NYPD Shooting Incidents

Alex

2025-05-08

Importing data

```
library(tidyverse)
```

```
NYPD_shooting <- read_csv("https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLO
head(NYPD_shooting,10)
```

```
## # A tibble: 10 x 21
##    INCIDENT_KEY OCCUR_DATE OCCUR_TIME BORO     LOC_OF_OCCUR_DESC PRECINCT
##           <dbl> <chr>      <time>     <chr>    <chr>                <dbl>
##  1    231974218 08/09/2021 01:06      BRONX    <NA>                    40
##  2    177934247 04/07/2018 19:48      BROOKLYN <NA>                    79
##  3    255028563 12/02/2022 22:57      BRONX    OUTSIDE                 47
##  4     25384540 11/19/2006 01:50      BROOKLYN <NA>                    66
##  5     72616285 05/09/2010 01:58      BRONX    <NA>                    46
##  6     85875439 07/22/2012 21:35      BRONX    <NA>                    42
##  7     79780323 07/12/2011 22:26      BROOKLYN <NA>                    71
##  8     85744504 07/14/2012 23:45      BROOKLYN <NA>                    69
##  9    142324890 04/21/2015 15:36      BROOKLYN <NA>                    75
## 10    152868707 05/07/2016 15:23      BROOKLYN <NA>                    69
## # 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>
```

Cleaning data

Removing columns

At first glance, I see some rows that do not seem that useful. These rows being:

- INCIDENT_KEY
- Precinct
- Jurisdiction Code

```
NYPD_shooting <- NYPD_shooting %>%
  select(-INCIDENT_KEY, -PRECINCT, -JURISDICTION_CODE)
colnames(NYPD_shooting)
```

```
##  [1] "OCCUR_DATE"              "OCCUR_TIME"
##  [3] "BORO"                    "LOC_OF_OCCUR_DESC"
##  [5] "LOC_CLASSFCTN_DESC"      "LOCATION_DESC"
##  [7] "STATISTICAL_MURDER_FLAG" "PERP_AGE_GROUP"
##  [9] "PERP_SEX"                "PERP_RACE"
## [11] "VIC_AGE_GROUP"           "VIC_SEX"
## [13] "VIC_RACE"                "X_COORD_CD"
## [15] "Y_COORD_CD"              "Latitude"
## [17] "Longitude"               "Lon_Lat"
```

Handling missing data

This shows the count of missing data within the dataset.

```r
colSums(is.na(NYPD_shooting))
```

```
##              OCCUR_DATE              OCCUR_TIME                    BORO
##                       0                       0                       0
##       LOC_OF_OCCUR_DESC      LOC_CLASSFCTN_DESC           LOCATION_DESC
##                   25596                   25596                   14977
## STATISTICAL_MURDER_FLAG          PERP_AGE_GROUP                PERP_SEX
##                       0                    9344                    9310
##               PERP_RACE           VIC_AGE_GROUP                 VIC_SEX
##                    9310                       0                       0
##                VIC_RACE              X_COORD_CD              Y_COORD_CD
##                       0                       0                       0
##                Latitude               Longitude                 Lon_Lat
##                      97                      97                      97
```

We can see that there are a lot missing from `LOC_OF_OCCUR_DESC` and `LOC_CLASSFCTN_DESC`

Seeing that there are only 29,734 rows in the dataset and both of those columns have 25,596 missing rows, it might be best to either just leave the column or delete it. This being the case, both `LOC_OF_OCCUR_DESC` and `LOC_CLASSFCTN_DESC` probably is not an important column so I conclude that I will delete the columns.

```r
NYPD_shooting <- NYPD_shooting %>%
  select(-LOC_OF_OCCUR_DESC, -LOC_CLASSFCTN_DESC)
colnames(NYPD_shooting)
```

```
##  [1] "OCCUR_DATE"              "OCCUR_TIME"
##  [3] "BORO"                    "LOCATION_DESC"
##  [5] "STATISTICAL_MURDER_FLAG" "PERP_AGE_GROUP"
##  [7] "PERP_SEX"                "PERP_RACE"
##  [9] "VIC_AGE_GROUP"           "VIC_SEX"
## [11] "VIC_RACE"                "X_COORD_CD"
## [13] "Y_COORD_CD"              "Latitude"
## [15] "Longitude"               "Lon_Lat"
```

I also want to take a look at the next largest missing data column which is `LOCATION_DESC`.

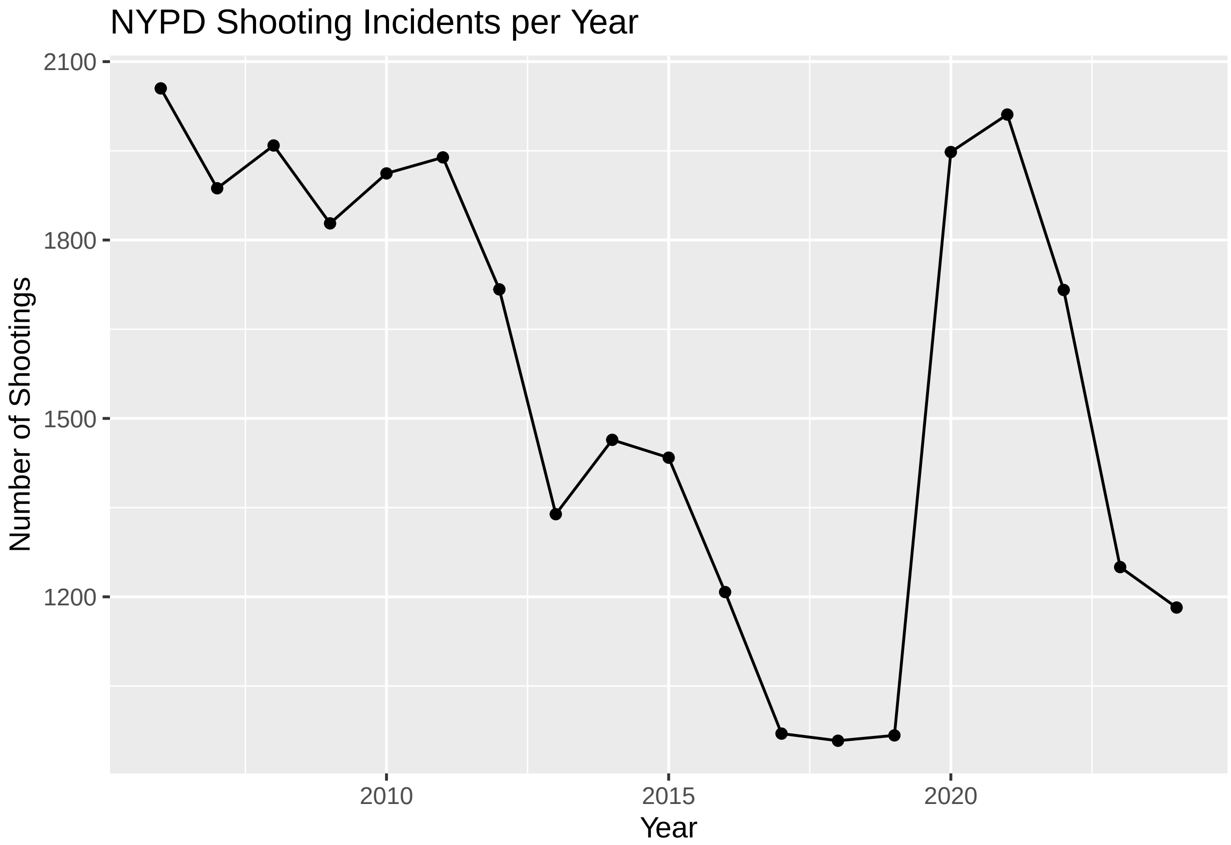
```r
unique(NYPD_shooting$LOCATION_DESC)
```

```
##  [1] NA                          "GROCERY/BODEGA"
##  [3] "PVT HOUSE"                 "MULTI DWELL - APT BUILD"
##  [5] "MULTI DWELL - PUBLIC HOUS" "(null)"
##  [7] "BAR/NIGHT CLUB"            "COMMERCIAL BLDG"
##  [9] "FAST FOOD"                 "HOSPITAL"
## [11] "BEAUTY/NAIL SALON"         "LIQUOR STORE"
## [13] "CHAIN STORE"               "RESTAURANT/DINER"
## [15] "SMALL MERCHANT"            "GAS STATION"
## [17] "JEWELRY STORE"             "GYM/FITNESS FACILITY"
## [19] "STORE UNCLASSIFIED"        "SOCIAL CLUB/POLICY LOCATI"
## [21] "DRY CLEANER/LAUNDRY"       "NONE"
## [23] "VIDEO STORE"               "SUPERMARKET"
## [25] "VARIETY STORE"             "FACTORY/WAREHOUSE"
## [27] "CLOTHING BOUTIQUE"         "SHOE STORE"
## [29] "HOTEL/MOTEL"               "CANDY STORE"
## [31] "DEPT STORE"                "BANK"
## [33] "TELECOMM. STORE"           "DRUG STORE"
## [35] "LOAN COMPANY"              "CHECK CASH"
## [37] "SCHOOL"                    "STORAGE FACILITY"
## [39] "PHOTO/COPY STORE"          "ATM"
## [41] "DOCTOR/DENTIST"
```

This column shows the specific location that the shooting takes place which can be useful. The problem is the missing data. I think I will leave it in case we want to look more into that specifically. The same can be said about `PERP_AGE_GROUP`, `PERP_SEX`, and `PERP_RACE`. Optionally, we can replace the missing rows with something like `Unknown` to make it easier to see on charts.

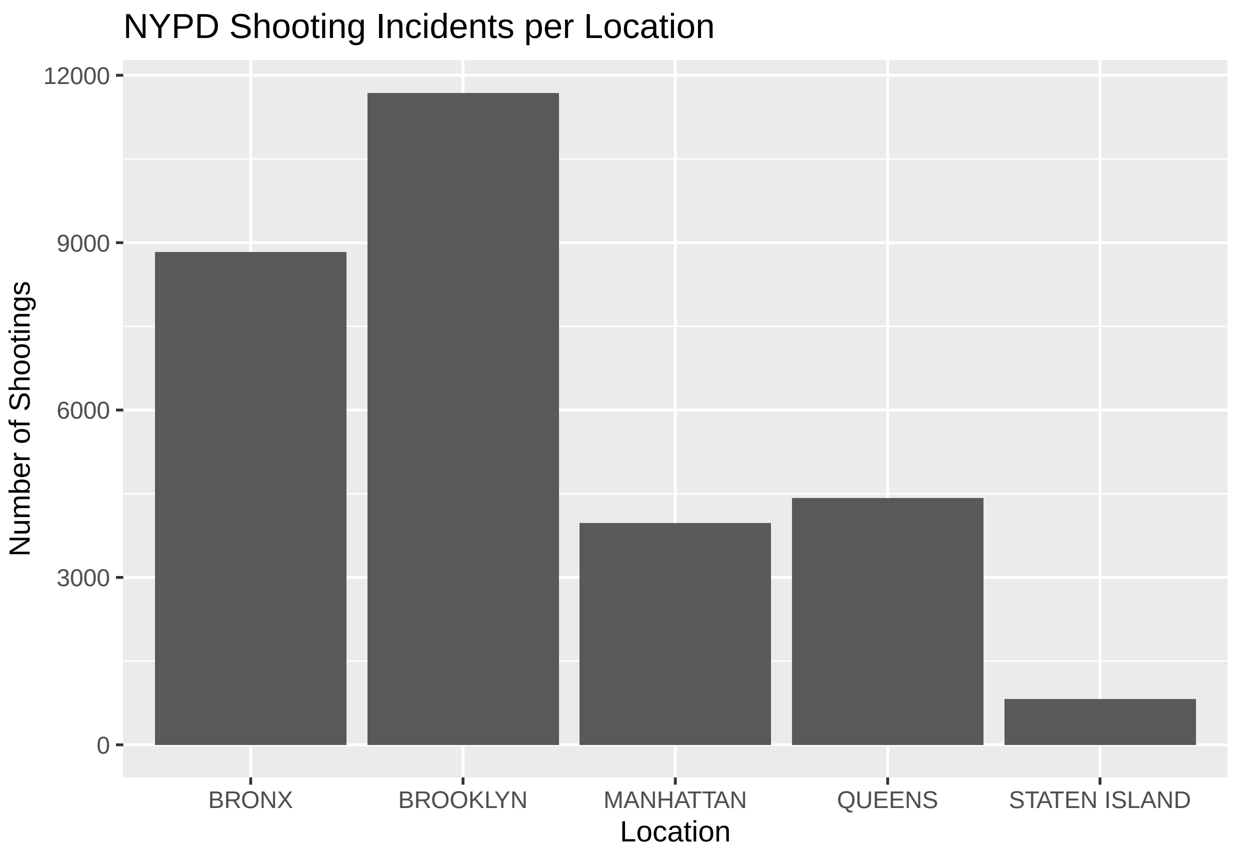
Plotting Data and Analysis

```r
NYPD_shooting %>%
  mutate(year = year(mdy(OCCUR_DATE))) %>%
  count(year) %>%
  ggplot(aes(x = year, y = n)) +
  geom_line() +
  geom_point() +
  labs(
  title = "NYPD Shooting Incidents per Year",
    x = "Year",
    y = "Number of Shootings"
  )
```



This graph shows the number of shooting incidents by year. What's interesting about this chart is that the number of shooting seems to go down over the years and spikes in 2020. What I find weird is that 2020 is around the Covid-19 incident and we were in lock down.

```
NYPD_shooting %>%
  count(BORO) %>%
  ggplot(aes(x = BORO, y = n)) +
  geom_col() +
  labs(
    title = "NYPD Shooting Incidents per Location",
    x = "Location",
    y = "Number of Shootings"
  )
```



This graph shows the amount of shooting incidents per location. We can see that there are more shooting incidents in Bronx and Brooklyn we also have to keep in mind that this is not per a certain amount of people so the data can be more skewed depending on the population density.

For this next model, I want my prediction `y` to be `STATISTICAL_MURDER_FLAG` and my predictors will be:

- `OCCUR_TIME`
- `BORO`
- `PERP_SEX`
- `PERP_AGE_GROUP`
- `PERP_RACE`

To do this, I first remove all the null columns.

```
NYPD_shooting_clean <- NYPD_shooting %>% drop_na()
colSums(is.na(NYPD_shooting_clean))
```

```
##              OCCUR_DATE              OCCUR_TIME                    BORO
##                       0                       0                       0
##           LOCATION_DESC STATISTICAL_MURDER_FLAG          PERP_AGE_GROUP
##                       0                       0                       0
##                PERP_SEX               PERP_RACE           VIC_AGE_GROUP
##                       0                       0                       0
##                 VIC_SEX                VIC_RACE              X_COORD_CD
##                       0                       0                       0
##              Y_COORD_CD                Latitude               Longitude
##                        0                        0                        0
##                  Lon_Lat
##                        0
```

Model

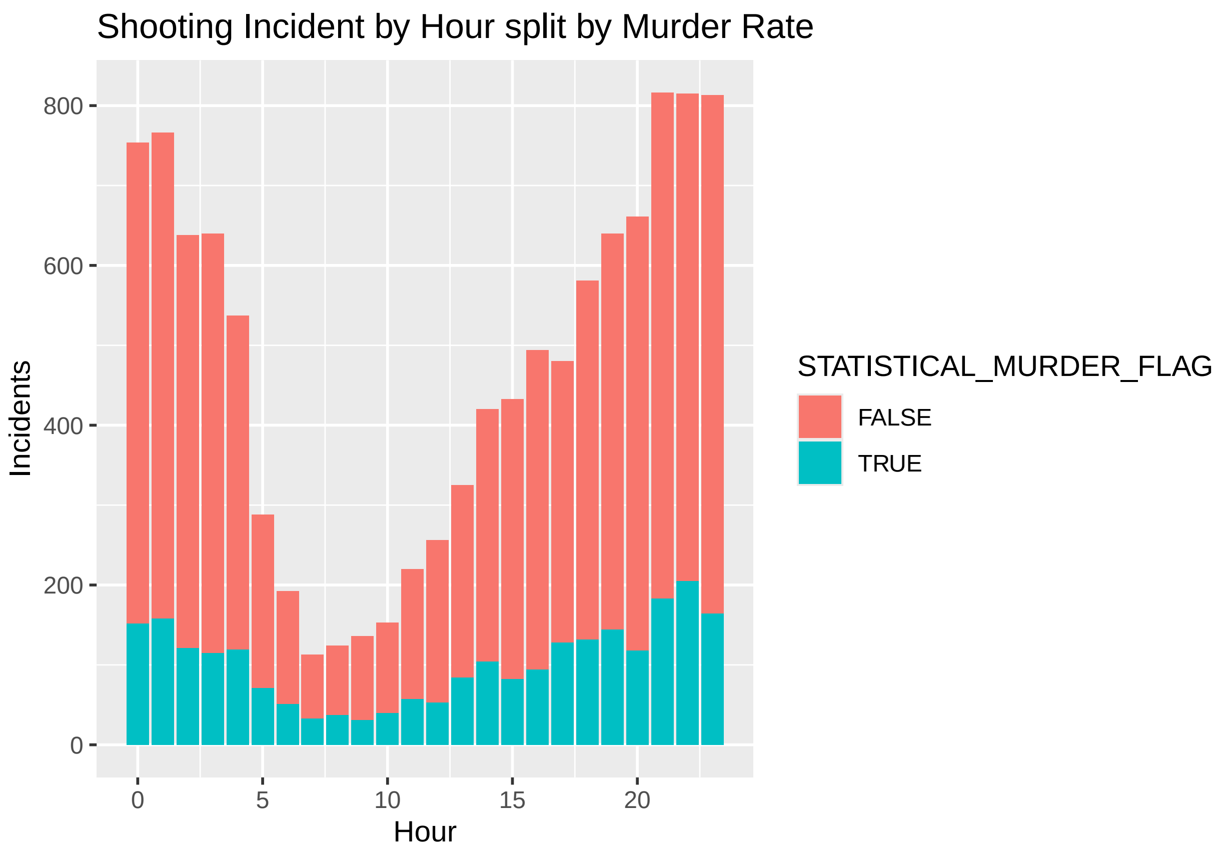
I want to see if the time of day and location matter for the murder rate. For this, I will use a GLM model

```
model <- glm(STATISTICAL_MURDER_FLAG ~ hour(OCCUR_TIME) + BORO,
             data=NYPD_shooting_clean,
             family="binomial")
summary(model)
```

```
##
## Call:
## glm(formula = STATISTICAL_MURDER_FLAG ~ hour(OCCUR_TIME) + BORO,
##     family = "binomial", data = NYPD_shooting_clean)
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -1.218197   0.054185 -22.482  < 2e-16 ***
## hour(OCCUR_TIME)   0.003794   0.002819   1.346  0.17836
## BOROBROOKLYN      -0.169282   0.055548  -3.047  0.00231 **
## BOROMANHATTAN     -0.118558   0.070638  -1.678  0.09327 .
## BOROQUEENS        -0.152269   0.072890  -2.089  0.03670 *
## BOROSTATEN ISLAND  0.034649   0.131691   0.263  0.79247
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 11880  on 11294  degrees of freedom
## Residual deviance: 11867  on 11289  degrees of freedom
## AIC: 11879
##
## Number of Fisher Scoring iterations: 4
```

From the p-value of `hour(OCCUR_TIME)`, the time doesn't seem to make a big difference to the murder chance.

```
ggplot(NYPD_shooting_clean, aes(hour(OCCUR_TIME))) + geom_bar(aes(fill=STATISTICAL_MURDER_FLAG)) +
  labs(
    title="Shooting Incident by Hour split by Murder Rate",
    x = "Hour",
    y = "Incidents"
  )
```



Visually we see that although we do see a trend of less shootings in the early hours of the day, the `TRUE` and `FALSE` are proportional at each hour. Conclusively it seems as though the murder rate doesn't change by time but by the number of incidents.

Conclusion

Throughout this report, I try to clean the data by removing unused columns and to combine columns that I see fit. I also try to present the data to gain some insight on the amount of shootings per location and per year. I also tried to fit the model with the hour of data and murder to see if murder rate to see if there is a significance.

Bias

A big bias that someone might be wary of is to be too cautious of unintentionally reinforcing stereotypes or misrepresenting groups. Because of this, one might avoid using the columns:

- `perp_sex`
- `vic_sex`
- `perp_race`
- `vic_race`

Data gathering is also a big bias. There could be shootings that go undocumented since this data set is only of those that we know of. There are also a lot of empty or null fields. This data set also includes data from when we got the Covid-19 lock down which is not mention here.