NYPD Shooting Incident Data (Historic)

Alana Hodge

2023-05-01

1. Description and importation of data.

This document uses a dataset containing a list of every shooting incident that occured in New York City going back to 2006 through the end of the previous calendar year.

This data is manually extracted every quarter and reviewed by the Office of Management Analysis and Planning before being posted on the NYPD website.

Each record in this dataset represents a shooting incident in New York City, and includes information about the event, location, time, and information related to suspect as well as victim demographics.

For this analysis, the *tidyverse* and *lubridate* packages will be utilized.

```
library(tidyverse)
library(lubridate)
```

We begin by importing the dataset as a CSV file.

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

2. Tidy and Transform your data

Remove columns from dataset that are unnecessary for the purposes of this analysis; X_COORD_CD,Y_COORD_CD, PRECINCT, Latitude, Longitude, JURISDICTION_CODE, and Lon_Lat

```
data_in = data_in %>% select(
  INCIDENT_KEY,
  OCCUR_DATE,
  OCCUR_TIME,
  BORO,
  STATISTICAL_MURDER_FLAG,
  PERP_AGE_GROUP,
  PERP_SEX,
  PERP_RACE,
  VIC_AGE_GROUP,
  VIC_SEX,
  VIC_RACE
  )

#Now we summarize the existing data to continue cleaning
summary(data_in)
```

```
##   INCIDENT_KEY        OCCUR_DATE         OCCUR_TIME           BORO
##  Min.   :  9953245   Length:27312       Length:27312      Length:27312
##  1st Qu.: 63860880   Class :character   Class1:hms        Class :character
##  Median : 90372218   Mode  :character   Class2:difftime   Mode  :character
##  Mean   :120860536                      Mode  :numeric
##  3rd Qu.:188810230
##  Max.   :261190187
##  STATISTICAL_MURDER_FLAG PERP_AGE_GROUP       PERP_SEX
##  Mode :logical           Length:27312       Length:27312
##  FALSE:22046             Class :character   Class :character
##  TRUE :5266              Mode  :character   Mode  :character
##
##
##
##   PERP_RACE         VIC_AGE_GROUP        VIC_SEX            VIC_RACE
##  Length:27312       Length:27312       Length:27312       Length:27312
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
```

Handling Missing Data

Next, we want to see any missing data that may be within the dataset. Similar to the Pandas library approach, we can use the sum of the *is.na(x)* function.

```
#Count of missing values by column:
sapply(data_in, function(x) sum(is.na(x)))
```

```
##            INCIDENT_KEY               OCCUR_DATE               OCCUR_TIME
##                       0                        0                        0
##                    BORO STATISTICAL_MURDER_FLAG           PERP_AGE_GROUP
##                       0                        0                     9344
##                PERP_SEX                PERP_RACE           VIC_AGE_GROUP
##                    9310                     9310                        0
##                 VIC_SEX                 VIC_RACE
##                       0                        0
```

We can see that the columns that contain missing values are PERP_AGE_GROUP, PERP_SEX and PERP_RACE.

For missing data,we don't want to simply remove the incident with missing data from the dataset. This is because the fact that there is information missing is information by itself; we can use missing data to draw further conclusions about the incident. In particular, noting that the missing data occurs in the PERP_AGE_GROUP, PERP_SEX and PERP_RACE columns, we can interpret this missing information as evidence that the perpetrator has yet be apprehended by authorities, and is thus unknown.

We'll simply impute a missing tokens **UNKNOWN** to fill in this missing information for the sake of visualization and modeling.

```
missing_token = "UNKNOWN"
data_in = data_in %>%
```

```
  replace_na(list(PERP_AGE_GROUP = missing_token, PERP_SEX = missing_token, PERP_RACE = missing_token))

#Sanity check to ensure that all missing values have been imputed correctly:
sapply(data_in, function(x) sum(is.na(x)))
```

```
##            INCIDENT_KEY               OCCUR_DATE               OCCUR_TIME
##                       0                        0                        0
##                    BORO STATISTICAL_MURDER_FLAG           PERP_AGE_GROUP
##                       0                        0                        0
##                PERP_SEX                PERP_RACE            VIC_AGE_GROUP
##                       0                        0                        0
##                 VIC_SEX                 VIC_RACE
##                       0                        0
```

We can see that all missing values have been imputed as intended. In the final cleaning step, we want to ensure that we change the appropriate variables to factors, and convert any column data types to the intended format.

Remove outliers and edge cases

We remove outliers from the dataset before continuing to visualization and modeling. This means identifying records of incidents where a variable was noticably inputted incorrectly. Failure to remove these outliers may skew the final results of this analysis.

Example: Values in PREP_AGE_GROUP should not exceed an age of 100 or so, based on common sense and intuition.

For this analysis, we'll focus on cleaning the categorical columns PERP_AGE_GROUP and VIC_AGE_GROUP.

First, let's view what values can occur in these columns.

```
#PERP_AGE_GROUP Unique Values
unique(data_in[["PERP_AGE_GROUP"]])
```

```
##  [1] "UNKNOWN" "25-44"   "18-24"   "45-64"   "<18"     "65+"     "940"
##  [8] "(null)"  "224"     "1020"
```

```
#VIC_AGE_GROUP Unique Values
unique(data_in[["VIC_AGE_GROUP"]])
```

```
## [1] "18-24"   "25-44"   "<18"     "45-64"   "65+"     "UNKNOWN" "1022"
```

We can see that in both age group columns, we have age-range values that do not seem to be accurate values: "1020","940", "224" and "1022".

We'll filter out these records for cleaner data.

```
data_in = data_in %>% filter(PERP_AGE_GROUP != "940",
                             PERP_AGE_GROUP != "1020",
                             PERP_AGE_GROUP != "224")

data_in = data_in %>% filter(VIC_AGE_GROUP != "1022")
```

```
#Return the unique values of these columns
#as a sanity check to ensure outlying data has been filtered out:

#PERP_AGE_GROUP Unique Values
unique(data_in[["PERP_AGE_GROUP"]])
```

```
## [1] "UNKNOWN" "25-44"   "18-24"   "45-64"   "<18"     "65+"     "(null)"
```

```
#VIC_AGE_GROUP Unique Values
unique(data_in[["VIC_AGE_GROUP"]])
```

```
## [1] "18-24"   "25-44"   "<18"     "45-64"   "65+"     "UNKNOWN"
```

We'll leave the *"(null)"* data values in PERP_AGE_GROUP alone, as the presence of these may indicate that there was no perpetrator as a result of a self-inflicted shooting.

```
numerator = nrow(data_in[data_in$STATISTICAL_MURDER_FLAG == FALSE &
 data_in$PERP_AGE_GROUP  == "(null)", ])

denomenator = nrow(data_in[data_in$PERP_AGE_GROUP == "(null)", ])

numerator / denomenator
```

```
## [1] 0.8515625
```

We can support the above hypothesis by noticing that 85% of records where "PERP_AGE_GROUP" was set to "null" are statistically unlikely to be murders, indicating that there was no perpetrator. Thus, we'll leave this value alone in cleaning.

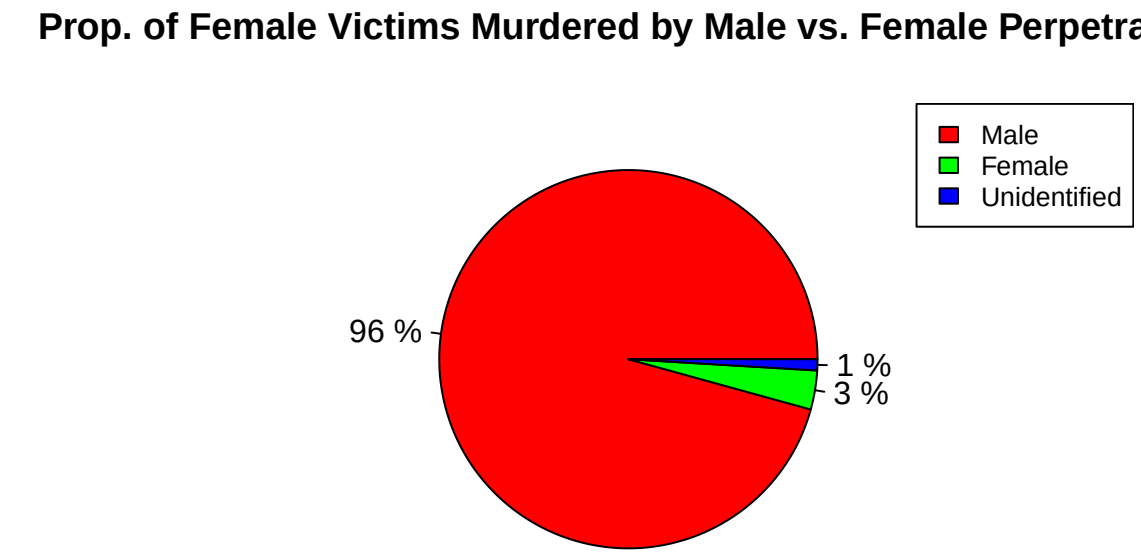
3. Add Visualizations and Analysis

Question 1:

How likely are women to be killed by men compared to women in a homicide shooting?

```
q1_data = data_in %>% filter(STATISTICAL_MURDER_FLAG == TRUE & VIC_SEX == 'F')

nFemale_Perps = nrow(q1_data[q1_data$PERP_SEX == "F", ])
nMale_Perps = nrow(q1_data[q1_data$PERP_SEX == "M", ])
nOther_Perps = nrow(q1_data[q1_data$PERP_SEX == "U", ])
total_Perps = nFemale_Perps + nMale_Perps + nOther_Perps

male_percent = paste(round(nMale_Perps / total_Perps*100), "%")
female_percent = paste(round(nFemale_Perps / total_Perps*100), "%")
other_percent = paste(round(nOther_Perps / total_Perps*100), "%")

x <- c(nMale_Perps, nFemale_Perps, nOther_Perps)
labels <- c(male_percent, female_percent, other_percent)

pie(x, labels, main ="Prop. of Female Victims Murdered by Male vs. Female Perpetrators",
```

```
    col=rainbow(length(x)))
legend("topright",
       c("Male", "Female", "Unidentified"), cex = 0.8,
       fill = rainbow(length(x)))
```

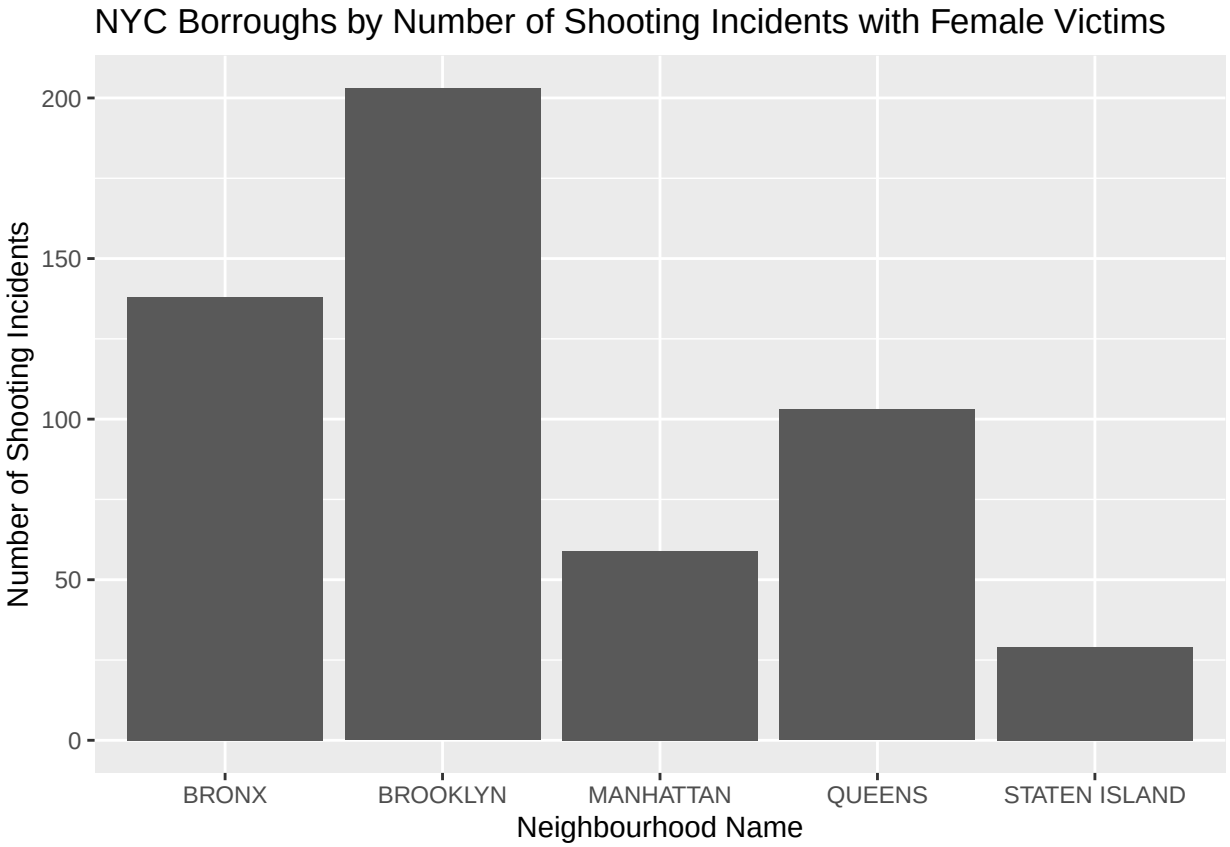


From above, we can see that a nearly all female victims in NYC that are involved in statistically determined murders have male perpetrators (~96%). This implies that women who are murdered in NYC are almost almost murdered by a man, making it very rare that the perpetrator of a shooting incident involving a female victim is also a female.

Question 2:

In which NYC Borroughs are female victims often murdered as a result of shooting incidents?

```
q2_data = q1_data
q2_plot <- ggplot(q2_data, aes(x = BORO)) +
  geom_bar() +
  labs(title = "NYC Borroughs by Number of Shooting Incidents with Female Victims",
       x = "Neighbourhood Name",
       y = "Number of Shooting Incidents")
q2_plot
```



From the visualization above, we can see that out of all incidents with female victims, the majority of these incidents occur in **Brooklyn**,with just over 200 incidents, followed closely by The **Bronx** Borrough, at just under 150.

From this analysis, we can determine that a woman is most likely to be murdered in a shooting incident in **Brooklyn** if she is invovled in a fatal shooting incident.

Moreover, we can gain an understanding of which borroughs are considered "safest" in terms of fatal shooting incidents involving woman:

In order of "Safest" to "Least Safest":

1. Staten Island
2. Manhattan
3. Queens
4. Bronx
5. Brooklyn

Modeling

We'll use a logistic regression model to see if we can predict which borough an incident occurs in based on the demographics of the victim and perpetrator.

Question: Can we predict which borough a shooting incident occurs in based on the demographic information of the victim and the perpetrator?

We'll *factor* the data that we need first, then call the generalized linear model (glm).

```r
#Factors:

data_in$BORO = as.factor(data_in$BORO)

data_in$PERP_AGE_GROUP = as.factor(data_in$PERP_AGE_GROUP)
data_in$PERP_RACE = as.factor(data_in$PERP_RACE)
data_in$PERP_SEX = as.factor(data_in$PERP_SEX)

data_in$VIC_AGE_GROUP = as.factor(data_in$VIC_AGE_GROUP)
data_in$VIC_RACE = as.factor(data_in$VIC_RACE)
data_in$VIC_SEX = as.factor(data_in$VIC_SEX)

#Call for glm model
glm.fit <- glm(BORO ~ PERP_RACE + PERP_SEX + PERP_AGE_GROUP + VIC_AGE_GROUP + VIC_RACE +
                 VIC_SEX, data = data_in, family=binomial)

summary(glm.fit)
```

```
##
## Call:
## glm(formula = BORO ~ PERP_RACE + PERP_SEX + PERP_AGE_GROUP +
##     VIC_AGE_GROUP + VIC_RACE + VIC_SEX, family = binomial, data = data_in)
##
## Coefficients: (2 not defined because of singularities)
##                                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)                           -0.02566    0.64218  -0.040 0.968121
## PERP_RACEAMERICAN INDIAN/ALASKAN NATIVE -0.79956    1.42460  -0.561 0.574623
## PERP_RACEASIAN / PACIFIC ISLANDER      0.68967    0.25295   2.726 0.006402
## PERP_RACEBLACK                         0.58599    0.14955   3.918 8.92e-05
## PERP_RACEBLACK HISPANIC               -0.11999    0.15892  -0.755 0.450210
## PERP_RACEUNKNOWN                       0.05382    0.09259   0.581 0.561041
## PERP_RACEWHITE                         0.97719    0.22335   4.375 1.21e-05
## PERP_RACEWHITE HISPANIC               -0.01064    0.15478  -0.069 0.945178
## PERP_SEXF                             -0.16480    0.16266  -1.013 0.310970
## PERP_SEXM                             -0.31609    0.11569  -2.732 0.006292
## PERP_SEXU                              0.04328    0.06433   0.673 0.501120
## PERP_SEXUNKNOWN                             NA         NA      NA       NA
## PERP_AGE_GROUP<18                     -0.20772    0.08124  -2.557 0.010557
## PERP_AGE_GROUP18-24                   -0.19226    0.06512  -2.952 0.003153
## PERP_AGE_GROUP25-44                   -0.16631    0.06636  -2.506 0.012202
## PERP_AGE_GROUP45-64                   -0.36503    0.10944  -3.336 0.000851
## PERP_AGE_GROUP65+                      0.43402    0.37239   1.165 0.243819
## PERP_AGE_GROUPUNKNOWN                       NA         NA      NA       NA
## VIC_AGE_GROUP18-24                     0.11499    0.04769   2.411 0.015909
## VIC_AGE_GROUP25-44                     0.22275    0.04772   4.668 3.05e-06
## VIC_AGE_GROUP45-64                     0.25901    0.06982   3.710 0.000207
## VIC_AGE_GROUP65+                       0.20671    0.18040   1.146 0.251883
## VIC_AGE_GROUPUNKNOWN                   0.32691    0.31230   1.047 0.295190
## VIC_RACEASIAN / PACIFIC ISLANDER       1.77293    0.65099   2.723 0.006461
## VIC_RACEBLACK                          1.15466    0.63493   1.819 0.068978
## VIC_RACEBLACK HISPANIC                 0.18254    0.63600   0.287 0.774101
## VIC_RACEUNKNOWN                        0.68932    0.69263   0.995 0.319623
## VIC_RACEWHITE                          1.44016    0.64322   2.239 0.025157
```

```
## VIC_RACEWHITE HISPANIC                  0.27446    0.63560   0.432 0.665879
## VIC_SEXM                               -0.15059    0.04854  -3.102 0.001920
## VIC_SEXU                               -0.17009    0.71072  -0.239 0.810862
##
## (Intercept)
## PERP_RACEAMERICAN INDIAN/ALASKAN NATIVE
## PERP_RACEASIAN / PACIFIC ISLANDER       **
## PERP_RACEBLACK                          ***
## PERP_RACEBLACK HISPANIC
## PERP_RACEUNKNOWN
## PERP_RACEWHITE                          ***
## PERP_RACEWHITE HISPANIC
## PERP_SEXF
## PERP_SEXM                               **
## PERP_SEXU
## PERP_SEXUNKNOWN
## PERP_AGE_GROUP<18                       *
## PERP_AGE_GROUP18-24                     **
## PERP_AGE_GROUP25-44                     *
## PERP_AGE_GROUP45-64                     ***
## PERP_AGE_GROUP65+
## PERP_AGE_GROUPUNKNOWN
## VIC_AGE_GROUP18-24                      *
## VIC_AGE_GROUP25-44                      ***
## VIC_AGE_GROUP45-64                      ***
## VIC_AGE_GROUP65+
## VIC_AGE_GROUPUNKNOWN
## VIC_RACEASIAN / PACIFIC ISLANDER        **
## VIC_RACEBLACK                           .
## VIC_RACEBLACK HISPANIC
## VIC_RACEUNKNOWN
## VIC_RACEWHITE                           *
## VIC_RACEWHITE HISPANIC
## VIC_SEXM                                **
## VIC_SEXU
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 32915  on 27307  degrees of freedom
## Residual deviance: 31223  on 27279  degrees of freedom
## AIC: 31281
##
## Number of Fisher Scoring iterations: 4
```

4. Add Bias Identification

Data Bias:

- Variables described here as relating to a victim or perpetrator's race may have bias. Definitions of racial-identity change overtime, is unclear whether or not the individual actually identifies as the race

they have been identified as in this dataset, or whether or not they are actually a combination of multiple races, and how mixed-race individuals are being identified in this dataset.

- The data does not contain any variable for understanding the context of the incident. For example, it's unclear whether or not the perpetrator acted out of self-defense. Context towards the full investigation of these shooting incidents could drastically change how this data and the resulting analysis is perceived.

Personal Bias:

In a topic like this that relies heavily on the demographics of the people involved, discrimination and implicit bias can seep into the perception of the data. I am aware that I am not immune to subconcious biases towards people due to their demographics, but in order to mitigate this, I avoided over-analyzing incidents relating to the demographics of those involved, and instead focused on just ensuring that the data was cleaned and handled with care, without diving into specifics or reading particular incident reports.

5. External Resources Used for Reference:

1. https://stackoverflow.com/questions/13613913/how-do-i-convert-certain-columns-of-a-data-frame-to-become-factors
2. https://ismayc.github.io/rbasics-book/5-rmdanal.html#data-structures
3. https://www.educative.io/answers/how-to-access-the-columns-of-a-data-frame-in-r
4. https://humansofdata.atlan.com/2018/03/when-delete-outliers-dataset/
5. https://dplyr.tidyverse.org/reference/filter.html
6. https://www.statology.org/r-count-values-in-column-with-condition/
7. http://statseducation.com/Introduction-to-R/modules/tidy%20data/filter/
8. https://www.tutorialspoint.com/r/r_pie_charts.htm

Session info (as output from R) for Reproducibility

```
sessionInfo()
```

```
## R version 4.3.0 (2023-04-21 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 11 x64 (build 22621)
##
## Matrix products: default
##
##
## locale:
## [1] LC_COLLATE=English_Canada.utf8  LC_CTYPE=English_Canada.utf8
## [3] LC_MONETARY=English_Canada.utf8 LC_NUMERIC=C
## [5] LC_TIME=English_Canada.utf8
##
## time zone: America/Toronto
## tzcode source: internal
##
## attached base packages:
```

```
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] lubridate_1.9.2 forcats_1.0.0   stringr_1.5.0   dplyr_1.1.2
##  [5] purrr_1.0.1     readr_2.1.4     tidyr_1.3.0     tibble_3.2.1
##  [9] ggplot2_3.4.2   tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
##  [1] bit_4.0.5        gtable_0.3.3     highr_0.10       crayon_1.5.2
##  [5] compiler_4.3.0   tidyselect_1.2.0 parallel_4.3.0   scales_1.2.1
##  [9] yaml_2.3.7       fastmap_1.1.1    R6_2.5.1         labeling_0.4.2
## [13] generics_0.1.3   curl_5.0.0       knitr_1.42       munsell_0.5.0
## [17] pillar_1.9.0     tzdb_0.3.0       rlang_1.1.1      utf8_1.2.3
## [21] stringi_1.7.12   xfun_0.39        bit64_4.0.5      timechange_0.2.0
## [25] cli_3.6.1        withr_2.5.0      magrittr_2.0.3   digest_0.6.31
## [29] grid_4.3.0       vroom_1.6.3      rstudioapi_0.14  hms_1.1.3
## [33] lifecycle_1.0.3  vctrs_0.6.2      evaluate_0.20    glue_1.6.2
## [37] farver_2.1.1     fansi_1.0.4      colorspace_2.1-0 rmarkdown_2.21
## [41] tools_4.3.0      pkgconfig_2.0.3  htmltools_0.5.5
```