NYPD_Shooting_Incident

Liang Yam

2022-09-25

Step 1 - Identify and import data

My first step is to import the data into R.

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

```
## Rows: 25596 Columns: 19
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (10): OCCUR_DATE, BORO, LOCATION_DESC, PERP_AGE_GROUP, PERP_SEX, PERP_R...
## dbl   (7): INCIDENT_KEY, PRECINCT, JURISDICTION_CODE, X_COORD_CD, Y_COORD_CD...
## lgl   (1): STATISTICAL_MURDER_FLAG
## time  (1): OCCUR_TIME
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
NYPD_Shootiing_Incident
```

```
## # A tibble: 25,596 x 19
##    INCID~1 OCCUR~2 OCCUR~3 BORO  PRECI~4 JURIS~5 LOCAT~6 STATI~7 PERP_~8 PERP_~9
##      <dbl> <chr>   <time>  <chr>   <dbl>   <dbl> <chr>   <lgl>   <chr>   <chr>
##  1  2.36e8 11/11/~ 15:04   BROO~      79       0 <NA>    FALSE   <NA>    <NA>
##  2  2.31e8 07/16/~ 22:05   BROO~      72       0 <NA>    FALSE   45-64   M
##  3  2.31e8 07/11/~ 01:09   BROO~      79       0 <NA>    FALSE   <18     M
##  4  2.38e8 12/11/~ 13:42   BROO~      81       0 <NA>    FALSE   <NA>    <NA>
##  5  2.24e8 02/16/~ 20:00   QUEE~     113       0 <NA>    FALSE   <NA>    <NA>
##  6  2.28e8 05/15/~ 04:13   QUEE~     113       0 <NA>    TRUE    <NA>    <NA>
##  7  2.27e8 04/14/~ 21:08   BRONX      42       0 COMMER~ TRUE    <NA>    <NA>
##  8  2.38e8 12/10/~ 19:30   BRONX      52       0 <NA>    FALSE   <NA>    <NA>
##  9  2.25e8 02/22/~ 00:18   MANH~      34       0 <NA>    FALSE   <NA>    <NA>
## 10  2.25e8 03/07/~ 06:15   BROO~      75       0 <NA>    TRUE    25-44   M
## # ... with 25,586 more rows, 9 more variables: 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>, and
## #   abbreviated variable names 1: INCIDENT_KEY, 2: OCCUR_DATE, 3: OCCUR_TIME,
## #   4: PRECINCT, 5: JURISDICTION_CODE, 6: LOCATION_DESC,
## #   7: STATISTICAL_MURDER_FLAG, 8: PERP_AGE_GROUP, 9: PERP_SEX
```

I will not use the X_COORD_CD, Y_COORD_CD, Latitude, Longitude in my analysis. In addition, JURISDICTION_CODE is the location of the incident, where 0 represents patrol, 1 represents transit, 2 represents housing, and anything above 2 is outside of NYPD jurisdiction.

```r
NYPD_Shootiing_Incident <- NYPD_Shootiing_Incident %>%
  select(-c(X_COORD_CD:Lon_Lat)) %>%
  mutate(OCCUR_DATE = mdy(OCCUR_DATE),
         JURISDICTION_CODE = case_when(JURISDICTION_CODE == 0 ~ 'Patrol',
                                       JURISDICTION_CODE == 1 ~ 'Transit',
                                       JURISDICTION_CODE == 2 ~ 'Housing',
                                       JURISDICTION_CODE > 2 ~ 'Non NYPD jurisdictions'))
NYPD_Shootiing_Incident
```

```
## # A tibble: 25,596 x 14
##    INCIDENT_KEY OCCUR_DATE OCCUR~1 BORO  PRECI~2 JURIS~3 LOCAT~4 STATI~5 PERP_~6
##           <dbl> <date>     <time>  <chr>   <dbl> <chr>   <chr>   <lgl>   <chr>
##  1    236168668 2021-11-11 15:04   BROO~      79 Patrol  <NA>    FALSE   <NA>
##  2    231008085 2021-07-16 22:05   BROO~      72 Patrol  <NA>    FALSE   45-64
##  3    230717903 2021-07-11 01:09   BROO~      79 Patrol  <NA>    FALSE   <18
##  4    237712309 2021-12-11 13:42   BROO~      81 Patrol  <NA>    FALSE   <NA>
##  5    224465521 2021-02-16 20:00   QUEE~     113 Patrol  <NA>    FALSE   <NA>
##  6    228252164 2021-05-15 04:13   QUEE~     113 Patrol  <NA>    TRUE    <NA>
##  7    226950018 2021-04-14 21:08   BRONX      42 Patrol  COMMER~ TRUE    <NA>
##  8    237710987 2021-12-10 19:30   BRONX      52 Patrol  <NA>    FALSE   <NA>
##  9    224701998 2021-02-22 00:18   MANH~      34 Patrol  <NA>    FALSE   <NA>
## 10    225295736 2021-03-07 06:15   BROO~      75 Patrol  <NA>    TRUE    25-44
## # ... with 25,586 more rows, 5 more variables: PERP_SEX <chr>, PERP_RACE <chr>,
## #   VIC_AGE_GROUP <chr>, VIC_SEX <chr>, VIC_RACE <chr>, and abbreviated
## #   variable names 1: OCCUR_TIME, 2: PRECINCT, 3: JURISDICTION_CODE,
## #   4: LOCATION_DESC, 5: STATISTICAL_MURDER_FLAG, 6: PERP_AGE_GROUP
```

I want to first look at a summary of this table and understand some descriptive statistics of each of the columns and validate the data.

```r
summary(NYPD_Shootiing_Incident)
```

```
##   INCIDENT_KEY         OCCUR_DATE          OCCUR_TIME           BORO
##  Min.   :  9953245   Min.   :2006-01-01   Length:25596      Length:25596
##  1st Qu.: 61593633   1st Qu.:2009-05-10   Class1:hms        Class :character
##  Median : 86437258   Median :2012-08-26   Class2:difftime   Mode  :character
##  Mean   :112382648   Mean   :2013-06-13   Mode  :numeric
##  3rd Qu.:166660833   3rd Qu.:2017-07-01
##  Max.   :238490103   Max.   :2021-12-31
##     PRECINCT      JURISDICTION_CODE  LOCATION_DESC      STATISTICAL_MURDER_FLAG
##  Min.   :  1.00   Length:25596       Length:25596       Mode :logical
##  1st Qu.: 44.00   Class :character   Class :character   FALSE:20668
##  Median : 69.00   Mode  :character   Mode  :character   TRUE :4928
##  Mean   : 65.87
##  3rd Qu.: 81.00
##  Max.   :123.00
##  PERP_AGE_GROUP       PERP_SEX          PERP_RACE         VIC_AGE_GROUP
##  Length:25596       Length:25596       Length:25596       Length:25596
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##    VIC_SEX            VIC_RACE
##  Length:25596       Length:25596
```

```
##  Class :character   Class :character
##  Mode  :character   Mode  :character
##
##
##
```

From the summary, I noticed that we have some data from Jan 2006 to December 2021. It also appears that majority of the columns are String variables.

Step 2 - Analysis

There are a few questions that intrigued me when looking at this data. My first analysis is understanding the fatal crimes, specifically the number of fatal crimes that are committed in each year for each borough.

I will first look at the count of yearly fatal crimes in each of the boroughs.

```
NYPD_borough_fatal <- NYPD_Shootiing_Incident %>%
  filter(STATISTICAL_MURDER_FLAG == TRUE) %>%
  mutate(year_occur = year(OCCUR_DATE)) %>%
  group_by(BORO, year_occur) %>%
  summarize(crimes = n())
```

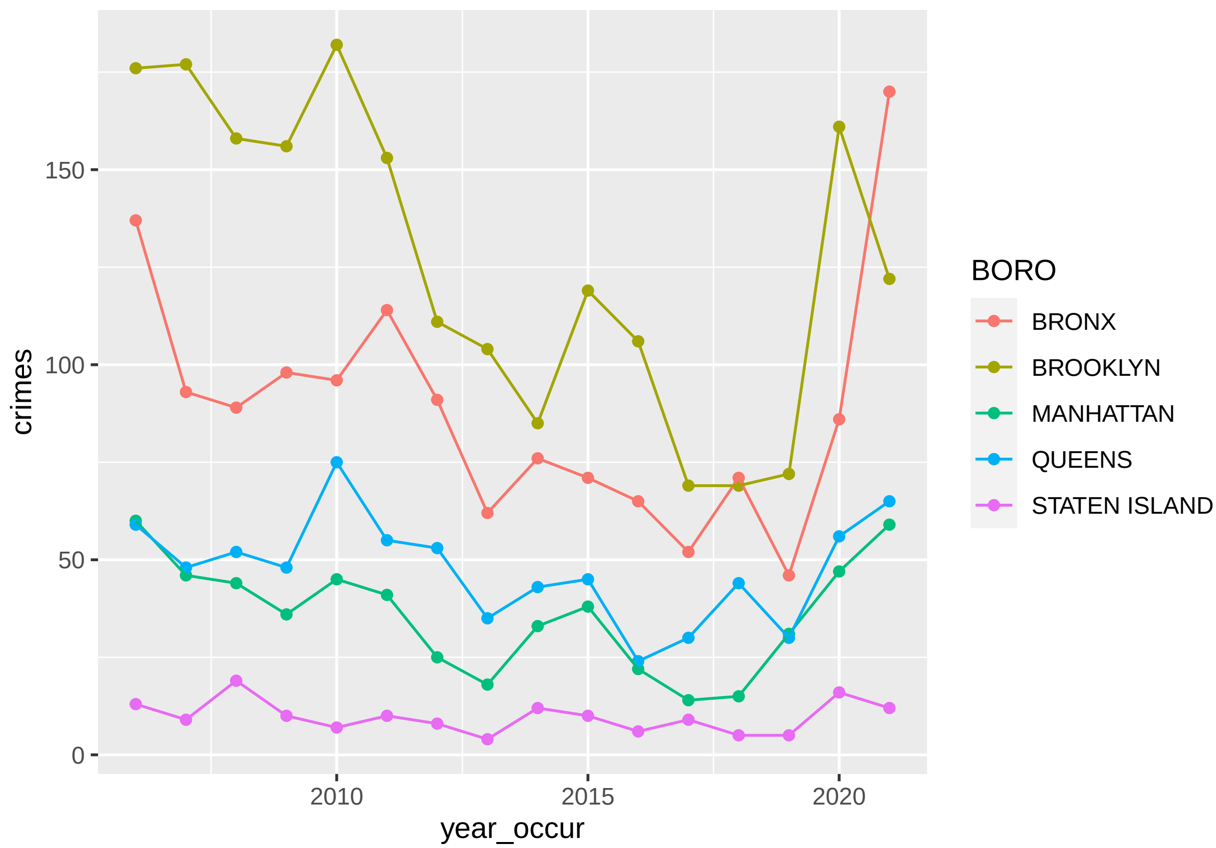
```
## `summarise()` has grouped output by 'BORO'. You can override using the
## `.groups` argument.
```

```
NYPD_borough_fatal
```

```
## # A tibble: 80 x 3
## # Groups:   BORO [5]
##    BORO  year_occur crimes
##    <chr>      <dbl>  <int>
##  1 BRONX       2006    137
##  2 BRONX       2007     93
##  3 BRONX       2008     89
##  4 BRONX       2009     98
##  5 BRONX       2010     96
##  6 BRONX       2011    114
##  7 BRONX       2012     91
##  8 BRONX       2013     62
##  9 BRONX       2014     76
## 10 BRONX       2015     71
## # ... with 70 more rows
```

I will graph the fatal crimes and identify if there are any trends in the data.

```
NYPD_borough_fatal %>%
  ggplot(aes(x = year_occur)) + geom_point(aes(y = crimes, color = BORO)) + geom_line(aes(y = crimes, co
```



Looking at the plot, there seems to be a decreasing trend of fatal crimes from 2006 to 2018, but started to increase and spike starting in 2019 and 2020. I am not surprised by this trend because of the movements and COVID in 2019 and 2020.

I will now look at all crimes (both fatal and non-fatal) and see if the trend matches to fatal crime. I will first count the number of crimes in each borough

```
NYPD_borough <- NYPD_Shootiing_Incident %>%
  mutate(year_occur = year(OCCUR_DATE)) %>%
  group_by(BORO, year_occur) %>%
  summarize(crimes = n())
```

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

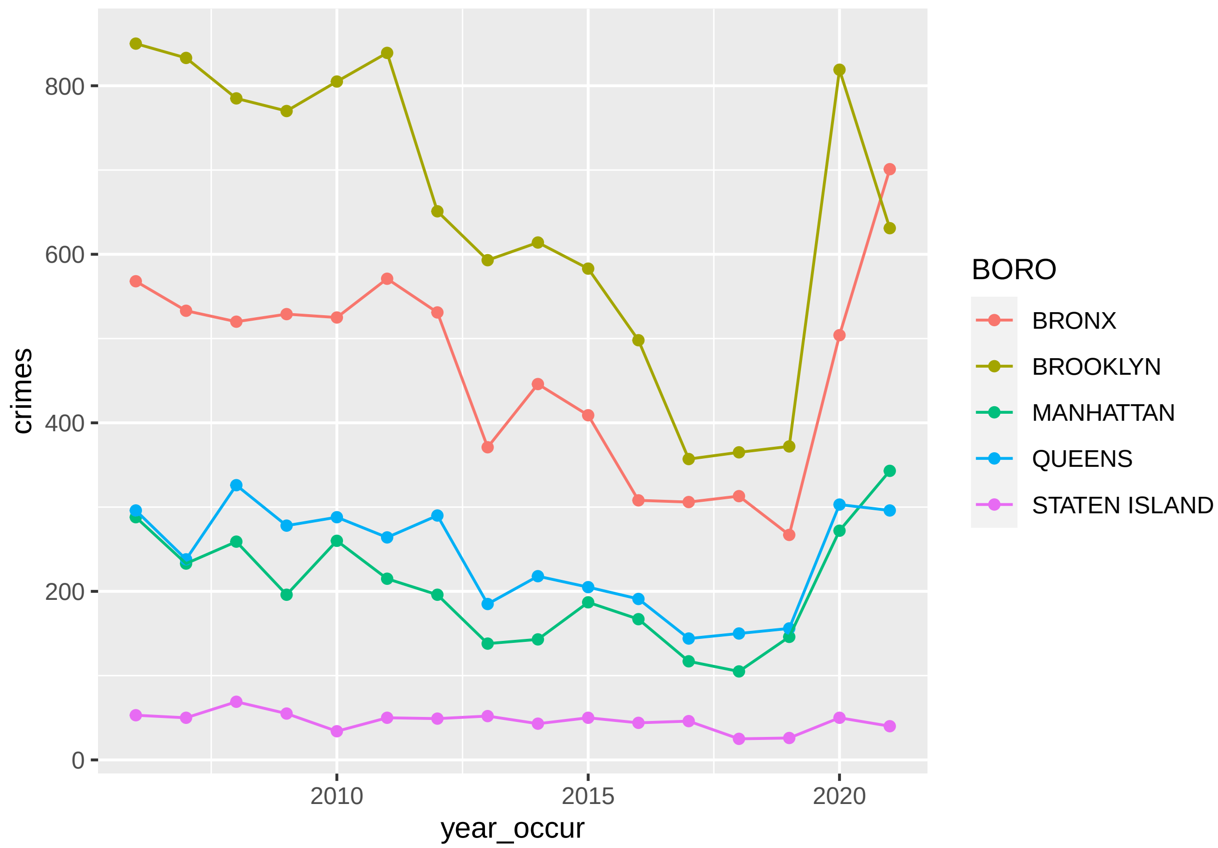
```
NYPD_borough
```

```
## # A tibble: 80 x 3
## # Groups:   BORO [5]
##    BORO  year_occur crimes
##    <chr>      <dbl>  <int>
##  1 BRONX       2006    568
##  2 BRONX       2007    533
##  3 BRONX       2008    520
##  4 BRONX       2009    529
##  5 BRONX       2010    525
##  6 BRONX       2011    571
##  7 BRONX       2012    531
```

```
## 8 BRONX       2013    371
## 9 BRONX       2014    446
## 10 BRONX      2015    409
## # ... with 70 more rows
```

And then graph each borough's annual number of crime

```
NYPD_borough %>%
  ggplot(aes(x = year_occur)) + geom_point(aes(y = crimes, color = BORO)) + geom_line(aes(y = crimes, co
```



It seems like the overall pattern remains the same, where there is a decreasing trend from 2006 to 2018, then an increasing trend from 2019 to 2021.

My second analysis will focus on understanding how victim's identity will correlate to crime's fatality.

Let's first take a look at how victim's age group correlate with fatal vs non-fatal crimes.

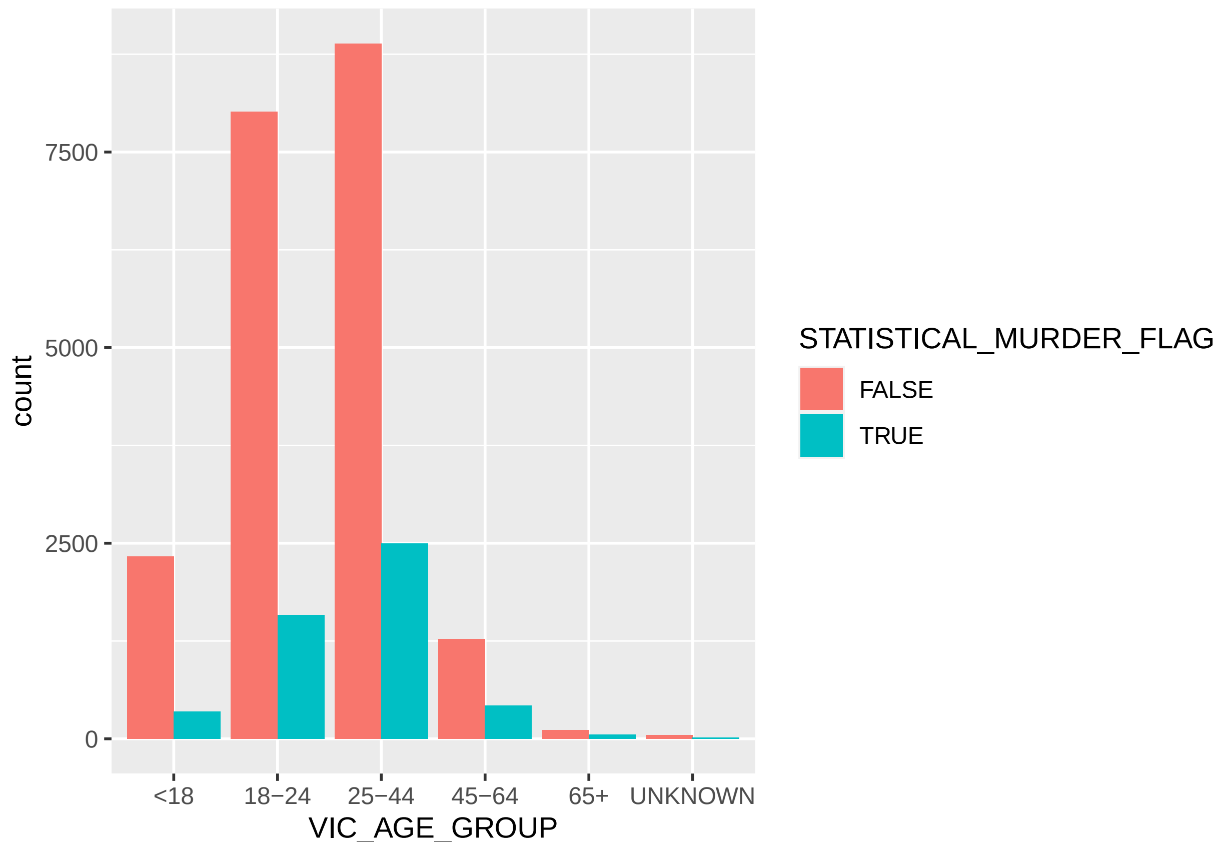
```
NYPD_Shootiing_Incident %>%
  group_by(VIC_AGE_GROUP, STATISTICAL_MURDER_FLAG) %>%
  summarize(count_age = n())
```

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

```
## # A tibble: 12 x 3
## # Groups:   VIC_AGE_GROUP [6]
##    VIC_AGE_GROUP STATISTICAL_MURDER_FLAG count_age
##    <chr>         <lgl>                       <int>
##  1 <18           FALSE                        2332
##  2 <18           TRUE                          349
```

```
##   3 18-24         FALSE                           8018
##   4 18-24         TRUE                            1586
##   5 25-44         FALSE                           8886
##   6 25-44         TRUE                            2500
##   7 45-64         FALSE                           1274
##   8 45-64         TRUE                             424
##   9 65+           FALSE                            113
##  10 65+           TRUE                              54
##  11 UNKNOWN       FALSE                             45
##  12 UNKNOWN       TRUE                              15
```

```
NYPD_Shootiing_Incident %>%
  ggplot(aes(x = VIC_AGE_GROUP, fill = STATISTICAL_MURDER_FLAG)) + geom_bar(position="dodge", stat = 'co
```



I notice that there is an overwhelming number of crime committed that are both fatal and non-fatal are committed by age groups below age 45, and number of fatal crimes to non-fatal crimes are closest for age group 65+. This is probably because sustaining a minor injury might cause severe damages to elders.

Let's also look at how victim's sex correlate with fatal vs non-fatal crimes.

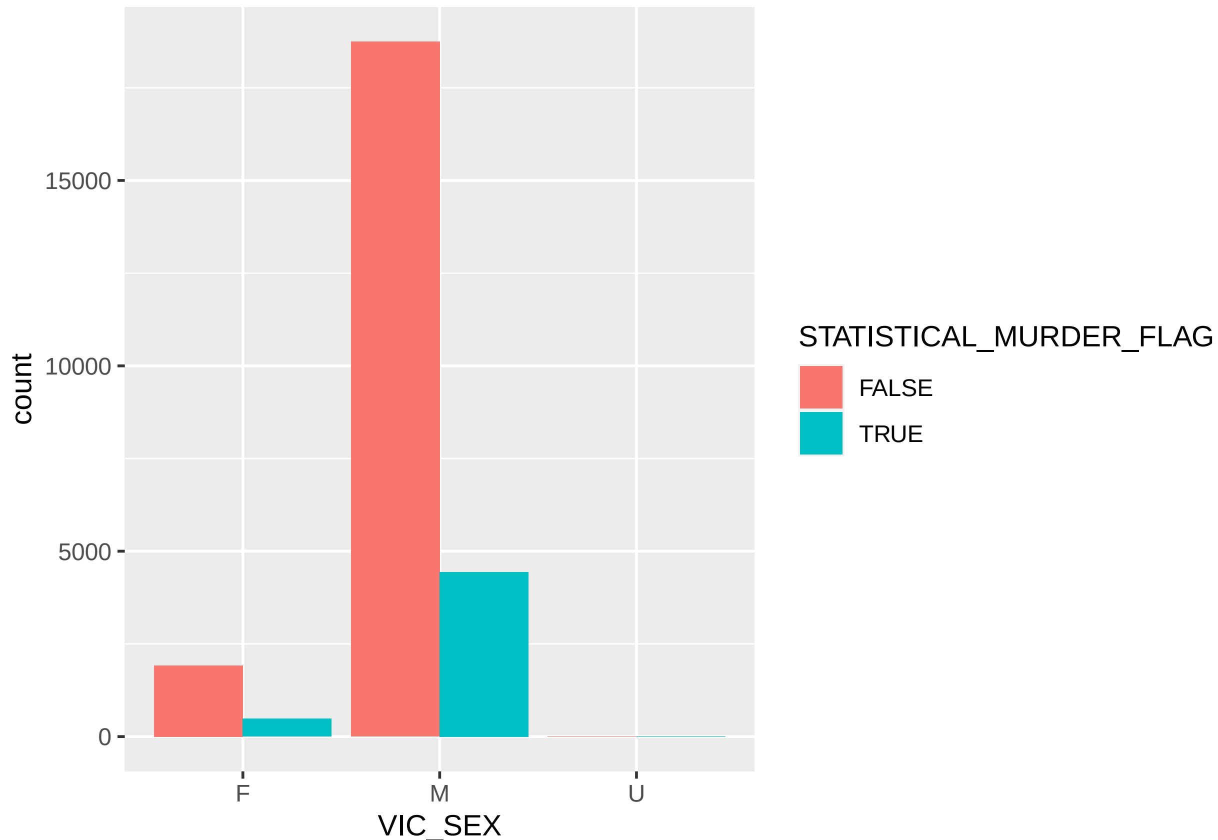
```
NYPD_Shootiing_Incident %>%
  group_by(VIC_SEX, STATISTICAL_MURDER_FLAG) %>%
  summarize(count_sex = n())
```

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

```
## # A tibble: 6 x 3
## # Groups:   VIC_SEX [3]
```

```
##   VIC_SEX STATISTICAL_MURDER_FLAG count_sex
##   <chr>   <lgl>                       <int>
## 1 F       FALSE                        1918
## 2 F       TRUE                          485
## 3 M       FALSE                       18740
## 4 M       TRUE                         4442
## 5 U       FALSE                          10
## 6 U       TRUE                            1
```

```
NYPD_Shootiing_Incident %>%
  ggplot(aes(x = VIC_SEX, fill = STATISTICAL_MURDER_FLAG)) + geom_bar(position="dodge", stat = 'count')
```



I notice that majority of the non-fatal crimes and fatal crimes are committed by men. And interesting note is that the percentage of fatal to non-fatal crime is roughly the same for female and male.

Step 3 - Model

I will then attempt to create a model to predict whether a crime is fatal using borough, victim age group, and victim sex.

```
model <- glm(STATISTICAL_MURDER_FLAG ~ BORO + VIC_SEX + VIC_AGE_GROUP,family=binomial(link='logit'),data
summary(model)
```

```
##
## Call:
## glm(formula = STATISTICAL_MURDER_FLAG ~ BORO + VIC_SEX + VIC_AGE_GROUP,
##     family = binomial(link = "logit"), data = NYPD_Shootiing_Incident)
##
```

```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9155  -0.7065  -0.6028  -0.5299   2.3771
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -1.850969   0.077799 -23.792  < 2e-16 ***
## BOROBROOKLYN          0.003543   0.038765   0.091  0.92717
## BOROMANHATTAN        -0.118233   0.054900  -2.154  0.03127 *
## BOROQUEENS            0.018741   0.050375   0.372  0.70987
## BOROSTATEN ISLAND     0.094918   0.095624   0.993  0.32090
## VIC_SEXM             -0.044962   0.054348  -0.827  0.40807
## VIC_SEXU             -1.074163   1.066401  -1.007  0.31380
## VIC_AGE_GROUP18-24    0.279120   0.063777   4.377 1.21e-05 ***
## VIC_AGE_GROUP25-44    0.631867   0.061853  10.216  < 2e-16 ***
## VIC_AGE_GROUP45-64    0.793878   0.080335   9.882  < 2e-16 ***
## VIC_AGE_GROUP65+      1.148047   0.175458   6.543 6.02e-11 ***
## VIC_AGE_GROUPUNKNOWN  0.861992   0.308796   2.791  0.00525 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 25077  on 25595  degrees of freedom
## Residual deviance: 24843  on 25584  degrees of freedom
## AIC: 24867
##
## Number of Fisher Scoring iterations: 4
```

Since our features were all classification, R had to encode our columns. However, looking at the summary, it appears that our intercept is negative, any age of over 18 will result in an increases to commit a crime, and being any other sex than Female actually lowers the estimate of committing a crime.

```
NYPD_Shootiing_Incident_w_pred <- NYPD_Shootiing_Incident %>%
  mutate(pred = predict(model, type='response')) %>%
  select(c(STATISTICAL_MURDER_FLAG, pred, BORO, VIC_AGE_GROUP, VIC_SEX)) %>%
  mutate(pred = ifelse(pred > .2,TRUE,FALSE))
NYPD_Shootiing_Incident_w_pred
```

```
## # A tibble: 25,596 x 5
##    STATISTICAL_MURDER_FLAG pred  BORO      VIC_AGE_GROUP VIC_SEX
##    <lgl>                   <lgl> <chr>     <chr>         <chr>
##  1 FALSE                   FALSE BROOKLYN  18-24         M
##  2 FALSE                   TRUE  BROOKLYN  25-44         M
##  3 FALSE                   TRUE  BROOKLYN  25-44         M
##  4 FALSE                   TRUE  BROOKLYN  25-44         M
##  5 FALSE                   TRUE  QUEENS    25-44         M
##  6 TRUE                    TRUE  QUEENS    25-44         M
##  7 TRUE                    FALSE BRONX     18-24         M
##  8 FALSE                   TRUE  BRONX     25-44         M
##  9 FALSE                   TRUE  MANHATTAN 25-44         M
## 10 TRUE                    TRUE  BROOKLYN  25-44         M
## # ... with 25,586 more rows
```

```r
misClasificError <- mean(NYPD_Shootiing_Incident_w_pred$pred != NYPD_Shootiing_Incident_w_pred$STATISTI
paste('Accuracy:',1-misClasificError)
```

```
## [1] "Accuracy: 0.521565869667135"
```

Our model was able to achieve an accuracy of 52% in predicting whether a victim was in a fatal crime given the borough of the crime, age and sex of the victim.

Bias

One possible bias in using this data is that this data is from known sources of crime. Some crime might not be reported to police out of fear, threats, or blackmail. Or there could be crime occurring but police were unable to detain any individuals or create a case. The crime listed in this data could be skewed towards less involved crime or only crimes that police were able to cite.

Appendix - Libraries

```r
sessionInfo()
```

```
## R version 4.2.0 (2022-04-22 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19044)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] lubridate_1.8.0 forcats_0.5.2   stringr_1.4.1   dplyr_1.0.10
##  [5] purrr_0.3.4     readr_2.1.2     tidyr_1.2.1     tibble_3.1.8
##  [9] ggplot2_3.3.6   tidyverse_1.3.2
##
## loaded via a namespace (and not attached):
##  [1] assertthat_0.2.1    digest_0.6.29       utf8_1.2.2
##  [4] R6_2.5.1            cellranger_1.1.0    backports_1.4.1
##  [7] reprex_2.0.2        evaluate_0.16       highr_0.9
## [10] httr_1.4.4          pillar_1.8.1        rlang_1.0.5
## [13] googlesheets4_1.0.1 curl_4.3.2          readxl_1.4.1
## [16] rstudioapi_0.14     rmarkdown_2.16      labeling_0.4.2
## [19] googledrive_2.0.0   bit_4.0.4           munsell_0.5.0
## [22] broom_1.0.1         compiler_4.2.0      modelr_0.1.9
## [25] xfun_0.33           pkgconfig_2.0.3     htmltools_0.5.3
## [28] tidyselect_1.1.2    fansi_1.0.3         crayon_1.5.1
## [31] tzdb_0.3.0          dbplyr_2.2.1        withr_2.5.0
## [34] grid_4.2.0          jsonlite_1.8.0      gtable_0.3.1
## [37] lifecycle_1.0.2     DBI_1.1.3           magrittr_2.0.3
```

```
## [40] scales_1.2.1        cli_3.4.0           stringi_1.7.8
## [43] vroom_1.5.7         farver_2.1.1        fs_1.5.2
## [46] xml2_1.3.3          ellipsis_0.3.2      generics_0.1.3
## [49] vctrs_0.4.1         tools_4.2.0         bit64_4.0.5
## [52] glue_1.6.2          hms_1.1.2           parallel_4.2.0
## [55] fastmap_1.1.0       yaml_2.3.5          colorspace_2.0-3
## [58] gargle_1.2.1        rvest_1.0.3         knitr_1.40
## [61] haven_2.5.1
```