

# NYPD Shooting Incident Data

Cody Hill

2023-04-04

## Setup

We will first begin by loading in the packages we intend to use.

Then, importing the data using a URL directly from the source. This ensures we will capture updates to the data as they come in whenever we run this again.

```
knitr::opts_chunk$set(echo = TRUE)
knitr::opts_chunk$set(fig.width=12, fig.height=8)

library(tidyverse)

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.1      v readr      2.1.4
## v forcats    1.0.0      v stringr   1.5.0
## v ggplot2    3.4.2      v tibble    3.2.1
## v lubridate  1.9.2      v tidyr     1.3.0
## v purrr      1.0.1
## -- 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(ggmap)

## i Google's Terms of Service: <https://mapsplatform.google.com>
## i Please cite ggmap if you use it! Use `citation("ggmap")` for details.

library(ggplot2)

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

## Exploratory Data Analysis (EDA)

Let's take a look at the dimension of this imported data.frame as well as the variable types of each column.

```
dim(import_URL)

## [1] 25596    19

str(import_URL)

## 'data.frame':   25596 obs. of  19 variables:
## $ INCIDENT_KEY      : int  236168668 231008085 230717903 237712309 224465521 228252164 2269500...
## $ OCCUR_DATE        : chr   "11/11/2021" "07/16/2021" "07/11/2021" "12/11/2021" ...
## $ OCCUR_TIME        : chr   "15:04:00" "22:05:00" "01:09:00" "13:42:00" ...
## $ BORO              : chr   "BROOKLYN" "BROOKLYN" "BROOKLYN" "BROOKLYN" ...
```

```
## $ PRECINCT          : int  79 72 79 81 113 113 42 52 34 75 ...
## $ JURISDICTION_CODE : int  0 0 0 0 0 0 0 0 0 0 ...
## $ LOCATION_DESC     : chr  "" "" "" "" ...
## $ STATISTICAL_MURDER_FLAG: chr  "false" "false" "false" "false" ...
## $ PERP_AGE_GROUP    : chr  "" "45-64" "<18" "" ...
## $ PERP_SEX          : chr  "" "M" "M" "" ...
## $ PERP_RACE         : chr  "" "ASIAN / PACIFIC ISLANDER" "BLACK" "" ...
## $ VIC_AGE_GROUP     : chr  "18-24" "25-44" "25-44" "25-44" ...
## $ VIC_SEX           : chr  "M" "M" "M" "M" ...
## $ VIC_RACE          : chr  "BLACK" "ASIAN / PACIFIC ISLANDER" "BLACK" "BLACK" ...
## $ X_COORD_CD        : num  996313 981845 996546 1001139 1050710 ...
## $ Y_COORD_CD        : num  187499 171118 187436 192775 184826 ...
## $ Latitude          : num  40.7 40.6 40.7 40.7 40.7 ...
## $ Longitude         : num  -74 -74 -74 -73.9 -73.8 ...
## $ Lon_Lat           : chr  "POINT (-73.95650899099996 40.68131820000008)" "POINT (-74.00866668
```

## Use GGMAP to Visualize Locations

Here we're going to visualize the location of each shooting incident using the coordinates given in the dataset, and superimposing them over a background image of the area. We can use the minimum and maximum values of each coordinate to find our map's bounding box then use 'ggmap()' to do the heavy lifting.

```
# Initialize the bounding box that will contain the map coordinates.
map_bounds <- c(left = min(import_URL$Longitude),
  bottom = min(import_URL$Latitude),
  right = max(import_URL$Longitude),
  top = max(import_URL$Latitude))

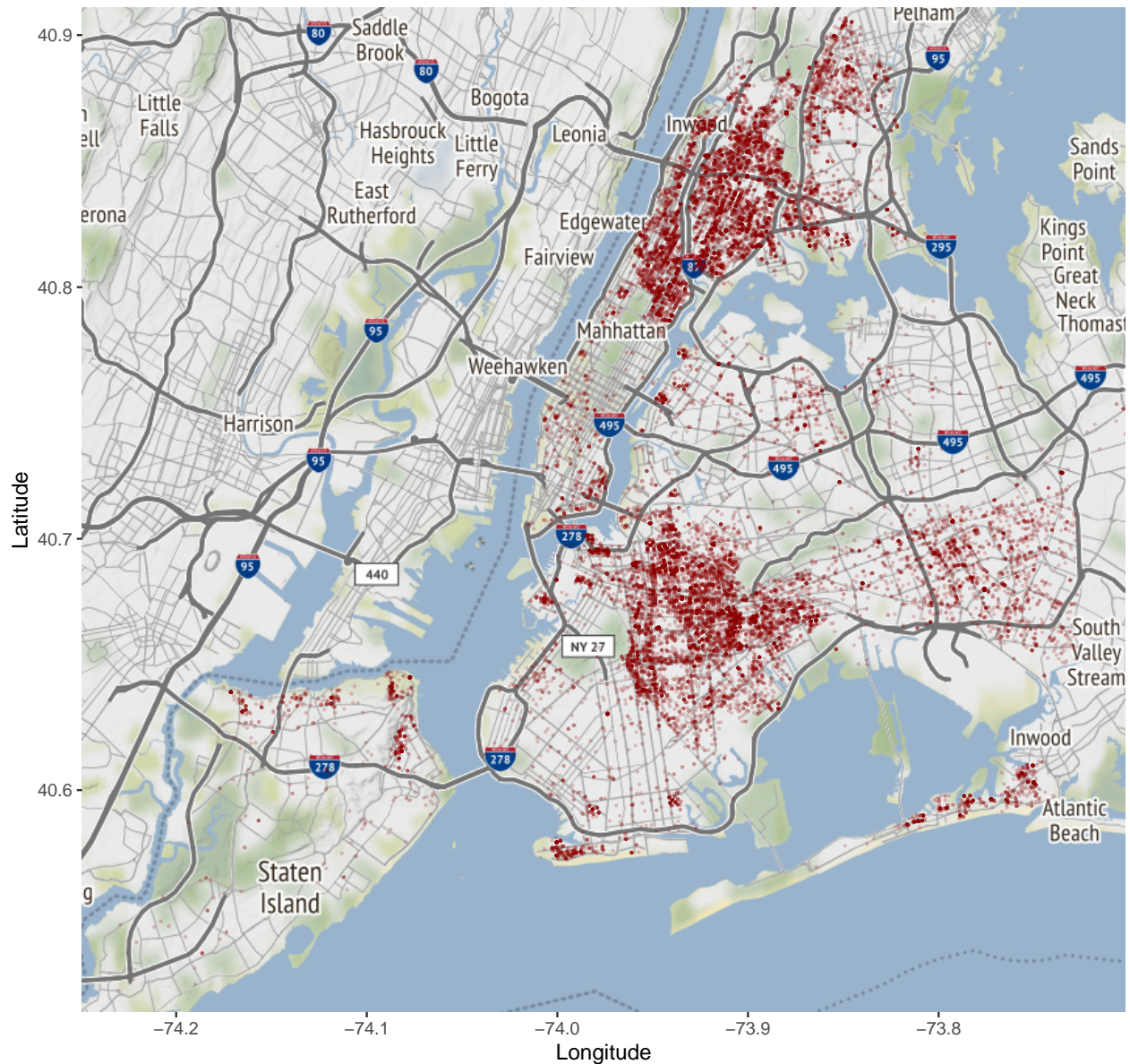
# Plot of incident coordinates
incident_map_point <- ggmap(get_stamenmap(map_bounds, maptype = 'terrain', zoom = 11)) +
  geom_point(data = import_URL,
    aes(x = Longitude, y = Latitude),
    color = 'darkred',
    size = 0.25,
    alpha = 0.2) +
  ggtitle('Point Plot of NYPD Shooting Incident Reporting 2006 - 2021\n Source:<https://catalog
  labs(x = 'Longitude', y = 'Latitude')
```

```
## i Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
```

```
incident_map_point
```

# Point Plot of NYPD Shooting Incident Reporting 2006 – 2021

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



*# Density Map to better visualize density of points.*

```
incident_map_density <- ggmap(get_stamenmap(map_bounds, maptype = 'terrain', zoom = 11)) +
  stat_density2d_filled(data = import_URL, contour_var = 'density',
    aes(x = Longitude, y = Latitude, fill = after_stat(level)),
    bins = 20,
    geom = 'polygon',
    alpha = 0.8) +
  geom_density_2d(data = import_URL,
    aes(x = Longitude, y = Latitude),
    bins = 20,
    alpha = 0.2,
    color = "white") +
  guides(fill = guide_legend(title = "Density")) +
  ggtitle('Density Plot of NYPD Shooting Incident Reporting 2006 - 2021\n Source:<

3


```

```
labs(x = 'Longitude', y = 'Latitude')
```

```
## i Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
```

```
incident_map_density
```

Density Plot of NYPD Shooting Incident Reporting 2006 – 2021

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

