NYPD Shooting Incident Data Report

Filsan Musa

Contents

1 Required Packages & Libraries **1**

2 Introduction **2**

2.1 Data Description 2

3 Methodology **3**

3.1 Import Dataset 3

3.2 Data Exploration 3

3.3 Data Cleaning 8

4 Model Evaluaion **15**

4.1 Spatial Analysis: Shooting Incidents 15

4.2 Time Series: Shooting Incidents Per Capita (by Borough) 2023-2024 17

5 Limitations & Challenges **24**

5.1 Bias 24

5.2 Other Factors 24

6 Conclusion **24**

7 References **25**

1 Required Packages & Libraries

```
#install.packages("tidyverse")
#install.packages("ggthemes")
#install.packages("dbscan")
#install.packages("scales")
library("tidyverse")
library("ggthemes")
library("dbscan")
library("scales")
```

Note: Uncomment by removing "#" to use.

2 Introduction

In this report, we explore the NYPD Shooting Incident dataset to uncover patterns and trend of these incidents across different boroughs. The analysis begins with a methodology section, in which we conduct a preliminary visual exploration of the dataset and perform essential data cleaning steps. We then outline our planned modeling approach, discussing the types of models to be used and the specific questions we aim to address in the model evaluation section. It is in this section, where we'll build and evaluate two models: the first model is a spatial clustering using DBSCAN, which examines the geographic dispersion of shooting incidents; in the second model, we perform a time series analysis that investigates borough-level trends in reported shootings (per capita of 100K residents) from 2023-2024. We then disclose potential biases, challenges and limitation faced throughout this project. Finally, we conclude with a brief discussion that reflects on the overall analytical process, from data pre-processing phase through model building and evaluation, to highlight key takeaways.

2.1 Data Description

The dataset is a compilation of all reported shooting incidents that occurred in New York City from 2006 through the end of the year prior, 2024. Each row contains information pertaining to a unique shooting incident and includes details such as location coordinates and borough of the incident, the time and date, as well as demographic information about the perpetrator and victim, including age, sex, and race. The data is manually extracted on a quarterly basis and is reviewed by the Office of Management Analysis and Planning prior to being posted on the NYPD website.

Data Source: https://catalog.data.gov/dataset/nypd-shooting-incident-data-historic

2.1.1 Data Dictionary

Variable	Description
INCIDENT_KEY	Synthetic generated key for each unique incident
OCCUR_DATE	Date of the shooting incident
OCCUR_TIME	Time of the incident
BORO	Borough in which the incident took place
LOC_OF_OCCUR_DESC	Whether or not the incident resulted in victim's murder
PRECINCT	Precinct the shooting was reported to
LOC_CLASSFCTN_DESC	Location classification description
LOCATION_DESC	Description of the location
STATISTICAL_MURDER_FLAG	Indicates whether or not the incident was classified as a murder
PERP_AGE_GROUP	Perpetrator's age group
PERP_SEX	Perpetrator's sex
PERP_RACE	Perpetrator's race
VIC_AGE_GROUP	Victim's age group
VIC_SEX	Victim's sex
VIC_RACE	Victim's race
X_COORD_CD	X coordinate (NYC spatial coordinate system)
Y_COORD_CD	Y coordinate (NYC spatial coordinate system)
Latitude	Location latitude
Longitude	Location longitude
Lon_Lat	Combined longitude and latitude coordinates
JURISDICTION_CODE	Jurisdiction responsible for arrest (0: NYPD Patrol, 1: NYPD Transit, and 2: NYPD Housing)

Table 1: Data dictionary for NYPD Shooting Incident Dataset (Historic)

3 Methodology

In this segment of the report, the main focus is to first conduct an exploration of the data by examining its properties, generating summaries, and performing a brief preliminary analysis of the dataset. The goal is then to clean the data by dropping irrelevant columns, adjusting data types, removing duplicates, identifying and addressing missing values, feature engineering and renaming the columns.

3.1 Import Dataset

```
#Defining url for the file we want to access
url_in <- "https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD"
#Importing and naming the datasets
data <- read.csv(url_in[1])
```

3.2 Data Exploration

First, let's look at the dimensions, and size of our dataset...

```
#Shows the # of row and columns in your dataset
dim(data)
```

```
[1] 29744    21
```

```
#Shows the # of entries in your dataset
prod(dim(data))
```

```
[1] 624624
```

Next, let's take a take a glance at our dataset to learn a bit more

```
#Shows first two rows of the dataset
head(data, 2)
```

```
  INCIDENT_KEY OCCUR_DATE OCCUR_TIME     BORO LOC_OF_OCCUR_DESC PRECINCT
1    231974218 08/09/2021   01:06:00    BRONX                        40
2    177934247 04/07/2018   19:48:00 BROOKLYN                        79
  JURISDICTION_CODE LOC_CLASSFCTN_DESC LOCATION_DESC STATISTICAL_MURDER_FLAG
1                 0                                                   false
2                 0                                                    true
  PERP_AGE_GROUP PERP_SEX      PERP_RACE VIC_AGE_GROUP VIC_SEX VIC_RACE
1                                          18-24       M    BLACK
2          25-44        M WHITE HISPANIC         25-44       M    BLACK
              X_COORD_CD             Y_COORD_CD Latitude Longitude
1                1006343                 234270 40.80967 -73.92019
2 1000082.937500000000000 189064.671875000000000 40.68561 -73.94291
                                    Lon_Lat
1  POINT (-73.92019278899994 40.80967347200004)
2 POINT (-73.94291302299996 40.685609672000055)
```

```
#Shows final two rows of the dataset
tail(data, 2)
```

```
      INCIDENT_KEY OCCUR_DATE OCCUR_TIME      BORO LOC_OF_OCCUR_DESC PRECINCT
29743    291163266 08/04/2024   15:41:00 MANHATTAN           OUTSIDE       23
29744    282706131 02/23/2024   20:06:00     BRONX           OUTSIDE       41
      JURISDICTION_CODE LOC_CLASSFCTN_DESC             LOCATION_DESC
29743                 0            HOUSING MULTI DWELL - PUBLIC HOUS
29744                 0             STREET                    (null)
      STATISTICAL_MURDER_FLAG PERP_AGE_GROUP PERP_SEX PERP_RACE VIC_AGE_GROUP
29743                   false          18-24        M     BLACK         18-24
29744                   false          25-44        M     BLACK         25-44
      VIC_SEX       VIC_RACE X_COORD_CD Y_COORD_CD Latitude Longitude
29743       M WHITE HISPANIC    998,480    225,704 40.78617 -73.94861
29744       M          BLACK  1,015,160    237,028 40.81721 -73.88833
                         Lon_Lat
29743 POINT (-73.948611 40.786171)
29744 POINT (-73.888326 40.817209)
```

Finally, let's perform a brief preliminary analysis before cleaning the data.

```
#Shows information about the # of rows and columns, and lists the columns along
#with their data types and contents
glimpse(data)
```

```
Rows: 29,744
Columns: 21
$ INCIDENT_KEY            <int> 231974218, 177934247, 255028563, 25384540, 726~
$ OCCUR_DATE              <chr> "08/09/2021", "04/07/2018", "12/02/2022", "11/~
$ OCCUR_TIME              <chr> "01:06:00", "19:48:00", "22:57:00", "01:50:00"~
$ BORO                    <chr> "BRONX", "BROOKLYN", "BRONX", "BROOKLYN", "BRO~
$ LOC_OF_OCCUR_DESC       <chr> "", "", "OUTSIDE", "", "", "", "", "", "", "",~
$ PRECINCT                <int> 40, 79, 47, 66, 46, 42, 71, 69, 75, 69, 40, 42~
$ JURISDICTION_CODE       <int> 0, 0, 0, 0, 0, 2, 0, 2, 0, 0, 0, 2, 0, 0, 2, 0~
$ LOC_CLASSFCTN_DESC      <chr> "", "", "STREET", "", "", "", "", "", "", "", ~
$ LOCATION_DESC           <chr> "", "", "GROCERY/BODEGA", "PVT HOUSE", "MULTI ~
$ STATISTICAL_MURDER_FLAG <chr> "false", "true", "false", "true", "true", "fal~
$ PERP_AGE_GROUP          <chr> "", "25-44", "(null)", "UNKNOWN", "25-44", "18~
$ PERP_SEX                <chr> "", "M", "(null)", "U", "M", "M", "", "", "M",~
$ PERP_RACE               <chr> "", "WHITE HISPANIC", "(null)", "UNKNOWN", "BL~
$ VIC_AGE_GROUP           <chr> "18-24", "25-44", "25-44", "18-24", "<18", "18~
$ VIC_SEX                 <chr> "M", "M", "M", "M", "F", "M", "M", "M", "M", "~
$ VIC_RACE                <chr> "BLACK", "BLACK", "BLACK", "BLACK", "BLACK", "~
$ X_COORD_CD              <chr> "1006343", "1000082.937500000000000", "1020691~
$ Y_COORD_CD              <chr> "234270", "189064.671875000000000", "257125", ~
$ Latitude                <dbl> 40.80967, 40.68561, 40.87235, 40.64249, 40.845~
$ Longitude               <dbl> -73.92019, -73.94291, -73.86823, -73.99691, -7~
$ Lon_Lat                 <chr> "POINT (-73.92019278899994 40.80967347200004)"~
```

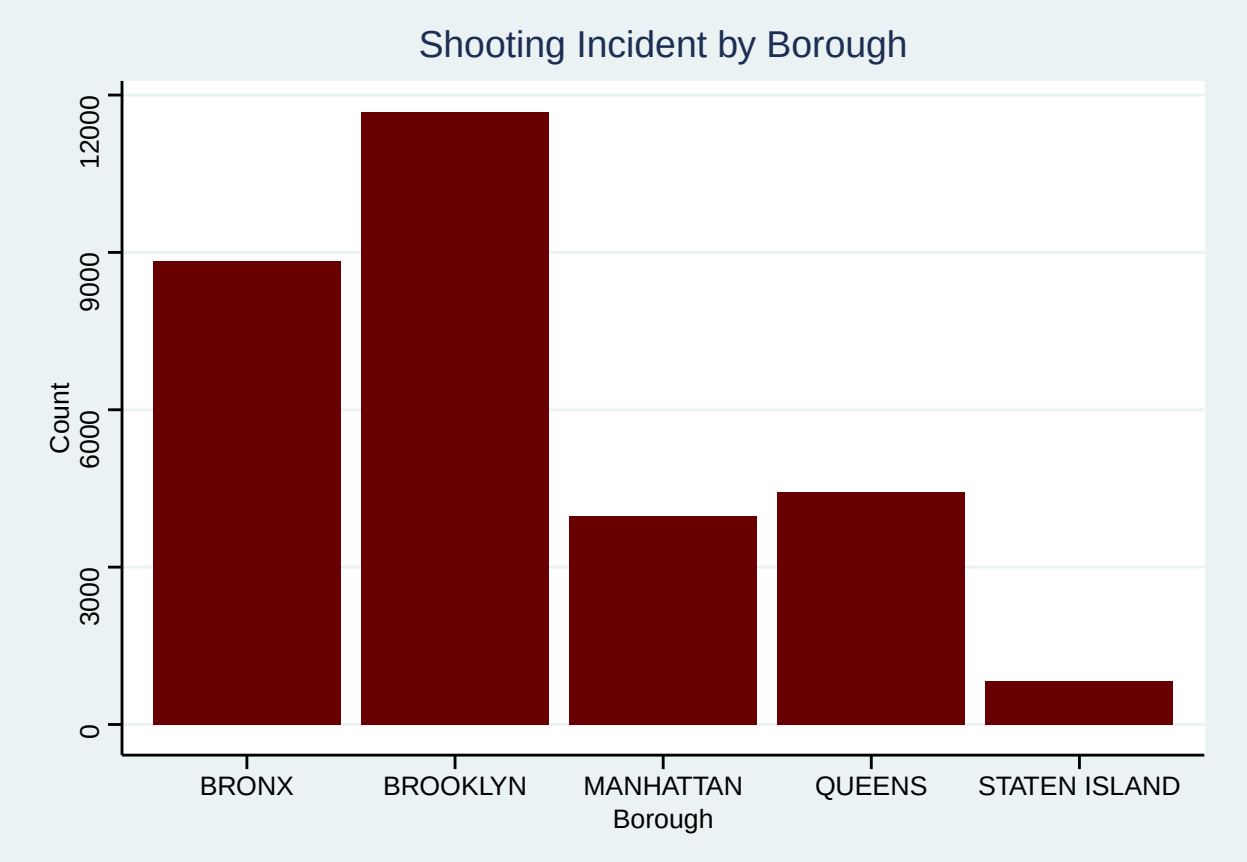
```
#Shows summary statistics for each column in the dataset
summary(data)
```

```
 INCIDENT_KEY        OCCUR_DATA          OCCUR_TIME            BORO
Min.   :  9953245   Length:29744       Length:29744       Length:29744
1st Qu.: 67321140   Class :character   Class :character   Class :character
Median :109291972   Mode  :character   Mode  :character   Mode  :character
Mean   :133850951
3rd Qu.:214741917
Max.   :299462478

LOC_OF_OCCUR_DESC     PRECINCT      JURISDICTION_CODE LOC_CLASSFCTN_DESC
Length:29744       Min.   :  1.00   Min.   :0.0000    Length:29744
Class :character   1st Qu.: 44.00   1st Qu.:0.0000    Class :character
Mode  :character   Median : 67.00   Median :0.0000    Mode  :character
                   Mean   : 65.23   Mean   :0.3181
                   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:29744       Length:29744            Length:29744
Class :character   Class :character        Class :character
Mode  :character   Mode  :character        Mode  :character



  PERP_SEX           PERP_RACE          VIC_AGE_GROUP        VIC_SEX
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_RACE           X_COORD_CD          Y_COORD_CD           Latitude
Length:29744       Length:29744       Length:29744       Min.   :40.51
Class :character   Class :character   Class :character   1st Qu.:40.67
Mode  :character   Mode  :character   Mode  :character   Median :40.70
                                                         Mean   :40.74
                                                         3rd Qu.:40.83
                                                         Max.   :40.91
                                                         NA's   :97
  Longitude         Lon_Lat
Min.   :-74.25   Length:29744
1st Qu.:-73.94   Class :character
Median :-73.91   Mode  :character
Mean   :-73.91
3rd Qu.:-73.88
Max.   :-73.70
NA's   :97
```

- Shooting Incidents Per Borough

```
#Shows a bar plot of the incident counts per borough
ggplot(data, aes(x = BORO)) +
  geom_bar(position = "dodge", fill = "#680001") +
```

```
  labs(
       title = "Shooting Incident by Borough",
       x = "Borough",
       y = "Count",
       fill = "Murder Flag") + theme_stata() +
  theme(plot.title = element_text(hjust = 0.5))
```



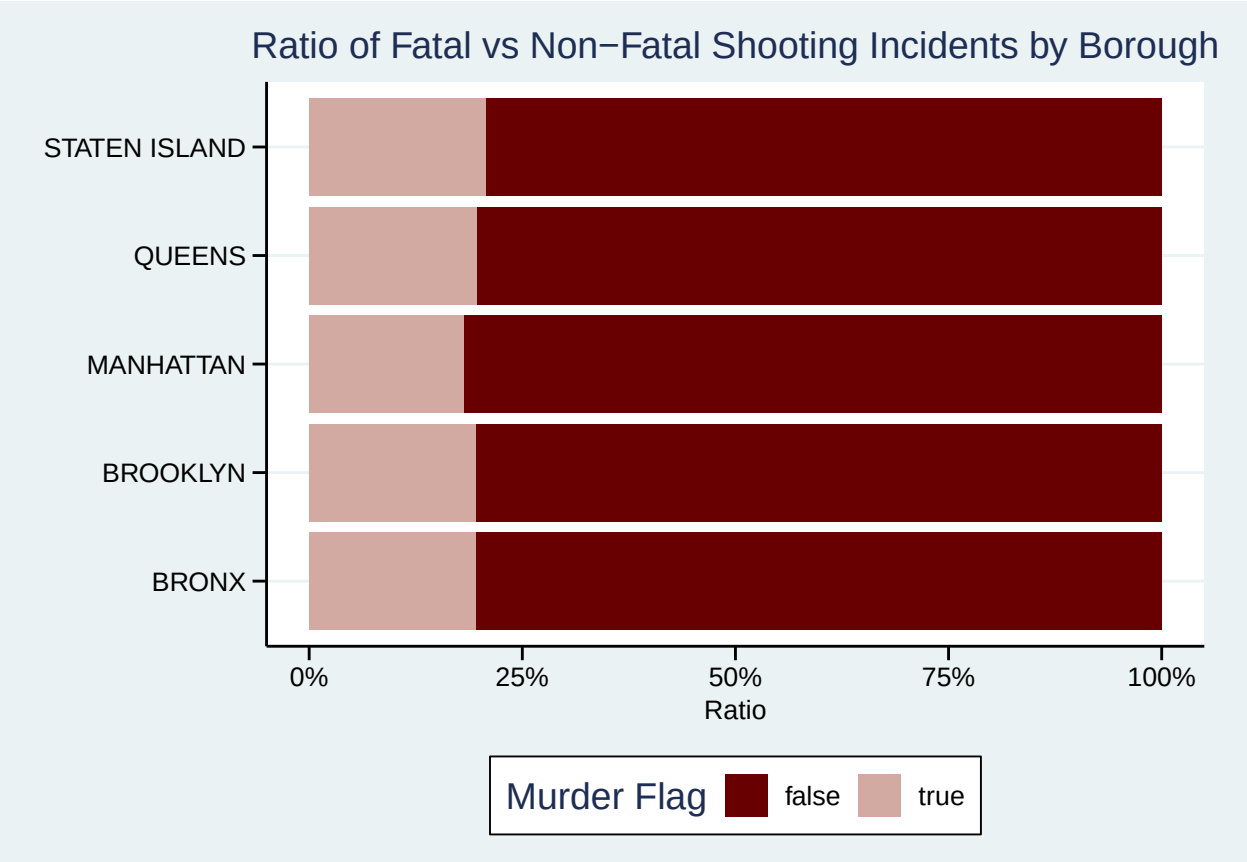

```
#Shows the actual number of incidents per borough
data %>%
  count(BORO, name = "COUNT_INCIDENTS")
```

```
##            BORO COUNT_INCIDENTS
## 1         BRONX            8834
## 2      BROOKLYN           11685
## 3     MANHATTAN            3977
## 4        QUEENS            4426
## 5 STATEN ISLAND             822
```

Note: Both the graph and table above illustrate why data cannot simply be interpreted in isolation. For instance, one could preemptively conclude that because it could appear that Brooklyn has the highest number of incidents is must have a higher crime rate. Though this conclusion might indeed be true, as of 2024 Brooklyn is the most populous of the five boroughs. In fact, here are the recorded populations: Brooklyn (2,617,631), Queens (2,316,841), Manhattan (1,660,664), The Bronx (1,384,724), and Staten Island (498,212). Perhaps looking at the per capita shooting incidents may provide a more accurate representation of crime distribution across the boroughs, yielding a better understanding of shooting related crime in New York City.

- Fatal vs. Non-Fatal Shooting Incidents Per Borough

```
#Shows a bar plot of the incidents grouped on murder flags per borough
ggplot(data, aes(x = BORO, fill = STATISTICAL_MURDER_FLAG)) +
  geom_bar(position = "fill") +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("false" = "#680001", "true" = "#d3aaa2")) +
  labs(
       title = "Ratio of Fatal vs Non-Fatal Shooting Incidents by Borough",
       x = NULL,
       y = "Ratio",
       fill = "Murder Flag") +
  coord_flip() + theme_stata() +
  theme(
    plot.title = element_text(hjust = 0.5),
    axis.text.y = element_text(angle = 0))
```



Note: The above graph is a visual representation of the ratio of fatal to non-fatal reported shooting incidents in each borough. (See following note for additional details)

```
#Creates a alt dataframe with numeric version of STATISTICAL_MURDER_FLAG
temp_data <- data %>%
  mutate(MURDER_FLAG = ifelse(tolower(STATISTICAL_MURDER_FLAG) == "true", 1, 0))

#Shows a table of the incident counts, fatality and non-fatality rates per borough
murder_boro <- temp_data %>%
```

```
  group_by(BORO) %>%
  summarise(COUNT_INCIDENTS = n(), COUNT_FATAL = sum(MURDER_FLAG, na.rm = TRUE)) %>%
  mutate(FATAL_RATE_PRCT = round(COUNT_FATAL / COUNT_INCIDENTS, 2) * 100) %>%
  mutate(NONFATAL_RATE_PRCT = round((COUNT_INCIDENTS-COUNT_FATAL)/COUNT_INCIDENTS, 2) *100)
print(murder_boro)
```

```
# A tibble: 5 x 5
  BORO          COUNT_INCIDENTS COUNT_FATAL FATAL_RATE_PRCT NONFATAL_RATE_PRCT
  <chr>                   <int>       <dbl>           <dbl>              <dbl>
1 BRONX                    8834        1728              20                 80
2 BROOKLYN                11685        2277              19                 81
3 MANHATTAN                3977         719              18                 82
4 QUEENS                   4426         871              20                 80
5 STATEN ISLAND             822         170              21                 79
```

Note: Upon examining the percentage of fatal and non-fatal shooting incidents, there seems to be little variation between the boroughs. These numbers could potentially signal that no specific borough experiences a disproportionately higher rate of lethally indented shootings. However, this may not be entirely accurate, since we cannot infer from the given data whether or not the perpetrator had an intent to kill. All that we can infer with the given data, is that the ratio of fatal to non-fatal shooting incidents appears to be relatively consistent across boroughs.

3.3 Data Cleaning

In this segment of the report, we're going to get the data ready for modeling. Specifically, we'll be doing the following:

- Feature Engineering
- Remove irrelevant columns
- Check for duplicate rows
- Check & deal w/ missing values
- Refactor column names

Before we get started with the data cleaning, let's create a copy of the initial dataset, and call it `nyc_shoot`.

```
nyc_shoot <- data
```

Note: We are creating this new table `nyc_shoot`, which will contain the processed data. And we may refer back to the original unaltered version `data`.

- Feature Engineering

```
#Creating new numeric variable based on STATISTICAL_MURDER_FLAG bool categories
nyc_shoot$MURDER_FLAG = ifelse(tolower(nyc_shoot$STATISTICAL_MURDER_FLAG) == "true", 1, 0)
```

```
glimpse(nyc_shoot)
```

```
## Rows: 29,744
## Columns: 22
## $ INCIDENT_KEY            <int> 231974218, 177934247, 255028563, 25384540, 726~
```

```
## $ OCCUR_DATE              <chr> "08/09/2021", "04/07/2018", "12/02/2022", "11/~
## $ OCCUR_TIME              <chr> "01:06:00", "19:48:00", "22:57:00", "01:50:00"~
## $ BORO                    <chr> "BRONX", "BROOKLYN", "BRONX", "BROOKLYN", "BRO~
## $ LOC_OF_OCCUR_DESC       <chr> "", "", "OUTSIDE", "", "", "", "", "", "", "",~
## $ PRECINCT                <int> 40, 79, 47, 66, 46, 42, 71, 69, 75, 69, 40, 42~
## $ JURISDICTION_CODE       <int> 0, 0, 0, 0, 0, 2, 0, 2, 0, 0, 0, 2, 0, 0, 2, 0~
## $ LOC_CLASSFCTN_DESC      <chr> "", "", "STREET", "", "", "", "", "", "", "", ~
## $ LOCATION_DESC           <chr> "", "", "GROCERY/BODEGA", "PVT HOUSE", "MULTI ~
## $ STATISTICAL_MURDER_FLAG <chr> "false", "true", "false", "true", "true", "fal~
## $ PERP_AGE_GROUP          <chr> "", "25-44", "(null)", "UNKNOWN", "25-44", "18~
## $ PERP_SEX                <chr> "", "M", "(null)", "U", "M", "M", "", "", "M",~
## $ PERP_RACE               <chr> "", "WHITE HISPANIC", "(null)", "UNKNOWN", "BL~
## $ VIC_AGE_GROUP           <chr> "18-24", "25-44", "25-44", "18-24", "<18", "18~
## $ VIC_SEX                 <chr> "M", "M", "M", "M", "F", "M", "M", "M", "M", "~
## $ VIC_RACE                <chr> "BLACK", "BLACK", "BLACK", "BLACK", "BLACK", "~
## $ X_COORD_CD              <chr> "1006343", "1000082.937500000000000", "1020691~
## $ Y_COORD_CD              <chr> "234270", "189064.671875000000000", "257125", ~
## $ Latitude                <dbl> 40.80967, 40.68561, 40.87235, 40.64249, 40.845~
## $ Longitude               <dbl> -73.92019, -73.94291, -73.86823, -73.99691, -7~
## $ Lon_Lat                 <chr> "POINT (-73.92019278899994 40.80967347200004)"~
## $ MURDER_FLAG             <dbl> 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1~
```

```
#Let's fix the some of the column data types
nyc_shoot <- nyc_shoot %>% mutate(
    BORO = as.factor(BORO),
    PERP_AGE_GROUP = as.factor(PERP_AGE_GROUP),
    PERP_SEX = as.factor(PERP_SEX),
    PERP_RACE = as.factor(PERP_RACE),
    VIC_AGE_GROUP = as.factor(VIC_AGE_GROUP),
    VIC_SEX = as.factor(VIC_SEX),
    VIC_RACE = as.factor(VIC_RACE),
    MURDER_FLAG = as.integer(MURDER_FLAG))
```

```
#Creating a timestamp variable by merging OCCUR_DATE and OCCUR_TIME
nyc_shoot$OCCUR_DATETIME <- mdy_hms(paste(nyc_shoot$OCCUR_DATE, nyc_shoot$OCCUR_TIME))
```

Note: Created two new variables: `MURDER_FLAG`, `OCCUR_DATETIME`. `MURDER_FLAG` is an encoded integer version of `STATISTICAL_MURDER_FLAG` (which is a character variable). `OCCUR_DATETIME` is a datetime variable combining `OCCUR_DATE` and `OCCUR_TIME` variables, so we can now extract all datetime info from a single column.

- Remove irrelevant columns

```
#OCCUR_DATE, and OCCUR_TIME are not represented by OCCUR_DATETIME
nyc_shoot <- subset(nyc_shoot, select = -c(OCCUR_DATE, OCCUR_TIME))
```

```
#STATISTICAL_MURDER_FLAG has been replaced by MURDER_FLAG
nyc_shoot <- subset(nyc_shoot, select = -c(STATISTICAL_MURDER_FLAG))
```

```
#Removed due to redundancy, we already have Latitude, Longitude, and BORO for location info
nyc_shoot <- subset(nyc_shoot, select = -c(X_COORD_CD, Y_COORD_CD, Lon_Lat))
```

```
#Not relevant, auto-generated identifiers
nyc_shoot <- subset(nyc_shoot, select = -c(INCIDENT_KEY))

#Removed columns irrelevant to planned analysis
nyc_shoot <- subset(nyc_shoot, select = -c(JURISDICTION_CODE))
```

- Check for duplicate rows

```
#Shows the total number of duplicated rows
sum(duplicated(nyc_shoot))
```

```
## [1] 0
```

Note: Since there are no duplicate rows within our dataset, there are no further steps are required.

- Check & deal w/ missing values

First, let's go through the entire dataset to see whether or not we have missing data

```
colSums(is.na(nyc_shoot))
```

```
##               BORO  LOC_OF_OCCUR_DESC           PRECINCT LOC_CLASSFCTN_DESC
##                  0                  0                  0                  0
##      LOCATION_DESC     PERP_AGE_GROUP           PERP_SEX          PERP_RACE
##                  0                  0                  0                  0
##      VIC_AGE_GROUP            VIC_SEX           VIC_RACE           Latitude
##                  0                  0                  0                 97
##          Longitude        MURDER_FLAG     OCCUR_DATETIME
##                 97                  0                  0
```

Let's deal with the missing values identified.

```
#Shows total number of missing values
sum(is.na(nyc_shoot$Latitude))
```

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

```
sum(is.na(nyc_shoot$Longitude))
```

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

```r
#Compute mean values for Latitude and Longitude by BORO
mean_lat_by_boro <- nyc_shoot %>%
  group_by(BORO) %>%
  summarise(mean_lat = mean(Latitude, na.rm = TRUE)) #Finds mean Latitude by BORO

mean_lon_by_boro <- nyc_shoot %>%
  group_by(BORO) %>%
  summarise(mean_lon = mean(Longitude, na.rm = TRUE)) #Finds mean Longitude by BORO

#Impute missing value using calculated means
nyc_shoot <- nyc_shoot %>%
  left_join(mean_lat_by_boro, by = "BORO") %>%
  mutate(Latitude = ifelse(is.na(Latitude), mean_lat, Latitude)) %>%
  select(-mean_lat) #Imputes missing Latitude values with means from prev. computations

nyc_shoot <- nyc_shoot %>%
  left_join(mean_lon_by_boro, by = "BORO") %>%
  mutate(Longitude = ifelse(is.na(Longitude), mean_lon, Longitude)) %>%
  select(-mean_lon) #Imputes missing Longitude values with means from prev. computations

#Verifying no missing values remain
sum(is.na(nyc_shoot$Latitude))
```

```
## [1] 0
```

```r
sum(is.na(nyc_shoot$Longitude))
```

```
## [1] 0
```

Next, let's look into some of the other variables (not identified as having missing values)

```r
#Looks through distinct values within the location related categorical variables
table(nyc_shoot$LOC_OF_OCCUR_DESC) #has 25596 missing values
```

```
##
##          INSIDE OUTSIDE
##   25596     682    3466
```

```r
table(nyc_shoot$LOC_CLASSFCTN_DESC) #has 25596 missing values
```

```
##
##                   (null)  COMMERCIAL    DWELLING     HOUSING       OTHER
##       25596            7         276         341         643          74
## PARKING LOT  PLAYGROUND      STREET     TRANSIT     VEHICLE
##          16          67        2639          52          33
```

```r
table(nyc_shoot$LOCATION_DESC) #has 14977 missing values
```

```
##
##                                      (null)                         ATM
##                       14977                2526                           1
##                        BANK         BAR/NIGHT CLUB          BEAUTY/NAIL SALON
##                           3                 695                         120
##                CANDY STORE          CHAIN STORE                 CHECK CASH
##                          10                    9                           1
##           CLOTHING BOUTIQUE       COMMERCIAL BLDG                 DEPT STORE
##                          14                  306                           9
##              DOCTOR/DENTIST           DRUG STORE         DRY CLEANER/LAUNDRY
##                           1                   14                          32
##           FACTORY/WAREHOUSE           FAST FOOD                 GAS STATION
##                           8                  131                          76
##              GROCERY/BODEGA  GYM/FITNESS FACILITY                    HOSPITAL
##                         775                    4                          84
##                HOTEL/MOTEL         JEWELRY STORE                LIQUOR STORE
##                          38                   14                          42
##                LOAN COMPANY MULTI DWELL - APT BUILD MULTI DWELL - PUBLIC HOUS
##                           1                 3042                        5188
##                        NONE      PHOTO/COPY STORE                   PVT HOUSE
##                         175                    2                        1010
##            RESTAURANT/DINER               SCHOOL                 SHOE STORE
##                         216                    1                          10
##              SMALL MERCHANT SOCIAL CLUB/POLICY LOCATI           STORAGE FACILITY
##                          46                   74                           1
##          STORE UNCLASSIFIED         SUPERMARKET            TELECOMM. STORE
##                          37                   21                          11
##                VARIETY STORE         VIDEO STORE
##                          11                    8
```

```
#Shows the total missing values in each column
sum(nyc_shoot$LOC_OF_OCCUR_DESC %in% c("(null)", ""), na.rm = TRUE)
```

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

```
sum(nyc_shoot$LOC_CLASSFCTN_DESC %in% c("(null)", ""), na.rm = TRUE)
```

```
## [1] 25603
```

```
sum(nyc_shoot$LOCATION_DESC %in% c("(null)", ""), na.rm = TRUE)
```

```
## [1] 17503
```

```
#Dropping columns w/ 50%+ missing values
nyc_shoot <- subset(nyc_shoot, select = -c(LOC_OF_OCCUR_DESC, LOC_CLASSFCTN_DESC, LOCATION_DESC))
```

Note: These columns have $>$ 50% missing values and will not be useful for our analysis. These will be dropped.

```
#Looks through distinct values within the demographic related categorical variables
table(nyc_shoot$PERP_SEX) #has 10938 missing values
```

```
##
##        (null)      F      M      U
##   9310   1628    461  16845   1500
```

```
table(nyc_shoot$PERP_AGE_GROUP) #has 10972 missing values
```

```
##
##          (null)     <18    1020    1028   18-24    2021     224   25-44   45-64
##    9344    1628    1805       1       1    6630       1       1    6342     775
##     65+     940 UNKNOWN
##      67       1    3148
```

```
table(nyc_shoot$PERP_RACE)  #has 10938 missing values
```

```
##
##                                                    (null)
##                              9310                    1628
## AMERICAN INDIAN/ALASKAN NATIVE       ASIAN / PACIFIC ISLANDER
##                                 2                     184
##                             BLACK                 BLACK HISPANIC
##                             12323                    1487
##                            UNKNOWN                  WHITE
##                              1838                    305
##                  WHITE HISPANIC
##                              2667
```

```
table(nyc_shoot$VIC_SEX) #no missing values
```

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

```
table(nyc_shoot$VIC_AGE_GROUP) #no missing values
```

```
##
##     <18    1022   18-24   25-44   45-64     65+ UNKNOWN
##    3081       1   10677   13563    2118     236      68
```

```
table(nyc_shoot$VIC_RACE) #no missing values
```

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

```r
#Shows the total missing values in each column
sum(nyc_shoot$PERP_SEX %in% c("(null)", ""), na.rm = TRUE)
```

```
## [1] 10938
```

```r
sum(nyc_shoot$PERP_AGE_GROUP %in% c("(null)", ""), na.rm = TRUE)
```

```
## [1] 10972
```

```r
sum(nyc_shoot$PERP_RACE %in% c("(null)", ""), na.rm = TRUE)
```

```
## [1] 10938
```

```r
sum(nyc_shoot$VIC_SEX %in% c("(null)", ""), na.rm = TRUE)
```

```
## [1] 0
```

```r
sum(nyc_shoot$VIC_AGE_GROUP %in% c("(null)", ""), na.rm = TRUE)
```

```
## [1] 0
```

```r
sum(nyc_shoot$VIC_RACE %in% c("(null)", ""), na.rm = TRUE)
```

```
## [1] 0
```

```r
#Filling missing values with "unknown"
nyc_shoot <- nyc_shoot %>%
  mutate(across(
    c(PERP_SEX, PERP_AGE_GROUP, PERP_RACE, VIC_SEX),
    ~replace(as.character(.), . == "" | . == "(null)" | . == "U" | is.na(.), "UNKNOWN"))) %>%
  mutate(across(
    c(PERP_SEX, PERP_AGE_GROUP, PERP_RACE, VIC_SEX),
    as.factor))
```

Note: Over 30% of the perpetrator demographic variables contain missing values, with some already labeled as "UNKNOWN". I have chosen to impute the remaining missing entries with "UNKNOWN", as it is possible for such information to be unavailable when the perpetrator is not identified by either the victim or the police.

Let's check the remaining columns to see whether there exist redundant categories, or missing values.

```r
unique(nyc_shoot$BORO)
```

```
## [1] BRONX         BROOKLYN      MANHATTAN     QUEENS        STATEN ISLAND
## Levels: BRONX BROOKLYN MANHATTAN QUEENS STATEN ISLAND
```

```
unique(nyc_shoot$PRECINCT)
```

```
##  [1]  40  79  47  66  46  42  71  69  75  76  34  41  70 113  52  77  43 103  73
## [20]  45  50  67  81  24  84  33 120   5  23 115  48 114  32  60  61  72  44   7
## [39] 105  30  14   9  63  78 106  68  83 110 107 100  25  28 109 101  90  49 102
## [58]  26  88 122  20  19 123  18 108  10  62  13 104 121  94 111   6   1 112  17
## [77]  22
```

```
unique(nyc_shoot$MURDER_FLAG)
```

```
## [1] 0 1
```

Note: Everything looks fine, so we'll move on to the next step.

- Refactor column names

```
#Renaming variables to match existing variable naming convention (SCREAMING_SNAKE_CASE)
nyc_shoot <- nyc_shoot %>% rename(
    LATITUDE = Latitude,
    LONGITUDE = Longitude)
```

Note: Renaming to match original `SCREAMING_SNAKE_CASE` used for the majority of variable names.

Let's also refomat `STATEN ISLAND` to `STATEN`

```
nyc_shoot$BORO <- recode(nyc_shoot$BORO, "STATEN ISLAND" = "STATEN")
```

4 Model Evaluaion

In this section, we build and evaluate two models to extract deeper insights from the dataset. The first, is a spatial clustering model using DBSCAN, which explores the geographic spread and density of shooting related incidents. The second, is a time series analysis that examines borough-level trends in reported shootings (per 100,000 residents) during the period 2023–2024. We'll interpret the outputs of both models to assess the patterns and implications they reveal.

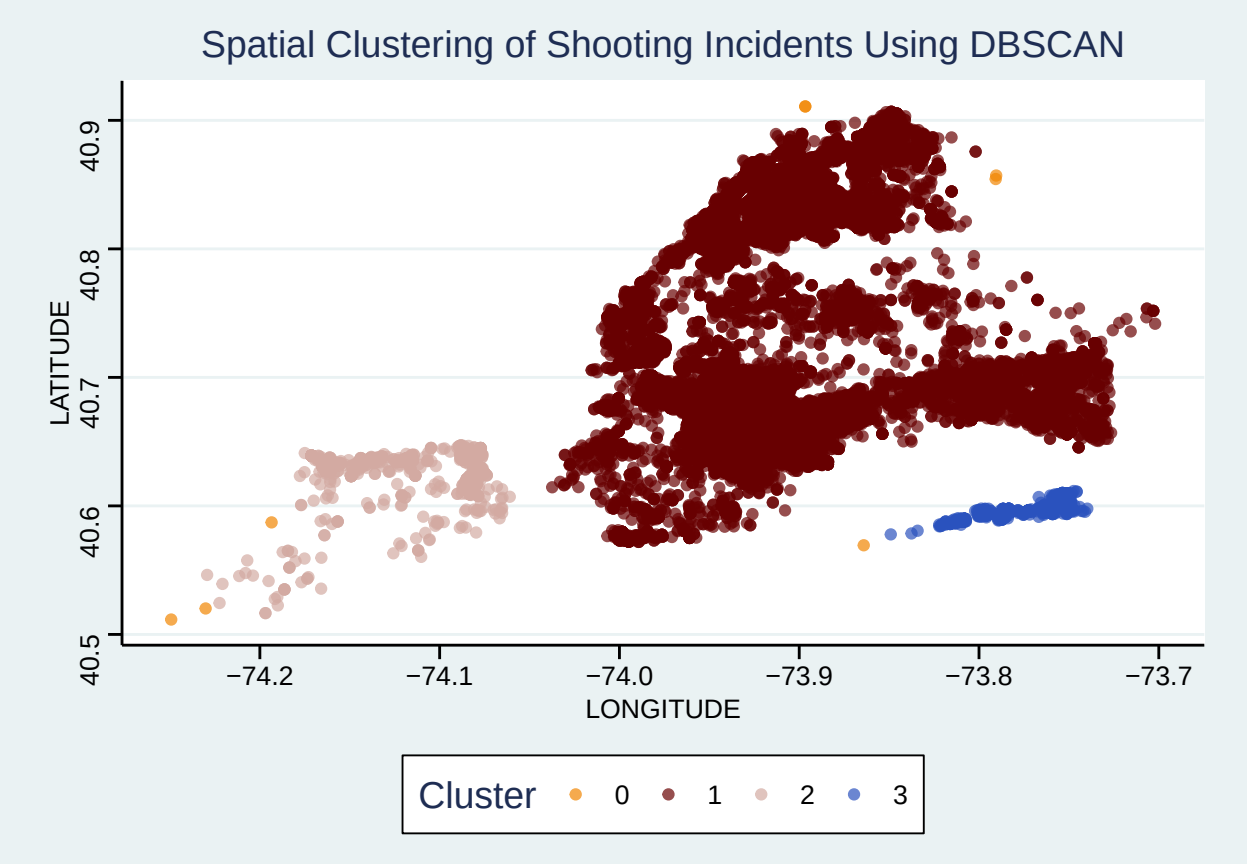
Key Questions:

- Model 1 (Spatial): What does the clustering reveal about the geographic distribution of incidents? Is there a discernible pattern in the clusterings of incident locations?
- Model 2 (Time Series): Do any of the boroughs show noticeable trends in the occurrences of shooting related incidents? Is there evidence of seasonality or recurring patterns?

4.1 Spatial Analysis: Shooting Incidents

We aim to build a model that maps the dispersion of crime across New York City. The objective of this analysis is to identify potential crime hotspots and uncover any observable patterns in the occurrence of these incidents. To achieve this, we will first apply DBSCAN (Density-Based Spatial Clustering of Applications with Noise), an unsupervised clustering algorithm that groups together spatially dense areas of incidents. This will help visualize the geographic spread of shootings. We will then examine a table showing the composition of each cluster by borough, including which boroughs are present in each cluster and the total number of incidents per borough within that cluster.

```r
#Selects columns for clustering
COORD <- nyc_shoot %>%
  select(LONGITUDE, LATITUDE)

#Applies DBSCAN clustering on the coordinate data
db <- dbscan(COORD, eps = 0.020,  minPts = 5)

#Creates a new column in the dataset to store cluster assignments (useful for later step)
nyc_shoot$cluster <- db$cluster

#Shows scatter plot of coordinates colour coded by assigned DBSCAN cluster
ggplot(nyc_shoot, aes(x = LONGITUDE, y = LATITUDE, color = factor(cluster))) +
  scale_color_manual( values = c( "#f28500", "#680001","#d3aaa2", "#2a52be"),na.value = "gray") +
  geom_point(alpha = 0.7) +
  labs(
    color = "Cluster",
    title = "Spatial Clustering of Shooting Incidents Using DBSCAN") + theme_stata()
```



Note: This looks like the map of new york, where the different clusters represent distinct land masses within NYC. The dark red mass is the mainland NYC.

```r
#Looks at the representation of each borough within the clusters (excluding a few outliers)
nyc_shoot %>%
  filter(cluster != 0) %>% #Suppresses outlier.
  count(cluster, BORO) %>%
  arrange(cluster, desc(n))
```

```
##   cluster      BORO     n
## 1       1  BROOKLYN 11685
## 2       1     BRONX  8830
## 3       1 MANHATTAN  3977
## 4       1    QUEENS  3725
## 5       2    STATEN   819
## 6       3    QUEENS   700
```

Note: The above table illustrates the number of shooting incidents per cluster, broken down by borough. Also, outliers have been suppressed.

Analysis:

The DBSCAN algorithm identifies a total of four clusters. The cluster 0 appears to consist of outliers, while the clusters 1 through 3 form groupings that closely resemble distinct land masses within New York City. The empty zones between cluster 1 through 3 resemble water pathways. When compared to a geographical map of NYC, the spatial distribution of these clusters aligns remarkably well with the city's layout, suggesting that the DBSCAN has effectively captured the underlying geographic structure of the data.

In the table following the DBSCAN visualization, the cluster 0 (the outliers) was excluded. The data shows that the vast majority of crimes, approximately 95 percent, occur within the cluster 1. This makes sense, as the cluster 1 includes four of the five boroughs: Manhattan, Brooklyn, the Bronx, and most of Queens, which together account for nearly 90 percent of the city's population. The cluster 2 corresponds to Staten Island, whilst the cluster 3 represents a small extension of Queens. These two clusters account for a much smaller portion of reported shooting incidents, perhaps suggesting that policing efforts should be less targeted to these areas.

4.2 Time Series: Shooting Incidents Per Capita (by Borough) 2023-2024

In order to better understand the distribution of shooting-related crime per capita across NYC boroughs, I decided to conduct a temporal analysis focused on the years 2023 and 2024. Whilst the dataset spans over two decades, conducting a per capita analysis for the entire period is impractical due to challenges in obtaining consistent population data over time, and the requirement for intensive feature engineering. Additionally, the years 2020 to 2022 were likely affected by COVID policies, which may have distorted crime patterns making trend analysis less reliable. By focusing on 2023 and 2024, this analysis captures recent and more stable patterns in shooting incidents, offering a clearer picture of current borough-level trends. The population estimates used are assumed to be relatively stable over this short period, making per capita comparisons more valid.

Let's first confirm that indeed the last reported date falls sometime toward the end of the previous year (2024)

```
#Confirming the latest incident date (should be end of prev. year)
max(nyc_shoot$OCCUR_DATETIME)
```

```
## [1] "2024-12-31 19:16:00 UTC"
```

Now, let's prep the data.

```
#Creating another dataframe for the time series data
timeseries <- nyc_shoot

#Filtering data on dates falling between start of 2023 to end of 2024
timeseries <- timeseries %>%
```

```
  mutate(OCCUR_RECENT = as.Date(OCCUR_DATETIME)) %>%
  filter(between(OCCUR_RECENT, as.Date("2023-01-01"), max(OCCUR_RECENT, na.rm = TRUE)))

#Creating a months column which is extracted from OCCUR_DATETIME
timeseries <- timeseries %>%
  mutate(OCCUR_MONTH = as.Date(floor_date(OCCUR_DATETIME, unit = "month")))

#Confirming the earliest and latest dates are indeed the start of 2023, and the end of 2024
max(timeseries$OCCUR_DATETIME)
```

```
## [1] "2024-12-31 19:16:00 UTC"
```

```
min(timeseries$OCCUR_DATETIME)
```

```
## [1] "2023-01-01 05:55:00 UTC"
```

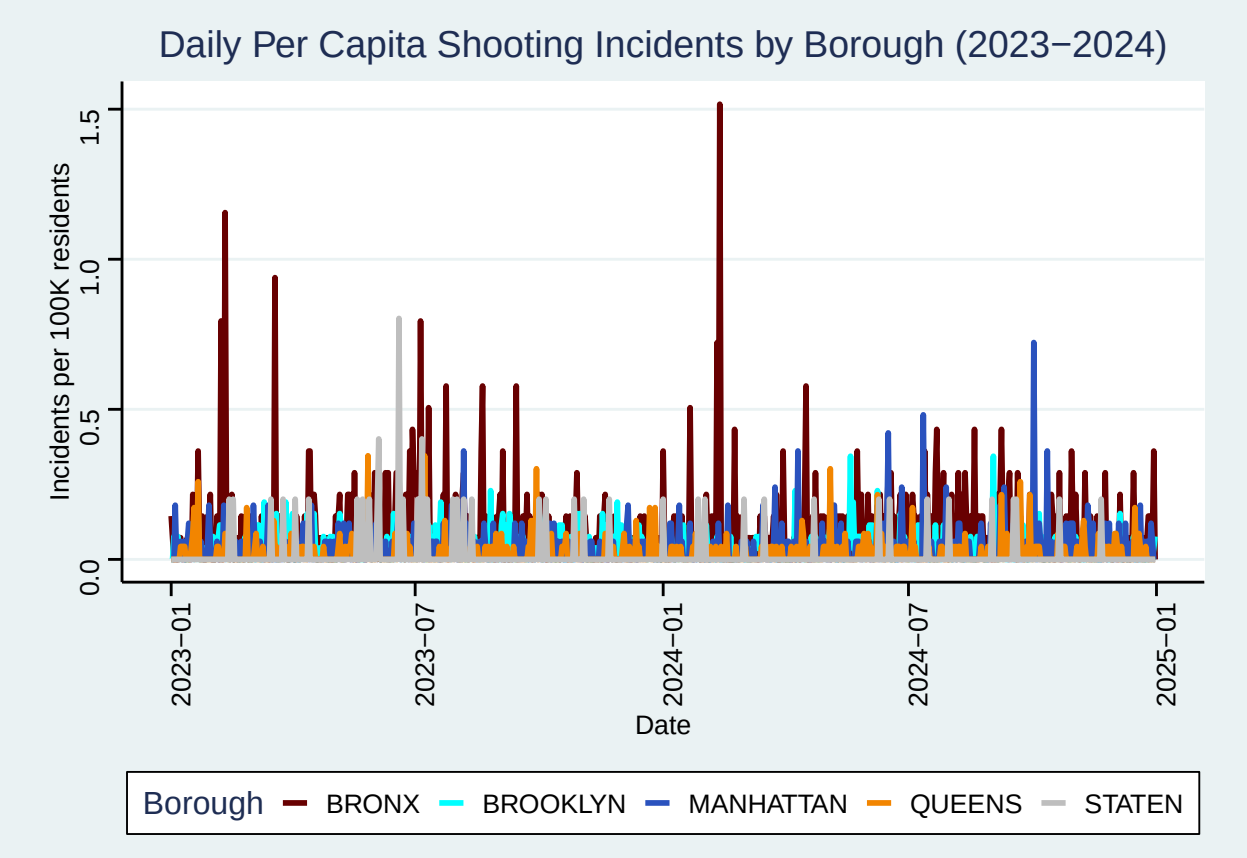
```
glimpse(timeseries)
```

```
## Rows: 2,432
## Columns: 15
## $ BORO           <fct> BROOKLYN, MANHATTAN, STATEN, MANHATTAN, MANHATTAN, BRON~
## $ PRECINCT       <int> 69, 33, 120, 5, 23, 52, 115, 24, 48, 81, 50, 79, 77, 47~
## $ PERP_AGE_GROUP <fct> UNKNOWN, UNKNOWN, <18, 25-44, 18-24, UNKNOWN, 45-64, 25~
## $ PERP_SEX       <fct> UNKNOWN, UNKNOWN, M, M, M, UNKNOWN, M, F, M, M, M, M, U~
## $ PERP_RACE      <fct> UNKNOWN, UNKNOWN, BLACK, BLACK, BLACK, UNKNOWN, BLACK, ~
## $ VIC_AGE_GROUP  <fct> 25-44, 25-44, 18-24, 25-44, 18-24, 18-24, 25-44, 25-44,~
## $ VIC_SEX        <fct> M, F, F, M, M, M, M, M, M, M, M, M, M, M, M, M, M, M, M~
## $ VIC_RACE       <fct> BLACK, BLACK, WHITE HISPANIC, WHITE HISPANIC, BLACK, BL~
## $ LATITUDE       <dbl> 40.66548, 40.80150, 40.62353, 40.80150, 40.80150, 40.84~
## $ LONGITUDE      <dbl> -73.93157, -73.95100, -74.11356, -73.95100, -73.95100, ~
## $ MURDER_FLAG    <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0~
## $ OCCUR_DATETIME <dttm> 2023-04-01 16:35:00, 2023-06-19 22:05:00, 2023-06-19 1~
## $ cluster        <int> 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ OCCUR_RECENT   <date> 2023-04-01, 2023-06-19, 2023-06-19, 2023-08-19, 2023-1~
## $ OCCUR_MONTH    <date> 2023-04-01, 2023-06-01, 2023-06-01, 2023-08-01, 2023-1~
```

And now let's build our model ...

```
#Setting static population estimates for 2024 (useful for per capita calculations)
boro_pop <- c(
  BRONX = 1384724,
  BROOKLYN = 2617631,
  MANHATTAN = 1660664,
  QUEENS = 2316841,
  STATEN = 498212)

#Creates full date range for 2023-2024 (to include dates with no incidents)
dates <- seq(as.Date("2023-01-01"), as.Date("2024-12-31"), by = "day")

#Shows the total number of daily incidents (and daily incident per 100k residents) grouped by borough
timeseries_1 <- timeseries %>%
```

```
  filter(BORO %in% names(boro_pop)) %>%
  group_by(OCCUR_RECENT, BORO) %>%
  summarise(INCIDENTS = n(), .groups = "drop") %>%
  complete(OCCUR_RECENT = dates, BORO = names(boro_pop), fill = list(INCIDENTS = 0)) %>%
  mutate(INCIDENT_PERCAP = INCIDENTS / (boro_pop[BORO] / 100000)) %>%
  arrange(BORO, OCCUR_RECENT)


#Shows time series plot on daily shooting incident per 100k resid. by borough
ggplot(timeseries_1, aes(x = OCCUR_RECENT, y = INCIDENT_PERCAP, color = BORO)) +
  geom_line(size = 1) +
  scale_color_manual( values = c( "#680001", "cyan", "#2a52be", "#f28500","gray"),na.value = "black") +
  labs(
    title = "Daily Per Capita Shooting Incidents by Borough (2023-2024)",
    x = "Date",
    y = "Incidents per 100K residents",
    color = "Borough") + theme_stata() +
    theme(
      axis.text.x = element_text(angle = 90, hjust = 1),
      legend.text = element_text(size = 10),
      legend.title = element_text(size = 12),
      legend.key.size = unit(0.4, "cm"))
```

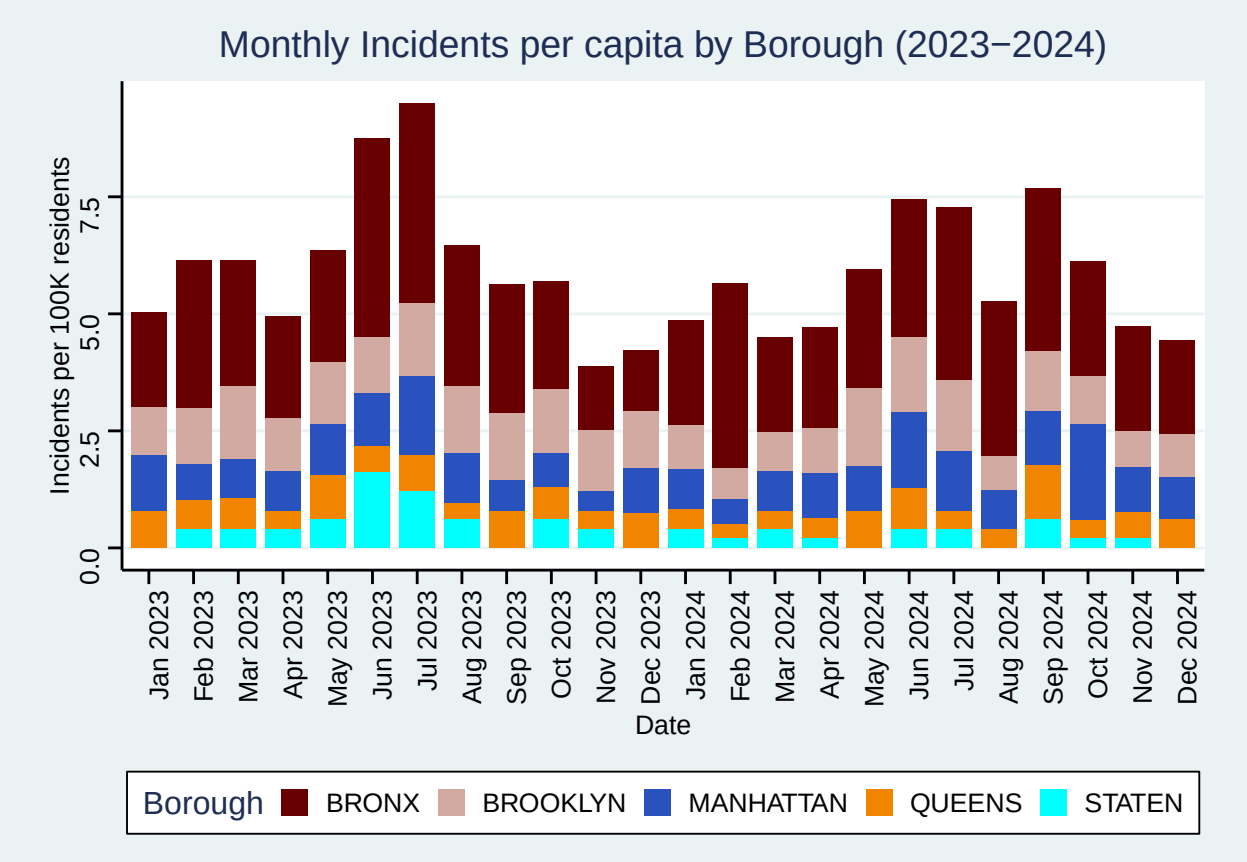


Note: From the above graph, it appears as though the Bronx has a notably higher number of reported shooting incidents per capita compared to the boroughs. On this scale it is a bit difficult to see how the other boroughs compare, to get a better view let's look at monthly incidents per capita. Also, to ensure continuity,

we included the full date range for both years, allowing even days with no reported incidents to appear in the graph.

Let's show the monthly incidents to perhaps get a clearer view of the trends.

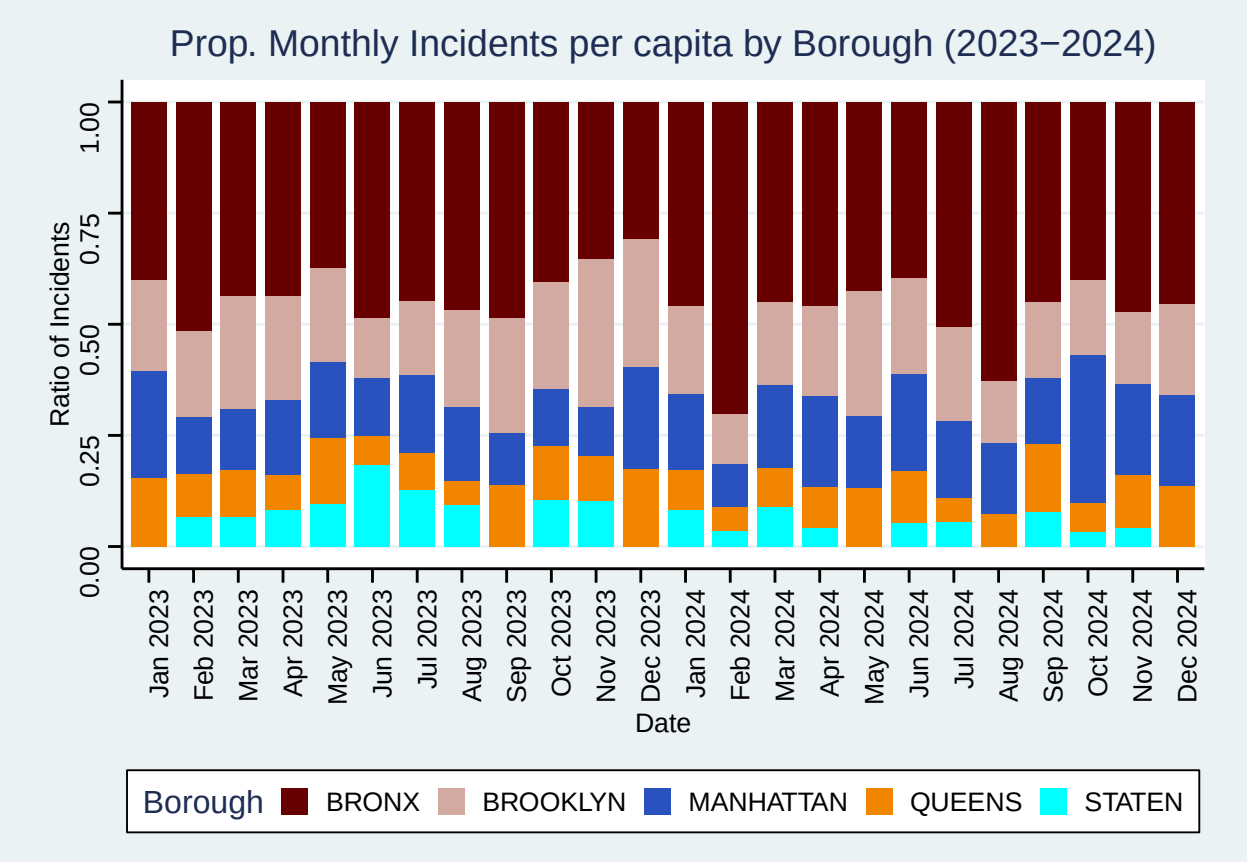
```
#Shows the number of monthly incidents (and monthly incident per 100k residents) grouped by borough
timeseries_2 <- timeseries %>%
  filter(BORO %in% names(boro_pop)) %>%
  group_by(OCCUR_MONTH, BORO) %>%
  summarise(INCIDENTS = n(), .groups = "drop") %>%
  mutate(INCIDENT_PERCAP = INCIDENTS / (boro_pop[BORO] / 100000))

#Filters the data to only include months within the 2023-2024 time frame and creates formatted labels
timeseries_2 <- timeseries_2 %>%
  filter(OCCUR_MONTH >= as.Date("2023-01-01") & OCCUR_MONTH <= as.Date("2024-12-31")) %>%
  mutate(MONTH_LABEL = format(OCCUR_MONTH, "%b %Y")) %>%
  mutate(MONTH_LABEL = factor(MONTH_LABEL, levels = unique(MONTH_LABEL)))

#Plots a bar chart of the monthly per capita incidents
ggplot(timeseries_2, aes(x = MONTH_LABEL, y = INCIDENT_PERCAP, fill = BORO)) +
  geom_col(position = "stack", width = 0.8) +
  labs(
    title = "Monthly Incidents per capita by Borough (2023-2024)",
    x = "Date",
    y = "Incidents per 100K residents",
    fill = "Borough") +
  scale_fill_manual(values = c( "#680001", "#d3aaa2", "#2a52be", "#f28500", "cyan"), na.value = "gray") +
  theme(
    axis.text.x = element_text(angle = 90, hjust = 1),
    legend.text = element_text(size = 10),
    legend.title = element_text(size = 12),
    legend.key.size = unit(0.4, "cm"))
```



Note: Examining the graph above, which shows the monthly per capita incidents per borough, it appears as though the data exhibits a somewhat cyclical pattern, with shooting incidents peaking around the summer months, and declining as winter approaches, bottoming out in the winter months.

```
#Shows the number of monthly incidents (and monthly incident per 100k residents) grouped by borough
timeseries_2 <- timeseries %>%
  filter(BORO %in% names(boro_pop)) %>%
  group_by(OCCUR_MONTH, BORO) %>%
  summarise(INCIDENTS = n(), .groups = "drop") %>%
  mutate(INCIDENT_PERCAP = INCIDENTS / (boro_pop[BORO] / 100000))

#Filters the data to only include months within the 2023-2024 time frame and creates formatted labels
timeseries_2 <- timeseries_2 %>%
  filter(OCCUR_MONTH >= as.Date("2023-01-01") & OCCUR_MONTH <= as.Date("2024-12-31")) %>%
  mutate(MONTH_LABEL = format(OCCUR_MONTH, "%b %Y")) %>%
  mutate(MONTH_LABEL = factor(MONTH_LABEL, levels = unique(MONTH_LABEL)))

#Creates a stacked proportional bar chart showing the monthly shooting incidents per capita by borough
ggplot(timeseries_2, aes(x = MONTH_LABEL, y = INCIDENT_PERCAP, fill = BORO)) +
  geom_col(position = "fill", width = 0.8) +
  labs(
    title = "Prop. Monthly Incidents per capita by Borough (2023-2024)",
    x = "Date",
    y = "Ratio of Incidents",
    fill = "Borough") +
  scale_fill_manual(values = c( "#680001", "#d3aaa2", "#2a52be", "#f28500", "cyan"),
```

```
                    na.value = "gray") + theme_stata() +
  theme(
    axis.text.x = element_text(angle = 90, hjust = 1),
    legend.text = element_text(size = 10),
    legend.title = element_text(size = 12),
    legend.key.size = unit(0.4, "cm"))
```




```
#Shows the annual per capita incidents by borough
timeseries_2 %>%
  mutate(YEAR = lubridate::year(OCCUR_MONTH)) %>%
  group_by(BORO, YEAR) %>%
  summarise(TOT_PERCAP = sum(INCIDENT_PERCAP, na.rm = TRUE), .groups = "drop")
```

```
## # A tibble: 10 x 3
##    BORO       YEAR TOT_PERCAP
##    <fct>     <dbl>      <dbl>
##  1 BRONX      2023       31.7
##  2 BRONX      2024       33.1
##  3 BROOKLYN   2023       15.8
##  4 BROOKLYN   2024       13.0
##  5 MANHATTAN  2023       11.4
##  6 MANHATTAN  2024       12.9
##  7 QUEENS     2023        7.64
##  8 QUEENS     2024        6.69
##  9 STATEN     2023        6.22
```

```
## 10 STATEN     2024       3.01
```

Note: Here, we're able to more clearly see how the boroughs compare in terms of shooting-related crime. In both 2023 and 2024, the Bronx consistently had significantly higher per capita shooting incidents than any other borough. Brooklyn and Manhattan are distant seconds, both with per capita incidents more than 50% lower than the Bronx. While Brooklyn had a higher number of reported incidents per cap in 2023, in 2024, its per capita rate was nearly identical to Manhattan's. Queens and Staten Island had the lowest shooting rates overall, with Staten Island standing out as the only borough to have entire months with zero reported shooting incidents.

Analysis:

In the first graph, we examine daily shooting incidents per 100,000 residents, broken down by borough. While the raw data only includes days with reported incidents, we explicitly included days with zero incidents for each borough. By covering the full date range, we ensure a continuous and accurate line chart, which is essential for interpreting trends over time. The graph shows some volatility but no clear upward or downward trend, indicating that although shootings fluctuate daily, there's no significant increase or decrease in per capita shooting incidents overall.

At the borough level, the Bronx stands out with noticeably higher per capita shootings compared to the other borough. Among the remaining boroughs, the differences are less distinct, though Manhattan appears to have more peaks in 2024, while Staten Island had more peaks in 2023. To better capture any underlying cyclical patterns, we turn to the next graph, which presents monthly shooting incidents per capita by borough.

In the second graph, a clear cyclical pattern emerges, we see a rising incidents per capita as summer approach, peaking in the summer months, and then declining steadily and bottoming out in the winter. This "cyclical" behaviour can be described as seasonality. Whilst we cannot say for certain what drives the seasonal shift, these patterns could indicate that harsher weather is a potential deterrent to criminal behaviour, while warmer weather creates a more conducive environment.

The final graph is a stacked bar chart designed to visualize how each borough compares in terms of monthly shooting incidents per capita. From this visualization, it's evident that the Bronx stands out with a notably prominent crime rate compared to the other boroughs. Brooklyn emerges as a distant second, although the gap has significantly narrowed between it and Manhattan in 2024. For the remaining boroughs, it is still quite difficult to confidently distinguish their values clearly in the graph, prompting a closer look at the accompanying table. The table confirms that the Bronx leads by a substantial margin in shootings per capita, followed by Brooklyn and then Manhattan. Queens and Staten Island report the lowest rates, with Staten Island notably having several months with zero reported shooting incidents.

Looking at the interannual changes, the Bronx and Manhattan were the only two boroughs that experienced an increase in shooting incidents per capita between 2023 and 2024. While this is a notable observation, we cannot confidently interpret it as part of a larger trend without data from additional years. In contrast, Brooklyn, Queens, and Staten Island all saw declines, with Staten Island experiencing a 50%+ dip. However, due to Staten Island's relatively small population, its per capita rates tend to be more volatile, meaning this large decline could be exaggerated and should be interpreted with a degree of skepticism.

Overall, the Bronx has consistently stood out in terms of having the highest shooting incidents per capita. Based on my observations, it is clear that more policing efforts and resources should be diverted toward the Bronx. Increased patrols can act as a deterrent, and stronger, positive community-police engagement may encourage residents to take an active role in enhancing the safety of their borough.

5 Limitations & Challenges

Finally, we discuss some of then biases, challenges, and limitations we encountered throughout this report.

5.1 Bias

I think it is easy for individuals to develop bias when analyzing this dataset, particularly because it includes information about the race of perpetrators. Whilst race can be viewed as just another identifier, like age or sex, the inclusion in crime data can have social repercussions. Although the intention behind including race may be to aid identification, it can inadvertently reinforce racial stereotypes and/or divide. In the case of this analysis, I chose to rather focus on geo-information of the crime, to try to identify whether there were any patterns or trends (cyclical or otherwise) in the occurrences of shooting incidents.

5.2 Other Factors

Assumptions:

Model 2 (Time Series)

- Excluded 2020-2022 data, since these were COVID-19 periods and would have been impacted by COVID-related policies, potentially leading to a distortion in crime patterns.
- Used an static estimate populations for 2024 to compute the per capita shooting incidents for each borough, which assumes that population change over a year course would likely not change drastically.

Challenges

- Identifying missing or "unknown" values: This was tricky in character and factor variables, especially in perpetrator demographic fields, which showed varying degrees of missingness.
- Deciding how to handle missing values: For the spatial model, a small portion (<1%) of coordinate data was missing, and I imputed these using the mean longitudes/lattitudes of the corresponding borough (based on the borough listed in each affected row).

6 Conclusion

Throughout this report we refer to "shooting incidents", which is really "reported shooting incidents" as the data pertains to, and is only relevant to reporting. We had build two models, the first was a spatial clustering model using DBSCAN, and the second was a time series model (daily, monthly, and annually) showing the per capita shooting incident (per 100K residents) by borough for 2023 to 2024. The spatial analysis revealed a heavy concentration of shooting incidents in more populous boroughs, which are the Bronx, Brooklyn, Manhattan, and most of Queens (see cluster 1). The time series analysis provided a detailed breakdown of incidents over time by borough, illustrating that the Bronx maintained the highest shooting rates over time. This reinforced by the monthly incident per capita graph, and the annual incident table breakdown. The Bronx, alongside Manhattan, were the only boroughs that exhibited an upward trend in annual per capita incidents. Meanwhile, Brooklyn, Queens, and Staten Island showed declining trends, with Staten Island's small population contributing to greater volatility in its per capita measures. Together, these analyses highlighted the Bronx as a persistent hotspot requiring targeted intervention. To address this, focused policing strategies, increased patrols, and enhanced community engagement efforts are required. Moreover, continued monitoring through spatial and temporal data will be critical to adapting interventions and evaluating their effectiveness over time. Ultimately, combining geographic insights with temporal pattern provides a robust framework to inform public safety policies and resource allocation aimed at reducing shooting-related incidents across NYC.

Further Work:

Model 1 (Spatial):

- Observe spatial patterns based on fatality.
- Overlay DBSCAN plot with actual nyc map to compare.
- Compare DBSCAN plot to a population density plot to derive further insights.

Model 2 (Time Series):

- Observe at a longer duration period to determine whether observations made still hold true.

7 References

[1]"NYPD Shooting Incident Data (Historic)," Data.gov, Nov. 22, 2021. https://catalog.data.gov/dataset/nypd-shooting-incident-data-historic

[2] RStudio, "Data Visualization with ggplot2," GitHub, Accessed Jun. 20, 2025. [Online]. Available: https://github.com/rstudio/cheatsheets/blob/main/data-visualization.pdf

[3] "New York City (United States): Boroughs - Population Statistics, Maps, Charts and Weather," City-Population, 2023. [Online]. Available: https://www.citypopulation.de/en/usa/newyorkcity/. [Accessed: 20-Jun-2025].