NPYD Shootings

C. White

2022-10-06

NYPD Shootings

Begin by reading in the data:

```
## Get currrent data
url_in <- "https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD"
```

Importing in the Data

Now we can read in the data and look at it:

```
library(tidyverse)
crime_data <- read_csv(url_in)
```

```
## 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.
```

Tidying the Data

```
crime_data <- crime_data %>%
  select(-c(INCIDENT_KEY, PRECINCT, JURISDICTION_CODE, LOCATION_DESC,
            X_COORD_CD, Y_COORD_CD, Latitude, Longitude, Lon_Lat))

crime <- crime_data %>%
  rename(date = 'OCCUR_DATE',
         time = 'OCCUR_TIME',
         borough = 'BORO',
         murder = 'STATISTICAL_MURDER_FLAG',
         perp_age = 'PERP_AGE_GROUP',
         perp_sex = 'PERP_SEX',
```

```
        perp_race = 'PERP_RACE',
        vic_age = 'VIC_AGE_GROUP',
        vic_sex = 'VIC_SEX',
        vic_race = 'VIC_RACE') %>%
  mutate(date = mdy(date),
        time = hms(time))
```

Taking a look at the data and a summary:

```
crime
```

```
## # A tibble: 25,596 x 10
##    date       time       borough   murder perp_age perp_sex perp_race    vic_age
##    <date>     <Period>   <chr>     <lgl>  <chr>    <chr>    <chr>         <chr>
##  1 2021-11-11 15H 4M 0S  BROOKLYN  FALSE  <NA>     <NA>     <NA>          18-24
##  2 2021-07-16 22H 5M 0S  BROOKLYN  FALSE  45-64    M        ASIAN / PAC~ 25-44
##  3 2021-07-11 1H 9M 0S   BROOKLYN  FALSE  <18      M        BLACK         25-44
##  4 2021-12-11 13H 42M 0S BROOKLYN  FALSE  <NA>     <NA>     <NA>          25-44
##  5 2021-02-16 20H 0M 0S  QUEENS    FALSE  <NA>     <NA>     <NA>          25-44
##  6 2021-05-15 4H 13M 0S  QUEENS    TRUE   <NA>     <NA>     <NA>          25-44
##  7 2021-04-14 21H 8M 0S  BRONX     TRUE   <NA>     <NA>     <NA>          18-24
##  8 2021-12-10 19H 30M 0S BRONX     FALSE  <NA>     <NA>     <NA>          25-44
##  9 2021-02-22 18M 0S     MANHATTAN FALSE  <NA>     <NA>     <NA>          25-44
## 10 2021-03-07 6H 15M 0S  BROOKLYN  TRUE   25-44    M        BLACK HISPA~ 25-44
## # ... with 25,586 more rows, and 2 more variables: vic_sex <chr>,
## #   vic_race <chr>
```

```
summary(crime)
```

```
##       date                 time                            borough
##  Min.   :2006-01-01   Min.   :0S                          Length:25596
##  1st Qu.:2009-05-10   1st Qu.:3H 23M 0S                   Class :character
##  Median :2012-08-26   Median :15H 10M 0S                  Mode  :character
##  Mean   :2013-06-13   Mean   :12H 39M 17.9910923581774S
##  3rd Qu.:2017-07-01   3rd Qu.:20H 45M 0S
##  Max.   :2021-12-31   Max.   :23H 59M 0S
##    murder          perp_age           perp_sex          perp_race
##  Mode :logical   Length:25596       Length:25596       Length:25596
##  FALSE:20668     Class :character   Class :character   Class :character
##  TRUE :4928      Mode  :character   Mode  :character   Mode  :character
##
##
##
##    vic_age            vic_sex            vic_race
##  Length:25596       Length:25596       Length:25596
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
```

Let's filter to where the data is greater than or equal to January 1st, 2020

```
crime <- crime %>% filter(date >= '2020-01-01')
```

Transforming the Data

We can observe from the summary that there are a fair number of categories with missing data. Let's calculate the precise amount of data that is missing for one of the dataset's features. The following code examines the percentage of missing data for a specific characteristic:

```
mean(is.na(crime$perp_age))
```

```
## [1] 0.5352362
```

We will see the total amount of values missing from the rows of data:

```
sum(is.na(crime))
```

```
## [1] 6357
```

Numerous variables lack several entries and some of them are missing more than fifty-percent of the data. There are a few approaches to handles this sort of predicament we are in where it is full of random data that is missing. Imputation is a technique whereby missing values are filled in using the values that are already there as a guide. This is helpful for lesser quantities of missing data, but it introduces too much bias when more than fifty-percent of the values for a feature are missing. Although there are still approaches to imputation for missing categorical data, it often works better for continuous values than for categorical ones.

With mode imputation, all missing values in a feature are given the most prevalent category. Nonetheless, much like with normal imputation, there is an increase in bias and a decrease in variance. If there had been fewer missing data, perpetrator sex might have been imputed using multinomial logistic regression as it can be utilized for features with few categories. On ordered categorical data, such as perpetrator age group, predictive mean matching imputation can be effective. However, because the percentage of missing data is excessive, we have to omit any observations that have data missing.

Given that the perpetrator is the subject of the majority of the severely missing data in this dataset, the answer relies on the significance of the analysis. If perp analysis is valued, then remove incomplete observations and maintain all the features; if not, remove those perp characteristics and keep all the observations.

Now, I am also interested in the correlations between the perputrator and victim's age, race, and sex. I will transform the related columns into factors:

```
crime <- crime %>% mutate(
  perp_age = as.factor(perp_age),
  perp_race = as.factor(perp_race),
  perp_sex = as.factor(perp_sex),
  vic_age = as.factor(vic_age),
  vic_race = as.factor(vic_race),
  vic_sex = as.factor(vic_sex)
)

crime_no_na <- crime %>%
  na.omit()

crime_no_na
```

```
## # A tibble: 1,840 x 10
##    date       time       borough   murder perp_age perp_sex perp_race    vic_age
##    <date>     <Period>   <chr>     <lgl>  <fct>    <fct>    <fct>         <fct>
##  1 2021-07-16 22H 5M 0S  BROOKLYN  FALSE  45-64    M        ASIAN / PAC~ 25-44
##  2 2021-07-11 1H 9M 0S   BROOKLYN  FALSE  <18      M        BLACK        25-44
##  3 2021-03-07 6H 15M 0S  BROOKLYN  TRUE   25-44    M        BLACK HISPA~ 25-44
##  4 2021-07-21 40M 0S     MANHATTAN FALSE  25-44    M        BLACK        25-44
##  5 2021-05-09 2H 50M 0S  BRONX     TRUE   25-44    M        BLACK        25-44
##  6 2021-06-16 23H 22M 0S BRONX     TRUE   25-44    M        BLACK        25-44
##  7 2021-01-12 22H 12M 0S BROOKLYN  FALSE  18-24    M        BLACK        18-24
##  8 2021-09-04 20H 18M 0S MANHATTAN FALSE  18-24    M        WHITE HISPA~ 18-24
##  9 2021-06-16 23H 22M 0S BRONX     FALSE  18-24    M        WHITE HISPA~ 25-44
## 10 2021-09-29 12H 50M 0S BRONX     FALSE  18-24    M        BLACK        <18
## # ... with 1,830 more rows, and 2 more variables: vic_sex <fct>, vic_race <fct>
```

```
summary(crime_no_na)
```

```
##       date                 time                            borough
##  Min.   :2020-01-01   Min.   :0S                          Length:1840
##  1st Qu.:2020-07-27   1st Qu.:5H 47M 45S                  Class :character
##  Median :2021-01-22   Median :16H 29M 30S                 Mode  :character
##  Mean   :2021-01-14   Mean   :14H 2M 58.9239130434798S
##  3rd Qu.:2021-07-02   3rd Qu.:20H 49M 0S
##  Max.   :2021-12-31   Max.   :23H 58M 0S
##    murder         perp_age   perp_sex                    perp_race
##  Mode :logical   <18  :172   F:  65   ASIAN / PACIFIC ISLANDER:  35
##  FALSE:1352      18-24:634   M:1775   BLACK                    :1254
##  TRUE :488       25-44:916            BLACK HISPANIC           : 191
##                  45-64:112            WHITE                    :  33
##                  65+  :  6            WHITE HISPANIC           : 327
##
##     vic_age     vic_sex                      vic_race
##  <18    : 123   F: 212   ASIAN / PACIFIC ISLANDER:  33
##  18-24  : 487   M:1628   BLACK                    :1178
##  25-44  :1063            BLACK HISPANIC           : 216
##  45-64  : 154            WHITE                    :  56
##  65+    :  11            WHITE HISPANIC           : 357
##  UNKNOWN:   2
```

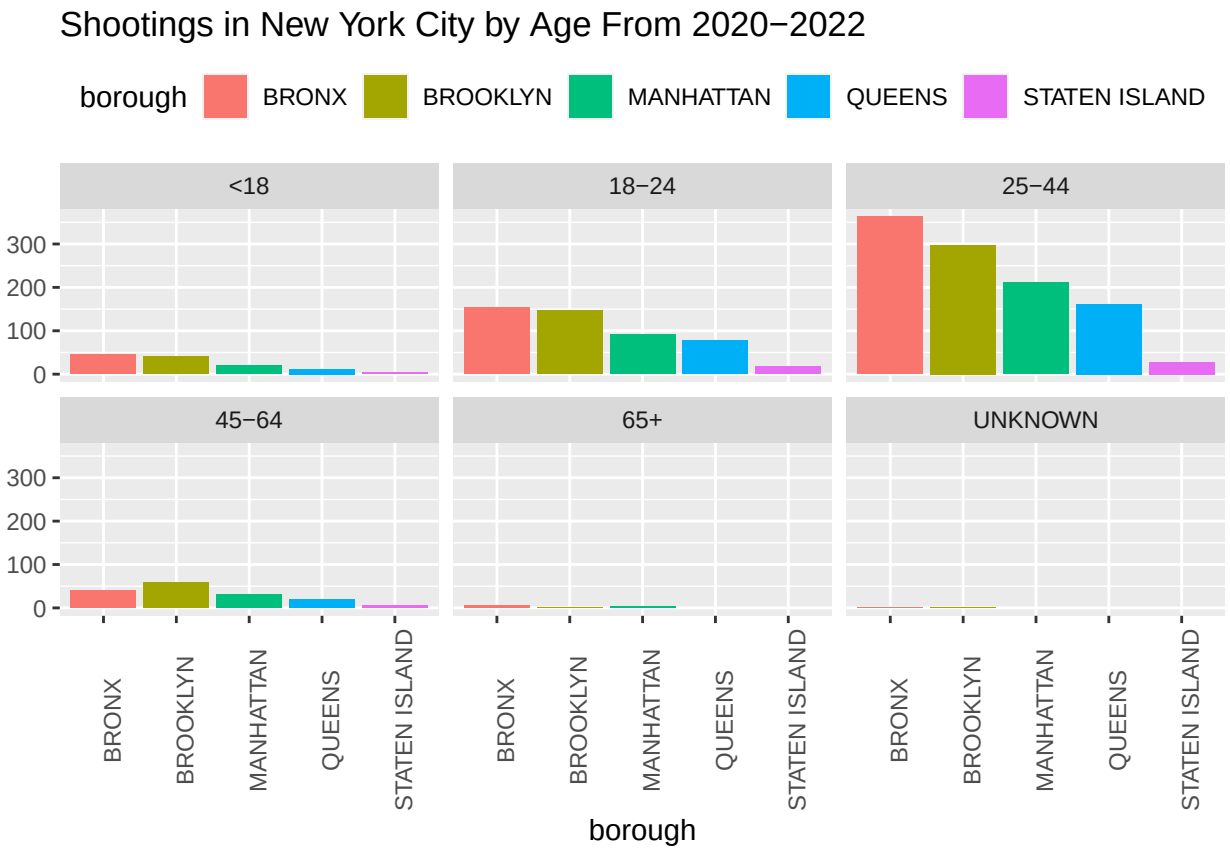
Visualizing Data

In order to preserve the bulk of the observations, I have opted to leave the characteristics that lacked sufficient data alone. Only a few observations required the dplyr function na.omit() since they were missing adequate variables.

We may observe an age breakdown for each of the five boroughs from 2020-2022 by factoring the number of gunshots by the victim's age. It is evident that the vast majority of gunshot victims in New York City are 25-44.

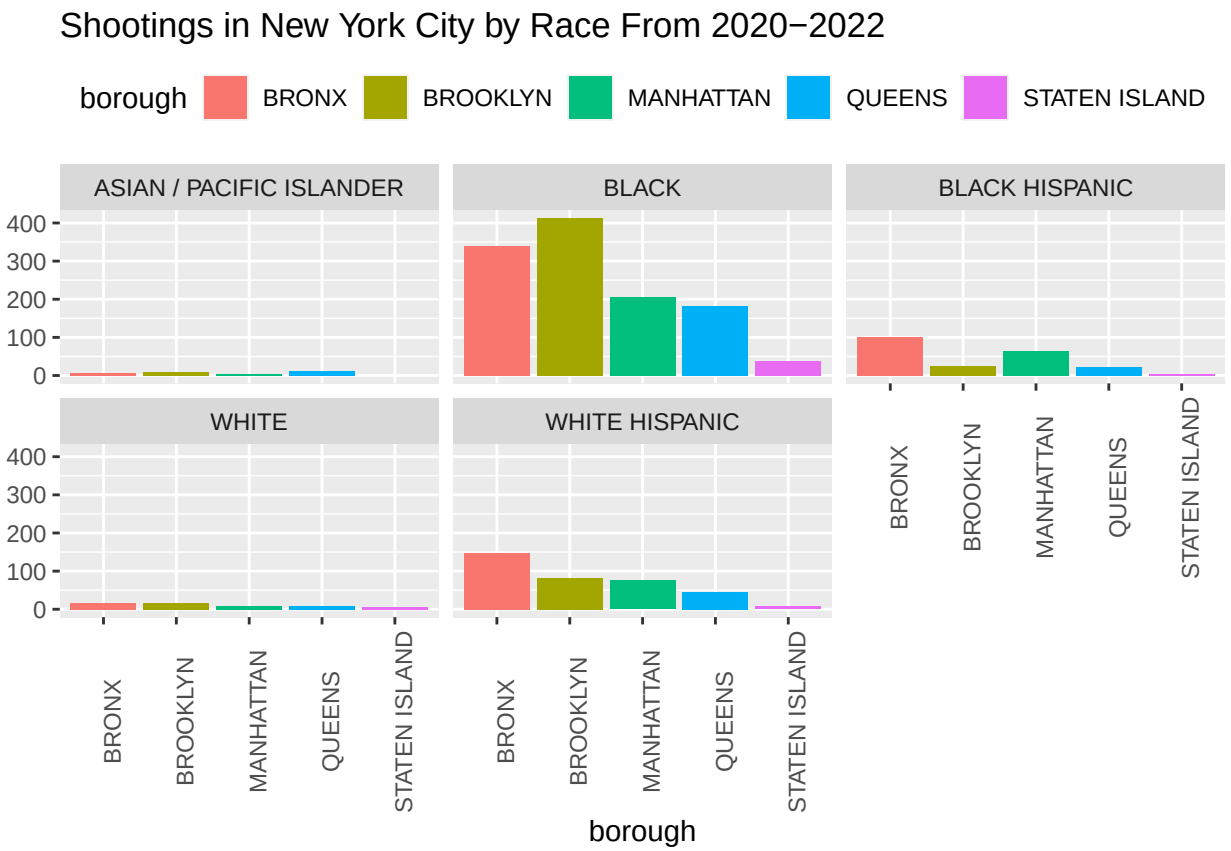
```
ggplot(crime_no_na) +
  geom_bar(aes(x = borough, fill = borough)) +
  facet_wrap(~vic_age) +
  theme(legend.position = "top",
```

```
        axis.text.x = element_text(angle = 90)) +
  labs(title = "Shootings in New York City by Age From 2020-2022", y = NULL)
```



In a similar vein, we may segment the shootings by borough for the racial characteristics of the victims, showing that the victims are likewise predominantly black:

```
ggplot(crime_no_na) +
  geom_bar(aes(x = borough, fill = borough)) +
  facet_wrap(~vic_race) +
  theme(legend.position = "top",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "Shootings in New York City by Race From 2020-2022", y = NULL)
```



Finally, using data from a section of the dataset to train a linear regression model, I will use the model to determine if a homicide victim was killed based on the victim's ethnicity, sex, age, and the borough where the incident took place:

```
train <- crime_no_na[1:410, ]
test <- crime_no_na[411:488, ]

model <- lm(murder ~ vic_age + vic_race + vic_sex + borough, data = train)
model
```

```
##
## Call:
## lm(formula = murder ~ vic_age + vic_race + vic_sex + borough,
##     data = train)
##
## Coefficients:
##             (Intercept)            vic_age18-24            vic_age25-44
##                0.361368                -0.002483                 0.002581
##           vic_age45-64              vic_age65+           vic_raceBLACK
##                -0.045341                -0.255958                -0.027123
## vic_raceBLACK HISPANIC           vic_raceWHITE  vic_raceWHITE HISPANIC
##                 0.068777                -0.078892                 0.041775
##                vic_sexM         boroughBROOKLYN        boroughMANHATTAN
##                -0.053364                -0.081452                -0.086132
##           boroughQUEENS    boroughSTATEN ISLAND
##                 0.003422                 0.145602
```

```
test$predict <- predict(model, test)
test <- test %>% mutate(murder_binary = case_when(
  murder == FALSE ~ 0,
  TRUE ~ 1
))
cor.test(test$murder_binary, test$predict)
```

```
##
##  Pearson's product-moment correlation
##
## data:  test$murder_binary and test$predict
## t = 0.47668, df = 76, p-value = 0.635
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.1699986  0.2738017
## sample estimates:
##        cor
## 0.05459757
```

About five-percent of the time, the model seems to have a respectable accuracy rate for forecasting the outcome. The model would never be accurate enough to forecast a victim dying even once, therefore the model would theoretically be more accurate if it just assumed that the victim lives every time.

Conclusion

Both of the representations appear to show that Staten Island is by far the least risky borough for gun violence, while Brooklyn is by far the most hazardous. Violence per capita statistics may have different findings since this simply considers the total number of recorded instances and ignores population density.

The reporting and recording of the data may have been biased. "Shooting" can be variously defined based on the individual precincts. For example, a gun being drawn at the victim could constitute a shooting in one precinct, but not in the other. The fact that not all shootings will be publicized is another instance of prejudice. Another example of bias data is victims of fatal gunshot wounds; victims of fatal gunshots will be reported at a higher rate than victims of non-lethal shootings.

The skewed reported statistics would probably exceed the actual population parameter for the gunshot fatality rate if the rate of shooting deaths were analyzed. I would argue that there is very little personal bias because the data set was picked for me, thus I have no personal relationship to it. Having said that, my method of tidying and cleaning the data was biased. I opted to leave out several characteristics because there was so much missing information, even though I could have retained them and left out the observations that lacked data. As a result, the characteristics of the shooters' perpetrators were substantially obscured, which forced my study to concentrate more on the shooting victims.