New York Shooting Incidents, When and Where

Yong Huang

2024-09-29

Contents

1. Introduction

The project will focus on the timing and locations of shooting incidents. It will investigate the specific times, months, and boroughs most affected by shootings and examine the patterns of these occurrences.

2. Summary

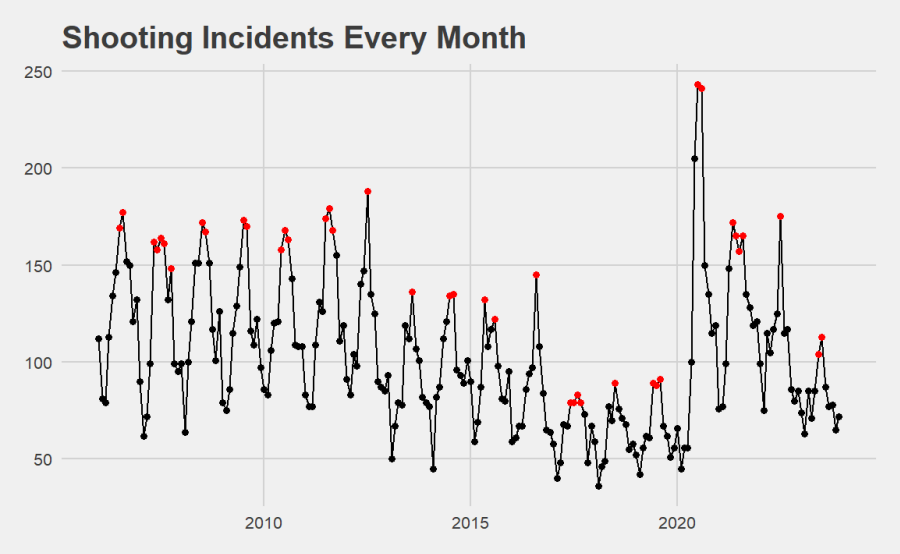
Possibly Bias

Bias can come from the data collection process, which only records shooting incidents with injured victims. It implies that there could be additional unrecorded shootings, casting uncertainty on how these incidents

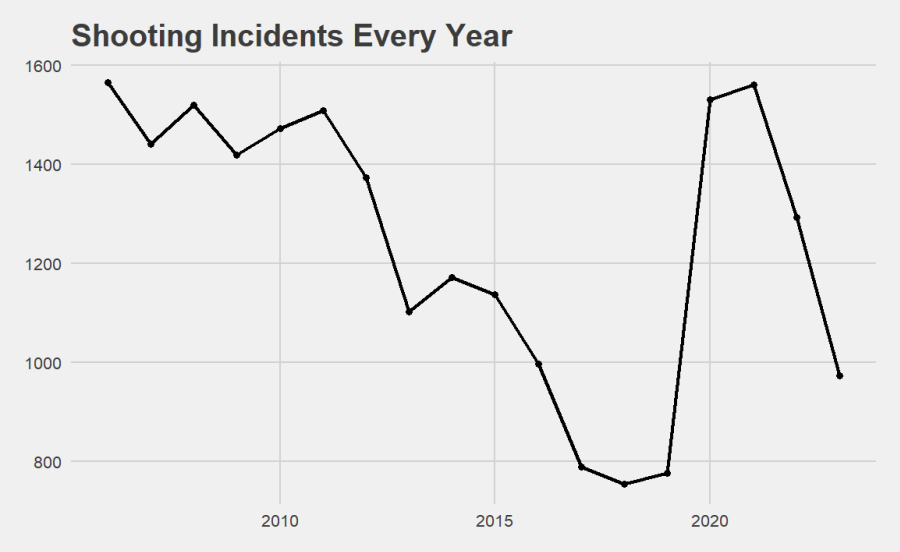
might influence the data. Another bias arises from the lack of population data for the city and its boroughs; a larger population could be a reason for higher shooting incidents.

Trends and Where did Shootings Happened?

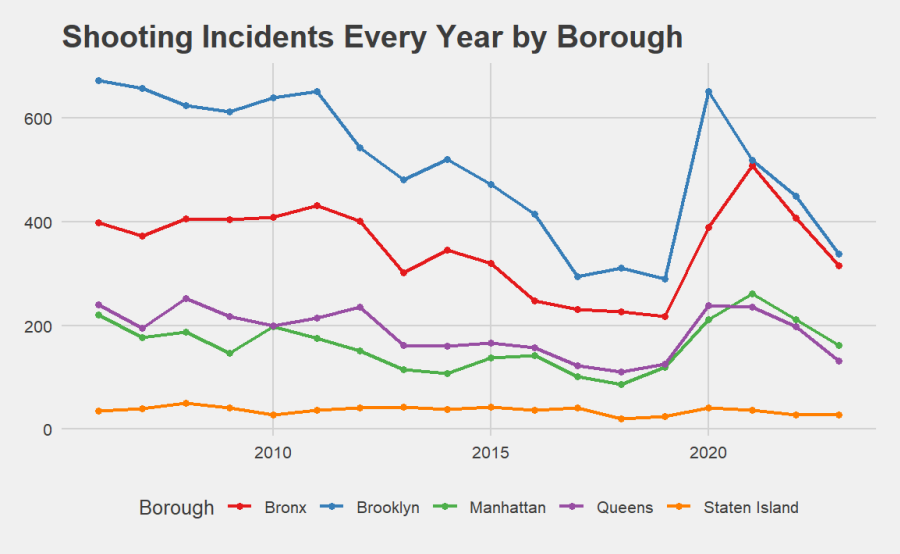
The plot illustrates the monthly occurrence of shooting incidents from 2006 to 2023. A yearly periodic pattern is evident, with incidents peaking in the summer and declining in the winter. Notably, there was a significant decrease after 2012 and 2016, followed by a sharp increase in 2020, with a gradual return to the pre-2020 pattern.



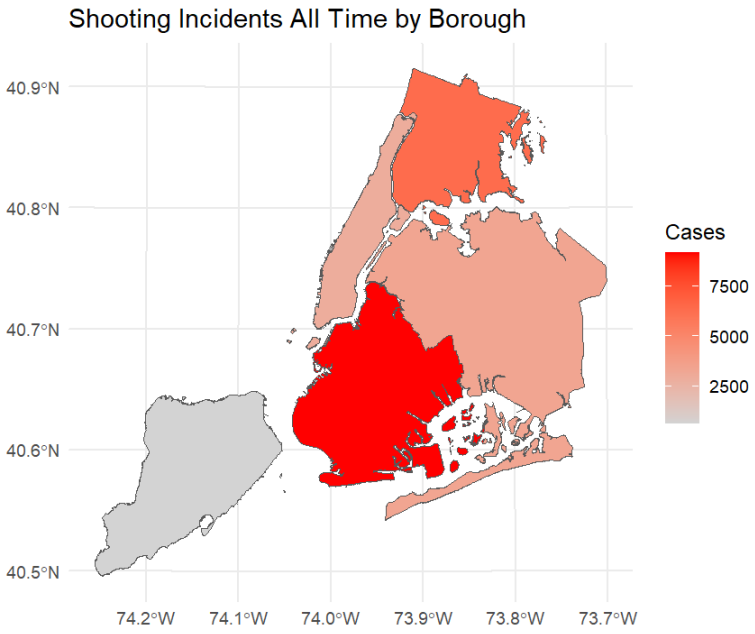
We can observe a distinct trend by plotting annual shooting incidents. After 2012, incidents decreased to approximately 75% and dropped to 50% after 2016. However, there was a resurgence in 2020, followed by a downward trend that returned to 50%.

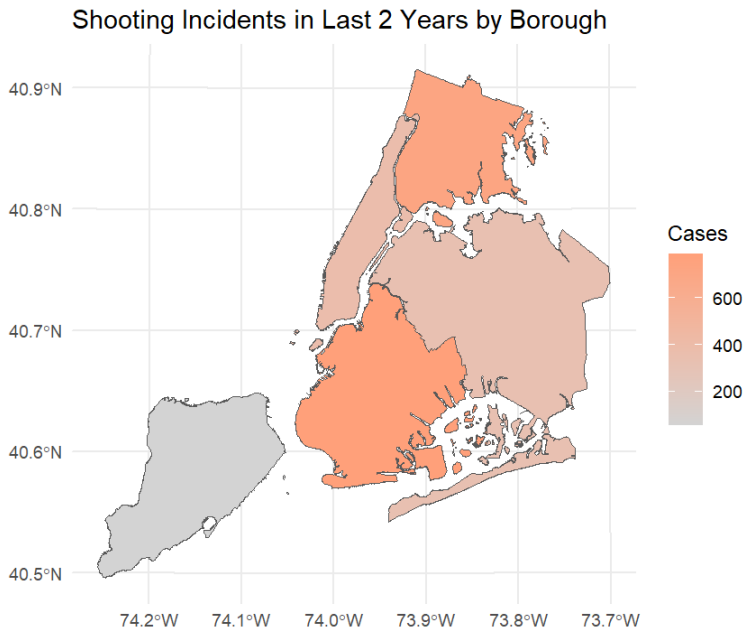


Further analysis of annual shooting incidents by borough reveals that initially, Brooklyn had a much higher number of shootings compared to other boroughs, with numbers later approaching those of the Bronx. Both Brooklyn and the Bronx reported twice as many shootings as the other boroughs.



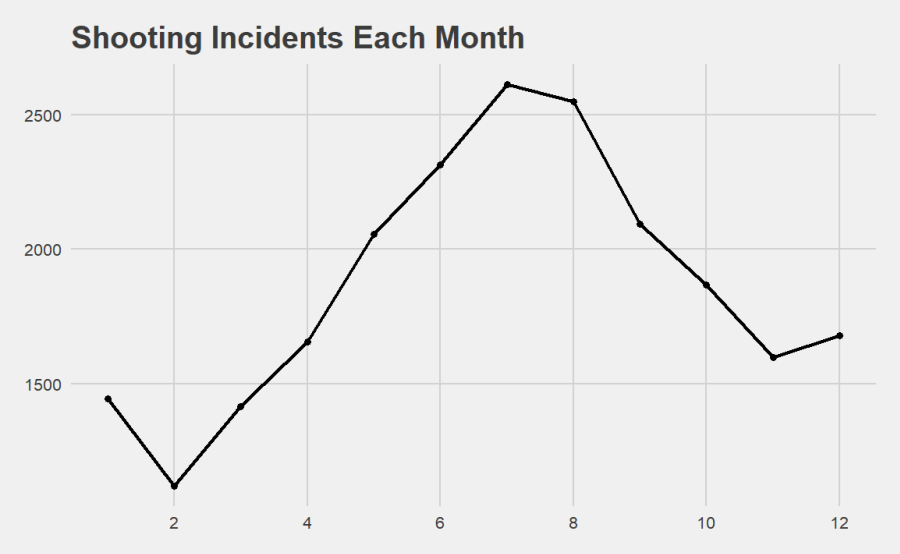
We can visualize shooting incidents on the map for all time and also for the past two years.



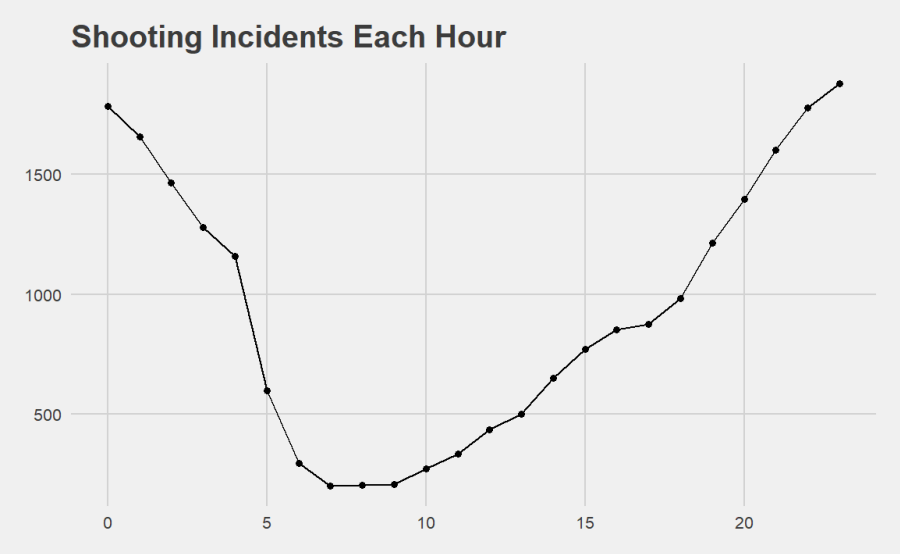


When did Shootings Happened?

Shootings peak between June and August and reach their lowest points from January to March.

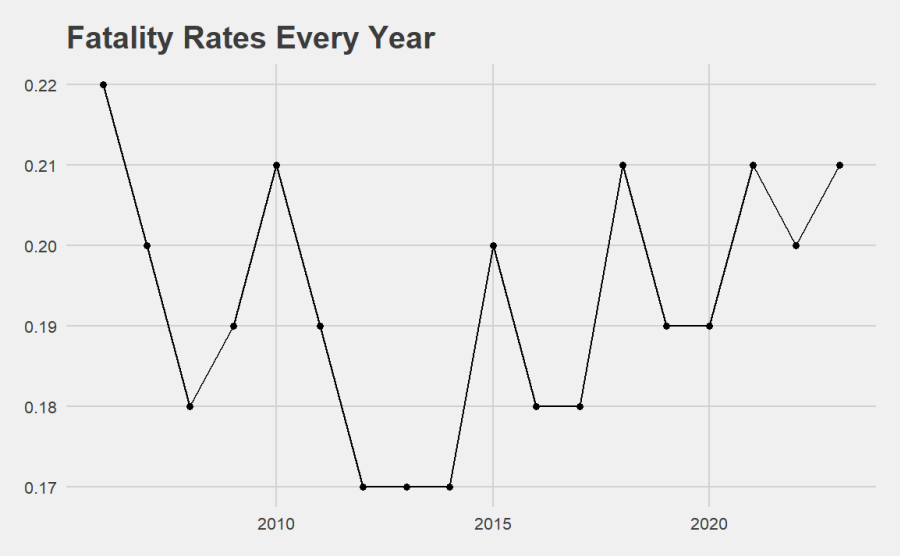


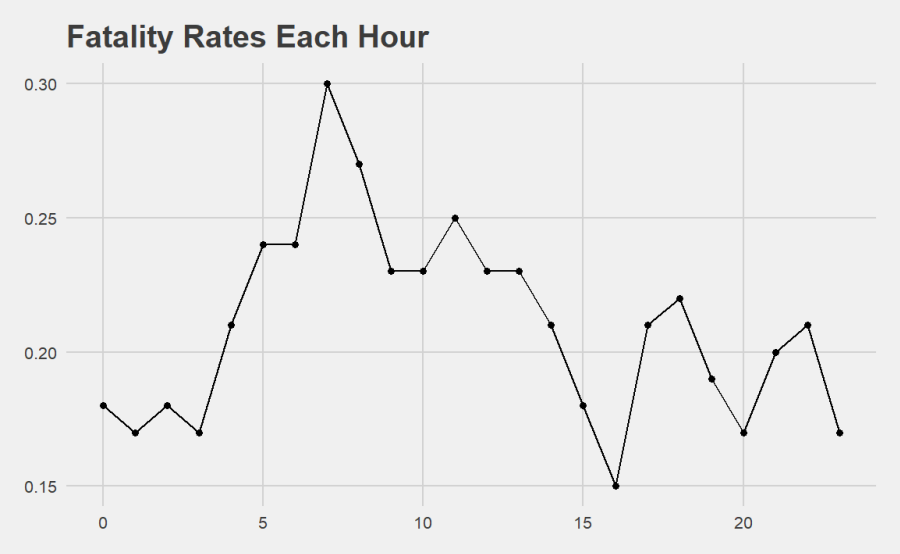
The shooting incidents is lowest in the morning and noon, then gradually increases throughout the day, reaching a peak at midnight. Following that, it begins to decline until morning.



How about Fatality Rates

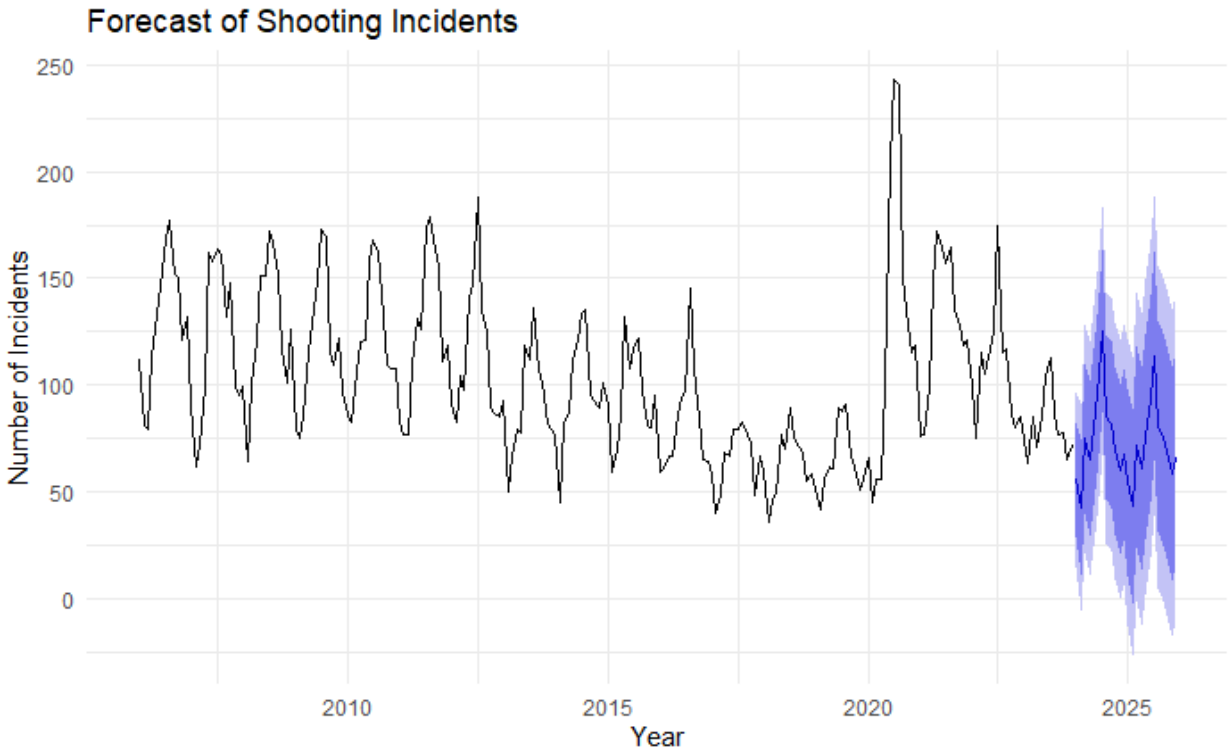
The overall fatality rate is approximately 20%. At seven in the morning, the fatality rate peaks at about 30%.





Forecast with ARIMA

Applied the Autoregressive Integrated Moving Average (ARIMA) method to forecast the next 24 months trend.



(width="450")

3. Data Source

Links

NYC Open Data Webpage: *NYPD Shooting Incident Data (Historic)*.

DATA.GOV Webpage: *NYPD Shooting Incident Data (Historic)*

CSV File: *NYPD Shooting Incident Data (Historic)*.

What's in this Dataset?

You can find below content in the attachments section of *NYPD Shooting Incident Data (Historic)*.

Field Names and Descriptions are as follows:

Field Name	Description
INCIDENT_KEY	Randomly generated persistent ID for each incident
OCCUR_DATE	Exact date of the shooting incident
OCCUR_TIME	Exact time of the shooting incident
BORO	Borough where the shooting incident occurred
PRECINCT	Precinct where the shooting incident occurred
JURISDICTION_CODE	Jurisdiction where the shooting incident occurred. Jurisdiction codes 0(Patrol), 1(Transit), and 2(Housing) represent NYPD whilst codes 3 and more represent non NYPD jurisdictions
LOCATION_DESC	Location of the shooting incident
STATISTICAL_MURDER_FLAG	Shooting resulted in the victim's death which would be counted as a murder
PERP_AGE_GROUP	Perpetrator's age within a category
PERP_SEX	Perpetrator's sex description
PERP_RACE	Perpetrator's race description
VIC_AGE_GROUP	Victim's age within a category
VIC_SEX	Victim's sex description
VIC_RACE	Victim's race description
X_COORD_CD	Midblock X-coordinate for New York State Plane Coordinate System, Long Island Zone, NAD 83, units feet (FIPS 3104)
Y_COORD_CD	Midblock Y-coordinate for New York State Plane Coordinate System, Long Island Zone, NAD 83, units feet (FIPS 3104)
Latitude	Latitude coordinate for Global Coordinate System, WGS 1984, decimal degrees (EPSG 4326)
Longitude	Longitude coordinate for Global Coordinate System, WGS 1984, decimal degrees (EPSG 4326)

1. Information is accurate as of the date it was queried from the system of record, but should be considered a close approximation of current records, due to revisions and updates.
2. Data is available as of the date that technological enhancements to information systems allowed for data capture. Null values appearing frequently in certain fields may be attributed to changes on official department forms where data was previously not collected. Null values may also appear in instances where information was not available or unknown at the time of the report and should be considered as either "Unknown/Not Available/Not Reported."
3. A shooting incident can have multiple victims involved and as a result duplicate INCIDENT_KEY's are produced. Each INCIDENT_KEY represents a victim but similar duplicate keys are counted as one incident.

4. Shooting incidents occurring near an intersection are represented by the X coordinate and Y coordinates of the intersection Shooting incidents occurring anywhere other than at an intersection are geo-located to the middle of the nearest street segment where appropriate.
5. Any attempt to match the approximate location of the incident to an exact address or link to other datasets is not recommended.
6. Many other shooting incidents that were not able to be geo-coded (for example, due to an invalid address) have been located as occurring at the police station house within the precinct of occurrence.
7. Shooting incidents occurring in open areas such as parks or beaches may be geo-coded as occurring on streets or intersections bordering the area.
8. Shooting incidents occurring on a moving train on transit systems are geo-coded as occurring at the train's next stop.
9. All shooting incidents occurring within the jurisdiction of the Department of Correction have been geo-coded as occurring on Riker's Island.
10. X and Y Coordinates are in NAD 1983 State Plane New York Long Island Zone Feet (EPSG 2263).
11. Latitude and Longitude Coordinates are provided in Global Coordinate System WGS 1984 decimal degrees (EPSG 4326).
12. Errors in data transcription may result in nominal data inconsistencies.
13. The CSV file should be opened using an appropriate tool for data exploration, e.g. SPSS, SAS, Tableau, etc. If using MS Excel, be sure to use the tools for importing external data, otherwise inconsistencies may occur when viewing the data.
14. Only valid shooting incidents resulting in an injured victim are included in this release. Shooting incidents not resulting in an injured victim are classified according to the appropriate offense according to NYS Penal Law.

4. Data Analyzing

Import Libraries

```
install.packages("dplyr")
install.packages("tidyverse")
install.packages("lubridate")
install.packages("ggplot2")
install.packages("scales")
install.packages("ggthemes")
install.packages("sf")
install.packages("devtools")
install.packages("forecast")
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v readr     2.1.5
## v ggplot2   3.5.1     v stringr   1.5.1
## v lubridate 1.9.3     v tibble    3.2.1
## v purrr     1.0.2     v tidyr     1.3.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(ggthemes)
library(lubridate)
library(ggplot2)
library(scales)
```

```
##
## Attaching package: 'scales'
##
## The following object is masked from 'package:purrr':
##
##     discard
##
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
library(devtools)
```

```
## Loading required package: usethis
```

```
library(sf)
```

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

```
devtools::install_github("mfherman/nycgeo")
```

```
## Using GitHub PAT from the git credential store.
## Skipping install of 'nycgeo' from a github remote, the SHA1 (4fee55c1) has not changed since last ins
##   Use 'force = TRUE' to force installation
```

```
library(nycgeo)
library(forecast)
```

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

Import Data

Store NYPD Shooting Incident Data (Historic) csv file in the nypd variable.

```
link <- "https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD"
nypd <- read.csv(link)
head(nypd)
```

```
##   INCIDENT_KEY OCCUR_DATE OCCUR_TIME      BORO LOC_OF_OCCUR_DESC PRECINCT
## 1    244608249 05/05/2022   00:10:00 MANHATTAN            INSIDE       14
## 2    247542571 07/04/2022   22:20:00     BRONX           OUTSIDE       48
## 3     84967535 05/27/2012   19:35:00    QUEENS                         103
## 4    202853370 09/24/2019   21:00:00     BRONX                          42
## 5     27078636 02/25/2007   21:00:00  BROOKLYN                          83
## 6    230311078 07/01/2021   23:07:00 MANHATTAN                          23
##   JURISDICTION_CODE LOC_CLASSFCTN_DESC             LOCATION_DESC
## 1                 0         COMMERCIAL               VIDEO STORE
## 2                 0             STREET                    (null)
## 3                 0
## 4                 0
## 5                 0
## 6                 2                    MULTI DWELL - PUBLIC HOUS
##   STATISTICAL_MURDER_FLAG PERP_AGE_GROUP PERP_SEX PERP_RACE VIC_AGE_GROUP
## 1                    true          25-44        M     BLACK         25-44
## 2                    true         (null)   (null)    (null)         18-24
## 3                   false                                             18-24
## 4                   false          25-44        M   UNKNOWN         25-44
## 5                   false          25-44        M     BLACK         25-44
## 6                   false                                             25-44
##   VIC_SEX VIC_RACE X_COORD_CD Y_COORD_CD Latitude Longitude
## 1       M    BLACK     986050   214231.0 40.75469 -73.99350
## 2       M    BLACK    1016802   250581.0 40.85440 -73.88233
## 3       M    BLACK    1048632   198262.0 40.71063 -73.76777
## 4       M    BLACK    1014493   242565.0 40.83242 -73.89071
## 5       M    BLACK    1009149   190104.7 40.68844 -73.91022
## 6       M    BLACK     999061   229912.0 40.79773 -73.94651
##                                    Lon_Lat
## 1                   POINT (-73.9935 40.754692)
## 2                  POINT (-73.88233 40.854402)
## 3  POINT (-73.76777349199995 40.71063412500007)
## 4 POINT (-73.89071440599997 40.832416753000075)
## 5  POINT (-73.91021857399994 40.68844345900004)
## 6  POINT (-73.94650786199998 40.79772716600007)
```

Data Exploration and Data Cleaning

Select and rename chosen columns. Sort the table by `key`.

```
nypd <- nypd %>%
    select(INCIDENT_KEY, OCCUR_DATE, OCCUR_TIME, BORO, STATISTICAL_MURDER_FLAG) %>%
    rename(key = INCIDENT_KEY,
           date = OCCUR_DATE,
```

```
            time = OCCUR_TIME,
            borough = BORO,
            murdered = STATISTICAL_MURDER_FLAG) %>%
    arrange(key)

summary(nypd)
```

```
##       key                date              time             borough
##  Min.   :  9953245   Length:28562       Length:28562       Length:28562
##  1st Qu.: 65439914   Class :character   Class :character   Class :character
##  Median : 92711254   Mode  :character   Mode  :character   Mode  :character
##  Mean   :127405824
##  3rd Qu.:203131993
##  Max.   :279758069
##    murdered
##  Length:28562
##  Class :character
##  Mode  :character
##
##
##
```

key Column

Each key represents a shooting incident. A shooting incident can have multiple victims.

```
summary(nypd$key)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##   9953245  65439914  92711254 127405824 203131993 279758069
```

```
cat("\n\n")
```

```
cat("Unique Incidents\n")
```

```
## Unique Incidents
```

```
length(unique(nypd$key))
```

```
## [1] 22394
```

date Column

Convert the character type to the date type. The date spans a range of 216 months from January 2006 to December 2023.

```
nypd <- nypd %>% mutate(date = mdy(date))
summary(nypd$date)
```

```
##         Min.      1st Qu.       Median         Mean      3rd Qu.         Max.
## "2006-01-01" "2009-09-04" "2013-09-20" "2014-06-07" "2019-09-29" "2023-12-29"
```

time Column

Keep the hour(0-23) only and rename the column to `hour`.

```
nypd$time <- hour(hms(nypd$time))
nypd <- nypd %>% rename(hour = time)
summary(nypd$hour)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    3.00   15.00   12.27   20.00   23.00
```

borough Column

Transform borough names to the title case.

```
nypd$borough <- str_to_title(nypd$borough)
count(nypd, borough)
```

```
##         borough     n
## 1         Bronx  8376
## 2      Brooklyn 11346
## 3     Manhattan  3762
## 4        Queens  4271
## 5 Staten Island   807
```

murdered Column

Transform the str type to the logical type.

```
nypd$murdered <- as.logical(nypd$murdered)
count(nypd, murdered)
```

```
##   murdered     n
## 1    FALSE 23036
## 2     TRUE  5526
```

Data Analyzing

To find out history shooting incidents, we keep the distinct incident keys and count the number of rows that group by months.

```
shooting_per_month <- nypd %>%
    # Keep one row for the same incident keys
    distinct(key, .keep_all=TRUE) %>%
    # Shooting incidents group by month
    mutate(floor_month = floor_date(date, "month")) %>%
    count(floor_month, name = "shooting") %>%
    # Get year and month
    mutate(year = year(floor_month),
           month = month(floor_month))

shooting_per_month
```

```
##     floor_month shooting year month
## 1    2006-01-01      112 2006     1
## 2    2006-02-01       81 2006     2
## 3    2006-03-01       79 2006     3
## 4    2006-04-01      113 2006     4
## 5    2006-05-01      134 2006     5
## 6    2006-06-01      146 2006     6
## 7    2006-07-01      169 2006     7
## 8    2006-08-01      177 2006     8
## 9    2006-09-01      152 2006     9
## 10   2006-10-01      150 2006    10
## 11   2006-11-01      121 2006    11
## 12   2006-12-01      132 2006    12
## 13   2007-01-01       90 2007     1
## 14   2007-02-01       62 2007     2
## 15   2007-03-01       72 2007     3
## 16   2007-04-01       99 2007     4
## 17   2007-05-01      162 2007     5
## 18   2007-06-01      158 2007     6
## 19   2007-07-01      164 2007     7
## 20   2007-08-01      161 2007     8
## 21   2007-09-01      132 2007     9
## 22   2007-10-01      148 2007    10
## 23   2007-11-01       99 2007    11
## 24   2007-12-01       95 2007    12
## 25   2008-01-01       99 2008     1
## 26   2008-02-01       64 2008     2
## 27   2008-03-01      100 2008     3
## 28   2008-04-01      121 2008     4
## 29   2008-05-01      151 2008     5
## 30   2008-06-01      151 2008     6
## 31   2008-07-01      172 2008     7
## 32   2008-08-01      167 2008     8
## 33   2008-09-01      151 2008     9
## 34   2008-10-01      117 2008    10
## 35   2008-11-01      101 2008    11
## 36   2008-12-01      126 2008    12
## 37   2009-01-01       79 2009     1
## 38   2009-02-01       75 2009     2
## 39   2009-03-01       86 2009     3
## 40   2009-04-01      115 2009     4
## 41   2009-05-01      129 2009     5
## 42   2009-06-01      149 2009     6
## 43   2009-07-01      173 2009     7
## 44   2009-08-01      170 2009     8
## 45   2009-09-01      116 2009     9
## 46   2009-10-01      109 2009    10
## 47   2009-11-01      122 2009    11
## 48   2009-12-01       97 2009    12
## 49   2010-01-01       86 2010     1
## 50   2010-02-01       83 2010     2
## 51   2010-03-01      106 2010     3
## 52   2010-04-01      120 2010     4
## 53   2010-05-01      121 2010     5
```

```
## 54   2010-06-01      158 2010     6
## 55   2010-07-01      168 2010     7
## 56   2010-08-01      163 2010     8
## 57   2010-09-01      143 2010     9
## 58   2010-10-01      109 2010    10
## 59   2010-11-01      108 2010    11
## 60   2010-12-01      108 2010    12
## 61   2011-01-01       83 2011     1
## 62   2011-02-01       77 2011     2
## 63   2011-03-01       77 2011     3
## 64   2011-04-01      109 2011     4
## 65   2011-05-01      131 2011     5
## 66   2011-06-01      126 2011     6
## 67   2011-07-01      174 2011     7
## 68   2011-08-01      179 2011     8
## 69   2011-09-01      168 2011     9
## 70   2011-10-01      155 2011    10
## 71   2011-11-01      111 2011    11
## 72   2011-12-01      119 2011    12
## 73   2012-01-01       91 2012     1
## 74   2012-02-01       83 2012     2
## 75   2012-03-01      104 2012     3
## 76   2012-04-01       98 2012     4
## 77   2012-05-01      140 2012     5
## 78   2012-06-01      147 2012     6
## 79   2012-07-01      188 2012     7
## 80   2012-08-01      135 2012     8
## 81   2012-09-01      125 2012     9
## 82   2012-10-01       90 2012    10
## 83   2012-11-01       87 2012    11
## 84   2012-12-01       85 2012    12
## 85   2013-01-01       93 2013     1
## 86   2013-02-01       50 2013     2
## 87   2013-03-01       67 2013     3
## 88   2013-04-01       79 2013     4
## 89   2013-05-01       78 2013     5
## 90   2013-06-01      119 2013     6
## 91   2013-07-01      112 2013     7
## 92   2013-08-01      136 2013     8
## 93   2013-09-01      107 2013     9
## 94   2013-10-01      101 2013    10
## 95   2013-11-01       82 2013    11
## 96   2013-12-01       79 2013    12
## 97   2014-01-01       77 2014     1
## 98   2014-02-01       45 2014     2
## 99   2014-03-01       82 2014     3
## 100  2014-04-01       87 2014     4
## 101  2014-05-01      112 2014     5
## 102  2014-06-01      121 2014     6
## 103  2014-07-01      134 2014     7
## 104  2014-08-01      135 2014     8
## 105  2014-09-01       96 2014     9
## 106  2014-10-01       93 2014    10
## 107  2014-11-01       89 2014    11
```

```
## 108  2014-12-01      101 2014    12
## 109  2015-01-01       90 2015     1
## 110  2015-02-01       59 2015     2
## 111  2015-03-01       69 2015     3
## 112  2015-04-01       87 2015     4
## 113  2015-05-01      132 2015     5
## 114  2015-06-01      108 2015     6
## 115  2015-07-01      117 2015     7
## 116  2015-08-01      122 2015     8
## 117  2015-09-01       98 2015     9
## 118  2015-10-01       81 2015    10
## 119  2015-11-01       80 2015    11
## 120  2015-12-01       95 2015    12
## 121  2016-01-01       59 2016     1
## 122  2016-02-01       61 2016     2
## 123  2016-03-01       67 2016     3
## 124  2016-04-01       67 2016     4
## 125  2016-05-01       86 2016     5
## 126  2016-06-01       94 2016     6
## 127  2016-07-01       97 2016     7
## 128  2016-08-01      145 2016     8
## 129  2016-09-01      108 2016     9
## 130  2016-10-01       84 2016    10
## 131  2016-11-01       65 2016    11
## 132  2016-12-01       64 2016    12
## 133  2017-01-01       58 2017     1
## 134  2017-02-01       40 2017     2
## 135  2017-03-01       48 2017     3
## 136  2017-04-01       68 2017     4
## 137  2017-05-01       67 2017     5
## 138  2017-06-01       79 2017     6
## 139  2017-07-01       79 2017     7
## 140  2017-08-01       83 2017     8
## 141  2017-09-01       79 2017     9
## 142  2017-10-01       73 2017    10
## 143  2017-11-01       48 2017    11
## 144  2017-12-01       67 2017    12
## 145  2018-01-01       59 2018     1
## 146  2018-02-01       36 2018     2
## 147  2018-03-01       46 2018     3
## 148  2018-04-01       49 2018     4
## 149  2018-05-01       77 2018     5
## 150  2018-06-01       70 2018     6
## 151  2018-07-01       89 2018     7
## 152  2018-08-01       76 2018     8
## 153  2018-09-01       71 2018     9
## 154  2018-10-01       68 2018    10
## 155  2018-11-01       55 2018    11
## 156  2018-12-01       58 2018    12
## 157  2019-01-01       52 2019     1
## 158  2019-02-01       42 2019     2
## 159  2019-03-01       56 2019     3
## 160  2019-04-01       62 2019     4
## 161  2019-05-01       61 2019     5
```

```
## 162  2019-06-01       89 2019     6
## 163  2019-07-01       88 2019     7
## 164  2019-08-01       91 2019     8
## 165  2019-09-01       67 2019     9
## 166  2019-10-01       62 2019    10
## 167  2019-11-01       51 2019    11
## 168  2019-12-01       56 2019    12
## 169  2020-01-01       66 2020     1
## 170  2020-02-01       45 2020     2
## 171  2020-03-01       56 2020     3
## 172  2020-04-01       56 2020     4
## 173  2020-05-01      100 2020     5
## 174  2020-06-01      205 2020     6
## 175  2020-07-01      243 2020     7
## 176  2020-08-01      241 2020     8
## 177  2020-09-01      150 2020     9
## 178  2020-10-01      135 2020    10
## 179  2020-11-01      115 2020    11
## 180  2020-12-01      119 2020    12
## 181  2021-01-01       76 2021     1
## 182  2021-02-01       77 2021     2
## 183  2021-03-01       99 2021     3
## 184  2021-04-01      148 2021     4
## 185  2021-05-01      172 2021     5
## 186  2021-06-01      165 2021     6
## 187  2021-07-01      157 2021     7
## 188  2021-08-01      165 2021     8
## 189  2021-09-01      135 2021     9
## 190  2021-10-01      128 2021    10
## 191  2021-11-01      119 2021    11
## 192  2021-12-01      121 2021    12
## 193  2022-01-01       99 2022     1
## 194  2022-02-01       75 2022     2
## 195  2022-03-01      115 2022     3
## 196  2022-04-01      105 2022     4
## 197  2022-05-01      117 2022     5
## 198  2022-06-01      125 2022     6
## 199  2022-07-01      175 2022     7
## 200  2022-08-01      115 2022     8
## 201  2022-09-01      117 2022     9
## 202  2022-10-01       86 2022    10
## 203  2022-11-01       80 2022    11
## 204  2022-12-01       85 2022    12
## 205  2023-01-01       74 2023     1
## 206  2023-02-01       63 2023     2
## 207  2023-03-01       85 2023     3
## 208  2023-04-01       71 2023     4
## 209  2023-05-01       85 2023     5
## 210  2023-06-01      104 2023     6
## 211  2023-07-01      113 2023     7
## 212  2023-08-01       87 2023     8
## 213  2023-09-01       77 2023     9
## 214  2023-10-01       78 2023    10
## 215  2023-11-01       65 2023    11
```

```
## 216  2023-12-01       72 2023    12
```

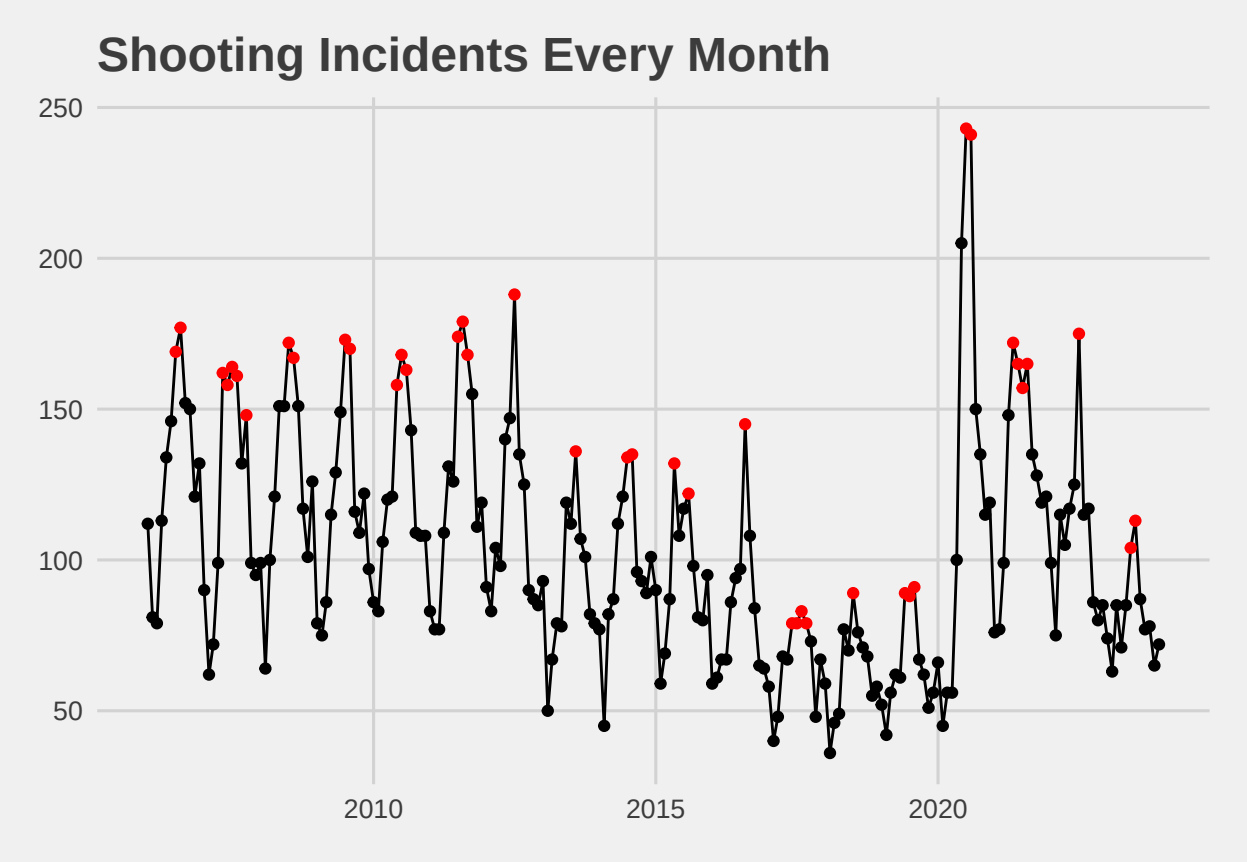
Calculate the maximum number of shooting incidents for each year and create a `highlight` column. This column should be set to TRUE if the number of shooting incidents in that month is greater than or equal to 90% of the maximum number of incidents for that year.

```
threshold <- 0.9

max_highlight <- shooting_per_month  %>%
    # Max shooting incidents each year
    # Highlight months that shooting incidents >= max_shooting
    group_by(year) %>%
    mutate(max_shooting = max(shooting),
           highlight = ifelse(shooting >= max_shooting * threshold, TRUE, FALSE))

max_highlight
```

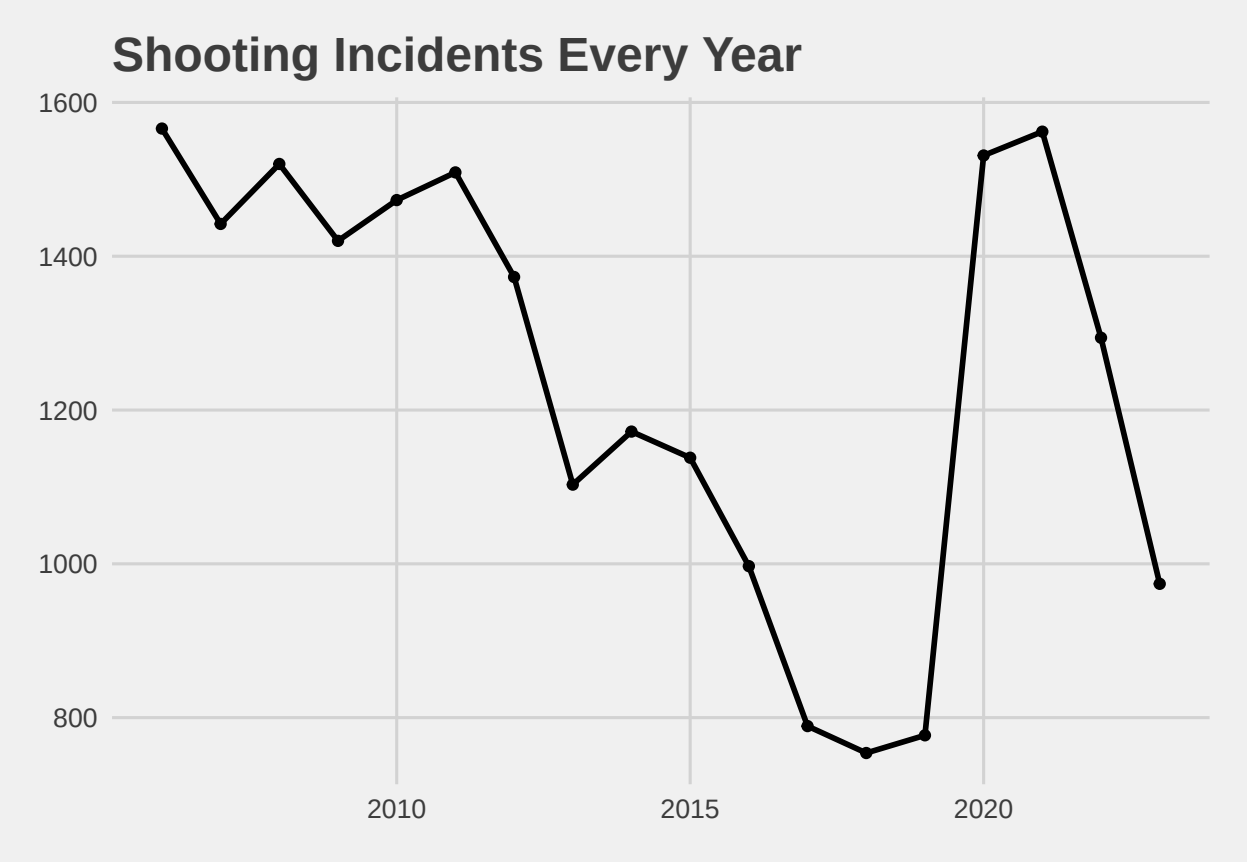
```
## # A tibble: 216 x 6
## # Groups:   year [18]
##    floor_month shooting  year month max_shooting highlight
##    <date>         <int> <dbl> <dbl>        <int> <lgl>
##  1 2006-01-01       112  2006     1          177 FALSE
##  2 2006-02-01        81  2006     2          177 FALSE
##  3 2006-03-01        79  2006     3          177 FALSE
##  4 2006-04-01       113  2006     4          177 FALSE
##  5 2006-05-01       134  2006     5          177 FALSE
##  6 2006-06-01       146  2006     6          177 FALSE
##  7 2006-07-01       169  2006     7          177 TRUE
##  8 2006-08-01       177  2006     8          177 TRUE
##  9 2006-09-01       152  2006     9          177 FALSE
## 10 2006-10-01       150  2006    10          177 FALSE
## # i 206 more rows
```

Create a line chart and emphasize the most severe months, where incidents exceed 90% of each year's maximum cases. The data indicates a cyclical trend, with the number of shooting incidents typically increasing during the summer and decreasing during the winter. There was a marked reduction in these incidents following the years 2012 and 2016, and a considerable increase at the start of 2020.

```
max_highlight %>%
    ggplot(aes(floor_month, shooting)) +
    geom_line() +
    geom_point(aes(color = highlight), show.legend = FALSE) +
    scale_color_manual(values = c("TRUE" = "red", "FALSE" = "black")) +
    labs(title = "Shooting Incidents Every Month") +
    theme_fivethirtyeight()
```

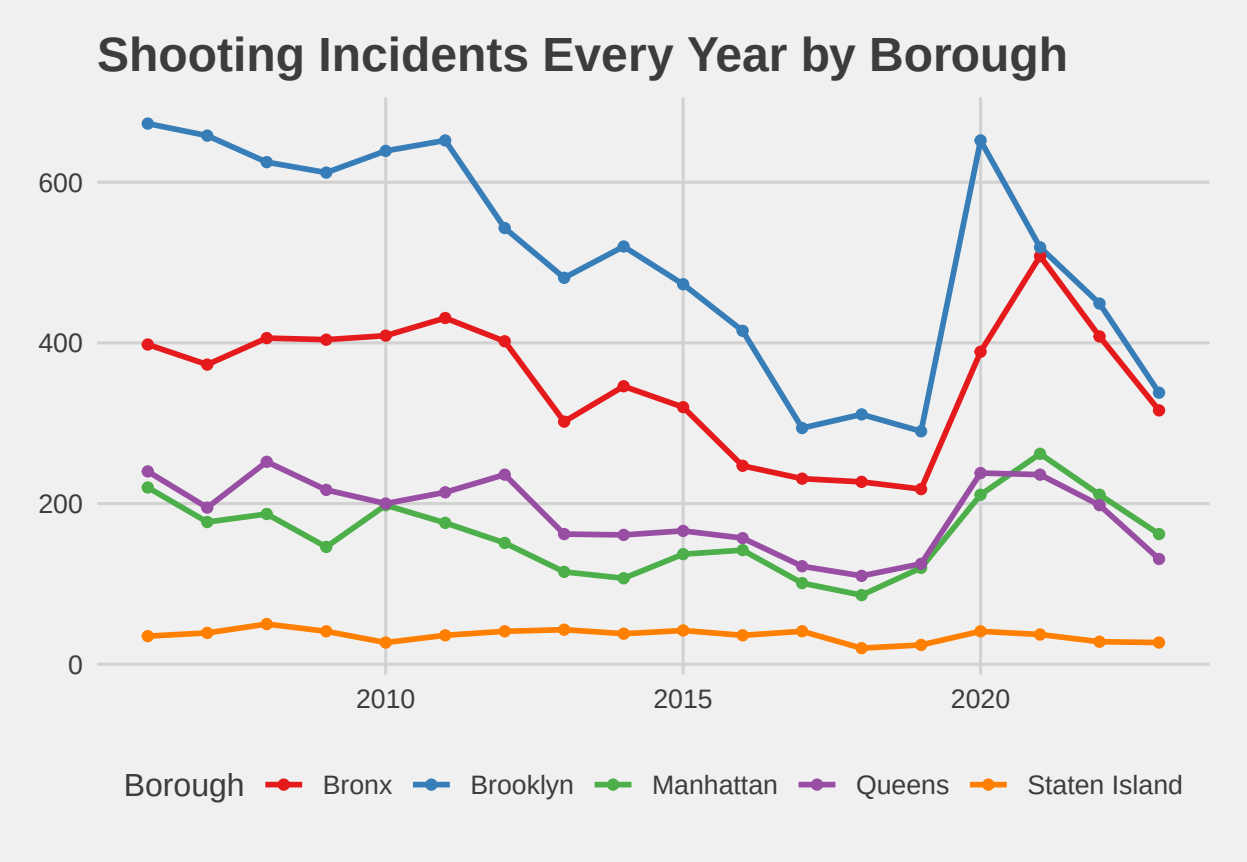



```
shooting_per_month %>%
    group_by(year) %>%
    summarise(n = sum(shooting)) %>%
    ggplot(aes(year, n)) +
    geom_line(linewidth = 1) +
    geom_point(size = 1.5) +
    labs(title = "Shooting Incidents Every Year") +
    theme_fivethirtyeight()
```

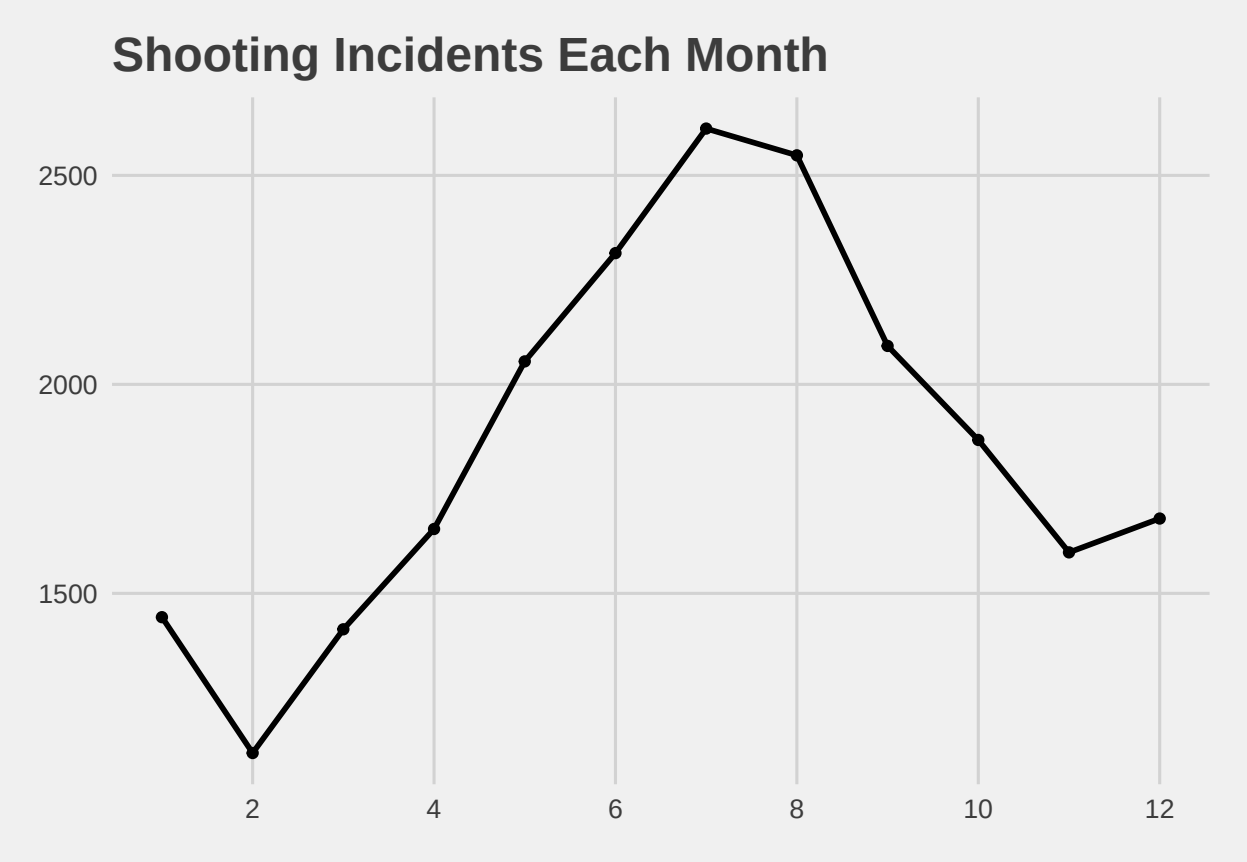



```
nypd %>%
    distinct(key, .keep_all=TRUE) %>%
    mutate(year = year(date)) %>%
    group_by(borough, year) %>%
    summarise(n = n()) %>%
    ggplot(aes(year, n, colour = borough)) +
    geom_line(linewidth = 1) +
    geom_point(size = 1.5) +
    scale_color_brewer(palette = "Set1") +
    labs(title = "Shooting Incidents Every Year by Borough", color = "Borough") +
    theme_fivethirtyeight()
```

```
## `summarise()` has grouped output by 'borough'. You can override using the
## `.groups` argument.
```

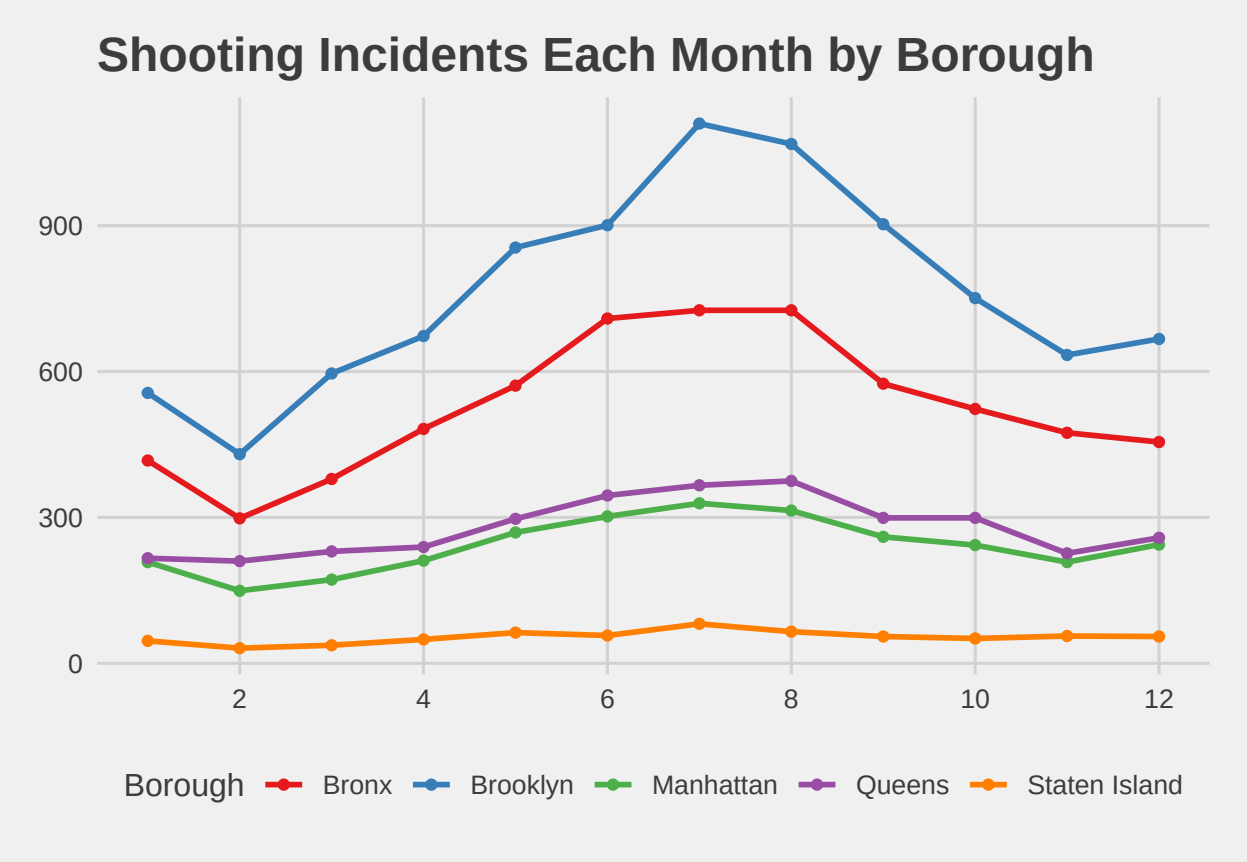



```
shooting_per_month %>%
    group_by(month) %>%
    summarise(n = sum(shooting)) %>%
    ggplot(aes(month, n)) +
    geom_line(linewidth = 1) +
    geom_point(size = 1.5) +
    scale_x_continuous(breaks = pretty_breaks(n = 5)) +
    labs(title = "Shooting Incidents Each Month") +
    theme_fivethirtyeight()
```

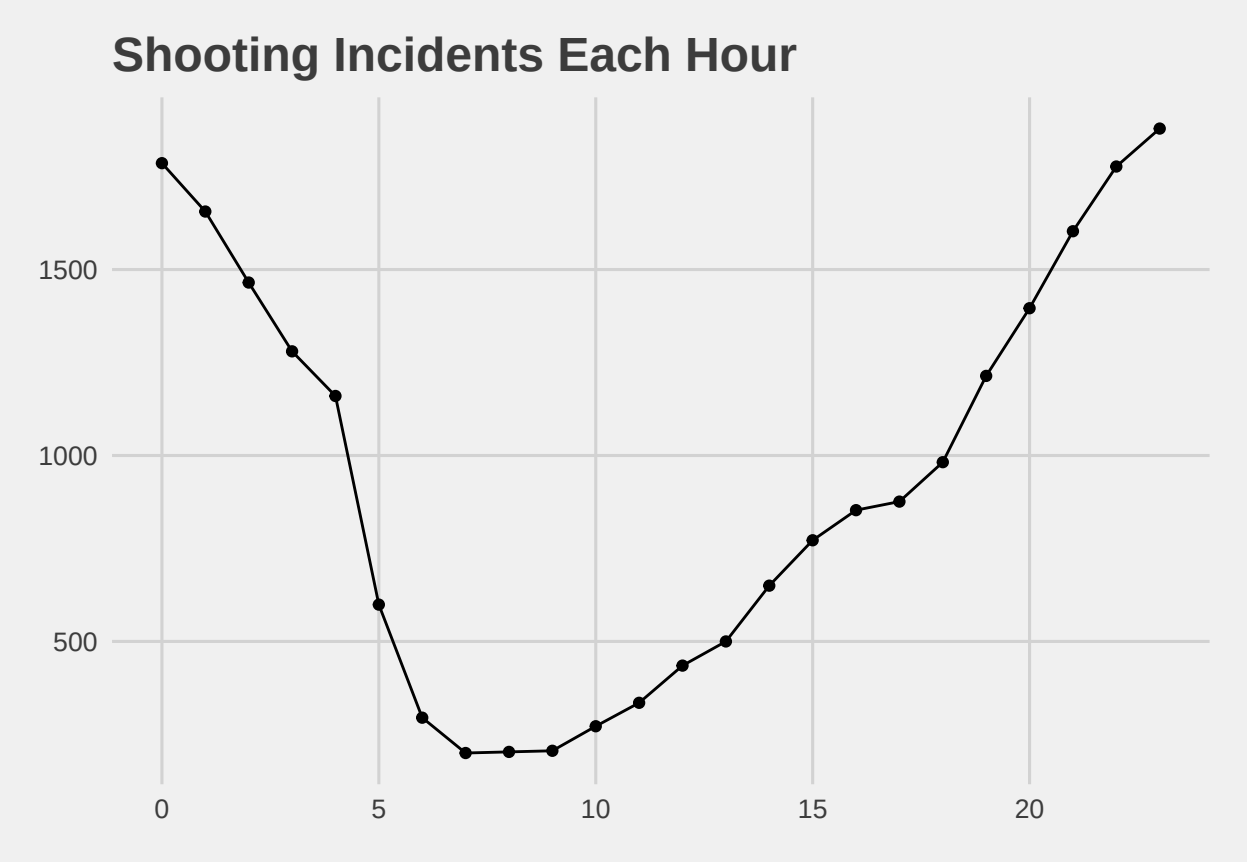



```
nypd %>%
    distinct(key, .keep_all=TRUE) %>%
    mutate(month = month(date)) %>%
    group_by(borough, month) %>%
    summarise(n = n()) %>%
    ggplot(aes(month, n, colour = borough)) +
    geom_line(linewidth = 1) +
    geom_point(size = 1.5) +
    scale_color_brewer(palette = "Set1") +
    scale_x_continuous(breaks = pretty_breaks(n = 5)) +
    labels(title = "Shooting Incidents Each Month by Borough", color = "Borough") +
    theme_fivethirtyeight()
```

```
## `summarise()` has grouped output by 'borough'. You can override using the
## `.groups` argument.
```

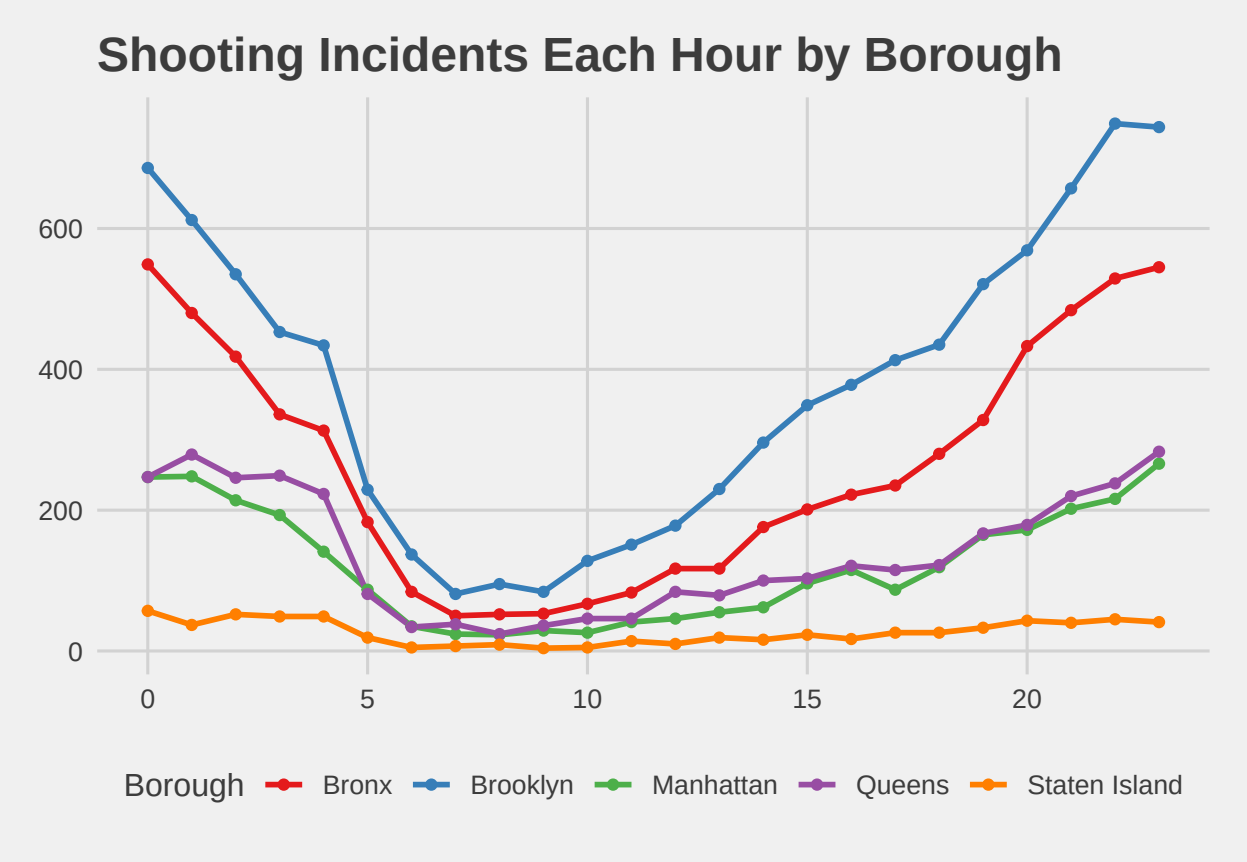



```
nypd %>%
    # Keep one row for the same incident keys
    distinct(key, .keep_all=TRUE) %>%
    count(hour, name = "case") %>%
    ggplot(aes(hour, case)) +
    geom_line() +
    geom_point() +
    labs(title = "Shooting Incidents Each Hour") +
    theme_fivethirtyeight()
```

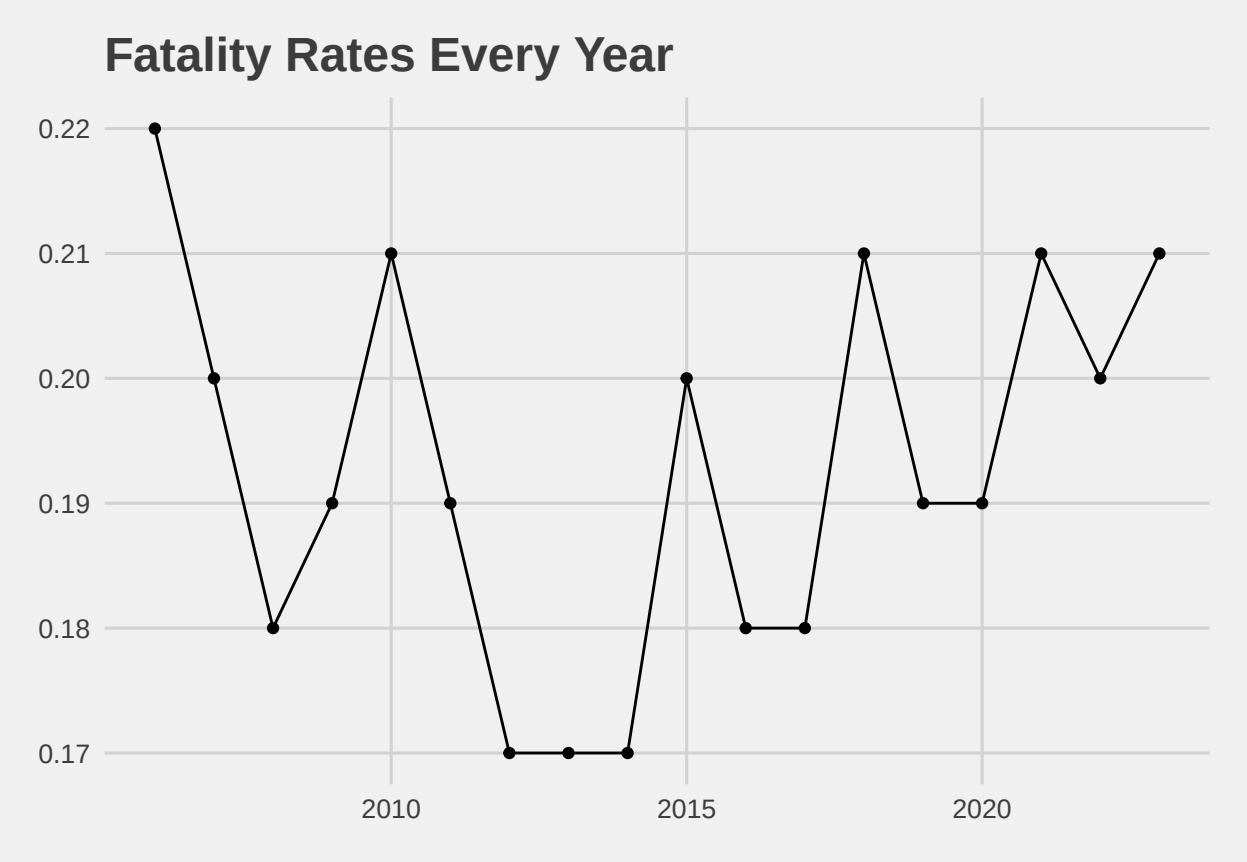



```
nypd %>%
    distinct(key, .keep_all=TRUE) %>%
    group_by(borough, hour) %>%
    summarise(n = n()) %>%
    ggplot(aes(hour, n, colour = borough)) +
    geom_line(linewidth = 1) +
    geom_point(size = 1.5) +
    scale_color_brewer(palette = "Set1") +
    labs(title = "Shooting Incidents Each Hour by Borough", color = "Borough") +
    theme_fivethirtyeight()
```

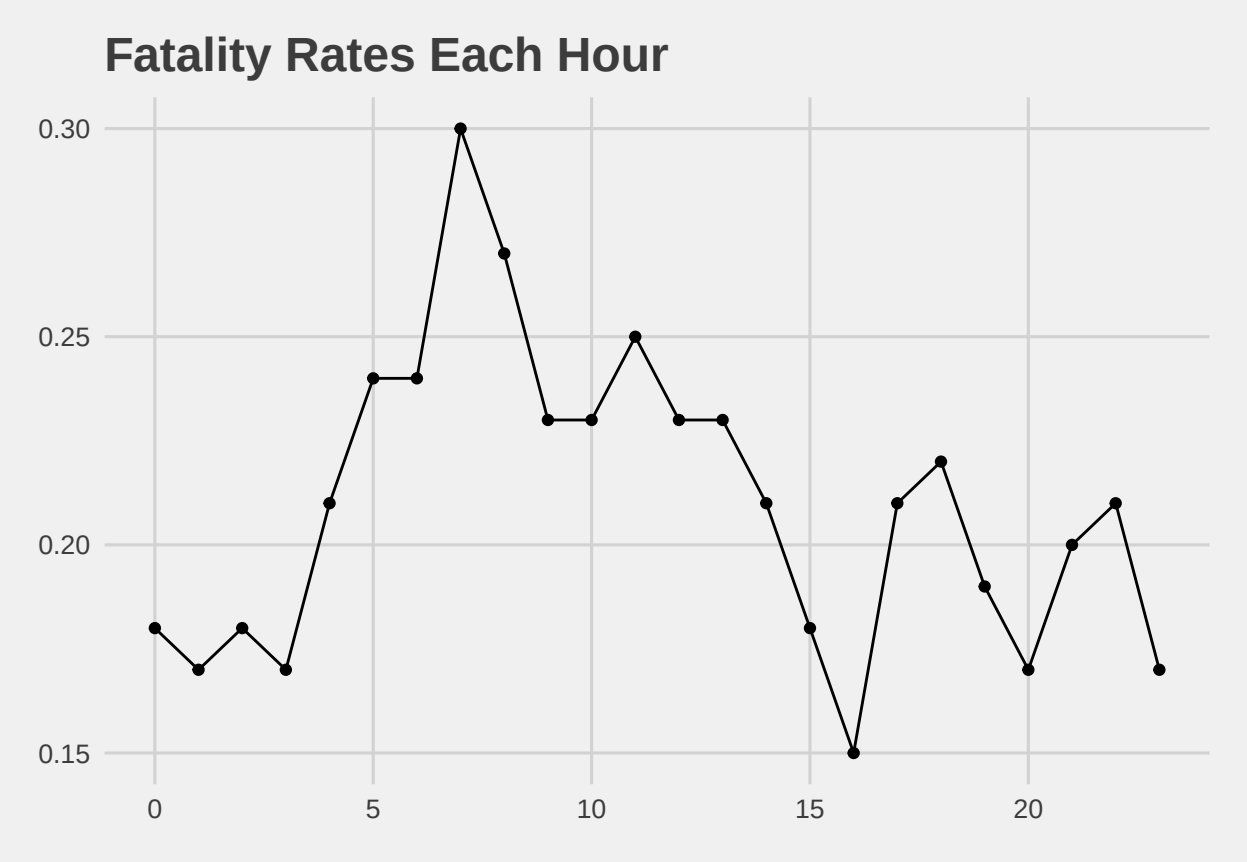
```
## `summarise()` has grouped output by 'borough'. You can override using the
## `.groups` argument.
```




```
nypd %>%
    mutate(year = year(date)) %>%
    group_by(year) %>%
    summarise(total = n(),
              killed = sum(murdered)) %>%
    mutate(rate = round(killed / total, 2)) %>%
    ggplot(aes(year, rate)) +
    geom_line() +
    geom_point() +
    labs(title = "Fatality Rates Every Year") +
    theme_fivethirtyeight()
```




```
nypd %>%
    group_by(hour) %>%
    summarise(total = n(),
              killed = sum(murdered)) %>%
    mutate(rate = round(killed / total, 2)) %>%
    ggplot(aes(hour, rate)) +
    geom_line() +
    geom_point() +
#     scale_x_continuous(breaks = pretty_breaks(n = 10)) +
    labs(title = "Fatality Rates Each Hour") +
    theme_fivethirtyeight()
```

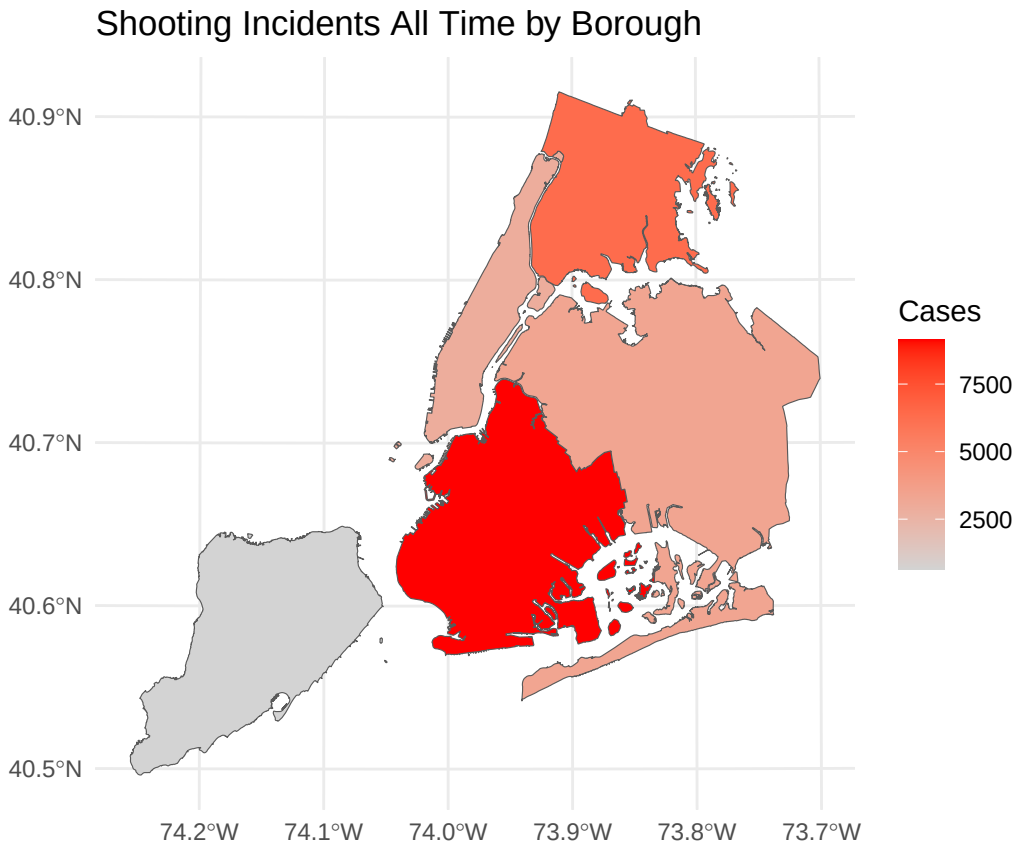



```
# Load the borough boundaries data
boroughs <- nycgeo::borough_sf
boroughs
```

```
## Simple feature collection with 5 features and 6 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: 913174.7 ymin: 120124.9 xmax: 1067382 ymax: 272847.3

## old-style crs object detected; please recreate object with a recent sf::st_crs()

## Projected CRS: NAD83 / New York Long Island (ftUS)

## old-style crs object detected; please recreate object with a recent sf::st_crs()

## # A tibble: 5 x 7
##   geoid state_fips county_fips county_name borough_name  borough_id
##   <chr> <chr>      <chr>       <chr>       <chr>         <chr>
## 1 36061 36         061         New York    Manhattan     1
## 2 36005 36         005         Bronx       Bronx         2
## 3 36047 36         047         Kings       Brooklyn      3
## 4 36081 36         081         Queens      Queens        4
## 5 36085 36         085         Richmond    Staten Island 5
## # i 1 more variable: geometry <MULTIPOLYGON [US_survey_foot]>
```

```
case_by_boro <- nypd %>%
    distinct(key, .keep_all = TRUE) %>%
    group_by(borough) %>%
    count() %>%
    rename(borough_name = borough)


boroughs %>%
    left_join(case_by_boro, by = "borough_name") %>%
    ggplot() +
    geom_sf(aes(fill = n)) +
    scale_fill_gradient(low = "lightgray", high = "red") +
    labs(title = "Shooting Incidents All Time by Borough", fill = "Cases") +
    theme_minimal()
```

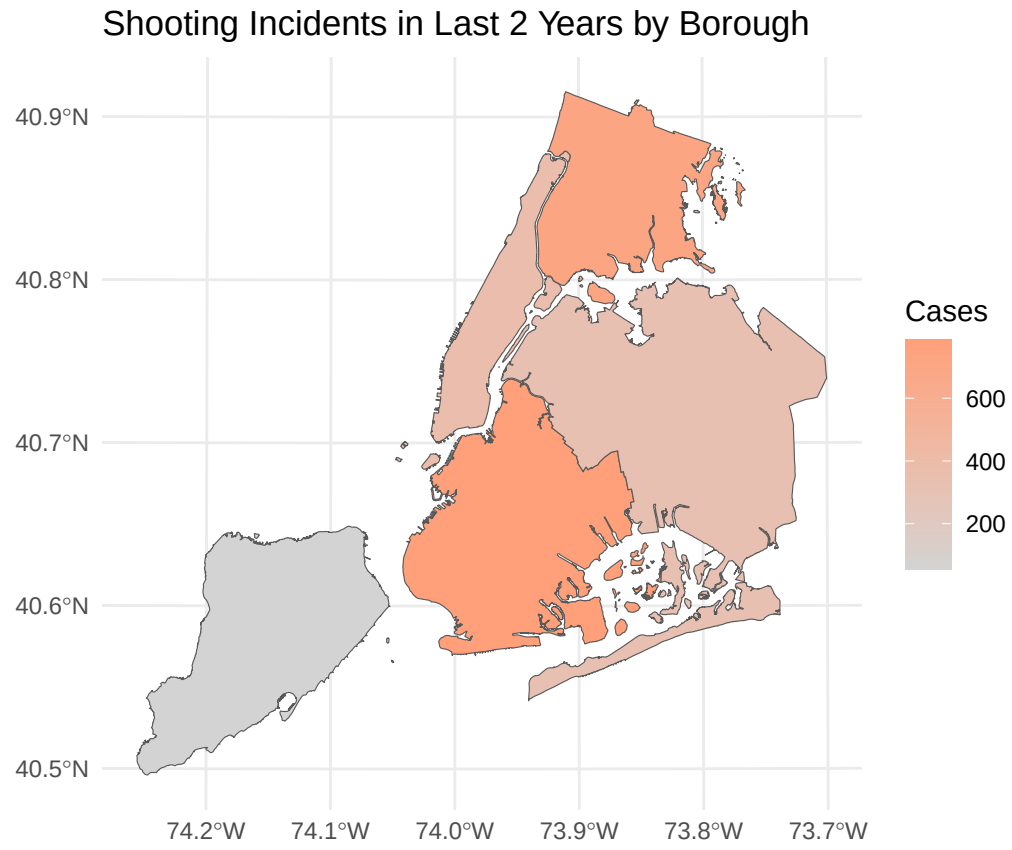
```
## old-style crs object detected; please recreate object with a recent sf::st_crs()
```



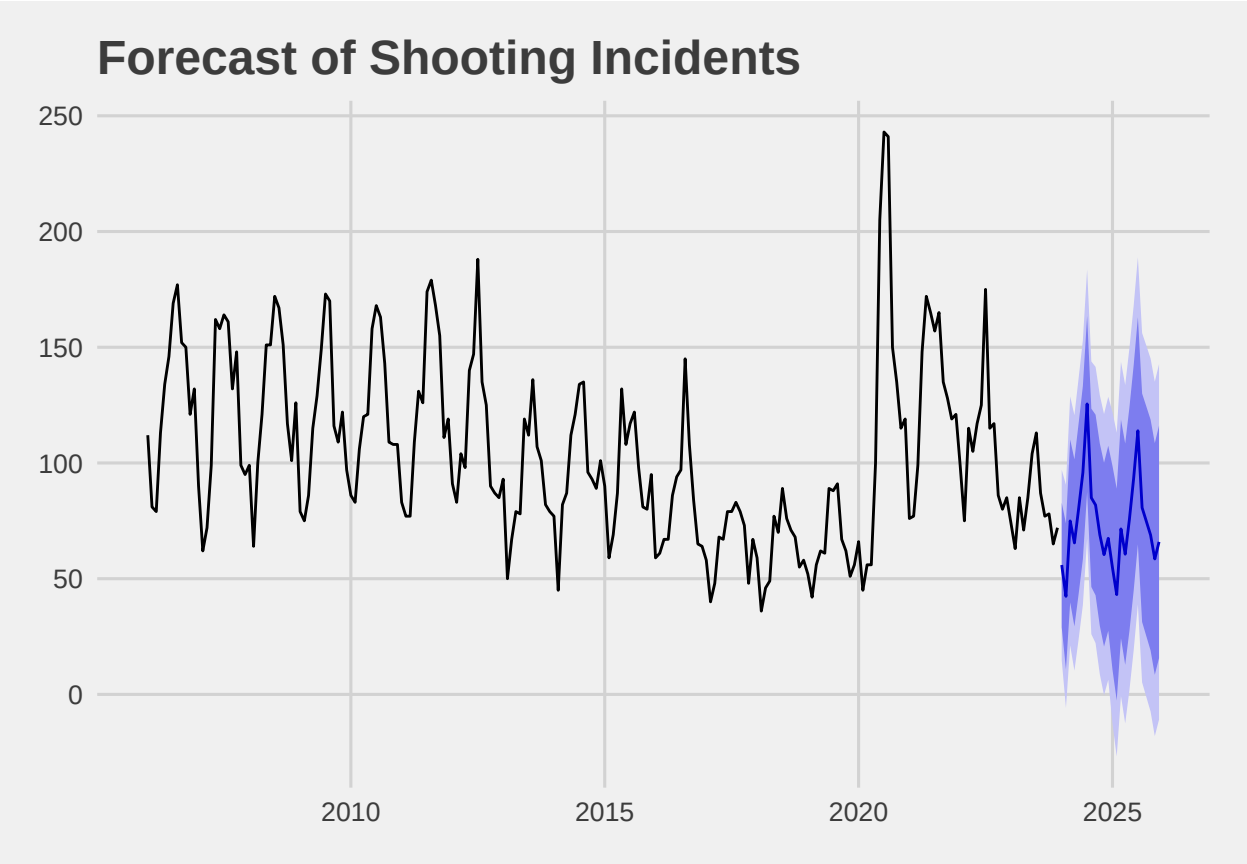

```
case_by_boro_last2y <- nypd %>%
    distinct(key, .keep_all = TRUE) %>%
    mutate(year = year(date)) %>%
    filter(year >= 2022) %>%
    group_by(borough) %>%
    count() %>%
    rename(borough_name = borough)
```

```
boroughs %>%
    left_join(case_by_boro_last2y, by = "borough_name") %>%
    ggplot() +
    geom_sf(aes(fill = n)) +
    scale_fill_gradient(low = "lightgray", high = "lightsalmon") +
    labs(title = "Shooting Incidents in Last 2 Years by Borough", fill = "Cases") +
    theme_minimal()
```

```
## old-style crs object detected; please recreate object with a recent sf::st_crs()
```




```
# Convert to time series object
ts_data <- ts(shooting_per_month$shooting, start = c(2006, 1), frequency = 12)

# Fit ARIMA model
fit <- auto.arima(ts_data)

# Forecast incidents for the next 24 months
forecast_data <- forecast(fit, h = 24)

# Plot the forecast
autoplot(forecast_data) +
    labs(title = "Forecast of Shooting Incidents",
         y = "Number of Incidents") +
    theme_fivethirtyeight()
```



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.