NYPD Shooting Incident Exploratory Analysis

Michael Christensen

2023-06-07

```
library(tidyverse)
library(crosstable) #crosstables for presentation of tabulated data
library(zoo) #for na.approx
library(flextable)
library(rcompanion)
```

NYPD Shooting Incident Data Report

In this report, we will be analyzing historical data regarding shootings in NYC from 2006 until now provided by the NYC OpenData repository. Our objective is to gain insight on the following:

1. Is there a difference in the rate of shooting incidents across the boroughs of New York City?
2. Is there a relationship between the demographic of the perpetrator and the demographic of the victims?

Importing Data

For our analysis, it would be helpful to have population data for each of the boroughs in New York City to normalize any difference in population. We will be using population data from the same repository where we have obtained the Shooting Incident Report Data.

https://data.cityofnewyork.us/Public-Safety/NYPD-Shooting-Incident-Data-Historic-/833y-fsy8 https://data.cityofnewyork.us/City-Government/New-York-City-Population-by-Borough-1950-2040/xywu-7bv9

```
shooting_data <-
  read.csv("https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv")
pop_data <-
  read.csv("New_York_City_Population_by_Borough__1950_-_2040.csv")
```

Tidying Data

A quick overview of our data shows that there are many unnecessary columns. We want to focus on the following:

- **Date** - date of the incident, particularly the year
- **Borough** - location of the incident by borough
- **Demographics** - age/sex/race of the perpetrator/victim

Additionally, we will convert the character columns into factors in order to remove junk data and analyze each category.

```
glimpse(shooting_data)
```

```
## Rows: 27,312
## Columns: 21
## $ INCIDENT_KEY            <int> 228798151, 137471050, 147998800, 146837977, 58~
## $ OCCUR_DATE              <chr> "05/27/2021", "06/27/2014", "11/21/2015", "10/~
## $ OCCUR_TIME              <chr> "21:30:00", "17:40:00", "03:56:00", "18:30:00"~
## $ BORO                    <chr> "QUEENS", "BRONX", "QUEENS", "BRONX", "BRONX",~
## $ LOC_OF_OCCUR_DESC       <chr> "", "", "", "", "", "", "", "", "", "", "", ""~
## $ PRECINCT                <int> 105, 40, 108, 44, 47, 81, 114, 81, 105, 101, 2~
## $ JURISDICTION_CODE       <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 2, 2~
## $ LOC_CLASSFCTN_DESC      <chr> "", "", "", "", "", "", "", "", "", "", "", ""~
## $ LOCATION_DESC           <chr> "", "", "", "", "", "", "", "", "", "MULTI DWE~
## $ STATISTICAL_MURDER_FLAG <chr> "false", "false", "true", "false", "true", "tr~
## $ PERP_AGE_GROUP          <chr> "", "", "", "", "25-44", "", "", "", "", "25-4~
## $ PERP_SEX                <chr> "", "", "", "", "M", "", "", "", "", "M", "", ~
## $ PERP_RACE               <chr> "", "", "", "", "BLACK", "", "", "", "", "BLAC~
## $ VIC_AGE_GROUP           <chr> "18-24", "18-24", "25-44", "<18", "45-64", "25~
## $ VIC_SEX                 <chr> "M", "M", "M", "M", "M", "M", "M", "M", "M", "~
## $ VIC_RACE                <chr> "BLACK", "BLACK", "WHITE", "WHITE HISPANIC", "~
## $ X_COORD_CD              <dbl> 1058925.0, 1005028.0, 1007667.9, 1006537.4, 10~
## $ Y_COORD_CD              <dbl> 180924.0, 234516.0, 209836.5, 244511.1, 262189~
## $ Latitude                <dbl> 40.66296, 40.81035, 40.74261, 40.83778, 40.886~
## $ Longitude               <dbl> -73.73084, -73.92494, -73.91549, -73.91946, -7~
## $ Lon_Lat                 <chr> "POINT (-73.73083868899994 40.662964620000025)~
```

```
#Removing/renaming columns
shooting_data <- shooting_data %>%
  rename(Date = "OCCUR_DATE", BOROUGH = "BORO",
         PERP_AGE = "PERP_AGE_GROUP", VIC_AGE = "VIC_AGE_GROUP") %>%
  select(-c(INCIDENT_KEY,OCCUR_TIME,LOC_OF_OCCUR_DESC, PRECINCT,
            JURISDICTION_CODE, LOC_CLASSFCTN_DESC,LOCATION_DESC,
            STATISTICAL_MURDER_FLAG,
            X_COORD_CD,Y_COORD_CD,Latitude,Longitude,Lon_Lat))

#Adjusting characters to factors
shooting_data <- as.data.frame(unclass(shooting_data), stringsAsFactors = TRUE)
sapply(shooting_data[-1], levels)
```

```
## $BOROUGH
```

```
## [1] "BRONX"         "BROOKLYN"      "MANHATTAN"     "QUEENS"
## [5] "STATEN ISLAND"
##
## $PERP_AGE
##  [1] ""        "(null)"  "<18"     "1020"    "18-24"   "224"     "25-44"
##  [8] "45-64"   "65+"     "940"     "UNKNOWN"
##
## $PERP_SEX
## [1] ""       "(null)" "F"      "M"      "U"
##
## $PERP_RACE
## [1] ""                                "(null)"
## [3] "AMERICAN INDIAN/ALASKAN NATIVE" "ASIAN / PACIFIC ISLANDER"
## [5] "BLACK"                           "BLACK HISPANIC"
## [7] "UNKNOWN"                         "WHITE"
## [9] "WHITE HISPANIC"
##
## $VIC_AGE
## [1] "<18"     "1022"    "18-24"   "25-44"   "45-64"   "65+"     "UNKNOWN"
##
## $VIC_SEX
## [1] "F" "M" "U"
##
## $VIC_RACE
## [1] "AMERICAN INDIAN/ALASKAN NATIVE" "ASIAN / PACIFIC ISLANDER"
## [3] "BLACK"                           "BLACK HISPANIC"
## [5] "UNKNOWN"                         "WHITE"
## [7] "WHITE HISPANIC"
```

```
#Cleaning empty/trash factors
shooting_data <- shooting_data %>%
  replace(shooting_data == "(null)", "") %>%
  replace(shooting_data == "1020", "UNKNOWN") %>%
  replace(shooting_data == "224", "UNKNOWN") %>%
  replace(shooting_data == "940", "UNKNOWN") %>%
  replace(shooting_data == "1022", "UNKNOWN")

shooting_data <- shooting_data %>%
  replace(shooting_data == "" | shooting_data == " ", NA) %>%
  replace_na(list(PERP_AGE="UNKNOWN", PERP_SEX = "U",
                  PERP_RACE = "UNKNOWN", VIC_AGE="UNKNOWN",
                  VIC_SEX = "U", VIC_RACE = "UNKNOWN")) %>%
  droplevels()

#Renaming Factor Levels for Future Tables
fct_count(shooting_data$PERP_RACE)
```

```
## # A tibble: 7 x 2
##   f                                n
##   <fct>                        <int>
## 1 AMERICAN INDIAN/ALASKAN NATIVE     2
## 2 ASIAN / PACIFIC ISLANDER         154
## 3 BLACK                          11432
## 4 BLACK HISPANIC                  1314
```

```
## 5 UNKNOWN                          11786
## 6 WHITE                              283
## 7 WHITE HISPANIC                    2341
```

```r
levels(shooting_data$PERP_RACE) <- c("AIAN", "AAPI", "B", "B H",
                                     "UNKNOWN", "W", "W H")
levels(shooting_data$VIC_RACE) <- c("AIAN", "AAPI", "B", "B H",
                                    "UNKNOWN", "W", "W H")

#Adjusting date format
shooting_data <- shooting_data %>%
  mutate(Date = mdy(Date)) %>%
  mutate(Year = year(Date)) %>%
  arrange(Date)

shooting_data <- shooting_data %>%
  group_by(BOROUGH, Year) %>%
  mutate(cases = n()) %>%
  ungroup()

shooting_data$Year <- as.integer(shooting_data$Year)
```

Appending Population Data

Now lets join the population data set to the main data set.

```
#Wrangling population data
pop_data <- pop_data %>%
  select(-c(Age.Group,
            X1950:X1990...Boro.share.of.NYC.total,
            X2000...Boro.share.of.NYC.total,
            X2010...Boro.share.of.NYC.total,
            X2020...Boro.share.of.NYC.total:X2040...Boro.share.of.NYC.total))%>%
  rename("BOROUGH" = "Borough", "2000" = "X2000",
         "2010" = "X2010", "2020" = "X2020")

pop_data$BOROUGH <- pop_data$BOROUGH %>%
  toupper() %>% trimws("l")

pop_data <- pop_data %>%
  pivot_longer(cols = -c(BOROUGH), names_to ="Year", values_to = "Population")

pop_data$Year <- as.integer(pop_data$Year)

#Interpolating Population Data
pop_data <- pop_data %>%
  group_by(BOROUGH) %>%
  complete(Year = full_seq(2000:2022,1)) %>%
    #In this line, we are interpolating the population between years.
    #Alternatively, we could use the nearest available
    #year without changing results significantly.
  mutate_all(na.approx, rule = 2)

#Appending Population Data to Main Data Set
shooting_data <- left_join(shooting_data, pop_data, by = c("BOROUGH" = "BOROUGH",
                                                           "Year" = "Year"))

#Cases by borough
shootings_by_borough <- shooting_data %>%
  group_by(BOROUGH, Year) %>%
  summarize(cases = max(cases), Population = round(max(Population),0)) %>%
  mutate(cases_per_thou = cases * 1000 / Population) %>%
  ungroup()

shooting_totals <- shooting_data %>%
  group_by(Year) %>%
  summarize(cases = max(cases), Population = round(max(Population),0)) %>%
  mutate(cases_per_thou = cases * 1000 / Population) %>%
  ungroup()
```

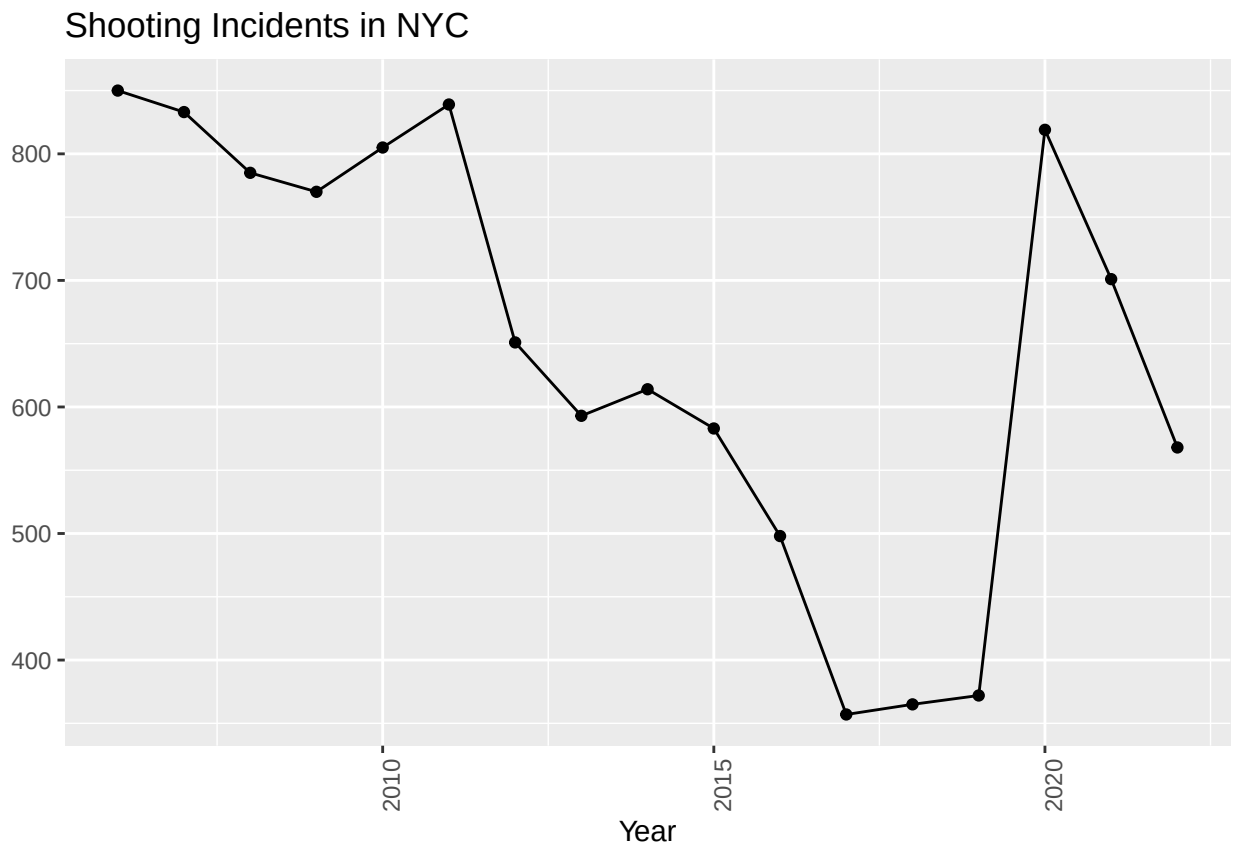
Visualizing the Data

The first question we would like to answer is: is there a difference in the rate of shooting incidents across the boroughs of New York City? We will do so by graphing the yearly cases for each borough and taking a quick look at key statistics to see how each borough compares.

Shooting Incidents over Time (Total)

Let's take a quick glance at the total cases in New York to see what our data looks like.

```
shooting_totals %>%
  ggplot(aes(x=Year,y=cases)) +
  geom_line() +
  geom_point() +
  theme(legend.position="bottom",
        axis.text.x = element_text(angle=90)) +
  labs(title = "Shooting Incidents in NYC", y=NULL)
```




```
summary(shooting_totals)
```

```
##       Year          cases          Population      cases_per_thou
##  Min.   :2006   Min.   :357.0   Min.   :2517877   Min.   :0.1363
##  1st Qu.:2010   1st Qu.:568.0   1st Qu.:2552911   1st Qu.:0.2145
##  Median :2014   Median :651.0   Median :2591127   Median :0.2531
```

```
##  Mean   :2014   Mean   :647.2   Mean   :2589909   Mean   :0.2507
##  3rd Qu.:2018   3rd Qu.:805.0   3rd Qu.:2629344   3rd Qu.:0.3096
##  Max.   :2022   Max.   :850.0   Max.   :2648452   Max.   :0.3376
```

Since our data doesn't appear linear, we will try to find a polynomial to approximately model our data, as the number of incidents per year. To do this, we will be doing an ANOVA test for polynomial models up to degree 5.

```
est_1 <- lm(cases ~ Year, data = shooting_totals)
est_2 <- lm(cases ~ poly(Year,2), data = shooting_totals)
est_3 <- lm(cases ~ poly(Year,3), data = shooting_totals)
est_4 <- lm(cases ~ poly(Year,4), data = shooting_totals)
est_5 <- lm(cases ~ poly(Year,5), data = shooting_totals)
anova(est_1,est_2,est_3,est_4,est_5)
```

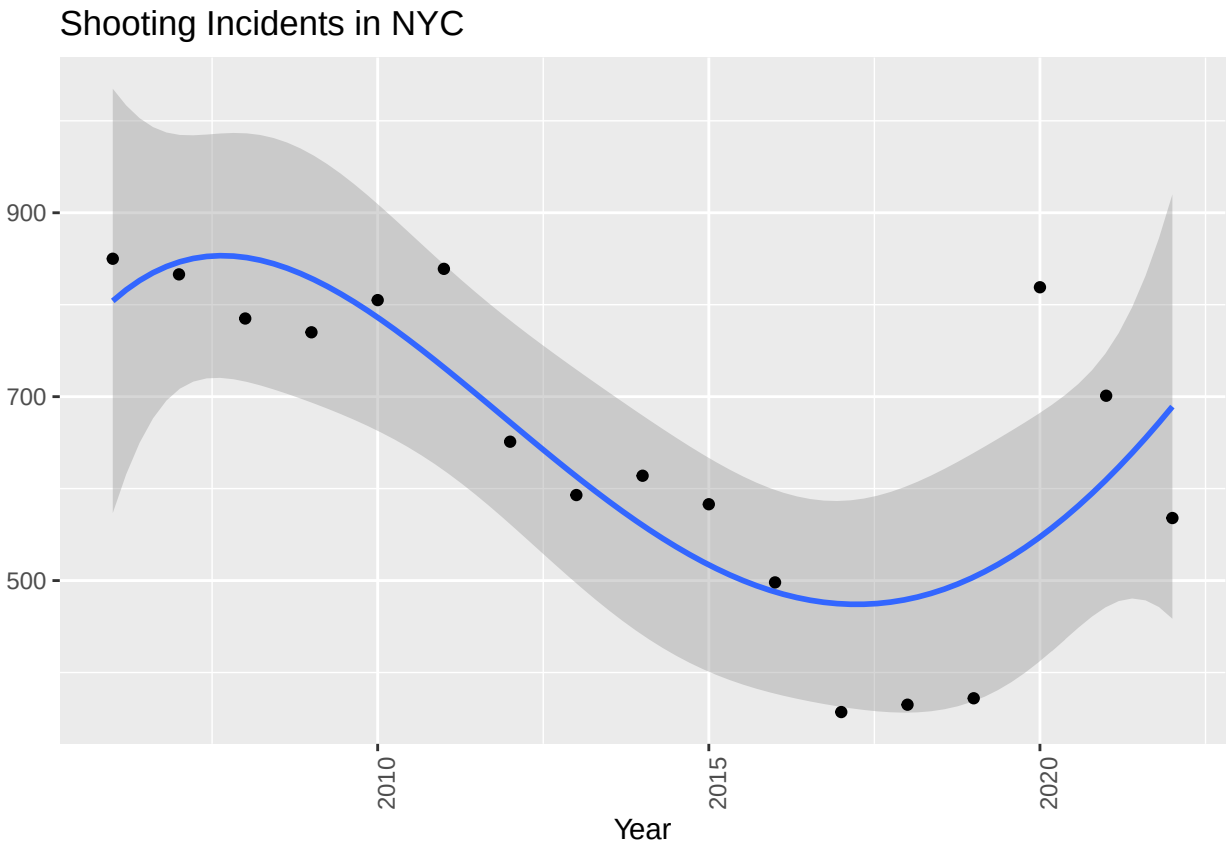
```
## Analysis of Variance Table
##
## Model 1: cases ~ Year
## Model 2: cases ~ poly(Year, 2)
## Model 3: cases ~ poly(Year, 3)
## Model 4: cases ~ poly(Year, 4)
## Model 5: cases ~ poly(Year, 5)
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1     15 303529
## 2     14 232919  1     70611 6.7367 0.02489 *
## 3     13 174210  1     58709 5.6013 0.03737 *
## 4     12 171395  1      2814 0.2685 0.61461
## 5     11 115295  1     56100 5.3523 0.04104 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(est_4)
```

```
##
## Call:
## lm(formula = cases ~ poly(Year, 4), data = shooting_totals)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -131.82  -66.48  -13.39   53.93  271.65
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)      647.24      28.99  22.329 3.84e-11 ***
## poly(Year, 4)1  -420.22     119.51  -3.516  0.00425 **
## poly(Year, 4)2   265.73     119.51   2.223  0.04615 *
## poly(Year, 4)3   242.30     119.51   2.027  0.06542 .
## poly(Year, 4)4   -53.05     119.51  -0.444  0.66503
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 119.5 on 12 degrees of freedom
## Multiple R-squared:  0.643,  Adjusted R-squared:  0.524
## F-statistic: 5.404 on 4 and 12 DF,  p-value: 0.01006
```

With a p value of 0.9384 for the quartic model, we can conclude that the quartic model is a reasonable fit for our data, with an r squared value of 0.643. Here is a look at the smoothed data:

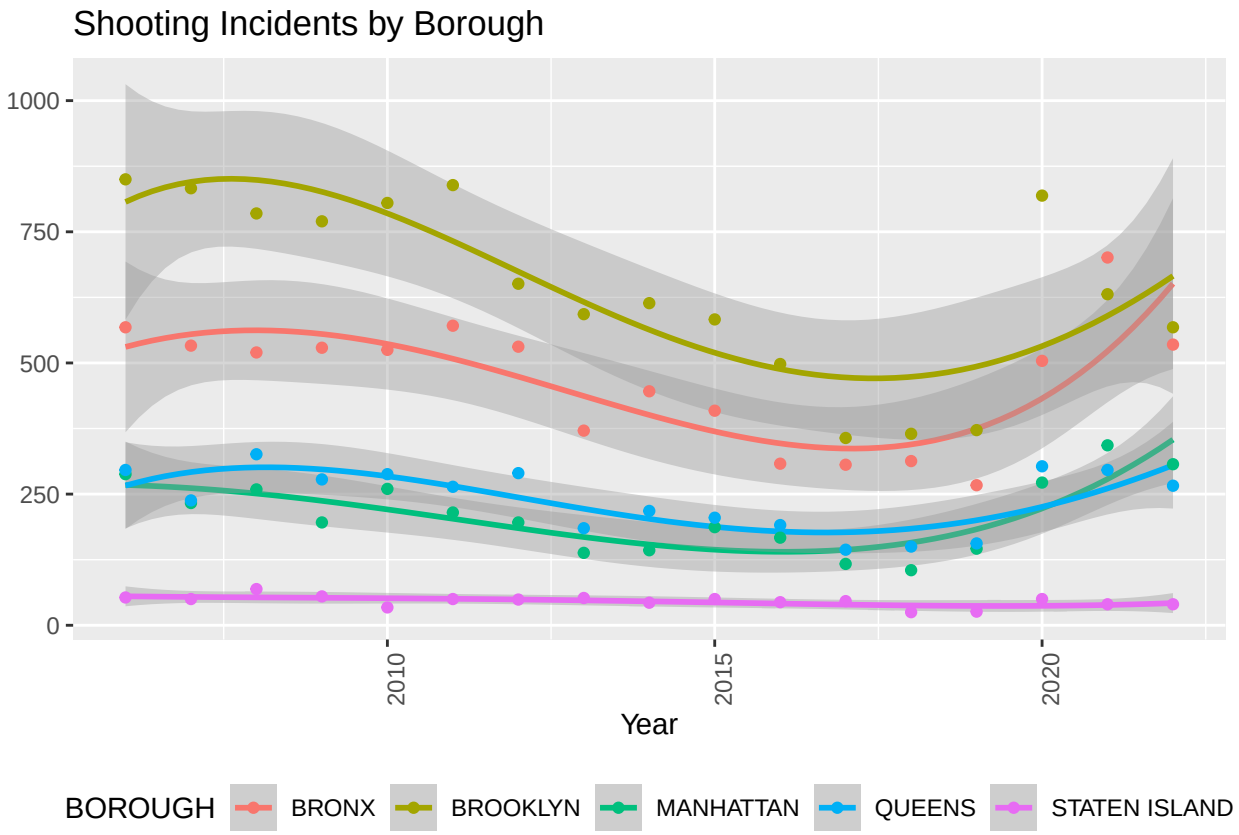
```
shooting_totals %>%
  ggplot(aes(x=Year,y=cases)) +
  geom_smooth(method = "lm", formula = y ~ poly(x,4)) +
  geom_point() +
  theme(legend.position="bottom",
        axis.text.x = element_text(angle=90)) +
  labs(title = "Shooting Incidents in NYC", y=NULL)
```



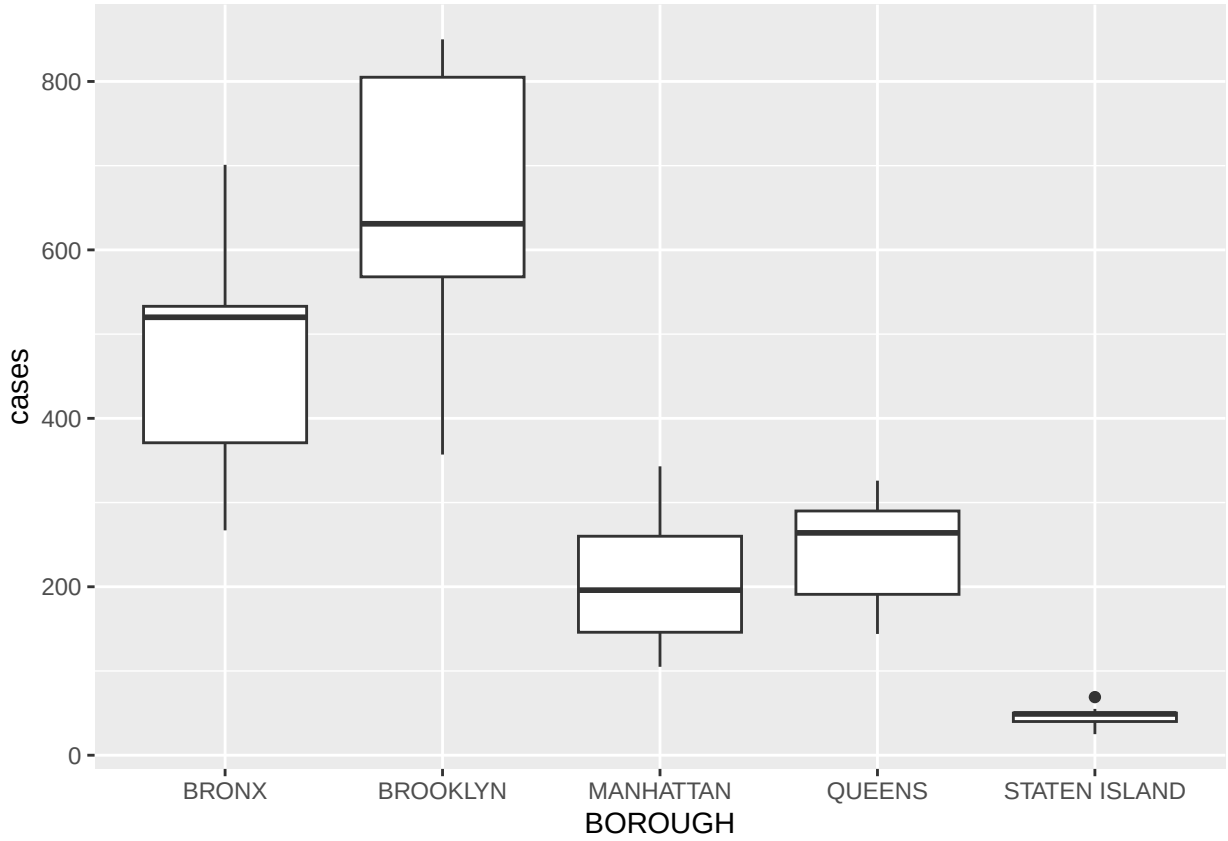
We can do a similar analysis for each of the boroughs, but for brevity we will assume the same quartic model for each.

Shooting Incidents by Borough over Time (Total)

```
shootings_by_borough %>%
  ggplot(aes(x=Year,y=cases,color = BOROUGH)) +
  geom_smooth(method = "lm", formula = y ~ poly(x,4)) +
  geom_point() +
  theme(legend.position="bottom",
        axis.text.x = element_text(angle=90)) +
  labs(title = "Shooting Incidents by Borough", y=NULL)
```

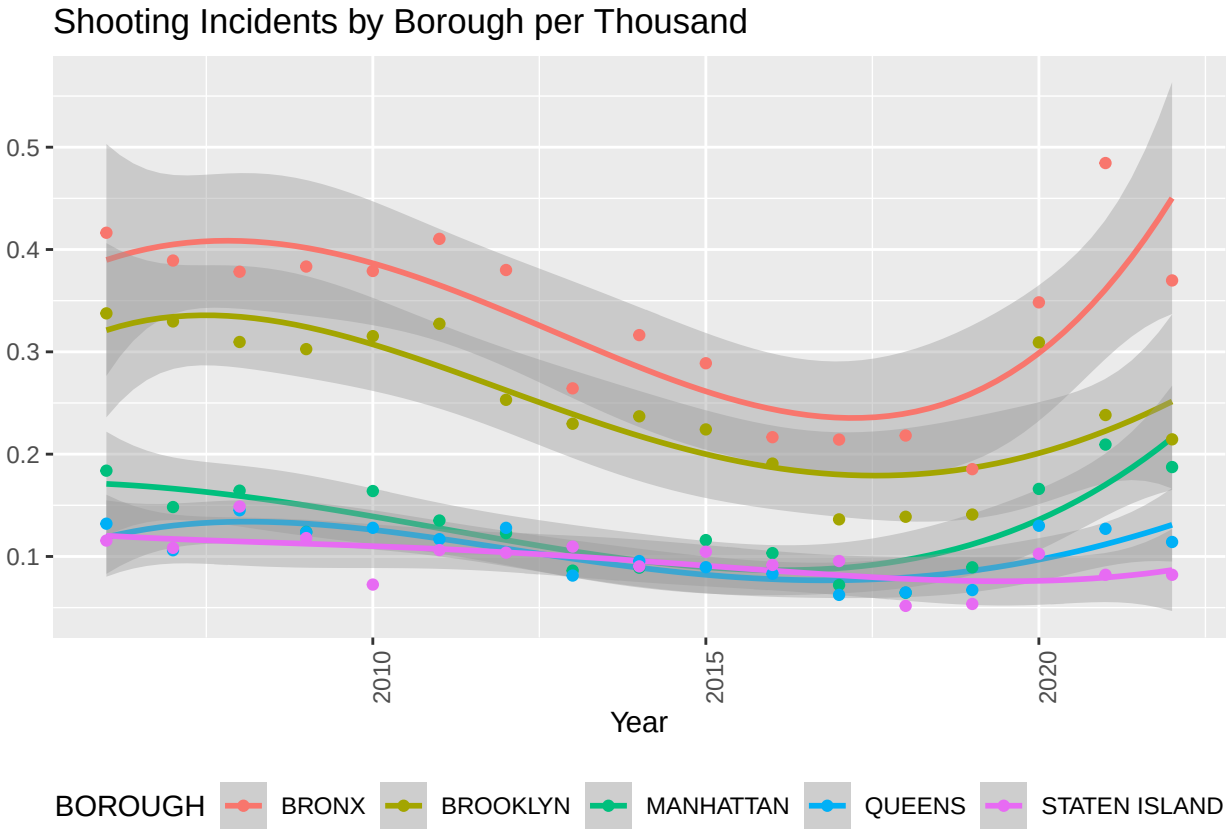



```
shootings_by_borough %>%
  ggplot(aes(x=BOROUGH,y=cases))+
  geom_boxplot()
```

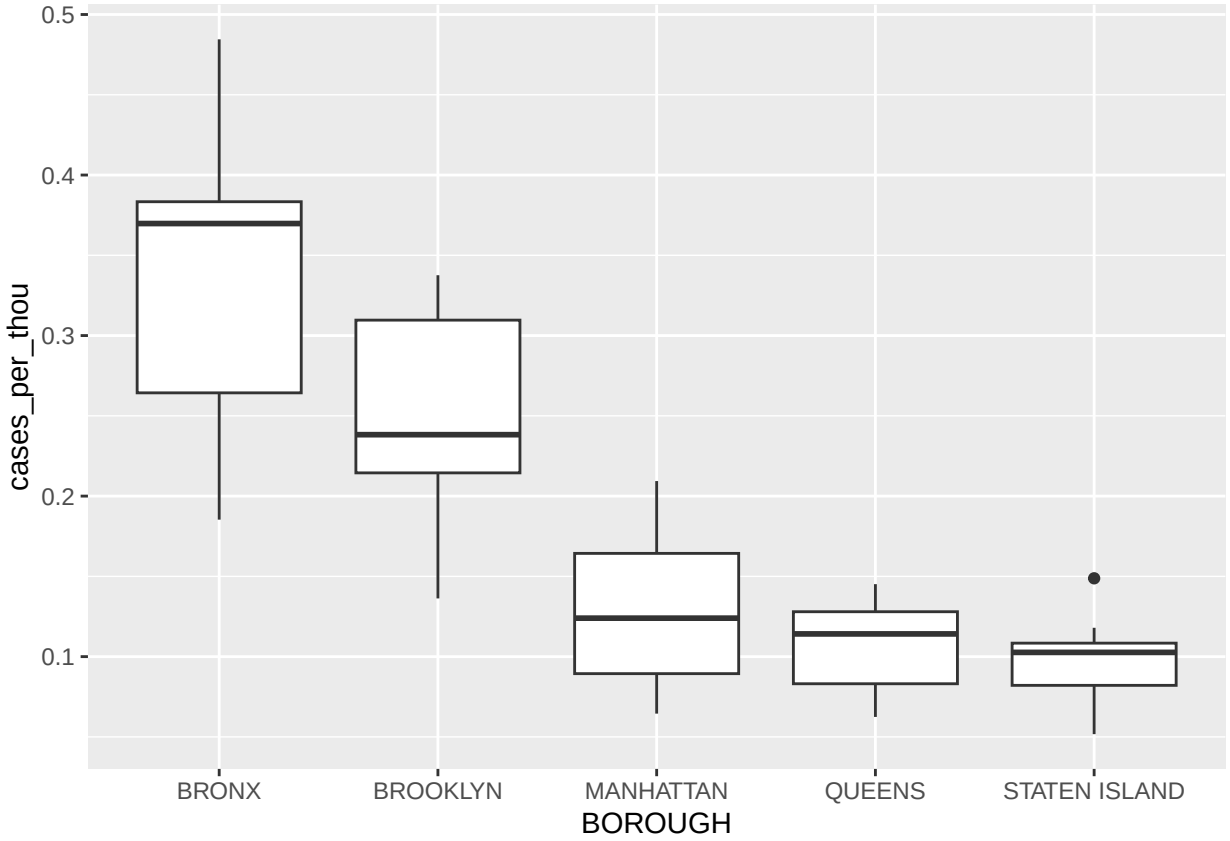


Shooting Incidents by Borough over Time (per Capita)

```
shootings_by_borough %>%
  ggplot(aes(x=Year,y=cases_per_thou,color = BOROUGH)) +
  geom_smooth(method = "lm", formula = y ~ poly(x,4)) +
  geom_point() +
  theme(legend.position="bottom",
        axis.text.x = element_text(angle=90)) +
  labs(title = "Shooting Incidents by Borough per Thousand", y=NULL)
```




```
shootings_by_borough %>%
  ggplot(aes(x=BOROUGH,y=cases_per_thou))+
  geom_boxplot()
```



A quick glance of the data shows that, although the Brooklyn has a higher number of shooting incidents, when corrected for by population, it appears the Bronx has the most shooting incidents per capita. We can verify this information by taking a look at the means by area, and noting the differences between each borough:

```
shootings_by_borough %>%
  group_by(BOROUGH) %>%
  summarise(casetot=sum(cases), mean = mean(cases),
            mean_thou = mean(cases_per_thou), borosd = sd(cases),
            sd_thou= sd(cases_per_thou)) %>%
  arrange(mean_thou)
```

```
## # A tibble: 5 x 6
##   BOROUGH       casetot  mean mean_thou borosd sd_thou
##   <chr>           <int> <dbl>     <dbl>  <dbl>   <dbl>
## 1 STATEN ISLAND     776  45.6    0.0963   10.7  0.0237
## 2 QUEENS           4094 241.     0.106    59.5  0.0266
## 3 MANHATTAN        3572 210.     0.131    70.1  0.0436
## 4 BROOKLYN        10933 643.     0.249   173.   0.0696
## 5 BRONX            7937 467.     0.332   119.   0.0863
```

Demographic Analysis

Recall that the other major question we want answered is: Is there a relationship between the demographic of the perpetrator and the demographic of the victims? There are many ways to approach this question - for example, are the age/race/sex of the victims and the perpetrators related in some way? One way to test that is through a chi-squared test of independence - we will be focusing on comparing within the same demographic category (i.e. age v. age, etc.). Another way is to do a Cramer's V test.

Chi-Squared Tests

Shootings by Age

Null Hypothesis: The age of the victim and the age of the perpetrator are **not related (independent)**.

Alternative Hypothesis: The age of the victim and the age of the perpetrator **are related (not independent)**.

We can cross tabulate our data to get an overhead view of the proportions for the victims' and perpetrators' ages:

```
age_table <- shooting_data %>%
  select(VIC_AGE, PERP_AGE) %>%
  filter_all(all_vars(. !="UNKNOWN")) %>%
  droplevels() %>%
  mutate(VIC_AGE = set_label(VIC_AGE, "Victim Age"),
         PERP_AGE = set_label(PERP_AGE, "Perpetrator Age"))

age_table %>%
  crosstable(c(PERP_AGE), by = VIC_AGE, total = "both") %>%
  as_flextable() %>% fit_to_width(6.5)
```

label	variable	Victim Age					Total
		<18	18-24	25-44	45-64	65+	
Perpetrator Age	<18	484 (30.46%)	621 (39.08%)	397 (24.98%)	77 (4.85%)	10 (0.63%)	1589 (11.25%)
	18-24	788 (12.69%)	2758 (44.42%)	2294 (36.95%)	329 (5.30%)	40 (0.64%)	6209 (43.97%)
	25-44	262 (4.64%)	1516 (26.82%)	3352 (59.31%)	479 (8.47%)	43 (0.76%)	5652 (40.02%)
	45-64	20 (3.27%)	76 (12.42%)	327 (53.43%)	177 (28.92%)	12 (1.96%)	612 (4.33%)
	65+	0 (0%)	1 (1.67%)	25 (41.67%)	23 (38.33%)	11 (18.33%)	60 (0.42%)
	Total	1554 (11.00%)	4972 (35.21%)	6395 (45.28%)	1085 (7.68%)	116 (0.82%)	14122 (100.00%)

We can see that the 65+ category is a very small proportion of the population compared to the rest of the categories - a low expected value would invalidate the assumptions we need to make when performing a chi-squared test, so let's collapse that data in the 45-64 age bracket.

```
fct_count(age_table$VIC_AGE)
```

```
## # A tibble: 5 x 2
##   f         n
##   <fct> <int>
## 1 <18    1554
## 2 18-24  4972
## 3 25-44  6395
## 4 45-64  1085
## 5 65+     116
```

```
  #Combine age groups
age_table$PERP_AGE <-
  fct_collapse(age_table$PERP_AGE, "45+" = c("45-64", "65+"))
age_table$VIC_AGE <-
  fct_collapse(age_table$VIC_AGE, "45+" = c("45-64", "65+"))

age_table %>%
  crosstable(c(PERP_AGE), by = VIC_AGE, total = "both") %>%
  as_flextable() %>% fit_to_width(6.5)
```

label	variable	Victim Age				Total
		<18	18-24	25-44	45+	
Perpetrator Age	<18	484 (30.46%)	621 (39.08%)	397 (24.98%)	87 (5.48%)	1589 (11.25%)
	18-24	788 (12.69%)	2758 (44.42%)	2294 (36.95%)	369 (5.94%)	6209 (43.97%)
	25-44	262 (4.64%)	1516 (26.82%)	3352 (59.31%)	522 (9.24%)	5652 (40.02%)
	45+	20 (2.98%)	77 (11.46%)	352 (52.38%)	223 (33.18%)	672 (4.76%)
	Total	1554 (11.00%)	4972 (35.21%)	6395 (45.28%)	1201 (8.50%)	14122 (100.00%)

```
chisq.test(table(age_table))
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(age_table)
## X-squared = 2230, df = 9, p-value < 2.2e-16
```

```
cramerV(table(age_table))
```

```
## Cramer V
##   0.2294
```

We **reject the null hypothesis** and can conclude that the age of the victim and the age of the perpetrator are not independent. We note a weak, but statistically significant association between these variables from the low Cramer V value.

Shootings by Race

We will do a similar analysis for race and sex.

Null Hypothesis: The race of the victim and the race of the perpetrator are **not related (independent)**.

Alternative Hypothesis: The race of the victim and the race of the perpetrator **are related (not independent)**.

```
#Race Data
race_table <- shooting_data %>%
  select(VIC_RACE, PERP_RACE) %>%
  filter_all(all_vars(. !="UNKNOWN")) %>%
  droplevels() %>%
  mutate(VIC_RACE = set_label(VIC_RACE, "Victim Race"),
         PERP_RACE = set_label(PERP_RACE, "Perpetrator Race"))


race_table %>%
  crosstable(c(PERP_RACE), by = VIC_RACE, total = "both") %>%
  as_flextable() %>% fit_to_width(6.5)
```

label	variable	Victim Race						Total
		AIAN	AAPI	B	B H	W	W H	
Perpetrator Race	AIAN	0 (0%)	0 (0%)	2 (100.00%)	0 (0%)	0 (0%)	0 (0%)	2 (0.01%)
	AAPI	0 (0%)	52 (33.77%)	53 (34.42%)	13 (8.44%)	12 (7.79%)	24 (15.58%)	154 (0.99%)
	B	4 (0.04%)	157 (1.38%)	9059 (79.42%)	803 (7.04%)	197 (1.73%)	1187 (10.41%)	11407 (73.67%)
	B H	0 (0%)	18 (1.38%)	531 (40.57%)	344 (26.28%)	36 (2.75%)	380 (29.03%)	1309 (8.45%)
	W	0 (0%)	13 (4.61%)	37 (13.12%)	23 (8.16%)	157 (55.67%)	52 (18.44%)	282 (1.82%)
	W H	0 (0%)	36 (1.55%)	788 (33.82%)	406 (17.42%)	97 (4.16%)	1003 (43.05%)	2330 (15.05%)
	Total	4 (0.03%)	276 (1.78%)	10470 (67.62%)	1589 (10.26%)	499 (3.22%)	2646 (17.09%)	15484 (100.00%)

Note that the Asian/Pacific Islander and American Indian and Alaskan Native categories have a very low proportion relative to the total population; we will need to omit them in order to perform an accurate chi-squared test.

```r
race_table <- race_table %>%
  filter_all(all_vars(. != "AAPI")) %>%
  filter_all(all_vars(. != "AIAN")) %>%
  droplevels() %>%
  mutate(VIC_RACE = set_label(VIC_RACE, "Victim Race"),
         PERP_RACE = set_label(PERP_RACE, "Perpetrator Race"))

race_table %>%
  crosstable(c(PERP_RACE), by = VIC_RACE, total = "both") %>%
  as_flextable() %>% fit_to_width(6.5)
```

label	variable	Victim Race				Total
		B	B H	W	W H	
Perpetrator Race	B	9059 (80.55%)	803 (7.14%)	197 (1.75%)	1187 (10.55%)	11246 (74.48%)
	B H	531 (41.13%)	344 (26.65%)	36 (2.79%)	380 (29.43%)	1291 (8.55%)
	W	37 (13.75%)	23 (8.55%)	157 (58.36%)	52 (19.33%)	269 (1.78%)
	W H	788 (34.35%)	406 (17.70%)	97 (4.23%)	1003 (43.72%)	2294 (15.19%)
	Total	10415 (68.97%)	1576 (10.44%)	487 (3.23%)	2622 (17.36%)	15100 (100.00%)

```r
chisq.test(table(race_table))
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(race_table)
## X-squared = 5386.8, df = 9, p-value < 2.2e-16
```

```r
cramerV(table(race_table))
```

```
## Cramer V
##   0.3448
```

We **reject the null hypothesis** and can conclude that the race of the victim and the race of the perpetrator are not independent. We note a moderate, but statistically significant association between these variables from the low Cramer V value.

Shootings by Sex

Null Hypothesis: The sex of the victim and the sex of the perpetrator are **not related (independent)**.

Alternative Hypothesis: The sex of the victim and the sex of the perpetrator **are related (not independent)**.

```
#Sex Data
sex_table <- shooting_data %>%
  select(VIC_SEX, PERP_SEX) %>%
  filter_all(all_vars(. !="U")) %>%
  droplevels() %>%
  mutate(VIC_SEX = set_label(VIC_SEX, "Victim Sex"),
         PERP_SEX = set_label(PERP_SEX, "Perpetrator Sex"))

sex_table %>%
  crosstable(c(PERP_SEX), by = VIC_SEX, total = "both") %>%
  as_flextable() %>% fit_to_width(6.5)
```

label	variable	Victim Sex		Total
		F	M	
Perpetrator Sex	F	72 (17.02%)	351 (82.98%)	423 (2.67%)
	M	1666 (10.80%)	13767 (89.20%)	15433 (97.33%)
	Total	1738 (10.96%)	14118 (89.04%)	15856 (100.00%)

```
chisq.test(table(sex_table))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(sex_table)
## X-squared = 15.722, df = 1, p-value = 7.337e-05
```

```
cramerV(table(sex_table))
```

```
## Cramer V
##  0.03212
```

We **reject the null hypothesis** and can conclude that the sex of the victim and the sex of the perpetrator are not independent. We note a weak, but statistically significant association between these variables from the low Cramer V value.

Bias Analysis

Let's quickly go over some potential sources of bias. Particularly notable is almost half of the information about the perpetrator of shooting incidents is missing:

```
fct_count(shooting_data$PERP_AGE)
```

```
## # A tibble: 6 x 2
##   f           n
##   <fct>   <int>
## 1 <18      1591
## 2 18-24    6222
## 3 25-44    5687
## 4 45-64     617
## 5 65+        60
## 6 UNKNOWN 13135
```

```
fct_count(shooting_data$PERP_RACE)
```

```
## # A tibble: 7 x 2
##   f           n
##   <fct>   <int>
## 1 AIAN        2
## 2 AAPI      154
## 3 B       11432
## 4 B H      1314
## 5 UNKNOWN 11786
## 6 W         283
## 7 W H      2341
```

```
fct_count(shooting_data$PERP_SEX)
```

```
## # A tibble: 3 x 2
##   f         n
##   <fct> <int>
## 1 F       424
## 2 M     15439
## 3 U     11449
```

This could clue us into a potential large source of recall bias. It is not known how the race, age, and sex of the perpetrator are verified and recorded - for example, if shooting victims or witnesses are asked to recall information about the perpetrator, or if the person recording the data carries their own bias, this could introduce bias into the data set. A large chunk of missing data can easily cause particular demographics to be under or over-represented. As for bias within this report, a cursory glance at the cross-tables may lead the reader to make a biased assumption about a particular race, sex, or age group. While we performed analysis on the demographics of the victims and perpetrators, we primarily focused on the relationship between the victim and the perpetrator without doing an in-depth analysis on a particular race, gender, or sex.

Final Thoughts

Recall that we sought to gain insight on the following:

1. Is there a difference in the rate of shooting incidents across the boroughs of New York City?
2. Is there a relationship between the demographic of the perpetrator and the demographic of the victims?

From our analysis, we have demonstrated that:

1. shootings occur, in order of prevalence, the most per capita in the Bronx, followed by Brooklyn, Manhattan, Queens, and lastly Staten Island and
2. there is a weak, but statistically significant relationship between the demographics of the perpetrator and victim, at least within each category (age, race, and sex).

```r
sessionInfo()
```

```
## R version 4.2.2 (2022-10-31 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19045)
##
## 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] rcompanion_2.4.30 flextable_0.9.1   zoo_1.8-12        crosstable_0.6.2
##  [5] lubridate_1.9.2   forcats_1.0.0     stringr_1.5.0     dplyr_1.1.2
##  [9] purrr_1.0.1       readr_2.1.4       tidyr_1.3.0       tibble_3.2.1
## [13] ggplot2_3.4.2     tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
##   [1] TH.data_1.1-2          colorspace_2.1-0       ellipsis_0.3.2
##   [4] class_7.3-20           modeltools_0.2-23      gld_2.6.6
##   [7] httpcode_0.3.0         rstudioapi_0.14        proxy_0.4-27
##  [10] farver_2.1.1           fansi_1.0.4            mvtnorm_1.2-2
##  [13] coin_1.4-2             xml2_1.3.4             codetools_0.2-18
##  [16] splines_4.2.2          rootSolve_1.8.2.3      libcoin_1.0-9
##  [19] knitr_1.43             jsonlite_1.8.4         shiny_1.7.4
##  [22] compiler_4.2.2         httr_1.4.6             backports_1.4.1
##  [25] Matrix_1.5-1           fastmap_1.1.1          cli_3.6.1
##  [28] later_1.3.1            htmltools_0.5.5        tools_4.2.2
##  [31] gtable_0.3.3           glue_1.6.2             lmom_3.0
##  [34] Rcpp_1.0.10            cellranger_1.1.0       fontquiver_0.2.1
##  [37] vctrs_0.6.2            crul_1.4.0             nlme_3.1-160
##  [40] lmtest_0.9-40          xfun_0.39              timechange_0.2.0
##  [43] mime_0.12              lifecycle_1.0.3        MASS_7.3-58.1
##  [46] scales_1.2.1           ragg_1.2.5             hms_1.1.3
##  [49] promises_1.2.0.1       parallel_4.2.2         sandwich_3.0-2
##  [52] expm_0.999-7           fontLiberation_0.1.0   yaml_2.3.7
##  [55] curl_5.0.0             Exact_3.2              gdtools_0.3.3
##  [58] stringi_1.7.12         fontBitstreamVera_0.1.1 highr_0.10
##  [61] nortest_1.0-4          e1071_1.7-13           checkmate_2.2.0
##  [64] boot_1.3-28            zip_2.3.0              rlang_1.1.1
##  [67] pkgconfig_2.0.3        systemfonts_1.0.4      matrixStats_1.0.0
##  [70] evaluate_0.21          lattice_0.20-45        labeling_0.4.2
##  [73] tidyselect_1.2.0       plyr_1.8.8             magrittr_2.0.3
##  [76] R6_2.5.1               DescTools_0.99.50      generics_0.1.3
##  [79] multcompView_0.1-9     multcomp_1.4-25        pillar_1.9.0
##  [82] withr_2.5.0            mgcv_1.8-41            survival_3.4-0
```

```
##  [85] crayon_1.5.2           gfonts_0.2.0           uuid_1.1-0
##  [88] utf8_1.2.3             tzdb_0.4.0             rmarkdown_2.22
##  [91] officer_0.6.2          grid_4.2.2             readxl_1.4.2
##  [94] data.table_1.14.8      digest_0.6.31          xtable_1.8-4
##  [97] httpuv_1.6.11          textshaping_0.3.6      openssl_2.0.6
## [100] stats4_4.2.2           munsell_0.5.0          askpass_1.1
```