   Max.   :-73.70
##                                        NA's   :97      NA's   :97
```

```r
# Split data

perp_data <- nypd_data %>% select(perp_age_group, perp_sex, perp_race)

vic_data <- nypd_data %>% select(vic_age_group, vic_sex, vic_race)

pv_data <- nypd_data %>% select(perp_age_group, perp_sex, perp_race, vic_age_group, vic_sex, vic_race)

location_data <- nypd_data %>% select(boro, precinct, Latitude, Longitude)
```

The years go from 2006 to 2024, so there a lot of years worth of data. The years also came in characters, so I had to mutate them to be a date object. There are many null values in the columns location desc, perp_age_group, perp_sex, perp_race, Latitude, and Longitude. Before removing these, it is worth noting how many null values there are.

```
# Number of null in location_desc: 25596 out of 29744
sum(is.na(nypd_data$location_desc))
```

```
## [1] 25596
```

```
# Number of null in perp_age_group: 9344 out of 29744
sum(is.na(nypd_data$perp_age_group))
```

```
## [1] 9344
```

```
# Number of null in perp_sex: 9310 out of 29744
sum(is.na(nypd_data$perp_sex))
```

```
## [1] 9310
```

```
# Number of null in perp_race: 9310 out of 29744
sum(is.na(nypd_data$perp_race))
```

```
## [1] 9310
```

```
# Number of null in Latitude: 97 out of 29744
sum(is.na(nypd_data$Latitude))
```

```
## [1] 97
```

```
# Number of null in Longitude: 97 out of 29744
sum(is.na(nypd_data$Longitude))
```

```
## [1] 97
```

The majority of the shooting incidents appear to lack a location description. This is interesting because the Latitude and Longitudes are only missing 97 values. This discrepancy could be because of 911 calls knowing the general area (Latitude and Longitude), but not whether the shooting occured inside or outside (location_desc)

With certain analysis, I will exclude the location_desc column due to it's lack of data.

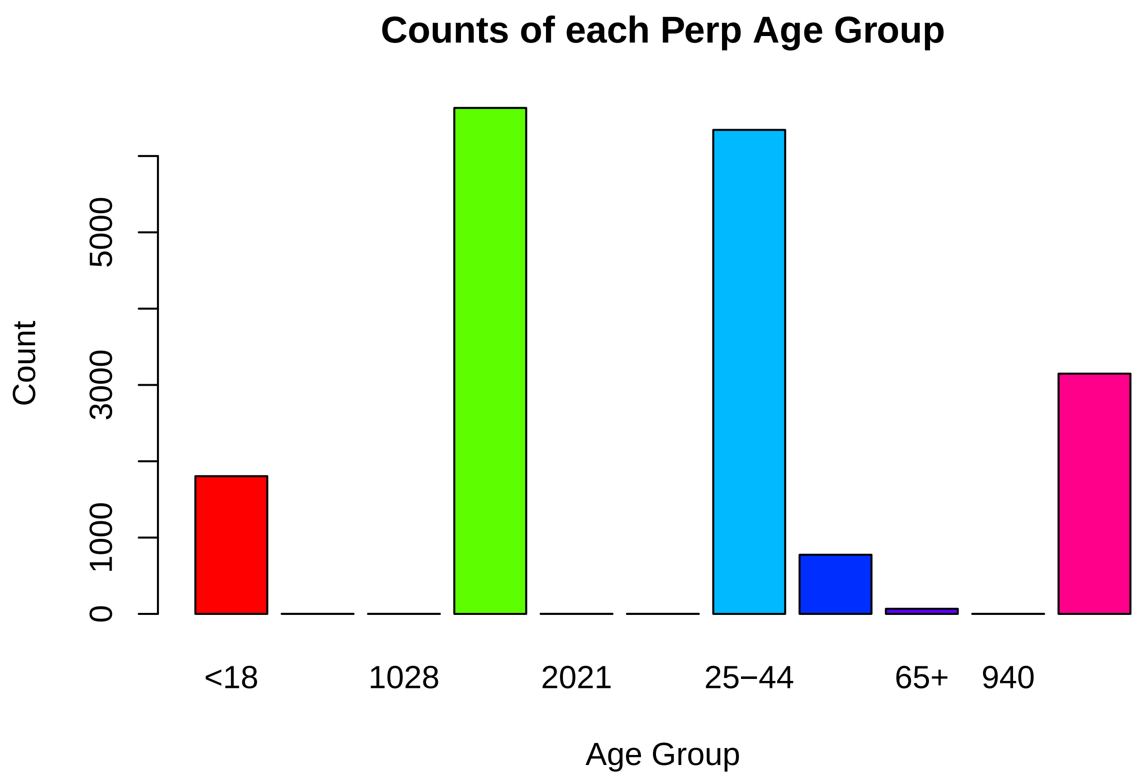
Perp and Vic analysis

Next, I am curious about the age groups of both the perp and the vic. I will create bar plots of both of them to get the frequencies of each age group in both columns

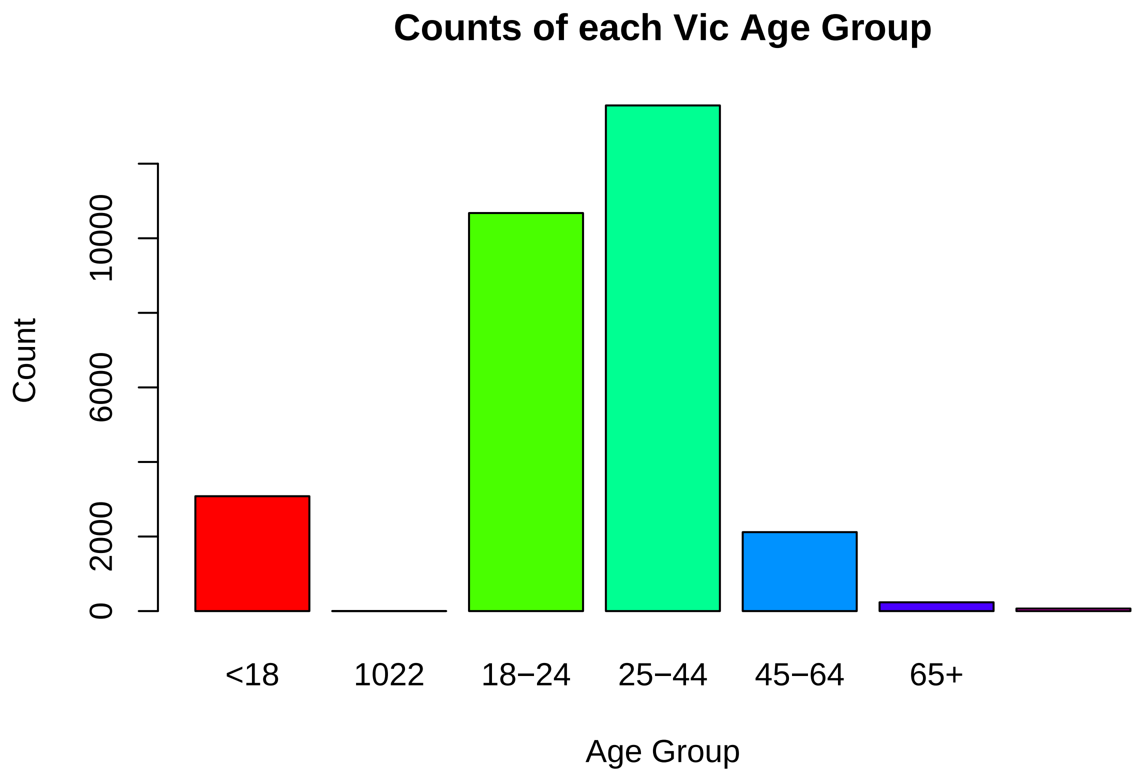
```r
# First drop na values from perp data set.
# There are also strings of (null) that appear 1628 times that should be removed.
perp_data <- perp_data %>% drop_na()
perp_data <- filter(perp_data, perp_age_group != "(null)")

# Drop na values and (null) strings from vic data set
vic_data <- vic_data %>% drop_na()
vic_data <- filter(vic_data, vic_age_group != "(null)")


# Visualize the bar plot of the age groups for PERPs
barplot(table(perp_data$perp_age_group), xlab = "Age Group", ylab = "Count", main="Counts of each Perp A
```




```r
# Visualize the bar plot of the age groups for VICs
barplot(table(vic_data$vic_age_group), xlab = "Age Group", ylab = "Count", main="Counts of each Vic Age
```



These bar plots tell us that the perps are mostly in the 18-24 and 25-44 age groups. This range does appear to cover a lot of ages though. However, something else to note is that there are roughly 7000 in the 18-24 age group and roughly 6500 in the 25-44 age group. There is a large portion in the UNKNOWN category and a lot of NA and (null) values were removed.

What we can say is that with the known data, there are more perps that are younger than older. The missing data tells us that a lot of the time, the perps get away with the killings.

The vic graph tells a different story. THe vic graph says that almost 13000-14000 of the vics are in the 25-44 age group and 10000-11000 are in the 18-24 age group. There are far fewer unknown and missing values.

What we can draw from this is that more of the vics are older, and since they are victims of a shooting, they do not necessarily run away. This is how the police have a report on the victims ages, sex, and race.

Next I want to see the counts of each sex and the counts of each race in each dataset.

```
# Counts of M/F in perp_sex
table(perp_data$perp_sex)
```

```
##
##     F     M     U
##   461 16845  1466
```

```
# Counts of M/F in vic_sex
table(vic_data$vic_sex)
```

```
##
```

```
##     F     M     U
##  2891 26841    12
```

```r
# Counts of each race in perp_sex
table(perp_data$perp_race)
```

```
##
## AMERICAN INDIAN/ALASKAN NATIVE       ASIAN / PACIFIC ISLANDER
##                              2                            184
##                          BLACK                 BLACK HISPANIC
##                          12323                           1487
##                        UNKNOWN                          WHITE
##                           1804                            305
##                 WHITE HISPANIC
##                           2667
```

```r
# Counts of each race in vic_sex
table(vic_data$vic_race)
```

```
##
## AMERICAN INDIAN/ALASKAN NATIVE       ASIAN / PACIFIC ISLANDER
##                             13                            478
##                          BLACK                 BLACK HISPANIC
##                          20999                           2930
##                        UNKNOWN                          WHITE
##                             72                            741
##                 WHITE HISPANIC
##                           4511
```

What we can see from these counts is the majority of both vics and perps are male. There are more unknown in the perps due to them potentially getting away with the shooting.

The race is also interesting to see because the pattern is also essentially the same for both vic and perp data. The largest values for perps from largest to smallest is BLACK, WHITE HISPANIC, UNKNOWN, BLACK HISPANIC, WHITE, ASIAN / PACIFIC ISLANDER, and AMERICAN INDIAN/ALASKAN NATIVE. The largest values for vics from largest to smallest is BLACK, WHITE HISPANIC, BLACK HISPANIC, WHITE, ASIAN / PACIFIC ISLANDER, UNKNOWN, AMERICAN INDIAN/ALASKAN NATIVE.

The values after the top 4 are much smaller than the rest. However, the top 3 for both categories are the same. This information could be very useful for the police, but any analysis using this information would only be speculation at best.

This wraps up my analysis of the perp and vic data.

Next I will look at the location data.

Location Analysis

Lets take a look at the counts of each variable in the boro and precinct columns. Then we will look at some more data from there.

```r
# Boro value counts
table(location_data$boro)
```

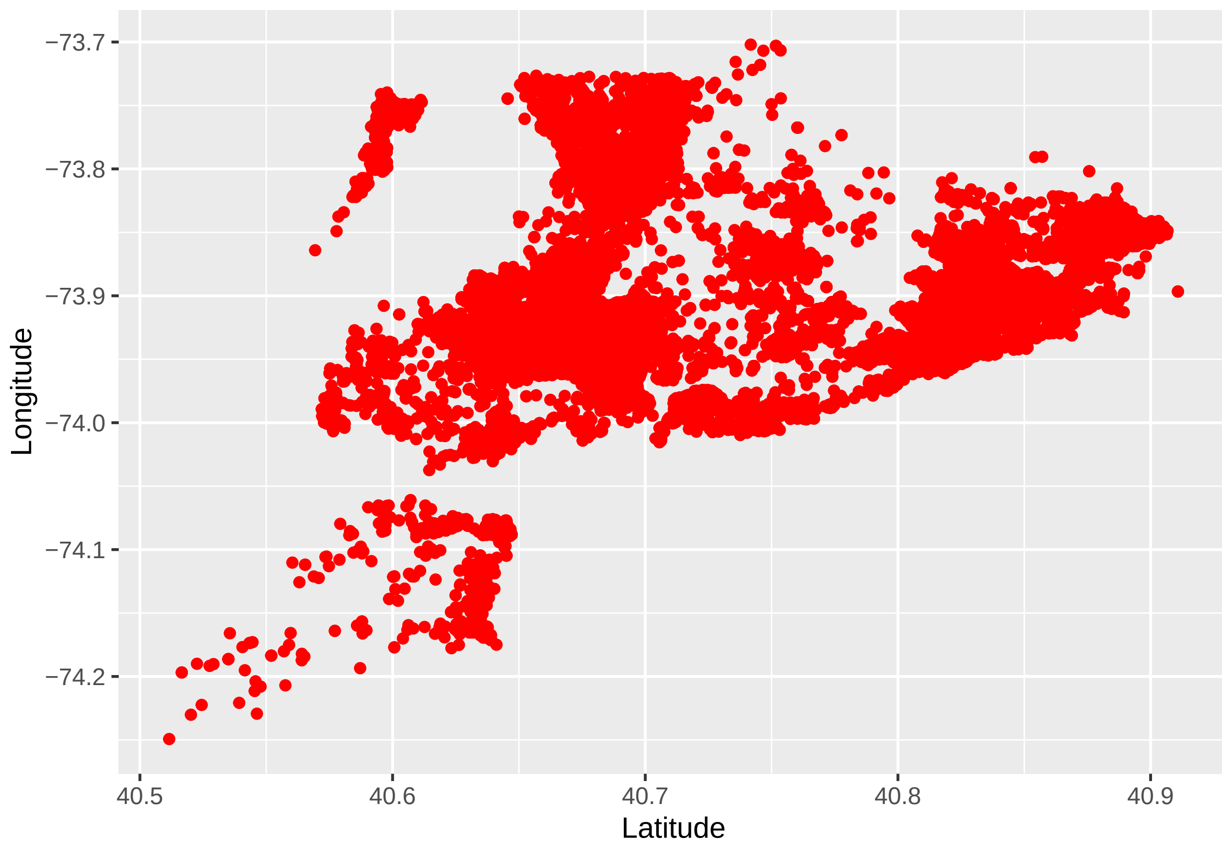
```
##
##         BRONX      BROOKLYN     MANHATTAN        QUEENS STATEN ISLAND
##          8834         11685          3977          4426           822
```

```r
# Precincy value counts
table(location_data$precinct)
```

```
##
##    1    5    6    7    9   10   13   14   17   18   19   20   22   23   24   25
##   29   74   29  127  128   76   64   69   10   48   27   50    1  525  117  515
##   26   28   30   32   33   34   40   41   42   43   44   45   46   47   48   49
##  164  372  245  686  258  363 1002  537  936  831 1159  199 1044 1048  879  385
##   50   52   60   61   62   63   66   67   68   69   70   71   72   73   75   76
##  169  645  389  165   73  295   53 1288   36  503  491  609  120 1561 1680  184
##   77   78   79   81   83   84   88   90   94  100  101  102  103  104  105  106
##  856   72 1073  839  528  133  308  339   90  184  520  249  633  111  508  237
##  107  108  109  110  111  112  113  114  115  120  121  122  123
##  110   81  131  176   13   23  853  406  191  608  117   64   33
```

```r
# Clean na values
location_data <- location_data %>% drop_na()

# Visualize longitude and latitude in scatter plot
location_data %>% ggplot() + geom_point(aes(x = Latitude, y = Longitude), color="red")
```



I also ended up creating a scatter plot of the locations. It looks interesting, but not a lot of information can be gathered from it. It is possible to see some high density areas though, which might be specific boroughs having higher number of shooting incidents.

I decided to google the population values for each of the boroughs in New York City (Source from https://datacommons.org). Brooklyn: 2.6 mil Bronx: 1.4 mil Queens: 2.3 mil Manhattan: 1.6 mil Staten Island 0.4 mil

Interestingly, Queens is third in the list of number of shooting incidents, but is second largest in population. It is much larger in population than Bronx with nearly half the number of shooting incidents.

I also looked at the number of incidents per precinct, but the data is very scattered and hard to follow.

I think what might be a better way to look at the data is by looking at incidents per day. This way I can find out number of incidents per year and potentially create a model to predict incidents. Since the data shows a single victim, I will be considering each entry to be a single incident.

```
# Add 1 incident column
incident_data <- nypd_data %>% select(date)
incident_data$incident_count <- 1

# Get year column
incident_data$year <- year(incident_data$date)
incident_data  <- incident_data %>% select(year, incident_count)

# Combine multiple dates to get total number of incidents per date
incident_data <- data.frame(table(incident_data)) %>% select(year, Freq) %>% rename(incident_count = "F

# Convert year object to int (from 1-19 instead of 2006-2024)
incident_data$year <- as.integer(incident_data$year)

# Summary stats per year
summary(incident_data)
```

```
##       year       incident_count
##  Min.   : 1.0   Min.   : 958
##  1st Qu.: 5.5   1st Qu.:1229
##  Median :10.0   Median :1716
##  Mean   :10.0   Mean   :1565
##  3rd Qu.:14.5   3rd Qu.:1926
##  Max.   :19.0   Max.   :2055
```

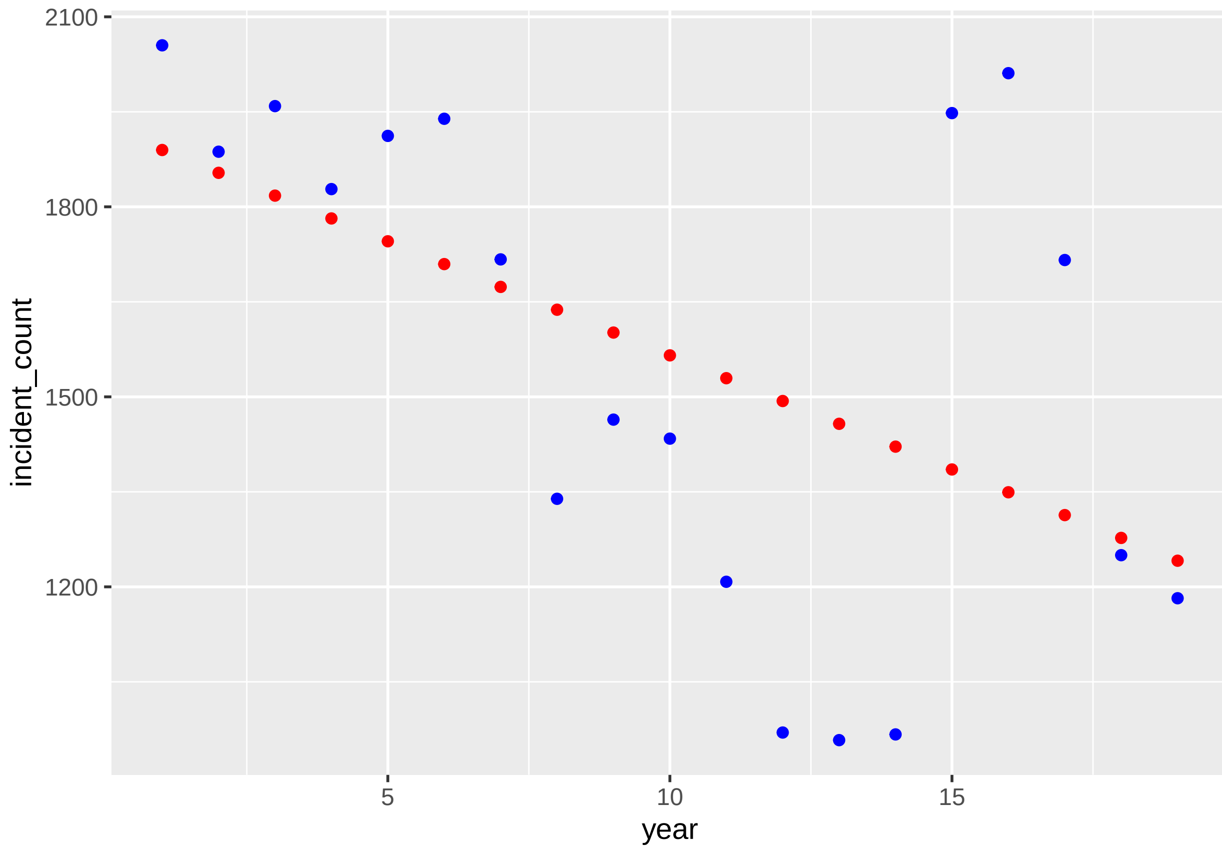
Interestingly, the max number of incidents in a given year is 2055 and the lowest is 958. This tells us that there is potential for some outliers, but not likely due to the quartiles being moderate increments apart from the other stats. This is something to consider when creating the model.

Data Model

```
# Deaths per thousand as represented by cases per thousand
mod <- lm(incident_count ~ year, data = incident_data)

# Predict on US state totals
test_pred <- incident_data %>% mutate(pred = predict(mod))
```

```r
# Plot predicted values compared to actual values
test_pred %>% ggplot() + geom_point(aes(x = year, y = incident_count), color="blue") +
                   geom_point(aes(x = year, y = pred), color = "red")
```



Bias and Conclusions

For my NYPD dataset assignment, I worked to analyse the relationships between locations and incidents, perpetrator and victim relationships, perpetrator and age-group relationships, and victim and age-group relationships. I was able to find that due to the data for perpetrators not being complete, the final analysis of the perpetrators was not likely to be complete either. However, continuing with the removing NA values, the perpetrator data showed a large number of incidents involved males and people between the ages of 18-24 and 25-44. The victims similarly had a large number of instances of males and being between the ages of 18-24 and 25-44. The perpetrator data not containing every data point due to null values shows far fewer counts. This could lead to more analysis of this and what the sex and ages of the perpetrators that got away were. I also looked at the races of the victims and perpetrators and the top 3 groups were the same for both. This could mean many things, but the most important is that they are likely the largest population groups of the area. Any further analysis is likely just speculation.

For the location information, I created a visualization of the latitude and longitude data on a cartesian coordinate system. This didn't provide too much context besides an interesting visual. Something that could be derived from the visual is there are certain areas with a higher density of incidents. This could also be from an increase in population in the areas and not much else. I also looked at the number of incidents per borough and found a couple of interesting things. The first is that the most number of incidents occured in the most populated borough, but the second and third in population are swapped in the incident counts.

The Bronx had nearly twice the number of incidents compared to Queens, but Queens had a population of nearly 1 million more people. This could show that the Bronx is not as safe as other areas in New York, that Queens is much safer as an area, or something else entirely.

For my model, I looked to model incidents as a function of time. I started off by using days, but instead transitioned to years. In this way, I was able to see that there is a general negative linear relationship between the number of incidents in New York city over time. This could be due to many factors, but one is that the NYPD is doing better to make New York City a safer place. There are some standout years in recent years. This is why I changed to years and changed the years to numbers from 1-19 instead of 2006-2024. Due to personal bias of Covid 19, I looked as only years and no other statistics. The recent downward trend with the exception of years 15 and 16 are a consistent pattern. After doing analysis though, going back to the actual years, we can see that years 15 and 16 are 2020 and 2021 respectively. This means that the early years of COVID had a potential large impact on shootings in New York City. Diving further into the reasoning for this would only lead to speculation and present more personal bias which is why I left the actual years out of it.

Appendix

```r
sessionInfo()
```

```
## R version 4.5.1 (2025-06-13 ucrt)
## Platform: x86_64-w64-mingw32/x64
## Running under: Windows 11 x64 (build 26100)
##
## Matrix products: default
##   LAPACK version 3.12.1
##
## locale:
## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## time zone: America/Denver
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] lubridate_1.9.4 forcats_1.0.0   stringr_1.5.1   dplyr_1.1.4
##  [5] purrr_1.1.0     readr_2.1.5     tidyr_1.3.1     tibble_3.3.0
##  [9] ggplot2_3.5.2   tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
##  [1] bit_4.6.0          gtable_0.3.6       crayon_1.5.3       compiler_4.5.1
##  [5] tidyselect_1.2.1   parallel_4.5.1     scales_1.4.0       yaml_2.3.10
##  [9] fastmap_1.2.0      R6_2.6.1           labeling_0.4.3     generics_0.1.4
## [13] knitr_1.50         pillar_1.11.0      RColorBrewer_1.1-3 tzdb_0.5.0
## [17] rlang_1.1.6        stringi_1.8.7      xfun_0.52          bit64_4.6.0-1
## [21] timechange_0.3.0   cli_3.6.5          withr_3.0.2        magrittr_2.0.3
```

```
## [25] digest_0.6.37      grid_4.5.1         vroom_1.6.5        rstudioapi_0.17.1
## [29] hms_1.1.3          lifecycle_1.0.4    vctrs_0.6.5        evaluate_1.0.4
## [33] glue_1.8.0         farver_2.1.2       rmarkdown_2.29     tools_4.5.1
## [37] pkgconfig_2.0.3    htmltools_0.5.8.1
```