Exploring NYPD Shooting Incident Data: Statistical Insights and Trends

Anh Mai

2024-10-27

Introduction and Objectives

Introduction

This project, titled "Exploring NYPD Shooting Incident Data: Statistical Insights and Trends," delves into the dynamics of shooting incidents reported by the New York Police Department (NYPD). By analyzing historical data on these incidents, we aim to uncover underlying patterns, trends, and statistical insights that illuminate the factors influencing shooting incidents in New York City. Our goal is to provide a data-driven understanding of how and when these incidents occur, identify significant trends over time, and explore any correlations with broader socio-economic and environmental factors. This analysis seeks to contribute valuable insights for policymakers, law enforcement agencies, and the community to better address and manage the issue of gun violence.

Objectives

1. **Analyze Incident Trends Over Time**: Examine the dataset to identify trends in shooting incidents over the years. Determine whether there is an increase or decrease in incidents and highlight any significant changes or patterns.

2. **Investigate Temporal Patterns**: Analyze shooting incidents by time of day, day of the week, and season to identify peak periods for shootings. Determine if certain times or seasons have higher incident rates.

3. **Examine Geographic Distribution**: Map the geographic distribution of shooting incidents across New York City to identify hotspots. Assess whether certain neighborhoods or areas experience higher frequencies of shootings.

4. **Study Demographic Information**: Analyze the demographics of suspects and victims involved in shooting incidents, including age, gender, and ethnicity. Explore any demographic trends or disparities in the data.

5. **Explore Correlations with Socio-Economic Factors**: Investigate potential correlations between shooting incidents and socio-economic factors such as income levels, unemployment rates, and community characteristics to understand broader social influences.

6. **Evaluate Police Response and Outcomes**: Assess the outcomes of shooting incidents, including police response times and resolution status. Evaluate the effectiveness of current policing strategies and interventions.

7. **Provide Recommendations for Policy and Action**: Based on the analysis, develop actionable recommendations for policymakers and law enforcement agencies aimed at reducing shooting incidents and enhancing public safety.

Through this analysis, we seek to enhance understanding of shooting incidents in New York City and support informed decision-making to address and mitigate gun violence in the community.

Data Source

The data for this analysis is sourced from this website "https://catalog.data.gov/dataset/nypd-shooting-incident-data-historic" Data is CSV file "https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD"

Scope of Analysis

This analysis will cover: The scope of this analysis includes:

1. **Data Range**: The dataset spans from 2006 through the end of the previous calendar year. Analysis will be limited to incidents reported within this time frame.
2. **Data Variables**: Focus will be on variables related to the timing, location, and demographic details of shooting incidents. This includes date, time, location, suspect demographics, and victim demographics.
3. **Geographic Focus**: Analysis will be concentrated on geographic patterns within New York City, with specific emphasis on identifying areas with high frequencies of shooting incidents.
4. **Temporal Analysis**: The study will assess temporal patterns such as time of day, day of the week, and seasonal variations in shooting incidents.
5. **Demographic Analysis**: Examination will include demographic details of suspects and victims to identify any notable trends or disparities.
6. **Socio-Economic Correlations**: The analysis will explore correlations between shooting incidents and socio-economic factors available in the dataset or inferred from external sources.
7. **Exclusions**: This analysis will not cover non-shooting crime incidents, historical context beyond the dataset range, or qualitative aspects not captured in the dataset.

By defining the scope clearly, this analysis aims to provide a focused and comprehensive understanding of shooting incidents in New York City, offering actionable insights while acknowledging the limitations and boundaries of the study.

Data Wrangling

Data Loading

```r
# Set CRAN mirror
options(repos = c(CRAN = "https://cran.rstudio.com/"))

# Load necessary libraries
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   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```r
library(knitr)
library(dplyr)
library(lubridate)
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
library(sf)
```

```
## Linking to GEOS 3.12.1, GDAL 3.8.4, PROJ 9.3.1; sf_use_s2() is TRUE
```

```r
# Set global options
opts_chunk$set(echo = TRUE)

# Load your dataset
url_in <- "https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD"

# Read data/load data into tables
raw_incidents <- read_csv(url_in[1])
```

```
## Rows: 28562 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (12): OCCUR_DATE, BORO, LOC_OF_OCCUR_DESC, LOC_CLASSFCTN_DESC, LOCATION...
## dbl   (7): INCIDENT_KEY, PRECINCT, JURISDICTION_CODE, X_COORD_CD, Y_COORD_CD...
## lgl   (1): STATISTICAL_MURDER_FLAG
## time  (1): OCCUR_TIME
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

Data Cleanup

```r
# Display the first few rows of the dataset
head(raw_incidents)
```

```
## # A tibble: 6 x 21
##   INCIDENT_KEY OCCUR_DATE OCCUR_TIME BORO      LOC_OF_OCCUR_DESC PRECINCT
##          <dbl> <chr>      <time>     <chr>     <chr>                <dbl>
## 1    244608249 05/05/2022 00:10      MANHATTAN INSIDE                  14
## 2    247542571 07/04/2022 22:20      BRONX     OUTSIDE                 48
## 3     84967535 05/27/2012 19:35      QUEENS    <NA>                   103
## 4    202853370 09/24/2019 21:00      BRONX     <NA>                    42
## 5     27078636 02/25/2007 21:00      BROOKLYN  <NA>                    83
## 6    230311078 07/01/2021 23:07      MANHATTAN <NA>                    23
## # i 15 more variables: JURISDICTION_CODE <dbl>, LOC_CLASSFCTN_DESC <chr>,
## #   LOCATION_DESC <chr>, STATISTICAL_MURDER_FLAG <lgl>, PERP_AGE_GROUP <chr>,
## #   PERP_SEX <chr>, PERP_RACE <chr>, VIC_AGE_GROUP <chr>, VIC_SEX <chr>,
## #   VIC_RACE <chr>, X_COORD_CD <dbl>, Y_COORD_CD <dbl>, Latitude <dbl>,
## #   Longitude <dbl>, Lon_Lat <chr>
```

```
summary(raw_incidents)
```

```
##   INCIDENT_KEY        OCCUR_DATE         OCCUR_TIME           BORO          
##  Min.   :  9953245   Length:28562       Length:28562      Length:28562      
##  1st Qu.: 65439914   Class :character   Class1:hms        Class :character  
##  Median : 92711254   Mode  :character   Class2:difftime   Mode  :character  
##  Mean   :127405824                      Mode  :numeric                      
##  3rd Qu.:203131993                                                          
##  Max.   :279758069                                                          
##                                                                             
##  LOC_OF_OCCUR_DESC     PRECINCT     JURISDICTION_CODE LOC_CLASSFCTN_DESC
##  Length:28562       Min.   :  1.0   Min.   :0.0000    Length:28562      
##  Class :character   1st Qu.: 44.0   1st Qu.:0.0000    Class :character  
##  Mode  :character   Median : 67.0   Median :0.0000    Mode  :character  
##                     Mean   : 65.5   Mean   :0.3219                      
##                     3rd Qu.: 81.0   3rd Qu.:0.0000                      
##                     Max.   :123.0   Max.   :2.0000                      
##                                     NA’s   :2                           
##  LOCATION_DESC      STATISTICAL_MURDER_FLAG PERP_AGE_GROUP    
##  Length:28562       Mode :logical           Length:28562      
##  Class :character   FALSE:23036             Class :character  
##  Mode  :character   TRUE :5526              Mode  :character  
##                                                               
##                                                               
##                                                               
##                                                               
##    PERP_SEX          PERP_RACE         VIC_AGE_GROUP        VIC_SEX         
##  Length:28562       Length:28562       Length:28562       Length:28562      
##  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:28562       Min.   : 914928   Min.   :125757   Min.   :40.51  
##  Class :character   1st Qu.:1000068   1st Qu.:182912   1st Qu.:40.67  
##  Mode  :character   Median :1007772   Median :194901   Median :40.70  
##                     Mean   :1009424   Mean   :208380   Mean   :40.74  
##                      3rd Qu.:1016807   3rd Qu.:239814   3rd Qu.:40.82
##                      Max.   :1066815   Max.   :271128   Max.   :40.91
##                                                         NA's   :59
##     Longitude         Lon_Lat
##  Min.   :-74.25   Length:28562
##  1st Qu.:-73.94   Class :character
##  Median :-73.92   Mode  :character
##  Mean   :-73.91
##  3rd Qu.:-73.88
##  Max.   :-73.70
##  NA's   :59
```

```
# Load data into INCIDENTS
incidents <- raw_incidents %>%
  rename(BOROUGH = `BORO`) %>%
  select (INCIDENT_KEY:VIC_RACE)

# Convert Date column to Date type
incidents$OCCUR_DATE <- mdy(incidents$OCCUR_DATE)

# Combine the date and time columns into one datetime column
incidents$OCCUR_DATETIME <- paste(incidents$OCCUR_DATE, incidents$OCCUR_TIME)

# Convert the new datetime column to POSIXct
incidents$OCCUR_DATETIME <- as.POSIXct(incidents$OCCUR_DATETIME, format = "%Y-%m-%d %H:%M:%S")

# Remove duplicates
incidents <- distinct(incidents)

#Verify OCCUR_DATE
str(incidents$OCCUR_DATE)
```

```
##  Date[1:28562], format: "2022-05-05" "2022-07-04" "2012-05-27" "2019-09-24" "2007-02-25" ...
```

```
class(incidents$OCCUR_DATE)
```

```
## [1] "Date"
```

```
# Add code for borough into incidents_by_borough

# Create a lookup table for borough codes and names
borough_lookup <- data.frame(
  Borough_Code = c(1, 2, 3, 4, 5),
  Borough = c("Manhattan", "Bronx", "Brooklyn", "Queens", "Staten Island")
)
```

Data Analysis

Summary

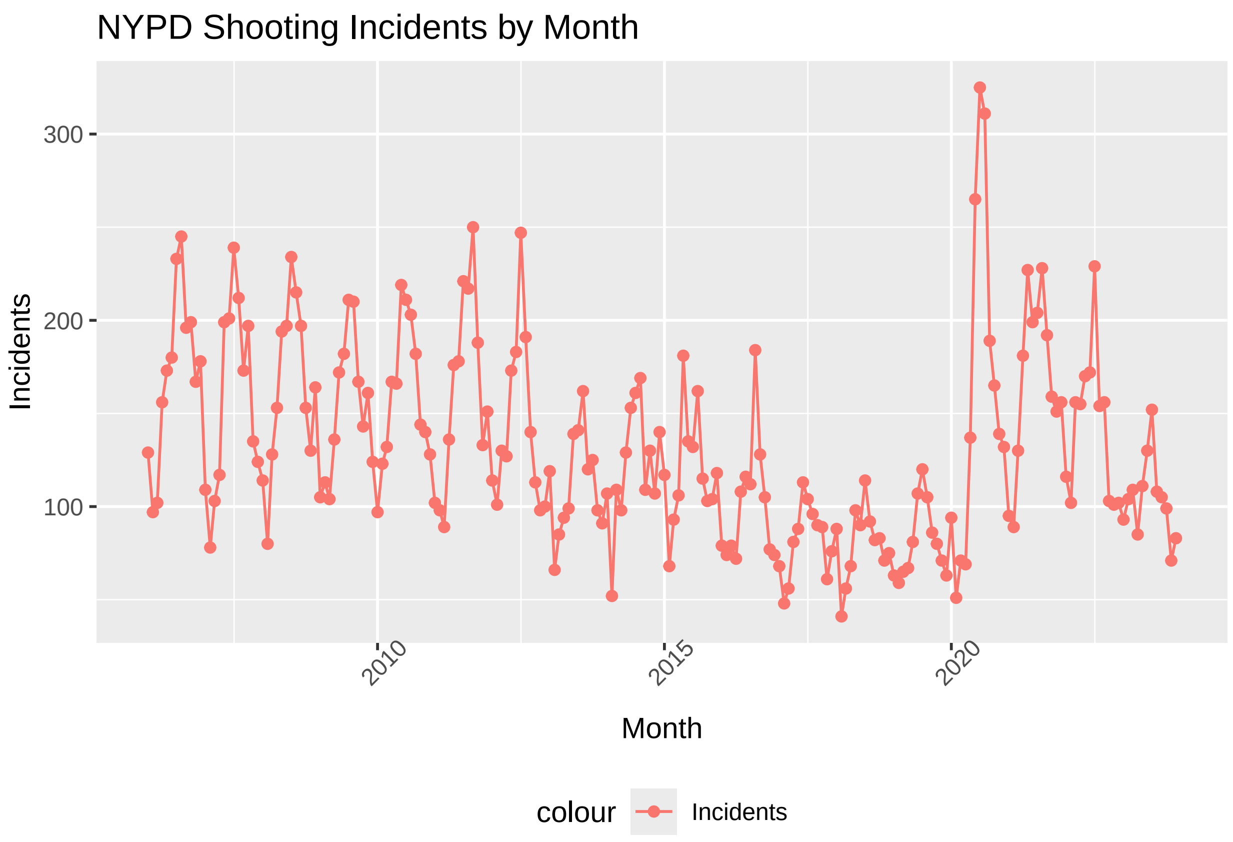
```r
# Summarize the dataset
summary(incidents)
```

```
##   INCIDENT_KEY          OCCUR_DATE          OCCUR_TIME         BOROUGH
##  Min.   :  9953245   Min.   :2006-01-01   Length:28562      Length:28562
##  1st Qu.: 65439914   1st Qu.:2009-09-04   Class1:hms        Class :character
##  Median : 92711254   Median :2013-09-20   Class2:difftime   Mode  :character
##  Mean   :127405824   Mean   :2014-06-07   Mode  :numeric
##  3rd Qu.:203131993   3rd Qu.:2019-09-29
##  Max.   :279758069   Max.   :2023-12-29
##
##  LOC_OF_OCCUR_DESC     PRECINCT    JURISDICTION_CODE LOC_CLASSFCTN_DESC
##  Length:28562       Min.   :  1.0   Min.   :0.0000    Length:28562
##  Class :character   1st Qu.: 44.0   1st Qu.:0.0000    Class :character
##  Mode  :character   Median : 67.0   Median :0.0000    Mode  :character
##                     Mean   : 65.5   Mean   :0.3219
##                     3rd Qu.: 81.0   3rd Qu.:0.0000
##                     Max.   :123.0   Max.   :2.0000
##                                     NA’s   :2
##  LOCATION_DESC      STATISTICAL_MURDER_FLAG PERP_AGE_GROUP
##  Length:28562       Mode :logical           Length:28562
##  Class :character   FALSE:23036             Class :character
##  Mode  :character   TRUE :5526              Mode  :character
##
##
##
##
##    PERP_SEX          PERP_RACE         VIC_AGE_GROUP        VIC_SEX
##  Length:28562       Length:28562       Length:28562       Length:28562
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    VIC_RACE         OCCUR_DATETIME
##  Length:28562       Min.   :2006-01-01 02:00:00.00
##  Class :character   1st Qu.:2009-09-04 07:15:00.00
##  Mode  :character   Median :2013-09-20 17:56:00.00
##                     Mean   :2014-06-07 20:04:22.43
##                     3rd Qu.:2019-09-30 10:10:30.00
##                     Max.   :2023-12-29 21:22:00.00
##
```

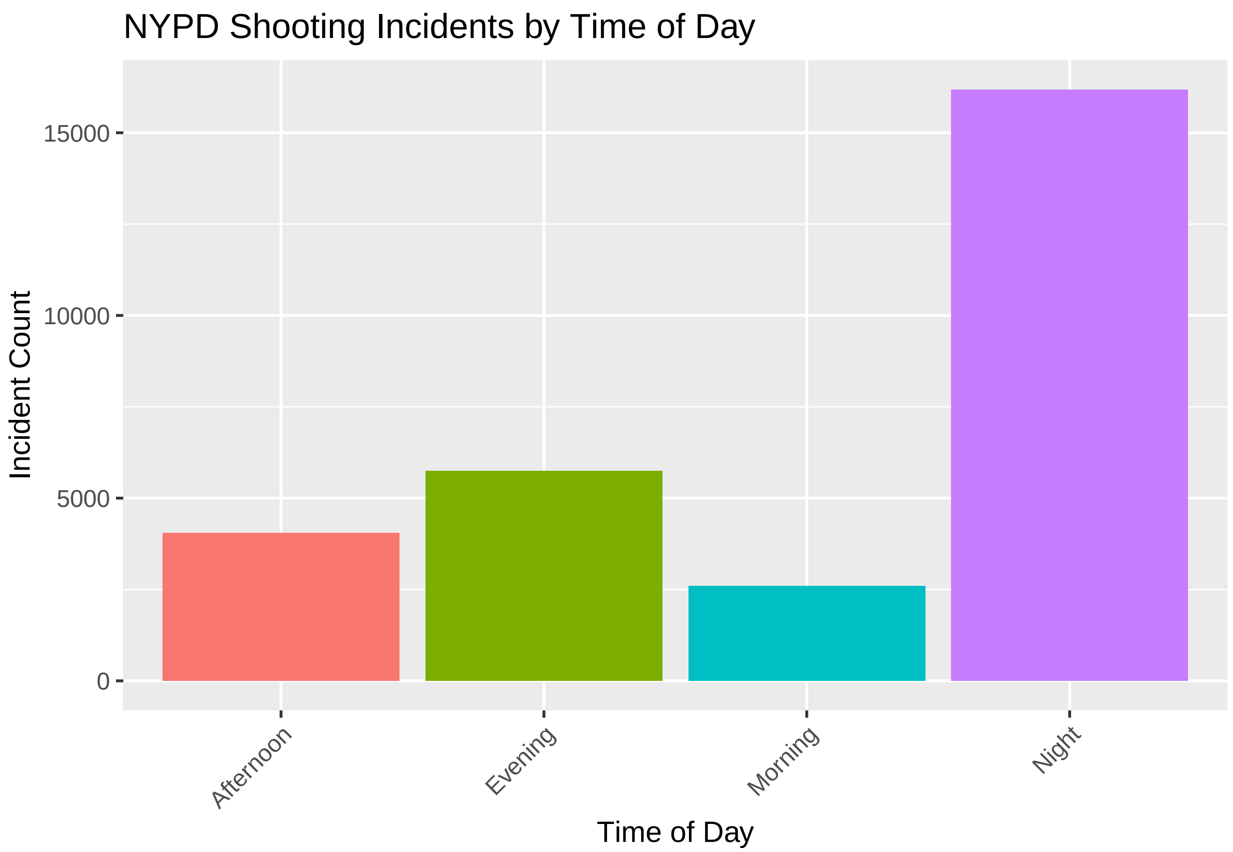
```r
# Trend Analysis
incidents_by_month <- incidents %>%
  group_by(Month = floor_date(OCCUR_DATE, unit = "month")) %>%
  summarize(Incident_Count = n()) %>%
  ungroup()

# Temporal Patterns
# Time of Day
incidents_by_time_of_day <- incidents %>%
  mutate(Hour = hour(OCCUR_DATETIME)) %>%
  mutate(Time_of_Day = case_when(
    Hour >= 5 & Hour < 12 ~ "Morning",
    Hour >= 12 & Hour < 17 ~ "Afternoon",
    Hour >= 17 & Hour < 21 ~ "Evening",
    TRUE ~ "Night"  # Covers from 21:00 to 04:59
  )) %>%
  group_by(Time_of_Day) %>%
  summarize(Incident_Count = n()) %>%
  ungroup()

# Incidents by Hour
incidents_by_hour <- incidents %>%
  group_by(Hour = hour(OCCUR_DATETIME)) %>%
  summarize(Incident_Count = n()) %>%
  ungroup() %>%
  arrange(desc(Incident_Count))

# Geographic Analysis
# Incidents by Borough
incidents_by_borough <- incidents %>%
  group_by(BOROUGH) %>%
  summarize(Incident_Count = n()) %>%
  ungroup() %>%
  arrange(desc(Incident_Count)) %>%
  mutate(Percentage = Incident_Count / sum(Incident_Count) * 100)

incidents_by_borough$BOROUGH <- str_to_title(incidents_by_borough$BOROUGH)

incidents_by_borough <- inner_join(incidents_by_borough, borough_lookup, by = c("BOROUGH" = "Borough"))

incidents_by_borough <- incidents_by_borough %>% select (Borough_Code, everything())

# Pattern Analysis
# Incidents by Time of Day and Murder Flag
incidents_by_time_murder <- incidents %>%
  mutate(Hour = hour(OCCUR_DATETIME)) %>%  # Extract hour from OCCUR_DATETIME
  mutate(Time_of_Day = case_when(
    Hour >= 5 & Hour < 12 ~ "Morning",
    Hour >= 12 & Hour < 17 ~ "Afternoon",
    Hour >= 17 & Hour < 21 ~ "Evening",
    TRUE ~ "Night"  # Covers from 21:00 to 04:59
  )) %>%
  group_by(Time_of_Day, STATISTICAL_MURDER_FLAG) %>%
  summarize(Incident_Count = n(), .groups = 'drop')   %>%
  rename (Murder_Flag = `STATISTICAL_MURDER_FLAG`)
```

Visualization

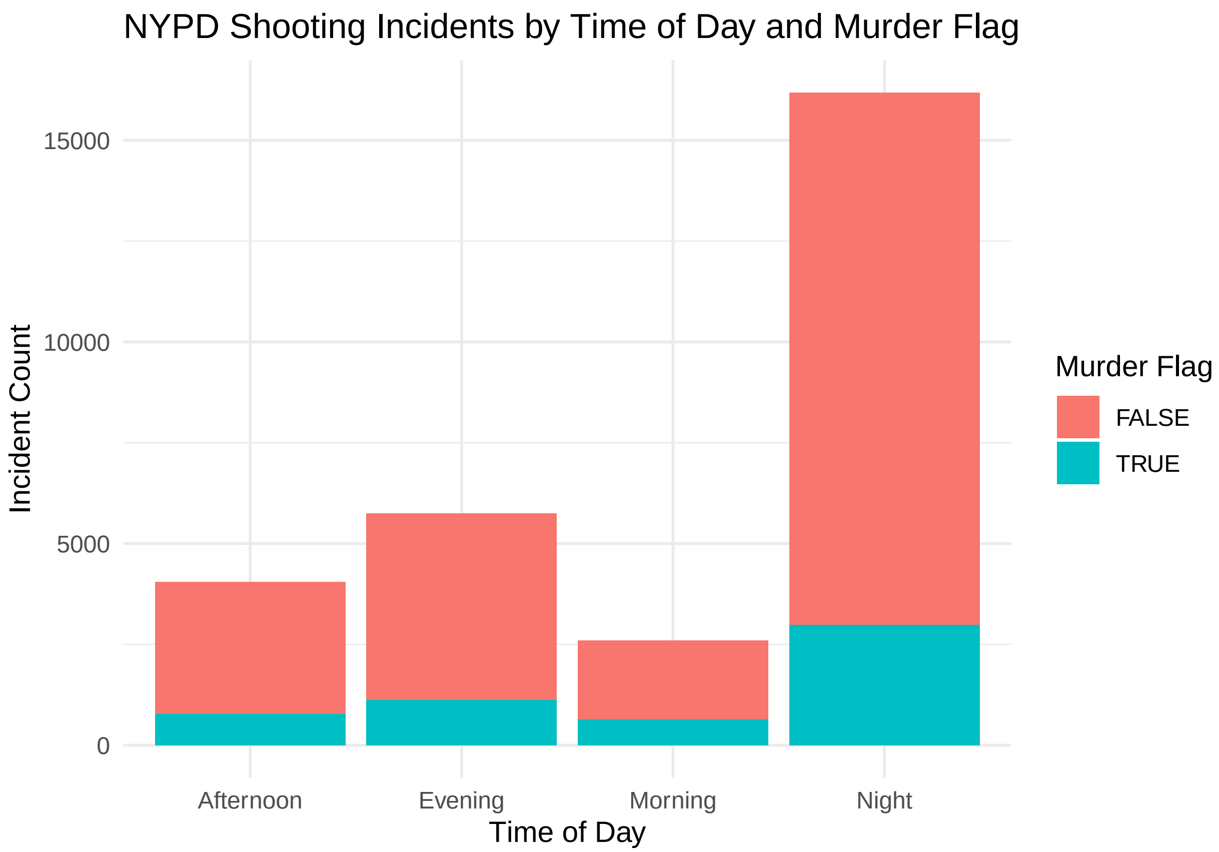
```
# Plot: Incidents by Month
incidents_by_month %>%
  ggplot(aes(x = Month)) +
  geom_line(aes(y = Incident_Count, color = "Incidents")) +
  geom_point(aes(y = Incident_Count, color = "Incidents")) +
  theme(legend.position = "bottom", axis.text.x = element_text(angle = 45)) +
  labs(title = "NYPD Shooting Incidents by Month", y = "Incidents")
```



```
# Plot: Incidents by Time of Day
incidents_by_time_of_day %>%
  ggplot(aes(x = Time_of_Day, y = Incident_Count, fill = Time_of_Day)) +  # Use fill to color bars by T
  geom_col() +  # Use geom_col() for a column chart
  theme(legend.position = "none",  # Remove the legend since fill is same as x-axis
        axis.text.x = element_text(angle = 45, hjust = 1)) +  # Adjust text angle
  labs(title = "NYPD Shooting Incidents by Time of Day",
       x = "Time of Day",  # Add x-axis label
       y = "Incident Count")  # Correct y-axis label
```



```
# Plot: Incidents by Time of Day and Murder Flag
# Create a stacked bar chart
ggplot(incidents_by_time_murder, aes(x = Time_of_Day, y = Incident_Count, fill = Murder_Flag)) +
  geom_bar(stat = "identity") +  # Use identity to stack bars according to Incident_Count
  theme_minimal() +  # Use a minimal theme for a clean look
  labs(title = "NYPD Shooting Incidents by Time of Day and Murder Flag",
       x = "Time of Day",
       y = "Incident Count",
       fill = "Murder Flag")  # Label the fill legend
```

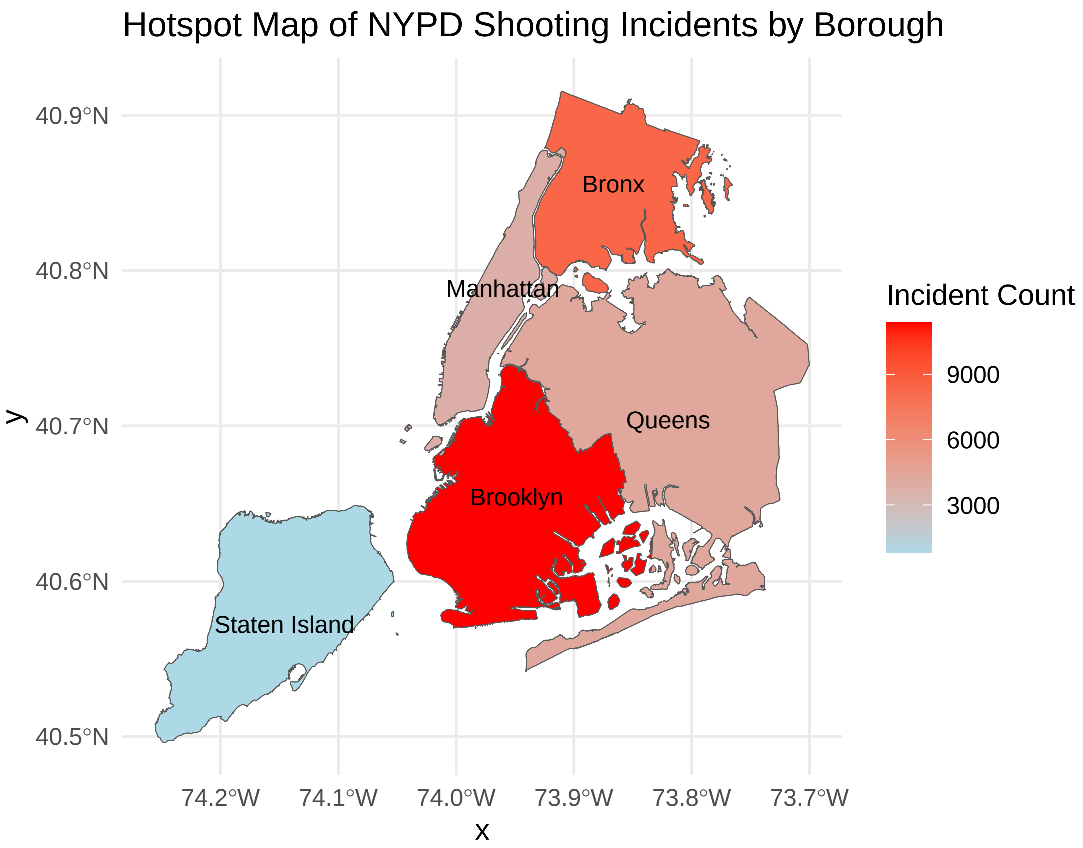


```r
# Plot: Incident Hotspot in New York map
# Load New York City borough shapefile
nyc_boroughs <- st_read("https://raw.githubusercontent.com/anhpmai/CU_MSDS/main/DTSA-5301/Assignments/Da
```

```
## Reading layer ‘new-york-city-boroughs’ from data source
##   ‘https://raw.githubusercontent.com/anhpmai/CU_MSDS/main/DTSA-5301/Assignments/Data/new-york-city-bo
##   using driver ‘GeoJSON’
## Simple feature collection with 5 features and 4 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -74.25559 ymin: 40.49612 xmax: -73.70001 ymax: 40.91553
## Geodetic CRS:  WGS 84
```

```r
# Merge incidents data with borough shapefile
nyc_boroughs <- nyc_boroughs %>%
  left_join(incidents_by_borough, by = c("name" = "BOROUGH"))

# Create the hotspot map
ggplot(data = nyc_boroughs) +
  geom_sf(aes(fill = Incident_Count)) +
  geom_sf_text(aes(label = name), size = 3, color = "black") +
  scale_fill_gradient(low = "lightblue", high = "red", na.value = "white") +
  theme_minimal() +
  labs(title = "Hotspot Map of NYPD Shooting Incidents by Borough",
       fill = "Incident Count")
```

```
## Warning in st_point_on_surface.sfc(sf::st_zm(x)): st_point_on_surface may not
## give correct results for longitude/latitude data
```



Data Modeling

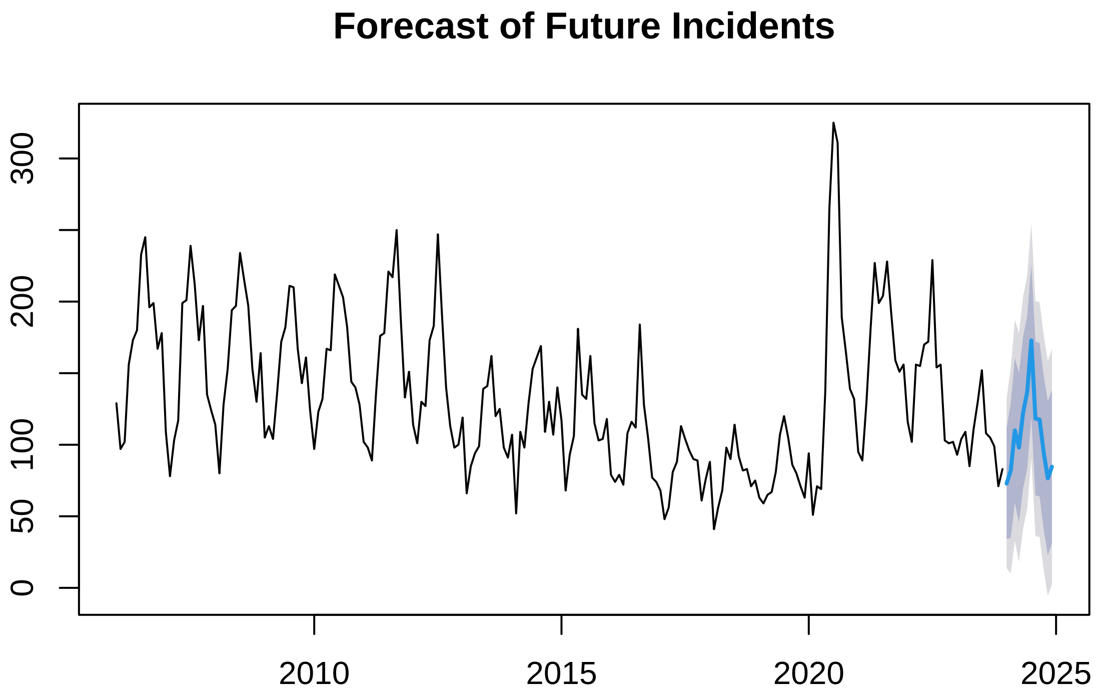
Modeling

```r
# Using ARIMA model to predict future shooting incidents based on historical data

# Example of aggregating data by month
monthly_incidents <- incidents %>%
  mutate(Month = floor_date(OCCUR_DATETIME, "month")) %>%
  group_by(Month) %>%
  summarize(Incident_Count = n()) %>%
  ungroup()

# Convert to a time series object
incident_ts <- ts(monthly_incidents$Incident_Count, start = c(year(min(monthly_incidents$Month)), month

# Fit the ARIMA model
fit <- auto.arima(incident_ts)
```

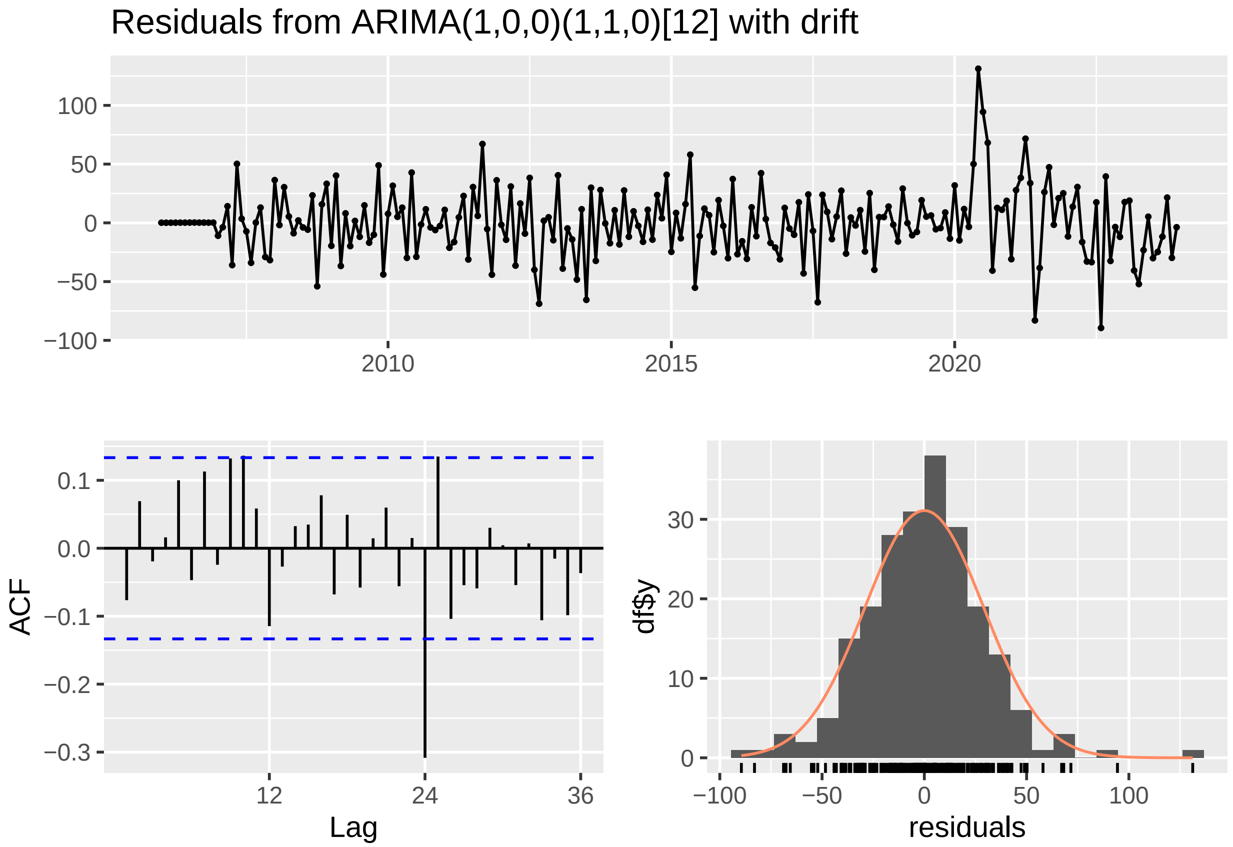
```
# Forecast future incidents
forecasted_incidents <- forecast(fit, h = 12)
```

Model Visualization

```
# Plot the forecast
plot(forecasted_incidents, main = "Forecast of Future Incidents")
```



```
# Check residuals
checkresiduals(fit)
```



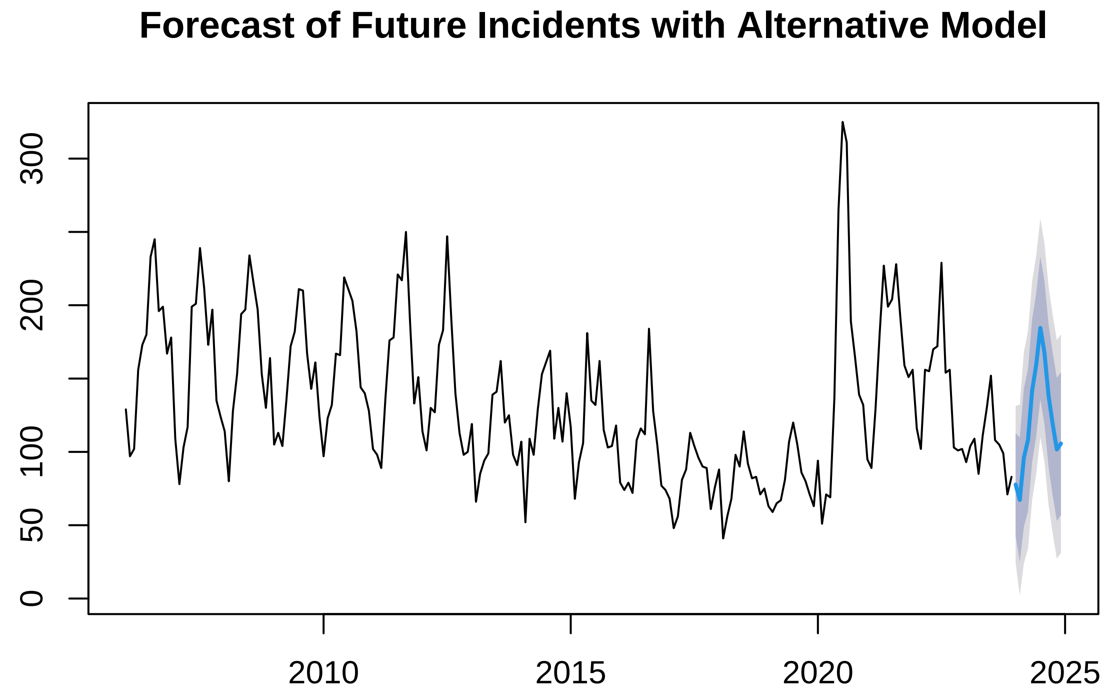
```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,0,0)(1,1,0)[12] with drift
## Q* = 49.883, df = 22, p-value = 0.0006081
##
## Model df: 2.   Total lags used: 24
```

Alternative Modeling

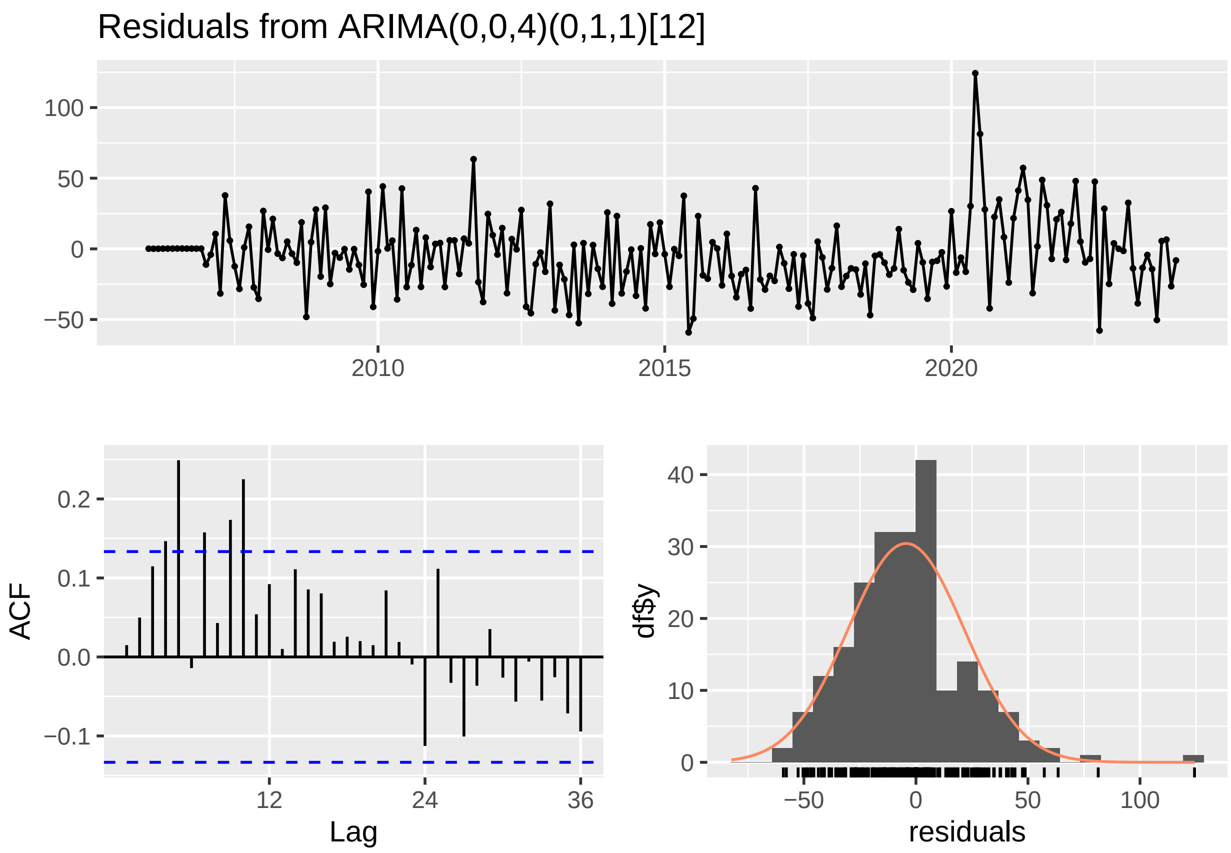
```
# Try fitting an ARIMA model with different parameters
fit_alternative <- auto.arima(incident_ts, seasonal = TRUE, stepwise = FALSE, approximation = FALSE)

# Forecast future incidents
forecasted_incidents_alt <- forecast(fit_alternative, h = 12)
```

Model Visualization

```
# Plot the forecast
plot(forecasted_incidents_alt, main = "Forecast of Future Incidents with Alternative Model")
```



```r
# Check the residuals of the new model
checkresiduals(fit_alternative)
```



```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,0,4)(0,1,1)[12]
## Q* = 60.697, df = 19, p-value = 3.001e-06
##
## Model df: 5.   Total lags used: 24
```

Results

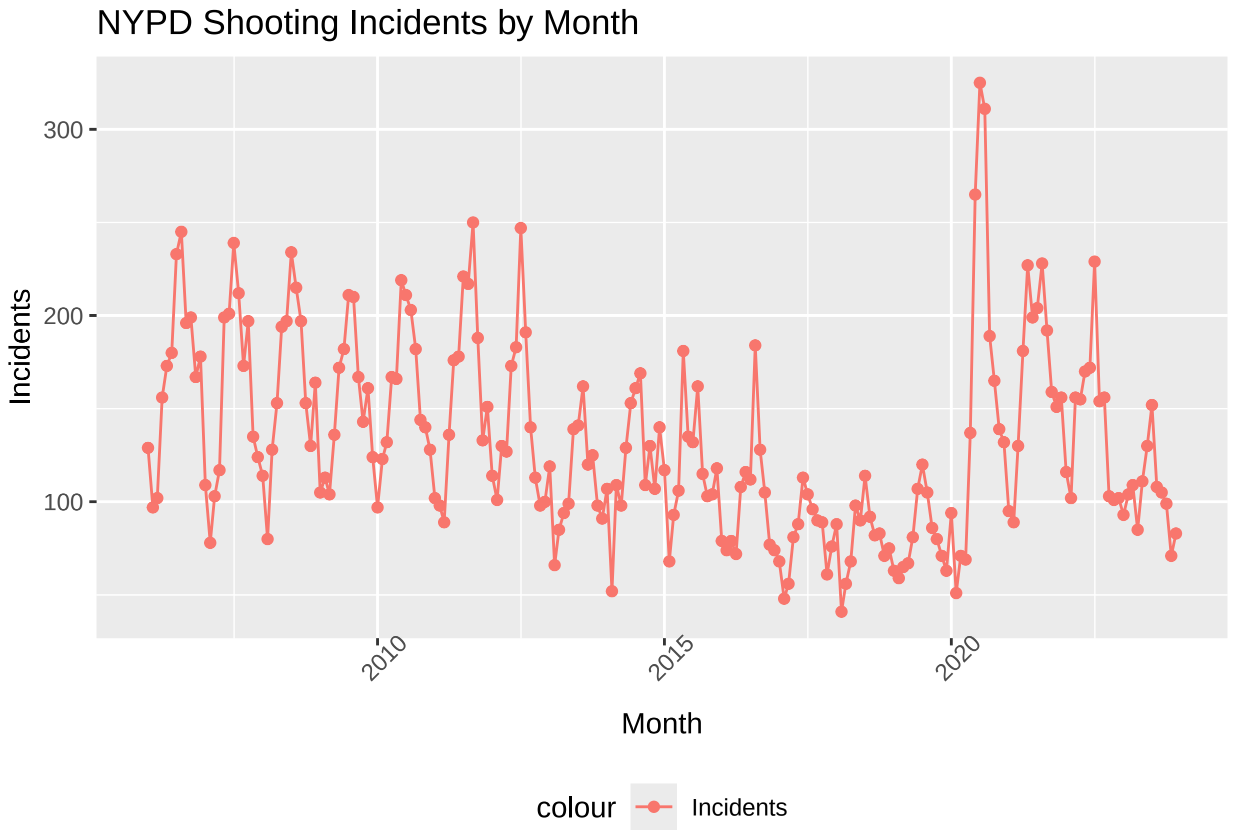
Tables and Figures

```
# Create a table of results
kable(head(incidents_by_month))
```

Month	Incident_Count
2006-01-01	129
2006-02-01	97
2006-03-01	102
2006-04-01	156
2006-05-01	173

Month	Incident_Count
2006-06-01	180

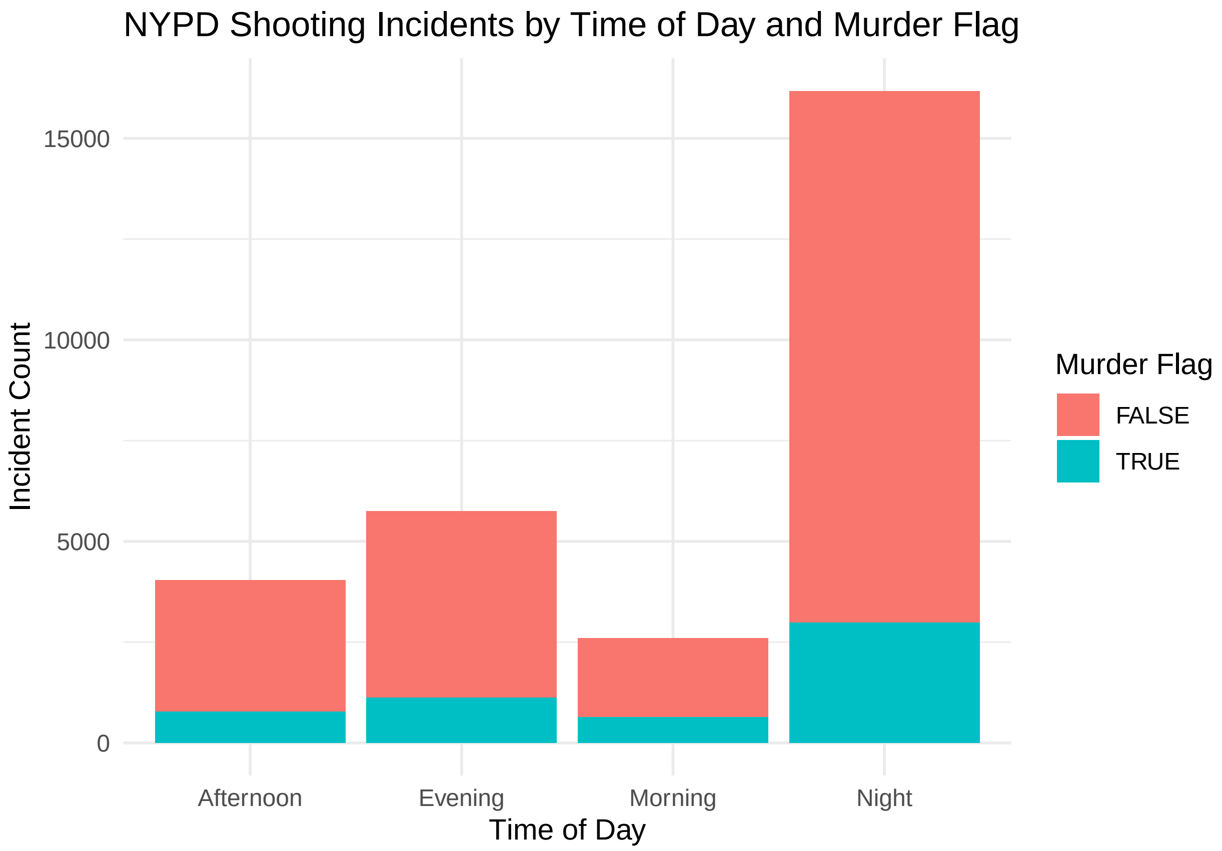
```
# Incidents by Month
incidents_by_month %>%
  ggplot(aes(x = Month)) +
  geom_line(aes(y = Incident_Count, color = "Incidents")) +
  geom_point(aes(y = Incident_Count, color = "Incidents")) +
  theme(legend.position = "bottom", axis.text.x = element_text(angle = 45)) +
  labs(title = "NYPD Shooting Incidents by Month", y = "Incidents")
```



```
kable(head(incidents_by_time_murder))
```

Time_of_Day	Murder_Flag	Incident_Count
Afternoon	FALSE	3269
Afternoon	TRUE	775
Evening	FALSE	4621
Evening	TRUE	1126
Morning	FALSE	1954
Morning	TRUE	642

```r
# Incidents by Time of Day and Murder Flag
ggplot(incidents_by_time_murder, aes(x = Time_of_Day, y = Incident_Count, fill = Murder_Flag)) +
  geom_bar(stat = "identity") +
  theme_minimal() +
  labs(title = "NYPD Shooting Incidents by Time of Day and Murder Flag",
       x = "Time of Day",
       y = "Incident Count",
       fill = "Murder Flag")
```



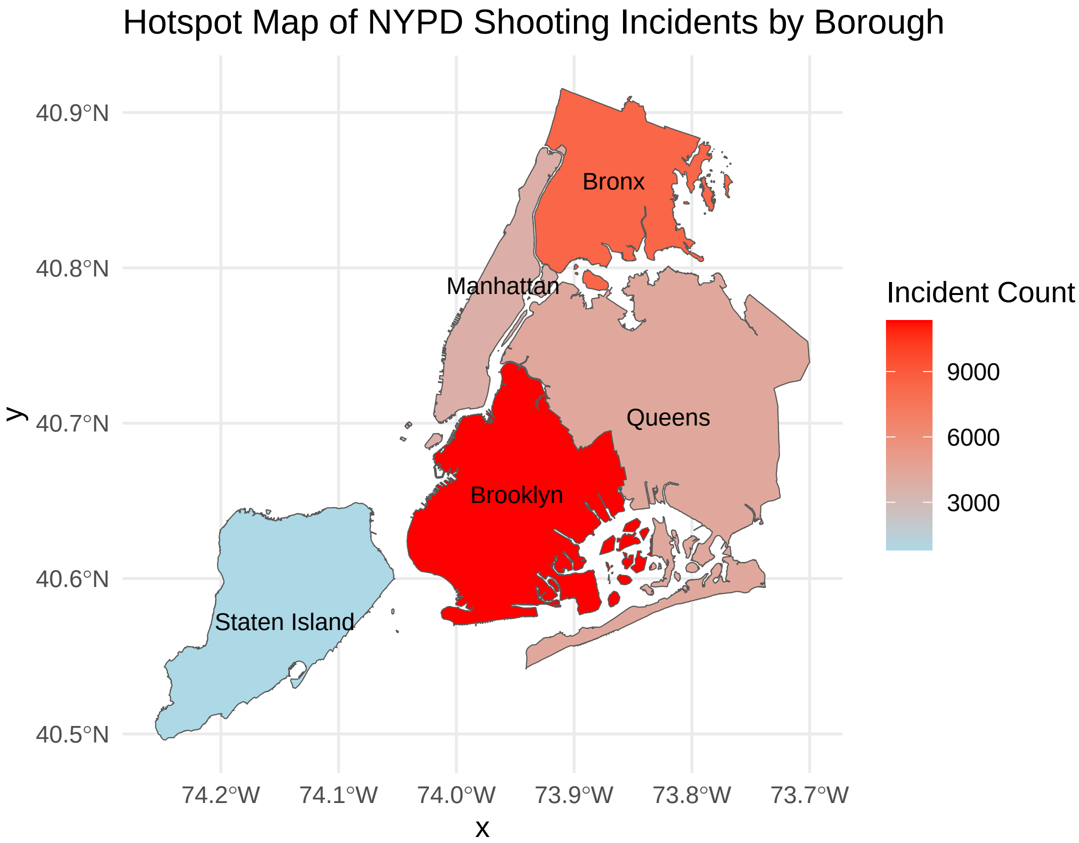
```r
kable(head(incidents_by_borough), caption = "Incidents by Borough")
```

Table 3: Incidents by Borough

Borough_Code	BOROUGH	Incident_Count	Percentage
3	Brooklyn	11346	39.724109
2	Bronx	8376	29.325678
4	Queens	4271	14.953435
1	Manhattan	3762	13.171347
5	Staten Island	807	2.825432

```r
# Incidents by Borough
ggplot(data = nyc_boroughs) +
  geom_sf(aes(fill = Incident_Count)) +
  geom_sf_text(aes(label = name), size = 3, color = "black") +
  scale_fill_gradient(low = "lightblue", high = "red", na.value = "white") +
  theme_minimal() +
  labs(title = "Hotspot Map of NYPD Shooting Incidents by Borough",
       fill = "Incident Count")
```

```
## Warning in st_point_on_surface.sfc(sf::st_zm(x)): st_point_on_surface may not
## give correct results for longitude/latitude data
```



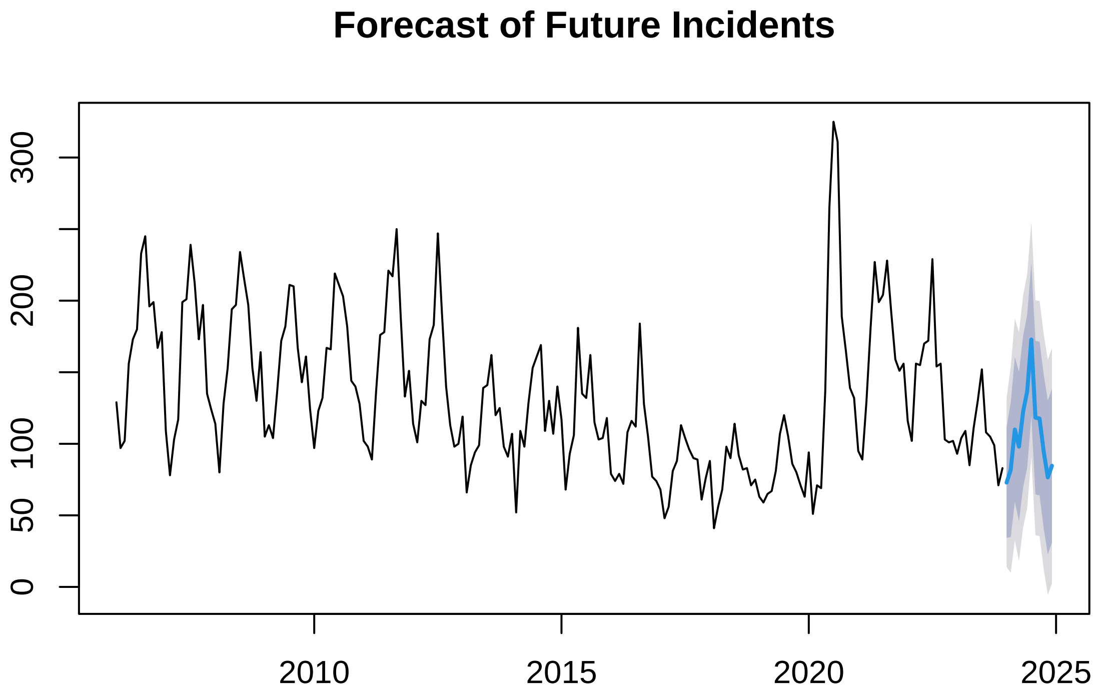
```
# Prepare the forecast data for display on FIT
# Extracting the start date and frequency from the time series object

# Display the table using kable
kable(forecasted_incidents, caption = "ARIMA Forecast Results", col.names = c("Month", "Forecast", "Lowe
```

Table 4: ARIMA Forecast Results

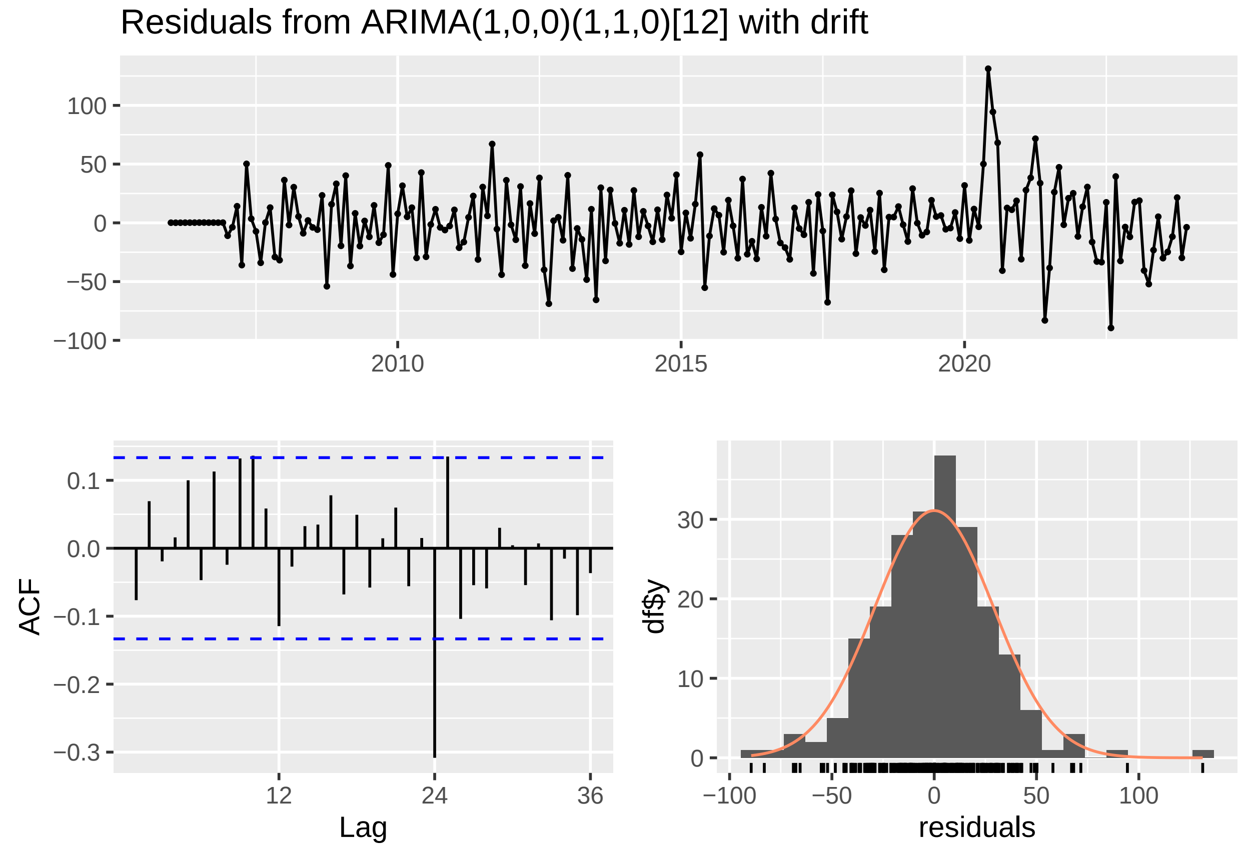
Month	Forecast	Lower 80%	Upper 80%	Lower 95%	Upper 95%
Jan 2024	72.88312	34.28915	111.4771	13.858744	131.9075
Feb 2024	81.83495	34.81989	128.8500	9.931618	153.7383
Mar 2024	109.99223	59.40225	160.5822	32.621539	187.3629
Apr 2024	97.99690	45.76445	150.2294	18.114259	177.8795
May 2024	122.35128	69.34213	175.3604	41.280782	203.4218

Month	Forecast	Lower 80%	Upper 80%	Lower 95%	Upper 95%
Jun 2024	136.74554	83.36453	190.1266	55.106327	218.3848
Jul 2024	172.87923	119.31916	226.4393	90.966175	254.7923
Aug 2024	118.21861	64.57210	171.8651	36.173345	200.2639
Sep 2024	117.63848	63.95017	171.3268	35.529297	199.7477
Oct 2024	94.63207	40.92354	148.3406	12.491964	176.7722
Nov 2024	76.53390	22.81559	130.2522	-5.621163	158.6890
Dec 2024	84.65255	30.92951	138.3756	2.490250	166.8149

```r
# Plot the forecast
plot(forecasted_incidents, main = "Forecast of Future Incidents")
```



```r
# Check residuals
checkresiduals(fit)
```



```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,0,0)(1,1,0)[12] with drift
## Q* = 49.883, df = 22, p-value = 0.0006081
##
## Model df: 2.   Total lags used: 24
```

Conclusion and Bias

In this analysis of the NYPD Shooting Incidents data, several key aspects were examined, including the distribution of incidents over time, by borough, and by time of day, along with the relevance of the Murder Flag. The following analyses were conducted:

1. **Incidents by Month:** The analysis revealed that the highest number of incidents occurred in the summer months, particularly in June, July, and August of 2020. This period showed a significant spike in incidents, suggesting a potential seasonal trend.
2. **Incidents by Borough:**
   Brooklyn emerged as the borough with the highest number of incidents, followed by the Bronx. In contrast, Staten Island recorded the lowest number of incidents. These findings highlight geographic disparities in the occurrence of shooting incidents across New York City.
3. **Incidents by Time of Day and Murder Flag:** The data showed that incidents are most frequent at night, with evening being the second most common time of day for shootings. Despite the high

number of incidents during these times, none were marked with the “Murder” flag. This indicates that the Murder Flag is not significantly related to the overall incident count and may not be a critical factor in understanding the frequency of shooting incidents.

Findings

Seasonal Peak: The highest number of incidents occurred in June, July, and August of 2020.

Geographic Disparities: Brooklyn had the highest incident count, followed by the Bronx, while Staten Island had the lowest.

Time of Day Influence: Nighttime had the highest incident count, followed by evening, but these incidents were not associated with the “Murder” flag, indicating that this flag is not a relevant predictor for incident frequency.

Predictive Modeling

The ARIMA model was selected and applied to the time series data to forecast future incidents. This model was found to be useful for predicting trends based on historical data, making it a valuable tool for anticipating and potentially mitigating future incidents.

Conclusion

The analysis provides insights into the temporal, geographic, and situational characteristics of shooting incidents in New York City. The ARIMA model, in particular, offers a robust method for forecasting future incidents, which could be crucial for law enforcement and policy-making decisions aimed at reducing shooting incidents. The data highlights the need for targeted interventions in specific boroughs and times of day, while also suggesting that the Murder Flag may not be a critical factor in understanding or predicting overall incident counts.

Sources of Bias

Several potential sources of bias may have influenced the results of this project:

Reporting Bias: - Underreporting or Overreporting: Not all shooting incidents may be reported or recorded accurately by law enforcement, leading to underreporting. Conversely, certain types of incidents might be more rigorously reported, leading to overrepresentation in the data. - Discretionary Reporting: Police officers may exercise discretion in reporting incidents, which could vary by borough, time of day, or other factors, leading to inconsistencies.

Geographic Bias: - Resource Allocation: Differences in police presence and resources across boroughs could influence the number of reported incidents. Boroughs with higher police presence might have more incidents recorded simply due to greater surveillance. - Population Density: Higher population densities in certain boroughs might naturally lead to more incidents, not necessarily because they are more dangerous but because more people live and interact in those areas.

Temporal Bias: - Seasonal Trends: The data shows a spike in incidents during the summer months, which could be influenced by factors such as weather, public events, or seasonal activities, rather than underlying trends in violence. - Year-Specific Anomalies: The year 2020 was marked by unique social, economic, and political factors (e.g., the COVID-19 pandemic, social unrest) that may have influenced crime rates. These factors might not be present in other years, limiting the generalizability of the findings.

Data Quality and Completeness: - Missing Data: Incomplete or missing data points can bias the analysis, especially if the missing data is not randomly distributed across the dataset. - Data Entry Errors: Mistakes in data entry or inconsistencies in how data is recorded can introduce bias into the analysis.

Temporal Scope Bias: - Short-Term Data: Analyzing data from only a specific year or short period might lead to conclusions that are not applicable over longer time frames, potentially missing longer-term trends or cyclical patterns.

Mitigating Bias

To mitigate these biases, it is crucial to: - Use multiple years of data to capture broader trends. - Cross-validate findings with different data sources or models. - Consider socioeconomic and demographic factors when interpreting results. - Be transparent about the limitations of the analysis and the potential impact of biases on the conclusions.

End of document.