DTSA 5301 - NYPD Shooting Incident Data Report

Ashish Bhutiani

6/20/2021

Question

Are shootings involving younger people more prevalent in different boroughs within New York City?

Data Source and Summary

In order to attempt to answer this question, we will be using a dataset provided by data.gov. This dataset is a breakdown of every shooting incident that occurred in NYC 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 represents a shooting incident in NYC and includes information about the event, the location and time of occurrence. In addition, information related to suspect and victim demographics is also included.

The URL for the dataset that will be loading in is: https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD

Each record is labeled by an Incident key, and originally contains the following information on the shooting:

- The date of the shooting
- The time the shooting occurred
- Which boro the shooting occured in
- The NYPD Precinct Number
- The Jurisdiction Code
- A description of the location
- A Statistical murder flag
- The Perpetrator's age group
- The Perpetrator's Sex
- The Perpetrator's Race
- The Victim's age group
- The Victim's Sex
- The Victim's Race
- The X coordinate
- The Y coordinate
- The Latitude
- The Longitude
- The longitude and latitude point

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

```
##   INCIDENT_KEY        OCCUR_DATE         OCCUR_TIME            BORO
##  Min.   :  9953245   Length:23568       Length:23568       Length:23568
##  1st Qu.: 55317014   Class :character   Class :character   Class :character
##  Median : 83365370   Mode  :character   Mode  :character   Mode  :character
##  Mean   :102218616
##  3rd Qu.:150772442
##  Max.   :222473262
##
##     PRECINCT      JURISDICTION_CODE LOCATION_DESC      STATISTICAL_MURDER_FLAG
##  Min.   :  1.00   Min.   :0.0000    Length:23568       Length:23568
##  1st Qu.: 44.00   1st Qu.:0.0000    Class :character   Class :character
##  Median : 69.00   Median :0.0000    Mode  :character   Mode  :character
##  Mean   : 66.21   Mean   :0.3323
##  3rd Qu.: 81.00   3rd Qu.:0.0000
##  Max.   :123.00   Max.   :2.0000
##                   NA's   :2
##  PERP_AGE_GROUP       PERP_SEX          PERP_RACE         VIC_AGE_GROUP
##  Length:23568       Length:23568       Length:23568       Length:23568
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    VIC_SEX            VIC_RACE          X_COORD_CD         Y_COORD_CD
##  Length:23568       Length:23568       Length:23568       Length:23568
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##     Latitude       Longitude        Lon_Lat
##  Min.   :40.51   Min.   :-74.25   Length:23568
##  1st Qu.:40.67   1st Qu.:-73.94   Class :character
##  Median :40.70   Median :-73.92   Mode  :character
##  Mean   :40.74   Mean   :-73.91
##  3rd Qu.:40.82   3rd Qu.:-73.88
##  Max.   :40.91   Max.   :-73.70
##
```

Data Cleanup

As we can see from the summary, there are a lot of columns in this dataset that we don't need. For our analysis, we need the boro and the victim's age group, but we can also keep the date, time, and murder flag for more information.

We will also rename the date time, and murder flag columns and do some type conversion for those fields as well. The cleaned dataset summary now looks like this:

```
cleaned_dataset <- dataset %>%
    select(OCCUR_DATE, OCCUR_TIME, BORO, VIC_AGE_GROUP, STATISTICAL_MURDER_FLAG) %>%
    rename(DATE = "OCCUR_DATE", TIME = "OCCUR_TIME", MURDER_FLAG = "STATISTICAL_MURDER_FLAG") %>%
    mutate(DATE = mdy(DATE), MURDER_FLAG = as.logical(MURDER_FLAG))
summary(cleaned_dataset)
```

```
##       DATE                TIME               BORO           VIC_AGE_GROUP     
##  Min.   :2006-01-01   Length:23568       Length:23568       Length:23568      
##  1st Qu.:2008-12-30   Class :character   Class :character   Class :character  
##  Median :2012-02-26   Mode  :character   Mode  :character   Mode  :character  
##  Mean   :2012-10-03                                                           
##  3rd Qu.:2016-02-28                                                           
##  Max.   :2020-12-31                                                           
##  MURDER_FLAG    
##  Mode :logical  
##  FALSE:19080    
##  TRUE :4488     
##                 
##                 
## 
```

Analysis and Visualization

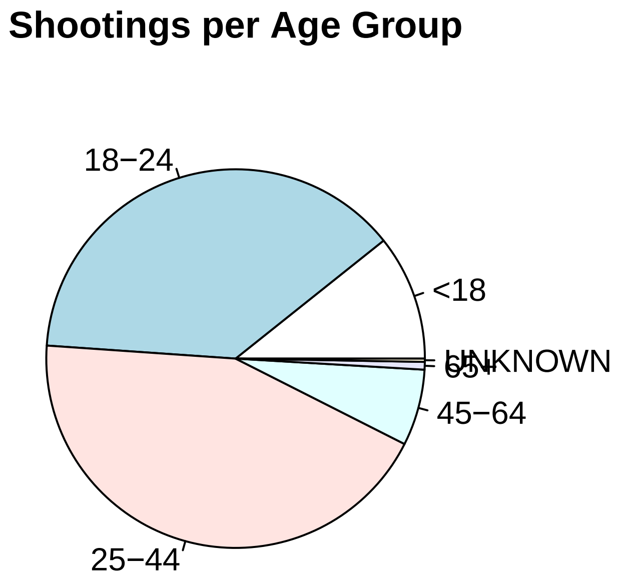
Now let's look at what values we have in our data:

```
head(cleaned_dataset)
```

```
##         DATE     TIME          BORO VIC_AGE_GROUP MURDER_FLAG
## 1 2019-08-23 22:10:00        QUEENS         25-44       FALSE
## 2 2019-11-27 15:54:00         BRONX         25-44       FALSE
## 3 2019-02-02 19:40:00     MANHATTAN         18-24       FALSE
## 4 2019-10-24 00:52:00 STATEN ISLAND         25-44        TRUE
## 5 2019-08-22 18:03:00         BRONX         18-24       FALSE
## 6 2019-06-07 17:50:00      BROOKLYN         25-44       FALSE
```

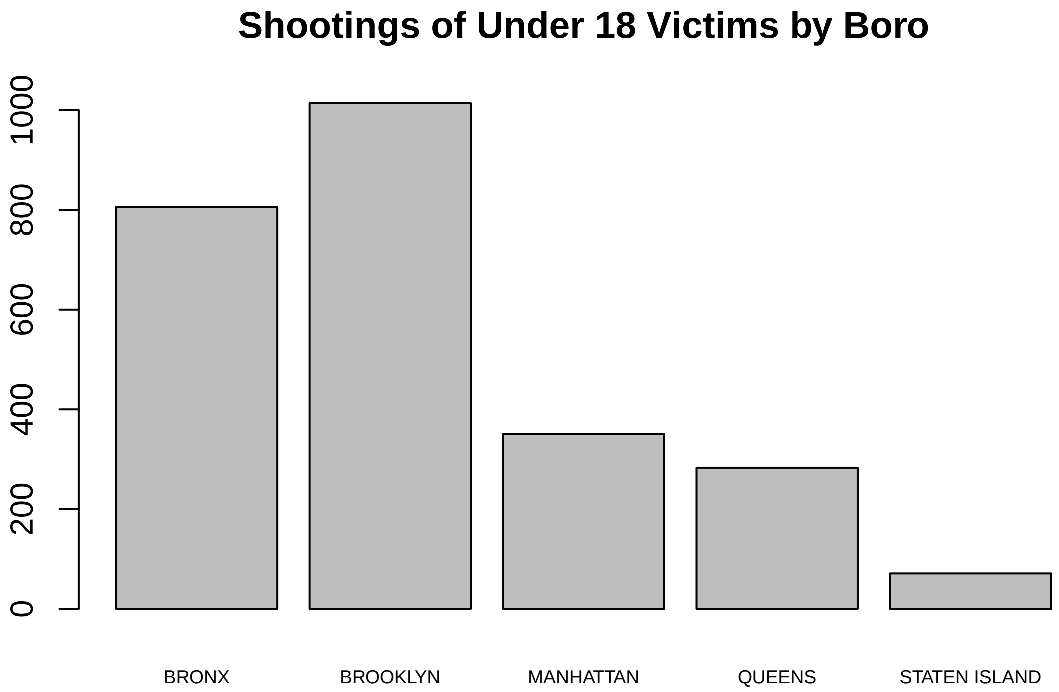
Let's see what values are in VIC_AGE_GROUP field by using a pie chart:

```
age_group_counts <- count(cleaned_dataset, VIC_AGE_GROUP = cleaned_dataset$VIC_AGE_GROUP)
pie(age_group_counts$n, main = "Shootings per Age Group", labels = age_group_counts$VIC_AGE_GROUP)
```



For our purposes we are only going to look at the Under 18 Age Group (< 18), so let's filter our dataset and see what the breakdown by boro is.

```
under18 <- filter(cleaned_dataset, cleaned_dataset$VIC_AGE_GROUP ==
    "<18")
under18_counts <- count(under18, BORO = under18$BORO)
barplot(under18_counts$n, main = "Shootings of Under 18 Victims by Boro",
    names.arg = under18_counts$BORO, cex.names = 0.6)
```

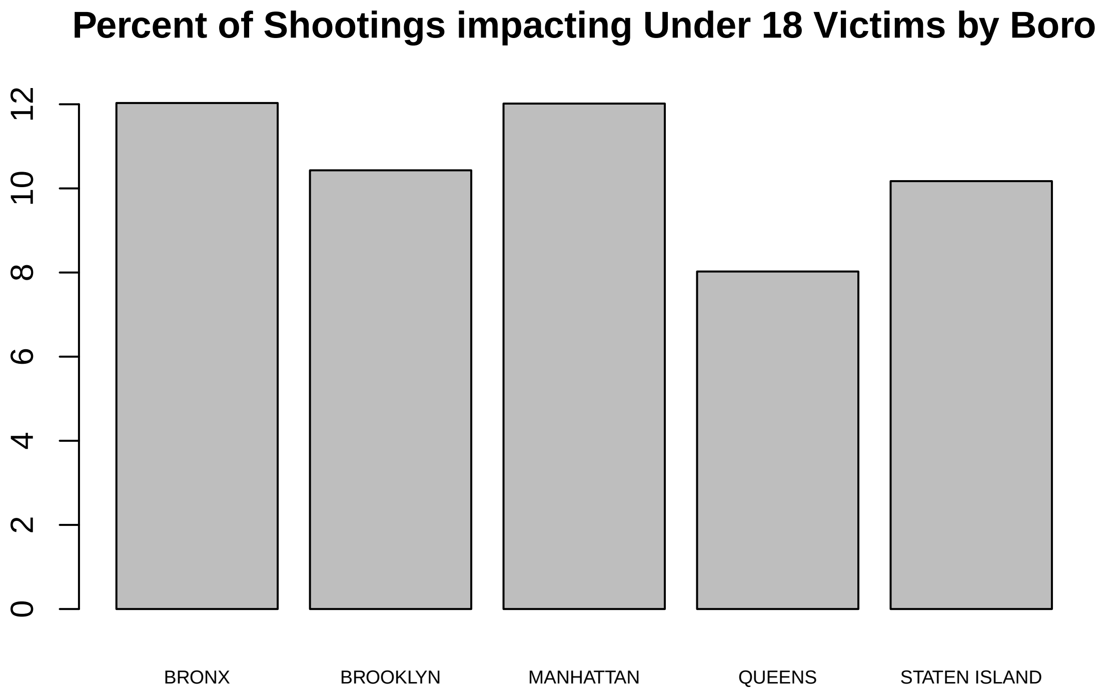


This shows that as absolute number, Brooklyn had the most shootings of children as a Boro, but to really see if those shootings are more prevalent, we need to see how these numbers compare to the total shootings in the boro, so we can do a percentage analysis.

```
boro_counts <- count(cleaned_dataset, BORO = cleaned_dataset$BORO)
under18_counts$total_shootings <- boro_counts$n
under18_counts <- transform(under18_counts, perc = n/total_shootings *
    100)
under18_counts
```

```
##            BORO    n total_shootings      perc
## 1         BRONX  806            6700 12.029851
## 2      BROOKLYN 1014            9722 10.429953
## 3     MANHATTAN  351            2921 12.016433
## 4        QUEENS  283            3527  8.023816
## 5 STATEN ISLAND   71             698 10.171920
```

```
barplot(under18_counts$perc, main = "Percent of Shootings impacting Under 18 Victims by Boro",
    names.arg = under18_counts$BORO, cex.names = 0.6)
```



Conclusion and Bias

When we accounted for the total number of shootings, we see that the large differences from the raw numbers go away and Brooklyn isn't as bad as it first seemed. Queens has the lowest percentage at 8%.

In terms of Bias, I didn't have a preconceived ideas of where the data would take me, but I think further analyis on the population differences between the boros might prove that Queens isn't safer if for example it has less kids as a whole. It would be an interesting discussion in the future.

Session Info

```
sessionInfo()
```

```
## R version 4.1.0 (2021-05-18)
## Platform: x86_64-apple-darwin20.4.0 (64-bit)
## Running under: macOS Big Sur 11.4
##
## Matrix products: default
## BLAS:   /usr/local/Cellar/openblas/0.3.15_1/lib/libopenblasp-r0.3.15.dylib
## LAPACK: /usr/local/Cellar/r/4.1.0/lib/R/lib/libRlapack.dylib
##
## locale:
```

```
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] lubridate_1.7.10 forcats_0.5.1    stringr_1.4.0    dplyr_1.0.7
##  [5] purrr_0.3.4      readr_1.4.0      tidyr_1.1.3      tibble_3.1.2
##  [9] ggplot2_3.3.4    tidyverse_1.3.1
##
## loaded via a namespace (and not attached):
##  [1] tidyselect_1.1.1  xfun_0.24         haven_2.4.1       colorspace_2.0-1
##  [5] vctrs_0.3.8       generics_0.1.0    htmltools_0.5.1.1 yaml_2.2.1
##  [9] utf8_1.2.1        rlang_0.4.11      pillar_1.6.1      glue_1.4.2
## [13] withr_2.4.2       DBI_1.1.1         dbplyr_2.1.1      modelr_0.1.8
## [17] readxl_1.3.1      lifecycle_1.0.0   munsell_0.5.0     gtable_0.3.0
## [21] cellranger_1.1.0  rvest_1.0.0       evaluate_0.14     knitr_1.33
## [25] fansi_0.5.0       highr_0.9         broom_0.7.7       Rcpp_1.0.6
## [29] formatR_1.11      scales_1.1.1      backports_1.2.1   jsonlite_1.7.2
## [33] fs_1.5.0          hms_1.1.0         digest_0.6.27     stringi_1.6.2
## [37] grid_4.1.0        cli_2.5.0         tools_4.1.0       magrittr_2.0.1
## [41] crayon_1.4.1      pkgconfig_2.0.3   ellipsis_0.3.2    xml2_1.3.2
## [45] reprex_2.0.0      assertthat_0.2.1  rmarkdown_2.9     httr_1.4.2
## [49] rstudioapi_0.13   R6_2.5.0          compiler_4.1.0
```