

# NYPD Shooting Incident Report

## Before you kint

### Required Packages

This analysis requires several R packages to be installed and loaded for proper execution. Before running the code, please ensure the following packages are installed on your system: `tidyverse`, `ggplot2`, `knitr`, `foercast`.

To check whether a package is installed, you can use the `require()` function. If the package is not installed, `require()` will return `FALSE`. You can then install the missing package using the `install.packages()` function.

Here is a code snippet to automate this process. Copy and paste the code in the console to execute it

```
## this chunk will not be executed when ran as part of this notebook
## please run the code in your console to install these packages

# required packages
required_packages <- c("tidyverse", "ggplot2", "knitr", "forecast")

# Install missing packages
for (pkg in required_packages) {
  if (!require(pkg, character.only = TRUE)) {
    install.packages(pkg)
  }
}

# Load all packages
lapply(required_packages, library, character.only = TRUE)
```

## Introduction

The primary objective of this analysis is to investigate the temporal and spatial patterns of shooting incidents in New York City from 2006 to 2023. Leveraging the NYPD Shooting Incident Dataset, this study examines trends in the frequency and distribution of incidents across boroughs and time periods. Key areas of focus include identifying high-risk hours and boroughs through heatmap visualizations and analyzing long-term trends to understand how incidents have evolved over time.

Additionally, the study employs time series forecasting to predict future shooting incidents, providing actionable insights for public safety strategies. By combining visualizations and statistical modeling, this analysis aims to highlight critical patterns, evaluate the effectiveness of past interventions, and identify opportunities for targeted resource allocation to improve public safety outcomes in New York City.

## About the Data

NYPD Shooting Incident Dataset contains a comprehensive record of every shooting incident in New York City from 2006 through the end of the previous calendar year. Updated quarterly, the data is manually extracted and reviewed by the NYPD Office of Management Analysis and Planning before publication. Each record includes details about the incident, such as location, time, and event type, along with demographic information about the victims and suspects. This dataset allows for public exploration of trends in shooting incidents and criminal activity in NYC.

The NYPD Shooting Incident Dataset consists of 21 variables and 28,562 observations. The variables include information on incident details, location, victim and perpetrator demographics, and geographic coordinates of each shooting incident. A full data dictionary and additional details can be found [here](#).

## Load the necessary packages

```
library(tidyverse)
library(ggplot2)
library(knitr)
library(forecast)
```

## Load the data

```
url_name1<- 'https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD'

nypd_data<-read_csv(url_name1, show_col_types = FALSE)
```

## Clean the data

To prepare the dataset for analysis, I performed several cleaning and preprocessing steps. First, I replaced placeholder values such as (null) and UNKNOWN with NA to indicate missing data. I also addressed odd or invalid values in variables like perpetrator and victim age groups, sex, and location descriptions by converting them to NA. Next, I formatted variables appropriately, converting dates to Date format, categorical variables to factors, and jurisdiction codes to descriptive labels. Age groups were standardized and ordered, and additional variables, such as the occurrence month, were derived for temporal analysis. These steps ensured the dataset was clean, consistent, and ready for analysis.

```

nypd_data_clean <- nypd_data %>%
  # Replace (null) and UNKNOWN values with NA
  mutate(
    across(
      where(is.character),
      ~ ifelse(.x %in% c("(null)", "UNKNOWN"), NA, .x)
    )
  ) %>%

  # Replace odd or invalid values with NA
  mutate(
    PERP_AGE_GROUP = ifelse(PERP_AGE_GROUP %in% c("1020", "1028", "224", "940"),
      NA, as.character(PERP_AGE_GROUP)),
    VIC_AGE_GROUP = ifelse(VIC_AGE_GROUP %in% c("1022"),
      NA, as.character(VIC_AGE_GROUP)),
    PERP_SEX = ifelse(PERP_SEX %in% c("U"),
      NA, as.character(PERP_SEX)),
    VIC_SEX = ifelse(VIC_SEX %in% c("U"),
      NA, as.character(VIC_SEX)),
    LOCATION_DESC = ifelse(LOCATION_DESC %in% c("NONE"),
      NA, as.character(LOCATION_DESC))
  ) %>%

  # Format and recode variables
  mutate(
    OCCUR_DATE = as.Date(OCCUR_DATE, format = "%m/%d/%Y"),
    BORO = as.factor(BORO),
    JURISDICTION = recode(
      JURISDICTION_CODE,
      `0` = "Patrol",
      `1` = "Transit",
      `2` = "Housing"
    ),
    JURISDICTION = as.factor(JURISDICTION),
    LOC_OF_OCCUR_DESC = as.factor(LOC_OF_OCCUR_DESC),
    PERP_AGE_GROUP = factor(PERP_AGE_GROUP,
      levels = c("<18", "18-24", "25-44", "45-64", "65+")),
    PERP_SEX = as.factor(PERP_SEX),
    PERP_RACE = as.factor(PERP_RACE),
    VIC_AGE_GROUP = factor(VIC_AGE_GROUP,
      levels = c("<18", "18-24", "25-44", "45-64", "65+")),
    VIC_SEX = as.factor(VIC_SEX),
    VIC_RACE = as.factor(VIC_RACE),
    LOC_CLASSFCTN_DESC = as.factor(LOC_CLASSFCTN_DESC),
    LOCATION_DESC = as.factor(LOCATION_DESC)
  )

  # Create additional variables
nypd_data_clean <- nypd_data_clean %>%
  mutate(
    OCCUR_MONTH = as.Date(floor_date(OCCUR_DATE))
  )

```

## Visualize the Data

To explore the dynamics of shooting incidents in New York City, I chose two key visualizations. The first is a heatmap that examines the distribution of incidents across boroughs and times of day, providing insights into temporal and spatial patterns. The second one is a trend analysis that tracks the number of incidents over time, offering a long-term perspective on changes in frequency of the incidents. Together, these visuals shed light on critical patterns and trends that can advice on resource allocation, policy decisions, and public safety strategies.

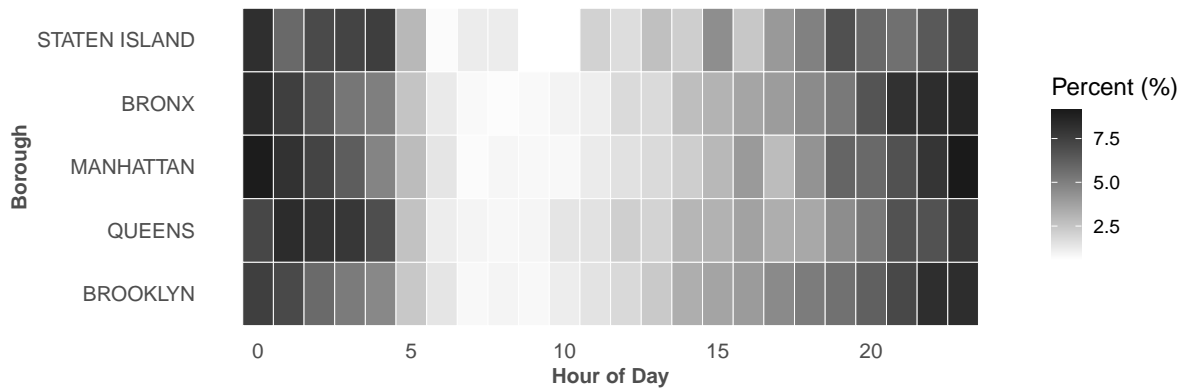
### Heatmap of Shooting Incidents by Borough and Time of Day

This heatmap highlights temporal and spatial patterns in shooting incidents, helping us identify when and where incidents are most likely to occur. By visualizing the data this way, we can uncover high-risk hours and boroughs, which can inform resource allocation for law enforcement and public safety efforts. For instance, if the heatmap shows a concentration of incidents during late-night hours in specific boroughs, targeted patrols or community interventions can be implemented to mitigate these risks.

```
# Calculate percentage of incidents by borough and hour
nypd_percentage <- nypd_data_clean %>%
  mutate(
    HOUR = hour(hms(OCCUR_TIME)) # Extract the hour from OCCUR_TIME
  ) %>%
  group_by(BORO, HOUR) %>% # Group by borough and hour
  summarise(
    INCIDENTS = n(), # Count the number of incidents
    .groups = "drop"
  ) %>%
  group_by(BORO) %>% # Group by borough to calculate percentages
  mutate(
    PERCENT = INCIDENTS / sum(INCIDENTS) * 100 # Calculate percentage within each borough
  )
# Manually reorder BORO from largest to smallest population
nypd_percentage <- nypd_percentage %>%
  mutate(BORO = factor(BORO, levels = c("BROOKLYN", "QUEENS", "MANHATTAN",
                                         "BRONX", "STATEN ISLAND")))
ggplot(nypd_percentage, aes(x = HOUR, y = BORO, fill = PERCENT)) +
  geom_tile(color = "white") + # Heatmap with white borders for separation
  scale_fill_gradient(low = "white", high = "gray10", name = "Percent (%)") +
  labs(
    title = "Heatmap of Shooting Incidents by Borough and Time of Day",
    subtitle = "Percentage distribution of incidents by hour within each borough",
    x = "Hour of Day", y = "Borough",
    caption = "Source: NYPD Shooting Incident Dataset"
  ) +
  theme_minimal(base_size = 14) +
  theme(
    plot.title = element_text(face = "bold", size = 18, hjust = 0.5),
    plot.subtitle = element_text(size = 12, hjust = 0.5, margin = margin(b = 10)),
    plot.caption = element_text(size = 10, hjust = 1, color = "gray40"),
    axis.text = element_text(size = 12, color = "gray30"),
    axis.title = element_text(size = 12, face = "bold", color = "gray30"),
    panel.grid.major = element_blank(), # Minimal gridlines
    panel.grid.minor = element_blank(),
    plot.margin = margin(20, 20, 20, 20)
  )
```

## Heatmap of Shooting Incidents by Borough and Time of Day

Percentage distribution of incidents by hour within each borough



Source: NYPD Shooting Incident Dataset

The heatmap reveals distinct temporal and spatial patterns in NYC shooting incidents, highlighting borough-specific risks and peak activity times. Across all boroughs, incidents are most frequent during nighttime hours, particularly between 10 PM and 2 AM, with a secondary spike occurring in the late evening between 8 PM and 11 PM.

The Bronx shows significant activity during late-night hours, with the highest percentages of incidents occurring at 11 PM (8.66%), midnight (8.40%), and 10 PM (8.27%). Brooklyn also experiences notable peaks at 11 PM (8.27%) and 10 PM (8.22%), but its incidents are more evenly distributed across the late evening. Manhattan displays a similar pattern, with a high concentration of incidents at 11 PM (9.14%) and midnight (9.04%), although it has fewer incidents during early morning hours compared to other boroughs. Queens, in contrast, exhibits a broader temporal distribution, with its highest activity occurring at 11 PM (7.77%) and 1 AM (8.34%). Staten Island has the smallest overall share of incidents but follows a similar nighttime trend, peaking at midnight (8.18%) and 11 PM (7.19%).

To address these trends, resources should be concentrated in the Bronx, Brooklyn, and Manhattan during peak hours, while community outreach efforts could target high-risk areas during late evenings. Further analysis could explore socioeconomic or event-driven factors contributing to these patterns, providing deeper insights for effective intervention strategies.

## Trend Analysis: Number of Shooting Incidents Over Time

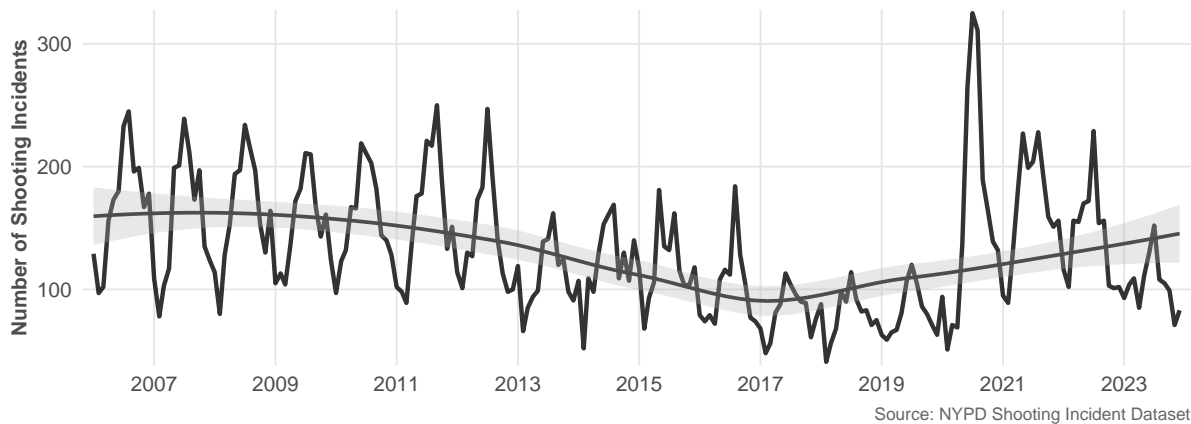
The trend analysis provides a macro-level perspective on how shooting incidents have evolved over the years. It helps identify whether incidents are increasing, decreasing, or stabilizing, offering insights into the effectiveness of policy changes or community interventions. Spikes or declines in certain periods might correlate with significant events (e.g., economic downturns, policy implementations, or public awareness campaigns), making this visualization crucial for long-term planning and evaluation.

```
# Aggregate data by month
incident_trends <- nypd_data_clean %>%
  mutate(YEAR_MONTH = floor_date(OCCUR_DATE, "month")) %>%
  group_by(YEAR_MONTH) %>%
  summarise(INCIDENTS = n())

# visualize the trend
ggplot(incident_trends, aes(x = YEAR_MONTH, y = INCIDENTS)) +
  # Main line plot for trends
  geom_line(color = "black", size = 1.2, alpha = 0.8) +
  # Smoothed trend line
  geom_smooth(method = "loess", color = "gray30", fill = "gray70",
             se = TRUE, size = 1, alpha = 0.3) +
  # Titles and labels
  labs(
    title = "Trend of NYC Shooting Incidents",
    subtitle = "Monthly shooting incidents from 2006 to 2023",
    x = NULL,
    y = "Number of Shooting Incidents",
    caption = "Source: NYPD Shooting Incident Dataset"
  ) +
  # NY Times-inspired theme
  theme_minimal(base_size = 14) +
  theme(
    plot.title = element_text(face = "bold", size = 18, hjust = 0.5),
    plot.subtitle = element_text(size = 12, hjust = 0.5, margin = margin(b = 10)),
    plot.caption = element_text(size = 10, hjust = 1, color = "gray40"),
    axis.text = element_text(size = 12, color = "gray30"),
    axis.title = element_text(size = 12, face = "bold", color = "gray30"),
    panel.grid.major = element_line(color = "gray90", size = 0.5),
    panel.grid.minor = element_blank(),
    plot.margin = margin(20, 20, 20, 20)
  ) +
  # Adjust x-axis for NYT-style clarity
  scale_x_date(
    date_labels = "%Y",
    date_breaks = "2 years",
    expand = expansion(mult = c(0.01, 0.01))
  ) +
  # Adjust y-axis to start from 0 for clarity
  scale_y_continuous(
    expand = expansion(mult = c(0.01, 0.01))
  )
```

## Trend of NYC Shooting Incidents

Monthly shooting incidents from 2006 to 2023



In 2020, incidents reached their highest, with 325 incidents recorded in July, potentially reflecting heightened tensions during the pandemic and civil unrest. Other significant peaks occurred in 2011 (September) and 2012 (July), with 250 and 247 incidents, respectively. From 2021 to 2023, July and August continued to dominate as the months with the most incidents, although the overall numbers have declined since the 2020 peak.

Occasional outliers break the summer pattern, such as May 2015 and June 2017, showing that while summer months generally experience the highest incidents, deviations occur, possibly driven by unique circumstances or events.

These trends underscore the importance of focusing intervention efforts during summer months, particularly in July and August, to prevent a surge in incidents. Exploring the factors driving these seasonal trends, such as weather patterns, holidays, or socio-economic conditions, could provide further insights to guide public safety strategies effectively.

## Forecasting incidents

To better understand and forecast the trends observed in shooting incidents, a time series analysis using an ARIMA model was used. While the observed patterns highlight seasonal surges and occasional potential outliers, quantifying these trends allows for a more precise examination of long-term changes and short-term variations. By capturing both seasonal and non-seasonal components, the ARIMA model provides a robust framework to analyze how incidents evolve over time and predict future occurrences. This analysis aims to guide proactive measures, especially during high-risk periods such as the summer months, and identify potential shifts in underlying patterns. The details of the fitted model and its implications are presented below.

```
# Convert the incident_trends data into a time series object
shooting_ts <- ts(incident_trends$INCIDENTS,
                  start = c(year(min(incident_trends$YEAR_MONTH)),
                           month(min(incident_trends$YEAR_MONTH))),
                  frequency = 12) # Monthly data

# Fit an ARIMA model to the time series
arma_model <- auto.arima(shooting_ts)

# Forecast the next 12 months
forecasted_incidents <- forecast(arma_model, h = 12)

arma_model

## Series: shooting_ts
## ARIMA(1,0,0)(1,1,0)[12] with drift
##
## Coefficients:
##          ar1      sar1      drift
##          0.6957 -0.3703 -0.3007
## s.e.    0.0507   0.0662   0.4204
##
## sigma^2 = 906.9: log likelihood = -983.79
## AIC=1975.58  AICc=1975.78  BIC=1988.85

# Convert forecast object to a data frame for ggplot
forecast_df <- data.frame(
  DATE = seq(from = tail(incident_trends$YEAR_MONTH, 1) + months(1),
             by = "month",
             length.out = 12),
  POINT_FORECAST = as.numeric(forecasted_incidents$mean),
  LOWER_80 = as.numeric(forecasted_incidents$lower[, 1]),
  UPPER_80 = as.numeric(forecasted_incidents$upper[, 1]),
  LOWER_95 = as.numeric(forecasted_incidents$lower[, 2]),
  UPPER_95 = as.numeric(forecasted_incidents$upper[, 2])
)

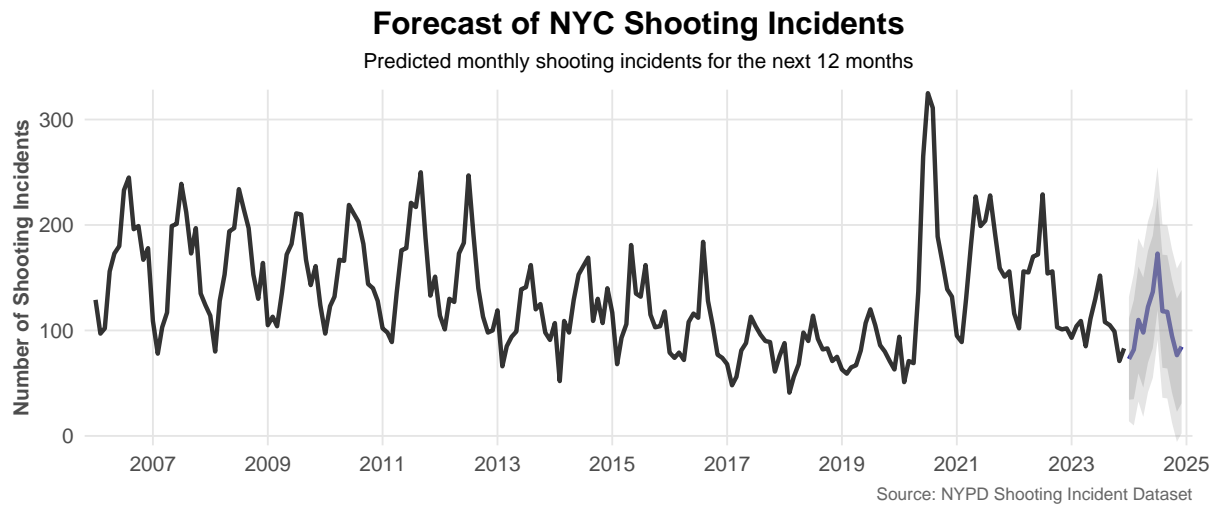
# Combine historical data with forecast data
combined_data <- rbind(
  data.frame(DATE = incident_trends$YEAR_MONTH, INCIDENTS = incident_trends$INCIDENTS),
  data.frame(DATE = forecast_df$DATE, INCIDENTS = forecast_df$POINT_FORECAST)
```

```

)

# Plot historical data and forecast
ggplot() +
  # Historical data line
  geom_line(data = incident_trends, aes(x = YEAR_MONTH, y = INCIDENTS),
            color = "black", size = 1.2, alpha = 0.8) +
  # Smoothed trend line
  geom_smooth(method = "loess", color = "gray30", fill = "gray70",
              se = TRUE, size = 1, alpha = 0.3) +
  # Forecasted data line
  geom_line(data = forecast_df, aes(x = DATE, y = POINT_FORECAST),
            color = "darkblue", size = 1.2, alpha = 0.8) +
  # 80% confidence interval
  geom_ribbon(data = forecast_df, aes(x = DATE, ymin = LOWER_80, ymax = UPPER_80),
             fill = "gray70", alpha = 0.4) +
  # 95% confidence interval
  geom_ribbon(data = forecast_df, aes(x = DATE, ymin = LOWER_95, ymax = UPPER_95),
             fill = "gray50", alpha = 0.2) +
  # Titles and labels
  labs(
    title = "Forecast of NYC Shooting Incidents",
    subtitle = "Predicted monthly shooting incidents for the next 12 months",
    x = NULL,
    y = "Number of Shooting Incidents",
    caption = "Source: NYPD Shooting Incident Dataset"
  ) +
  # NY Times-inspired theme
  theme_minimal(base_size = 14) +
  theme(
    plot.title = element_text(face = "bold", size = 18, hjust = 0.5),
    plot.subtitle = element_text(size = 12, hjust = 0.5, margin = margin(b = 10)),
    plot.caption = element_text(size = 10, hjust = 1, color = "gray40"),
    axis.text = element_text(size = 12, color = "gray30"),
    axis.title = element_text(size = 12, face = "bold", color = "gray30"),
    panel.grid.major = element_line(color = "gray90", size = 0.5),
    panel.grid.minor = element_blank(),
    plot.margin = margin(20, 20, 20, 20)
  ) +
  # Adjust x-axis for NYT-style clarity
  scale_x_date(
    date_labels = "%Y",
    date_breaks = "2 years",
    expand = expansion(mult = c(0.01, 0.01))
  ) +
  # Adjust y-axis to start from 0 for clarity
  scale_y_continuous(
    expand = expansion(mult = c(0.01, 0.01))
  )

```



The fitted  $ARIMA(1,0,0)(1,1,0)[12]$  model effectively captures both the seasonal and non-seasonal patterns in the shooting incidents data from 2006 to 2023. The autoregressive components highlight the influence of past values, with a significant relationship between incidents in consecutive months and a negative seasonal autoregressive effect across years. This suggests that while incidents in a given month are moderately tied to the previous month, there is a slight tendency for year-over-year fluctuations to counterbalance each other.

The model also indicates a slight downward trend in incident counts over time, as captured by the negative drift coefficient. This aligns with observed declines in overall incidents post-2020, potentially reflecting the impact of interventions or socio-environmental changes. However, the residual variance suggests that some variability in the data remains unexplained, indicating potential influences from external factors not accounted for in the model.

Overall, the ARIMA model provides a useful framework for forecasting future incidents. Its ability to identify seasonal surges, particularly during the summer months, reinforces the importance of targeted public safety strategies during these high-risk periods. While the model performs well for short-term forecasting, incorporating additional external variables could further improve its predictive accuracy and provide deeper insights into the factors driving changes in incident patterns.

## Conclusion and Discussion

This analysis reveals clear seasonal, temporal, and spatial patterns in NYC shooting incidents from 2006 to 2023. The ARIMA model effectively captures these trends, highlighting surges during the summer months, particularly in July and August, and the need for targeted public safety interventions during these high-risk periods. Borough-specific patterns, as shown in the heatmap, emphasize the importance of geographically focused strategies, with late-night hours consistently showing the highest activity, especially in the Bronx, Brooklyn, and Manhattan.

A slight downward trend in incidents since 2020 aligns with the potential impact of intervention efforts or broader socio-environmental changes. However, residual variance and occasional outliers, such as the spikes during the pandemic in 2020, suggest the influence of external factors not captured by the model. Future work should focus on refining model assumptions, incorporating additional external variables, and exploring community-level interventions to enhance public safety strategies further.

## Potential Sources of Bias

### 1. Data Bias:

- The dataset is limited to reported incidents, which may underrepresent actual occurrences due to unreported events or inaccuracies in data collection.
- The manual extraction and quarterly review process could introduce inconsistencies in the dataset.

### 2. Model Bias:

- The ARIMA model assumes linear relationships and stationarity, which might oversimplify complex, non-linear interactions between incidents and external factors.
- Seasonal differencing effectively removes seasonality but might obscure subtle recurring patterns.

### 3. Analyst Bias:

- Personal assumptions about the importance of certain factors, such as summer weather or night-time activity, may have influenced the interpretation of results.
- The focus on temporal and spatial patterns potentially overlooked other critical dimensions, such as demographics or policy impacts.

## Mitigation of Bias

### 1. Data Validation

Efforts were made to clean the dataset, replacing null values with NA and correcting anomalous entries to ensure a more accurate analysis.

### 2. Exploration of Patterns

Multiple statistical methods were used, including heatmaps and time series modeling, to triangulate findings and reduce over-reliance on a single perspective.

### 3. Awareness of Assumptions

Recognizing potential blind spots, I avoided drawing causal conclusions and emphasized the limited and exploratory nature of this study.

## Appendix

```
sessionInfo()
```

```
## R version 4.4.2 (2024-10-31)
## Platform: aarch64-apple-darwin20
## Running under: macOS Sonoma 14.6
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.4-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/Los_Angeles
## tzcode source: internal
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods   base
##
## other attached packages:
## [1] forecast_8.23.0 knitr_1.48      lubridate_1.9.3 forcats_1.0.0
## [5] stringr_1.5.1   dplyr_1.1.4     purrr_1.0.2     readr_2.1.5
## [9] tidyr_1.3.1     tibble_3.2.1    ggplot2_3.5.1   tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
## [1] utf8_1.2.4      generics_0.1.3  stringi_1.8.4   lattice_0.22-6
## [5] hms_1.1.3       digest_0.6.37   magrittr_2.0.3  evaluate_1.0.1
## [9] grid_4.4.2      timechange_0.3.0 fastmap_1.2.0    Matrix_1.7-1
## [13] nnet_7.3-19     mgcv_1.9-1      fansi_1.0.6     scales_1.3.0
## [17] cli_3.6.3       crayon_1.5.3    rlang_1.1.4     splines_4.4.2
## [21] bit64_4.5.2     munsell_0.5.1   withr_3.0.2     yaml_2.3.10
## [25] tools_4.4.2     parallel_4.4.2  tzdb_0.4.0      colorspace_2.1-1
## [29] curl_5.2.3      vctrs_0.6.5     R6_2.5.1        zoo_1.8-12
## [33] lifecycle_1.0.4 tseries_0.10-58 bit_4.5.0        vroom_1.6.5
## [37] urca_1.3-4       pkgconfig_2.0.3 pillar_1.9.0     gtable_0.3.6
## [41] quantmod_0.4.26 glue_1.8.0      Rcpp_1.0.13-1    xfun_0.49
## [45] lmtest_0.9-40    tidyselect_1.2.1 rstudioapi_0.17.1 farver_2.1.2
## [49] nlme_3.1-166     htmltools_0.5.8.1 labeling_0.4.3   xts_0.14.1
## [53] rmarkdown_2.28   timeDate_4041.110 fracdiff_1.5-3   compiler_4.4.2
## [57] quadprog_1.5-8   TTR_0.24.4
```

## Months with the highest number of shooting incidents

The table reveals the months with the highest number of shooting incidents from 2006 to 2023, highlighting clear seasonal and temporal trends. Most peak months fall during the summer, particularly in July and August, suggesting a strong seasonal pattern. Warmer weather, increased outdoor activities, or other socio-environmental factors likely contribute to this summer surge in incidents.

```
# Process data to find the month with the most incidents by year
summary_by_month <- nypd_data_clean %>%
  # Extract year and month from the occurrence date
  mutate(
    YEAR = year(OCCUR_DATE),
    MONTH = month(OCCUR_DATE, label = TRUE)
  ) %>%
  # Group by year and month to count incidents
  group_by(YEAR, MONTH) %>%
  summarise(
    INCIDENTS = n(),
    .groups = "drop"
  ) %>%
  # Identify the month with the most incidents per year
  group_by(YEAR) %>%
  slice_max(order_by = INCIDENTS, n = 1) %>%
  ungroup() %>%
  # Summarize by month across all years
  group_by(MONTH) %>%
  summarise(
    YEARS_WITH_HIGHEST_INCIDENTS = n(),
    YEARS = paste(YEAR, collapse = ", "),
    .groups = "drop"
  ) %>%
  # Arrange by the number of years
  arrange(desc(YEARS_WITH_HIGHEST_INCIDENTS))

# Display the results with column names
kable(
  summary_by_month,
  col.names = c("Month", "# of Years", "Years"),
  caption = "Summary of Months with the Most Incidents"
)
```

Table 1: Summary of Months with the Most Incidents

Month	# of Years	Years
Jul	9	2007, 2008, 2009, 2012, 2018, 2019, 2020, 2022, 2023
Aug	5	2006, 2013, 2014, 2016, 2021
Jun	2	2010, 2017
May	1	2015
Sep	1	2011