Project 1: NYPD Shooting Incident

Welly Wong

2023-04-02

Load Packages

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.5
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(treemapify)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##     combine
```

Import Data

We will import the data in csv format from a url, from this site: https://catalog.data.gov/dataset/nypd-shooting-incident-data-historic

A footnote containing incident level data and description on each field name can be found here: https://data.cityofnewyork.us/api/views/5ucz-vwe8/files/ec9fa5b4-2cfa-4af0-af44-594b85ace55b?download=true&filename=NYPD_Shootings_Incident_Level_Data_Footnotes.pdf

```
url = "https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD"
nypd = 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.
```

Introduction

This is an analysis on NYPD shooting incidents data that seek to identify which age group, race and gender of New York's population who are the most likely to get involved in a shooting incident.

When a shooting incident is reported, the New York City Police Department documents information about the incident, such as the borough in which the shooting happened, whether the shooting resulted in the death of the victim, as well as the age, race and gender of both the victim and the perpetrator's. My goal is to identify which groups of people are at a higher risk of becoming a victim, to identify concentrated gun violence locations across New York City, and to pinpoint what days of the week and what hours when most shootings occurs.

Understanding these facts may help allocate law enforcement resources, community and health services to where they are most needed.

Tidying data For ease of typing variable names, convert them to lower case.

```
nypd = nypd %>% rename_all(tolower)
```

Sum up NA's for each column

```
sapply(nypd, function(x) sum(is.na(x)))
```

```
##            incident_key              occur_date              occur_time
##                       0                       0                       0
##                   boro                precinct       jurisdiction_code
##                       0                       0                       2
##          location_desc statistical_murder_flag          perp_age_group
##                   14977                       0                    9344
##               perp_sex               perp_race           vic_age_group
##                   9310                    9310                       0
##                vic_sex                vic_race              x_coord_cd
##                       0                       0                       0
##             y_coord_cd                latitude               longitude
##                       0                       0                       0
##                lon_lat
##                       0
```

We will impute NA with "unknown"

```
nypd = nypd %>%
  replace_na(list(location_desc = "U", perp_age_group = "U", perp_sex = "U", perp_race = "U"))
```

Change data type: * incident_key: from dbl to chr * occur_date: from chr to date * boro, precinct, location_desc, perp_age_group, perp_sex, perp_race, vic_age_group, vic_sex, vic_race: from chr to factor

Discard columns: * jurisdiction_code to simplify analysis * x_coord_cd, y_coord_cd, lon_lat (this information is already contained in latitude and longitude). Recode perp_sex and vic_sex, from U to "UNKNOWN"

```
nypd = nypd %>%
  mutate(incident_key = as.character(incident_key),
         occur_date = mdy(occur_date),
         boro = as.factor(boro),
         precinct = as.factor(precinct),
         location_desc = as.factor(location_desc),
         perp_age_group = as.factor(perp_age_group),
         perp_sex = as.factor(perp_sex),
         perp_race = as.factor(perp_race),
         vic_age_group = as.factor(vic_age_group),
         vic_sex = as.factor(vic_sex),
         vic_race = as.factor(vic_race)) %>%
  select(incident_key:vic_race, latitude, longitude) %>%
  select(-jurisdiction_code)

nypd$perp_race = recode(nypd$perp_race, "UNKNOWN" = "U")
nypd$vic_race = recode(nypd$vic_race, "UNKNOWN" = "U")
nypd$perp_age_group = recode(nypd$perp_age_group, "UNKNOWN" = "U")
```

Summary

```
summary(nypd)
```

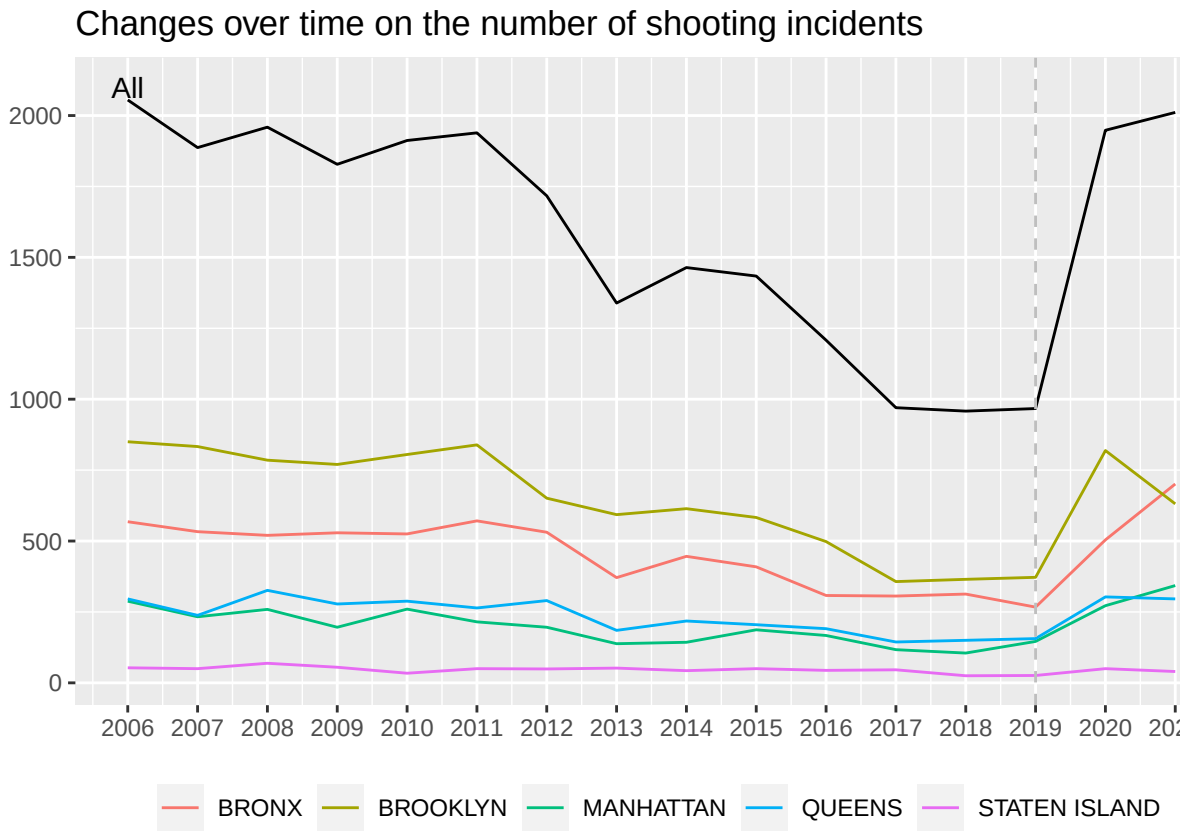
```
##  incident_key         occur_date          occur_time
##  Length:25596       Min.   :2006-01-01   Length:25596
##  Class :character   1st Qu.:2009-05-10   Class1:hms
##  Mode  :character   Median :2012-08-26   Class2:difftime
##                     Mean   :2013-06-13   Mode  :numeric
##                     3rd Qu.:2017-07-01
##                     Max.   :2021-12-31
##
##             boro          precinct                       location_desc
##  BRONX        : 7402   75     : 1470   U                         :14977
##  BROOKLYN     :10365   73     : 1372   MULTI DWELL - PUBLIC HOUS: 4559
##  MANHATTAN    : 3265   67     : 1160   MULTI DWELL - APT BUILD  : 2664
##  QUEENS       : 3828   79     :  982   PVT HOUSE                :  893
##  STATEN ISLAND:  736   44     :  949   GROCERY/BODEGA           :  622
##                        47     :  903   BAR/NIGHT CLUB           :  588
##                        (Other):18760   (Other)                  : 1293
##  statistical_murder_flag perp_age_group  perp_sex
##  Mode :logical           U      :12492   F:  371
```

```
##  FALSE:20668             18-24  : 5844   M:14416
##  TRUE :4928              25-44  : 5202   U:10809
##                          <18    : 1463
##                          45-64  :  535
##                          65+    :   57
##                          (Other):    3
##                           perp_race     vic_age_group   vic_sex
##  AMERICAN INDIAN/ALASKAN NATIVE:    2   <18    : 2681   F: 2403
##  ASIAN / PACIFIC ISLANDER      :  141   18-24  : 9604   M:23182
##  BLACK                         :10668   25-44  :11386   U:   11
##  BLACK HISPANIC                : 1203   45-64  : 1698
##  U                             :11146   65+    :  167
##  WHITE                         :  272   UNKNOWN:   60
##  WHITE HISPANIC                : 2164
##                             vic_race        latitude       longitude
##  AMERICAN INDIAN/ALASKAN NATIVE:    9   Min.   :40.51   Min.   :-74.25
##  ASIAN / PACIFIC ISLANDER      :  354   1st Qu.:40.67   1st Qu.:-73.94
##  BLACK                         :18281   Median :40.70   Median :-73.92
##  BLACK HISPANIC                : 2485   Mean   :40.74   Mean   :-73.91
##  U                             :   65   3rd Qu.:40.82   3rd Qu.:-73.88
##  WHITE                         :  660   Max.   :40.91   Max.   :-73.70
##  WHITE HISPANIC                : 3742
```

Visualizing and Analyzing data

```
nypd_historic = nypd %>% group_by(boro, year=year(ymd(occur_date))) %>% count()
all = nypd_historic %>% group_by(year) %>% summarise(n=sum(n))

ggplot(nypd_historic, aes(x=year, y=n, color=boro)) +
  geom_line() + geom_line(data=all, color="black") +
  geom_vline(xintercept = 2019, color = "gray", linetype = "dashed") +
  scale_x_continuous(breaks = 2006:2021) +
  annotate("text", x = 2006, y = 2100, label = "All") + xlab(NULL) + ylab(NULL) +
  labs(title = "Changes over time on the number of shooting incidents") +
  theme(legend.position = "bottom", legend.title = element_blank())
```



Historical Trends

There had been a general trend of a declining shooting incidents from 2006 to 2017, this is perhaps aided by the City investments in violence prevention efforts. Staten Island stands out as having a generally consistent number of incidents over the years. There was a sharp increase post Covid-19 pandemic, then the rate of increase appeared to be slowing down.

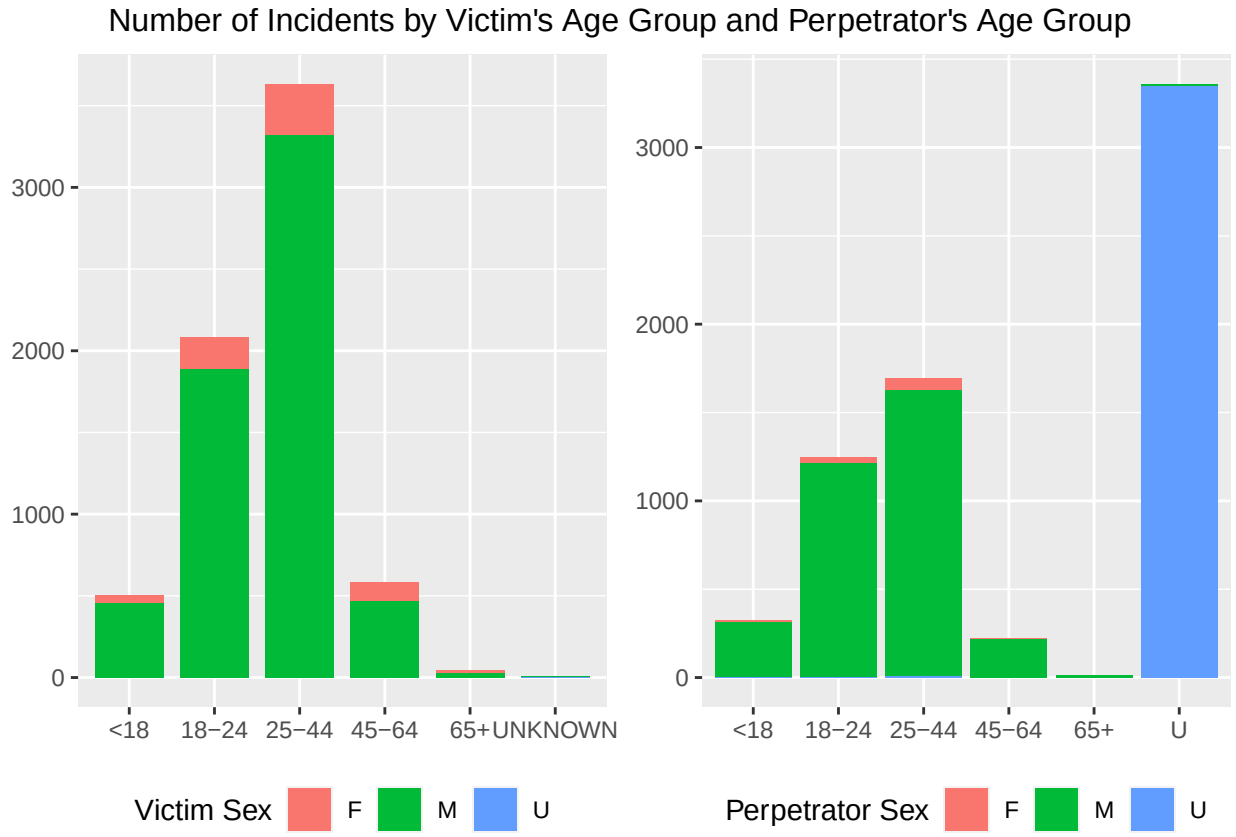
Next, We'll look at data from the past 5 years, which I believe is more relevant. Afterwards, we will analyze data for the 2 years prior to the pandemic then compared them to the 2 years after the pandemic.

```
nypd_5yr = nypd %>% filter(year(ymd(occur_date)) %in% 2017:2021)

p1 = nypd_5yr %>% group_by(vic_sex, vic_age_group) %>% count() %>%
  ggplot(aes(x = vic_age_group, y = n, fill=vic_sex)) +
    geom_bar(stat = "identity", width=0.8) + xlab(NULL) + ylab(NULL) +
    labs(fill="Victim Sex") + theme(legend.position = "bottom")

p2 = nypd_5yr %>% group_by(perp_sex, perp_age_group) %>% count() %>%
  ggplot(aes(x = perp_age_group, y = n, fill=perp_sex)) +
    geom_bar(stat = "identity") + xlab(NULL) + ylab(NULL) +
    labs(fill="Perpetrator Sex") + theme(legend.position = "bottom")

grid.arrange(p1, p2, top = "Number of Incidents by Victim's Age Group and Perpetrator's Age Group",nrow=
```



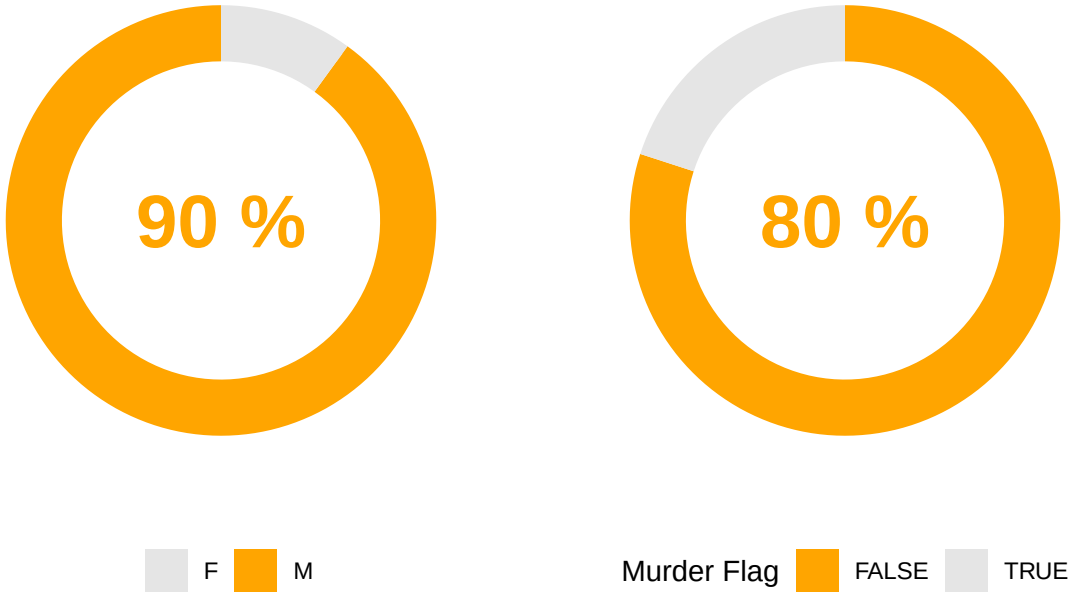

```
vic_sex_5yr = nypd_5yr %>% group_by(vic_sex) %>% summarise(counts = n()) %>%
  mutate(perct = round(counts/sum(counts), 1)) %>% arrange(desc(perct)) %>% slice_max(counts, n=2)
#levels(vic_sex_5yr$vic_sex) = c("Female", "Male", "Unknown")
center_label = paste(100*vic_sex_5yr$perct[1], "%")

p1 = ggplot(vic_sex_5yr, aes(x=1, y=perct, fill=vic_sex)) +
  geom_col() + coord_polar(theta = "y", direction = -1) + xlim(c(-2, 2)) +
  theme_void() + scale_fill_manual(values = c("grey90", "orange")) +
  labs(title = "Percentage of Victim by Gender", fill=NULL) +
  annotate("text", label = center_label, fontface = "bold", color = "orange", size = 10, x = -2, y = 0) +
  theme(plot.title = element_text(size = 15, face = "bold"), legend.position = "bottom")
```

```
fatal_5yr = nypd_5yr %>% group_by(statistical_murder_flag) %>%
  summarise(counts = n()) %>% mutate(perct = round(counts/sum(counts), 1)) %>% arrange(desc(perct))
#levels(fatal_5yr$statistical_murder_flag) = c("Non-Fatal", "Fatal")
center_label = paste(100*fatal_5yr$perct[1], "%")

p2 = ggplot(fatal_5yr, aes(x=1, y=perct, fill=statistical_murder_flag)) +
  geom_col() + coord_polar(theta = "y", direction = -1) + xlim(c(-2, 2)) +
  theme_void() + scale_fill_manual(values = c("orange", "grey90")) +
  labs(title = "Percentage of Victim Survived", fill="Murder Flag") +
  annotate("text", label = center_label, fontface = "bold", color = "orange", size = 10, x = -2, y = 0) +
  theme(plot.title = element_text(size = 15, face = "bold"), legend.position = "bottom")

grid.arrange(p1, p2, nrow=1)
```



Here we looked at the profile of the victims for the past 5 years. We can tell that most victims belong to the 25-44 and 18-24 age group, and they were mostly male. If we want to quantify it's roughly about 90% male. We also noticed that most shooting incidents were non fatal with only about 20% of all incidents resulting in the death of the victim, in other word, 80% did survived.

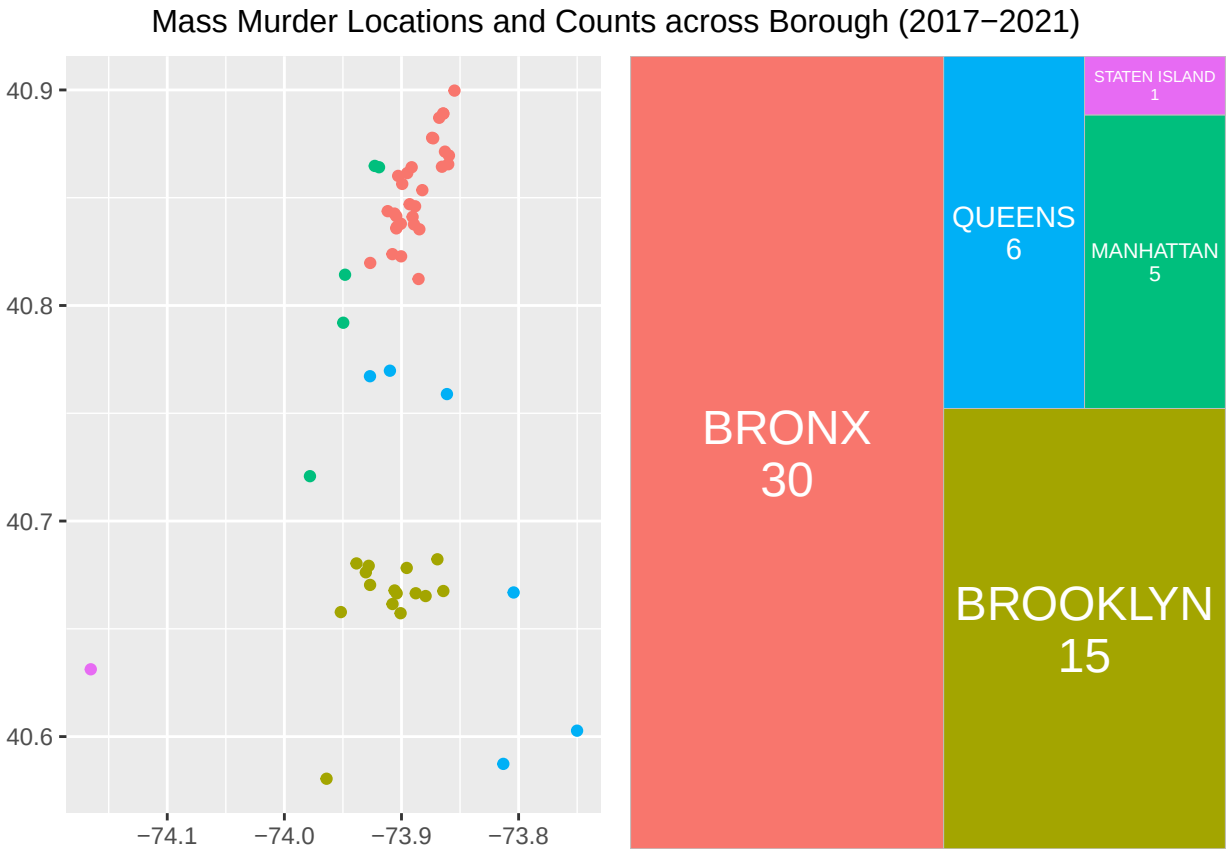
Identifying mass murder events This information is from the footnote that came with the data: A shooting incident can have multiple victims involved and as a result duplicate incident key's are produced. A shooting incident can have multiple victims involved and as a result duplicate incident key's are produced. We can identify mass murder, using events having the same incident key and TRUE statistical_murder_flag count greater than or equal to 3. Note: A mass murder is defined as the killing of three or more people at one time and in one location.

```
mass_murder_5yr = nypd_5yr %>% filter(statistical_murder_flag==TRUE) %>%
  mutate(incident_key = as.factor(incident_key)) %>% group_by(incident_key) %>%
  filter(n() >= 3) %>% ungroup() %>% distinct(incident_key, .keep_all = TRUE)

p1 = ggplot(mass_murder_5yr, aes(x = longitude, y = latitude, color = boro)) +
  geom_point() + xlab(NULL) + ylab(NULL) +
  theme(legend.position = "none")

p2 = mass_murder_5yr %>% group_by(boro) %>% summarise(mass_incident = n()) %>%
  ggplot(aes(fill=boro, area=mass_incident, label=paste0(boro, "\n", mass_incident))) +
    geom_treemap() + geom_treemap_text(color="white", place="centre") +
    theme(legend.position = "none")

grid.arrange(p1, p2, top="Mass Murder Locations and Counts across Borough (2017-2021)", nrow=1)
```



Shooting Incidents distribution across Boroughs Next, we will plot the locations, the number of shooting incidents and their percentages per million of population across Boroughs in 2021. For this, we will need population data in 2021. This population number was an estimate from 2020 Cencus. https://www.citypopulation.de/en/usa/newyorkcity/

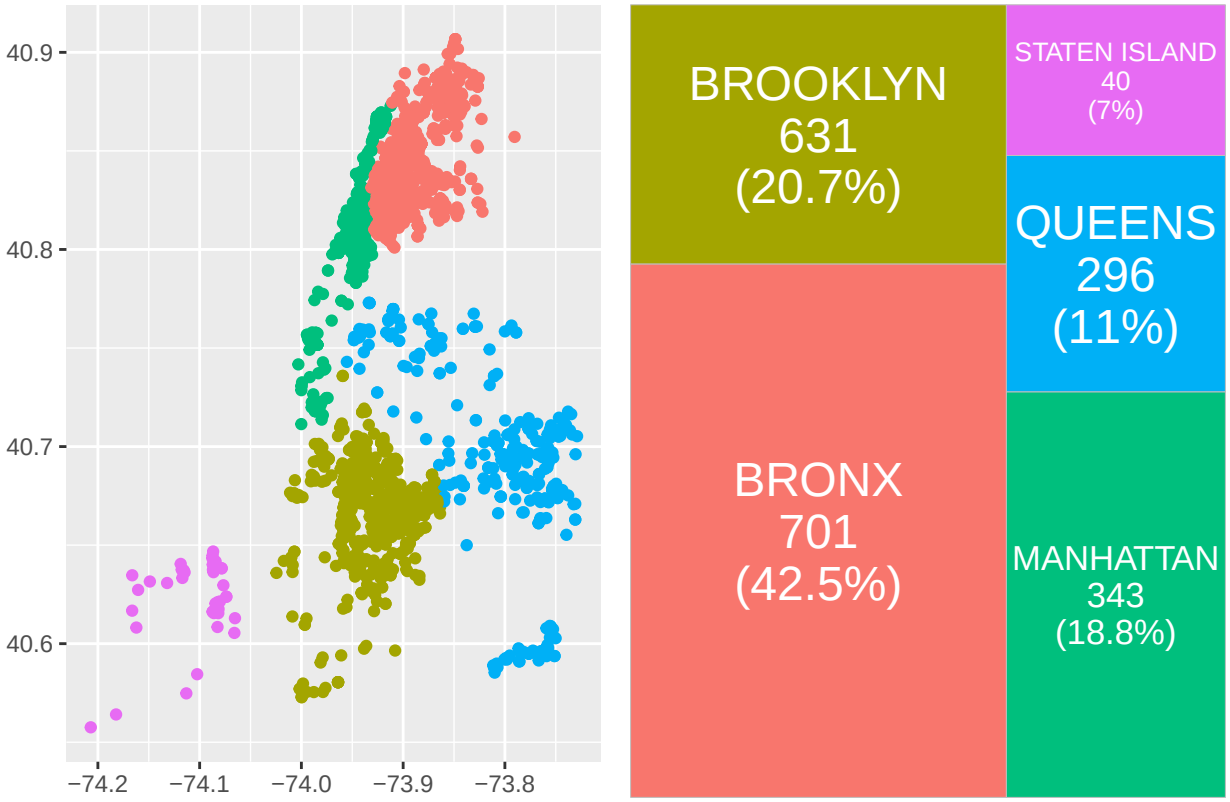
```
p1 = nypd %>% mutate(year=year(occur_date)) %>% filter(year > 2020) %>%
  ggplot(aes(x=longitude, y=latitude, color=boro)) + geom_point() +
  xlab(NULL) + ylab(NULL) +
  theme(legend.position = "none")

nypd_2021 = nypd %>% filter(year(ymd(occur_date)) %in% 2021) %>%
  group_by(boro) %>% summarise(incident = n()) %>%
  mutate(pop=c(1424948, 2641052, 1576876, 2331143, 493494), incident_per_mill = round((incident/pop)*100
         percentage = round(incident_per_mill/sum(incident_per_mill) * 100, 1))

p2 = ggplot(nypd_2021, aes(fill=boro, area=incident_per_mill,
                      label=paste0(boro, "\n", incident, "\n(", percentage, "%)"))) +
  geom_treemap() + geom_treemap_text(color="white", place="centre") +
  theme(legend.position = "none")

grid.arrange(p1, p2, top="Locations, Number of Incidents, Percentage of Incidents per population, across
```

ations, Number of Incidents, Percentage of Incidents per population, across Borough in 2



Identifying the most at risk group Next we will look at murder victims distribution across race and gender.

```
nypd_precovid = nypd %>% mutate(year = year(ymd(occur_date))) %>%
  filter(year %in% c(2018, 2019))

p1 = nypd_precovid %>% filter(statistical_murder_flag==TRUE) %>%
  group_by(vic_race, vic_sex) %>% summarise(count = n()) %>% ungroup() %>%
  mutate(pct = round(count*100/sum(count), 1)) %>%
  ggplot(aes(x = vic_race, y = vic_sex, size = count, color = vic_race)) +
  geom_point(alpha = 0.5) + scale_size(range = c(8, 60), guide = "none") +
  geom_text(aes(label=paste0(pct, "%")), size=4, color="white") +
  labs(color=NULL, title = "Murder Victims by Race and Gender Pre-Pandemic (2018-2019)",
       subtitle = "(Bubble size is proportional to the number of incidents)") + xlab(NULL) + ylab(NULL)
  theme_dark() + guides(color="none")
```

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

```
nypd_postcovid = nypd %>% mutate(year = year(ymd(occur_date))) %>%
  filter(year > 2019)

p2 = nypd_postcovid %>% filter(statistical_murder_flag==TRUE) %>%
  group_by(vic_race, vic_sex) %>% summarise(count = n()) %>% ungroup() %>%
```

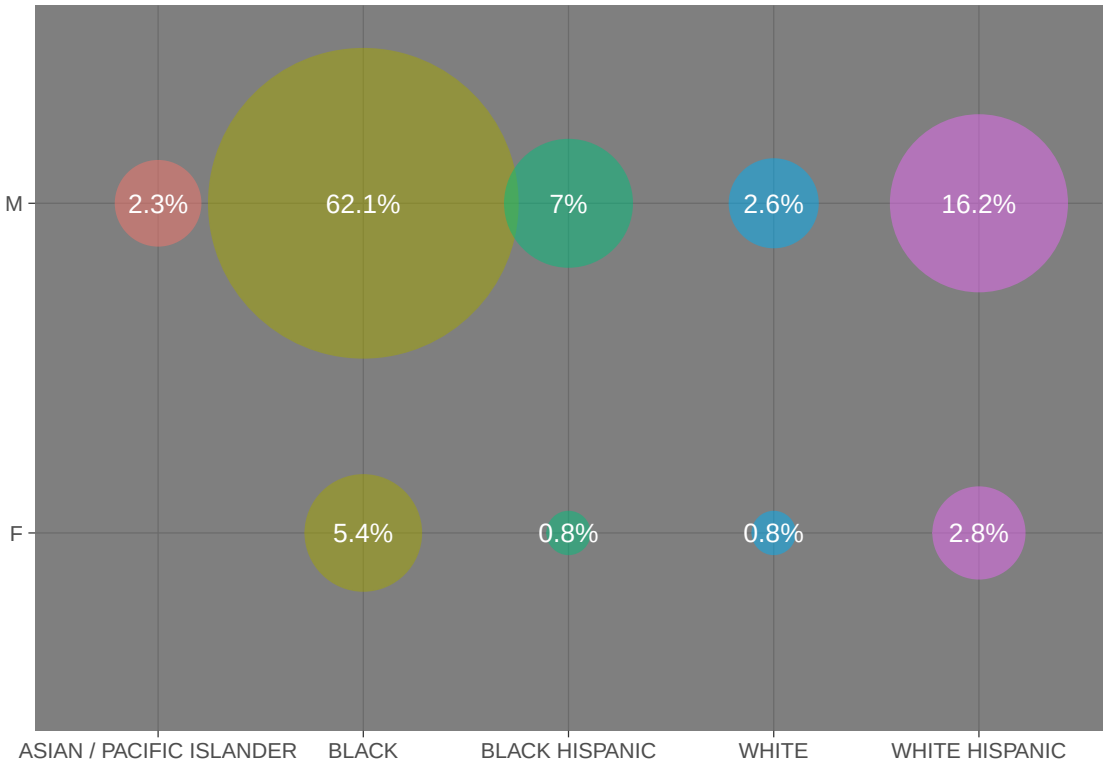
```
  mutate(pct = round(count*100/sum(count), 1)) %>%
  ggplot(aes(x = vic_race, y = vic_sex, size = count, color = vic_race)) +
  geom_point(alpha = 0.5) + scale_size(range = c(8, 60), guide = "none") +
  geom_text(aes(label=paste0(pct, "%")), size=4, color="white") +
  labs(color=NULL, title = "Post-Pandemic (2020-2021)") + xlab(NULL) + ylab(NULL) +
  theme_dark() + guides(color="none")
```

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

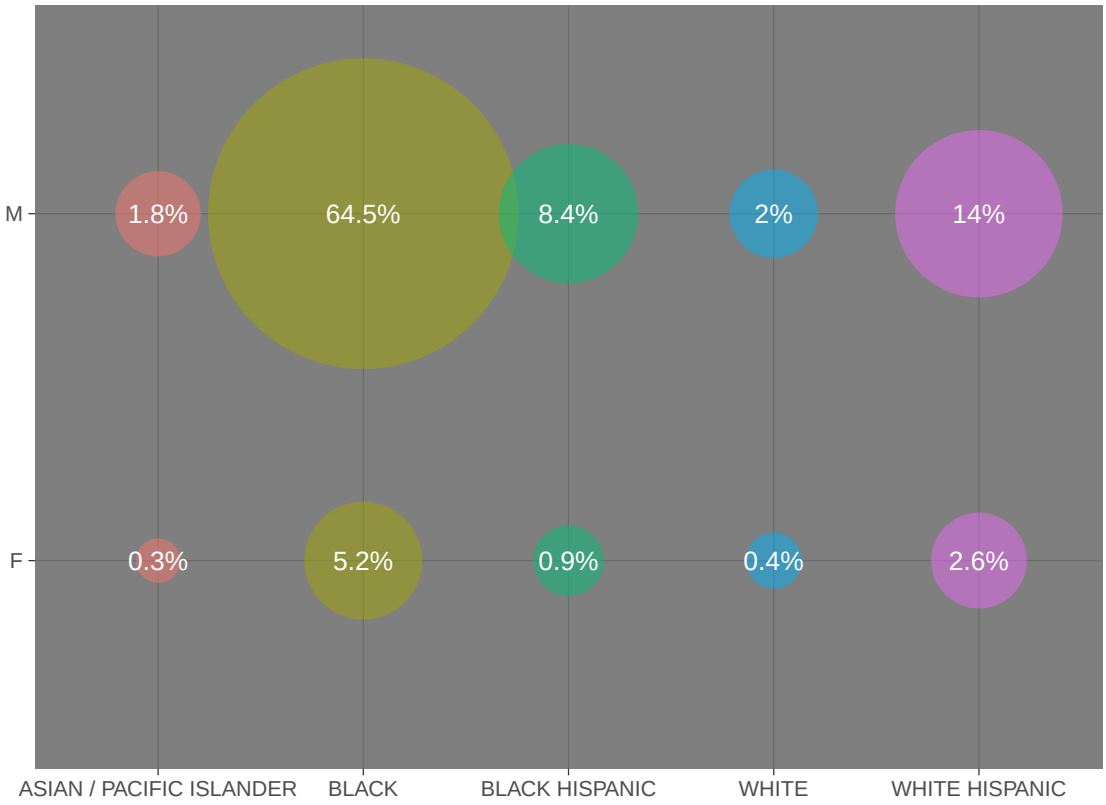
```
grid.arrange(p1, p2, ncol=1)
```

Murder Victims by Race and Gender Pre−Pandemic (2018−2019)

(Bubble size is proportional to the number of incidents)



Post−Pandemic (2020−2021)



location_desc	pre_covid_incident
U	864
MULTI DWELL - PUBLIC HOUS	287
MULTI DWELL - APT BUILD	79
BAR/NIGHT CLUB	32
PVT HOUSE	28
GROCERY/BODEGA	24
COMMERCIAL BLDG	8

location_desc	post_covid_incident
U	1922
MULTI DWELL - PUBLIC HOUS	543
MULTI DWELL - APT BUILD	171
PVT HOUSE	73
GROCERY/BODEGA	65
COMMERCIAL BLDG	34
BAR/NIGHT CLUB	23

Black-Male, White Hispanic-Male, Black Hispanic-Male consistently had the highest number of incidents where they became murder victims.

Gun violence hotspots We want to find out more specific locations where most shooting incidents occurred for the most at risk group.

```
at_risk_race = c("BLACK", "wHITE HISPANIC", "BLACK HISPANIC")
loc_precovid = nypd_precovid %>% filter(vic_race %in% at_risk_race) %>%
  filter(vic_sex == "M") %>% group_by(location_desc) %>%
  summarise(pre_covid_incident = n()) %>% ungroup() %>% arrange(desc(pre_covid_incident))

loc_postcovid = nypd_postcovid %>% filter(vic_race %in% at_risk_race) %>%
  filter(vic_sex == "M") %>% group_by(location_desc) %>%
  summarise(post_covid_incident = n()) %>% ungroup() %>% arrange(desc(post_covid_incident))

#t1 = head(loc_precovid, 7); t2 = head(loc_postcovid, 7)
knitr::kable(list(head(loc_precovid, 7), head(loc_postcovid, 7)), align = "c")
```

We will remove U (unknown) location because it gave no information on where an incident occurred. Top 6 known locations for most at risk group, are the same pre and post Covid.

```
location = c("MULTI DWELL - PUBLIC HOUS", "MULTI DWELL - APT BUILD",
      "PVT HOUSE", "GROCERY/BODEGA", "BAR/NIGHT CLUB", "COMMERCIAL BLDG")
preCovid = nypd_precovid %>%
  filter(location_desc %in% location) %>% filter(vic_race %in% at_risk_race) %>%
  filter(vic_sex=="M") %>% group_by(boro, location_desc) %>% summarise(n=n()) %>% ungroup()
```

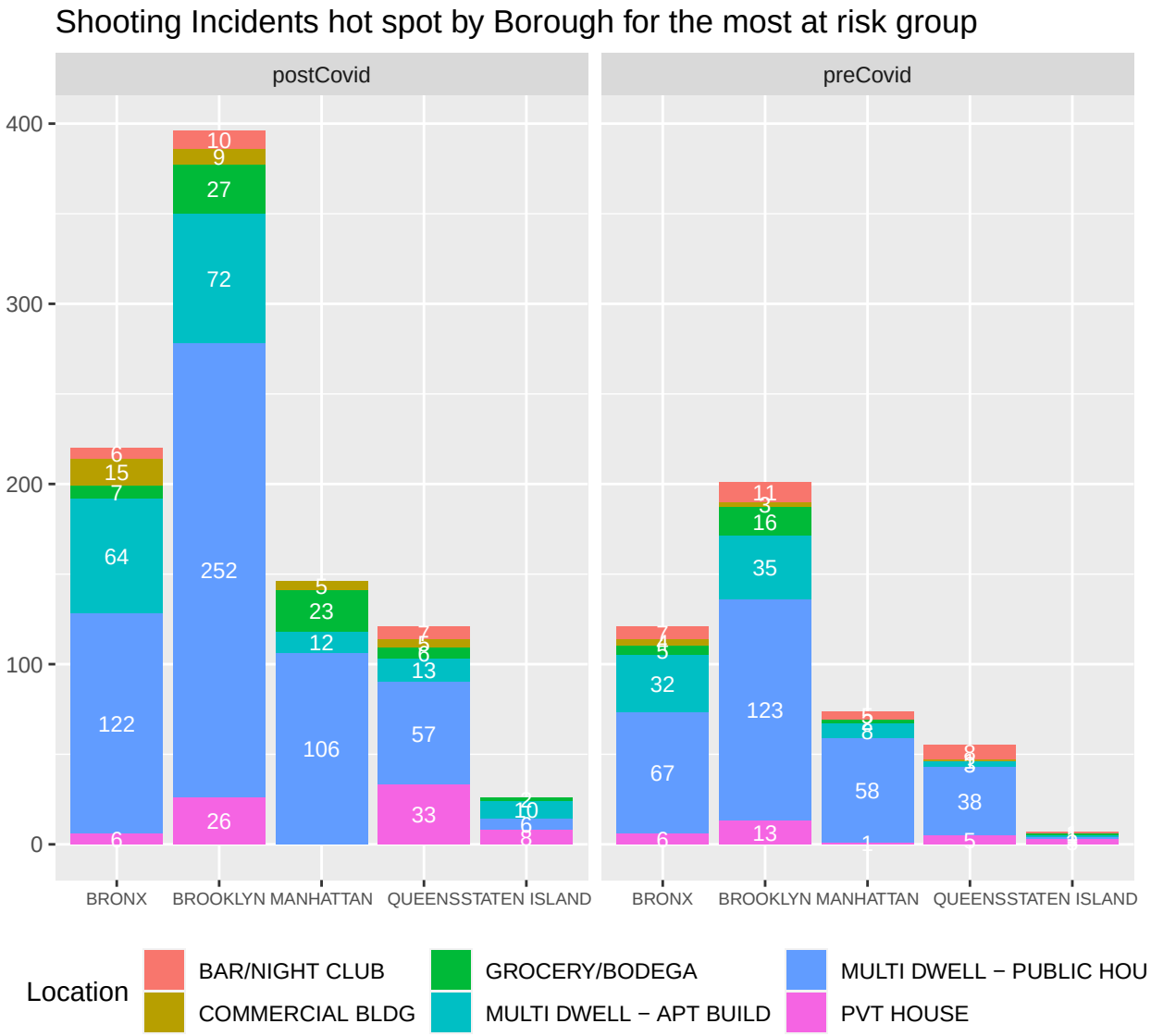
```
## `summarise()` has grouped output by 'boro'. You can override using the
## `.groups` argument.
```

```
postCovid = nypd_postcovid %>%
  filter(location_desc %in% location) %>% filter(vic_race %in% at_risk_race) %>%
  filter(vic_sex=="M") %>% group_by(boro, location_desc) %>% summarise(n=n()) %>% ungroup()
```

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

```
loc_risk = bind_rows(list(preCovid=preCovid, postCovid=postCovid), .id = 'source')

ggplot(loc_risk, aes(x=boro, y=n, fill=location_desc)) +
  geom_bar(position="stack", stat="identity") +
  geom_text(aes(label=n), position = position_stack(vjust = 0.5), size=3, color="white") +
  facet_grid(.~source) + xlab(NULL) + ylab(NULL) +
  labs(title = "Shooting Incidents hot spot by Borough for the most at risk group", fill="Location") +
  theme(legend.position = "bottom", legend.direction = "horizontal", axis.text.x = element_text(size=6.8
```



The number of incidents at Bar/night club appears to have decreased significantly in Manhattan post pandemic.

The number of incidents at Pvt House increased quite sharply (from 5 to 33) in Queens post pandemic, perhaps more resources could be allocated there.

The majority of incidents occured in multi-dwelling public housing. People living in public housing tend to have lower incomes, lower education levels, and higher unemployment rates. More study needed in order to understand why so many people in this most vulnerable group are becoming the victim of shooting incidents.

```
hr = c(paste(c(12,1:11),"AM"), paste(c(12,1:11),"PM"))
p1 = nypd_precovid %>%
  mutate(day = wday(occur_date, label=TRUE), hour = hour(hms(as.character(occur_time)))) %>%
  mutate(hour = factor(hour, level = 0:23, label = hr)) %>% select(-c(occur_date, occur_time)) %>%
  group_by(day, hour) %>% summarise(incidents = n()) %>% ungroup() %>%
  ggplot(aes(x = hour, y = day, fill = incidents)) + xlab(NULL) + ylab(NULL) +
  geom_tile() + scale_fill_gradient(low = "white", high = "darkblue") +
  labs(title = paste("Heatmap on the number of Incidents - Days of the week vs Hours \n", "Pre-Pandemic
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5), legend.title = element_text(), legend.posi
        legend.direction = "horizontal", legend.key.width = unit(2, "cm"))
```

Heatmap pre and post Pandemic

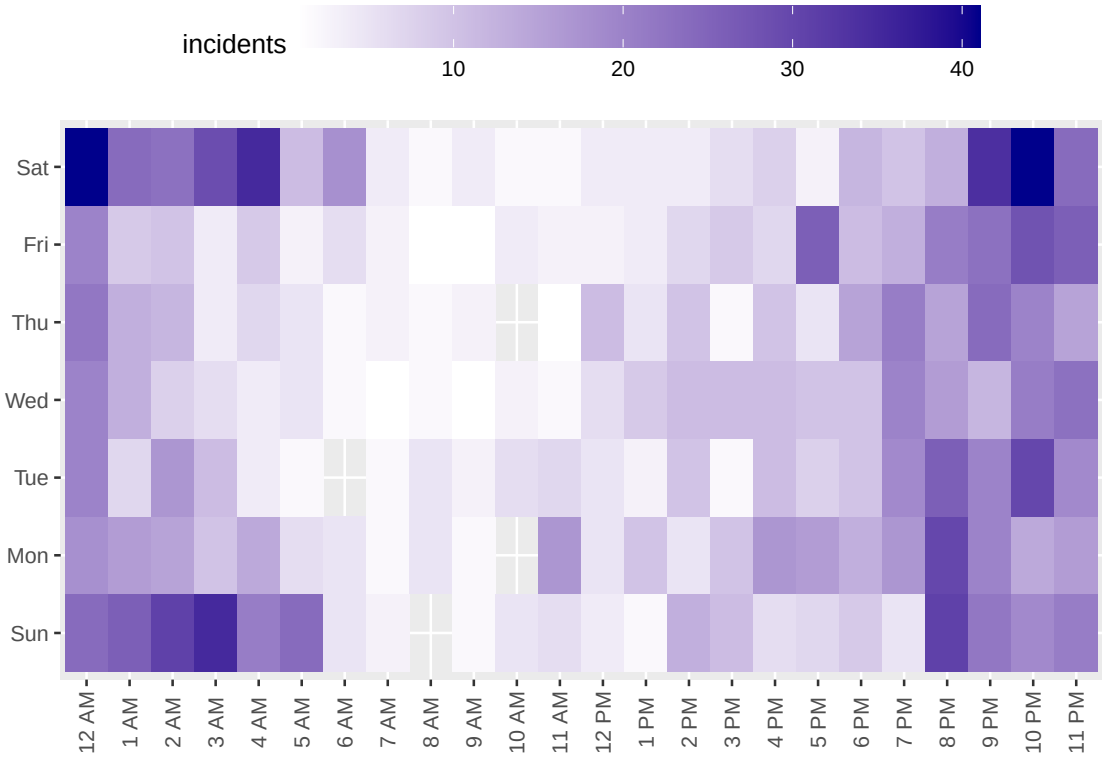
```
## `summarise()` has grouped output by 'day'. You can override using the `.groups`
## argument.
```

```
p2 = nypd_postcovid %>%
  mutate(day = wday(occur_date, label=TRUE), hour = hour(hms(as.character(occur_time)))) %>%
  mutate(hour = factor(hour, level = 0:23, label = hr)) %>% select(-c(occur_date, occur_time)) %>%
  group_by(day, hour) %>% summarise(incidents = n()) %>% ungroup() %>%
  ggplot(aes(x = hour, y = day, fill = incidents)) +
  geom_tile() + scale_fill_gradient(low = "white", high = "darkblue") +
  labs(title = "\n Post-Pandemic (2020-2021)") + xlab(NULL) + ylab(NULL) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5), legend.title = element_text(), legend.posi
        legend.direction = "horizontal", legend.key.width = unit(2, "cm"))
```

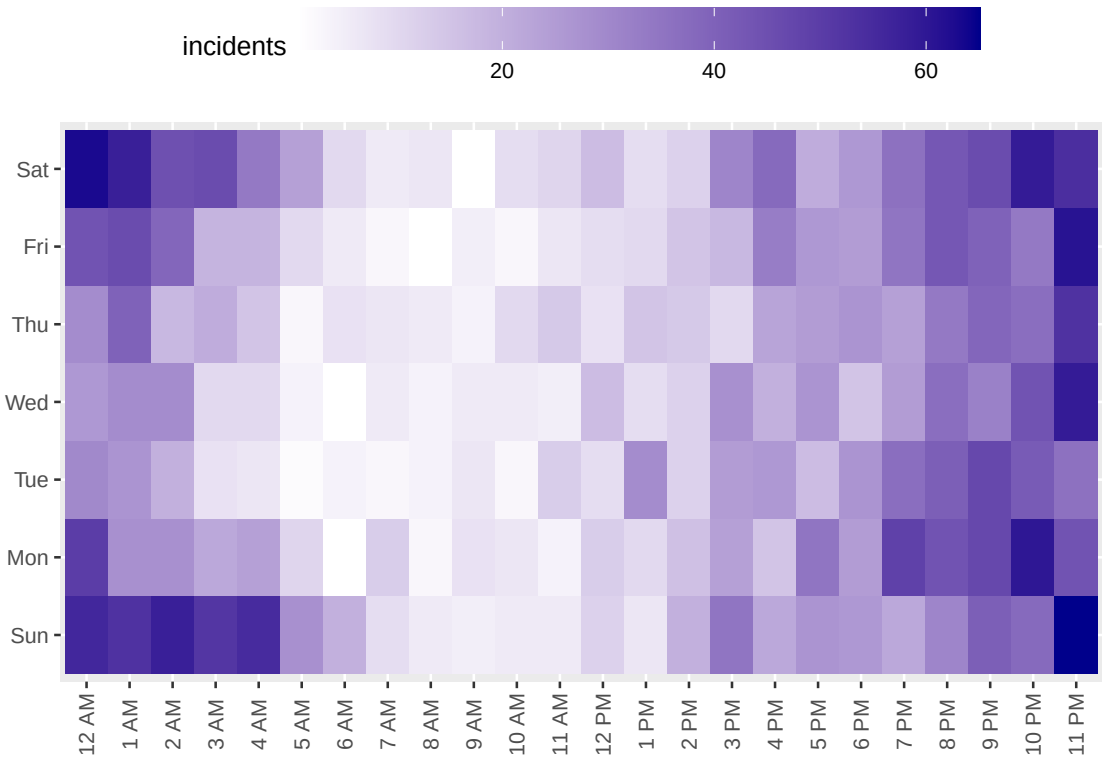
```
## `summarise()` has grouped output by 'day'. You can override using the `.groups`
## argument.
```

```
grid.arrange(p1, p2, nrow = 2)
```

Heatmap on the number of Incidents − Days of the week vs Hours
Pre−Pandemic (2018−2019)



Post−Pandemic (2020−2021)



Heatmap pre-pandemic observation: Most shooting incidents happened on a Saturday or a Sunday, during the hours between 8PM to 4 AM

Heatmap post-pandemic observation: Clearly more shooting incidents, apparent by more areas with darker color indicating more incidents. More incidents on a Monday between 10PM to 12AM, also more incidents on a Friday between 11PM to 1AM.

Information pertaining to time and locations can be used to allocate law enforcement to those areas during peak shooting times.

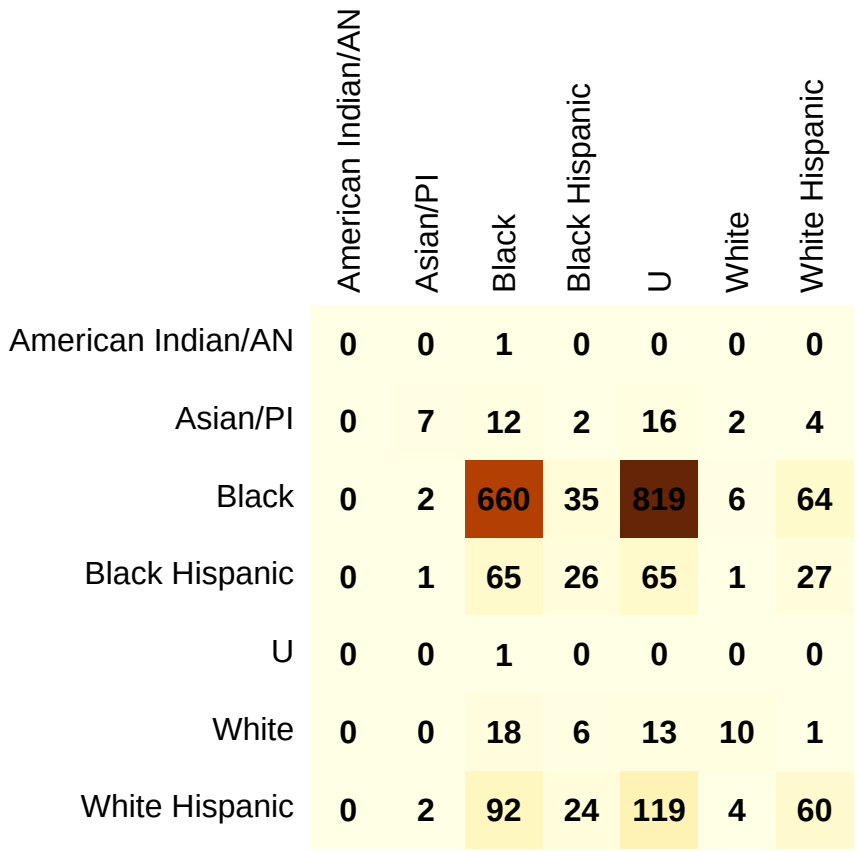
Chi-Square Test and Logistic Regression Model We will be using data from the last 5 years for our statistical model, since again it is more relevant. We will use days of the week and hours, the 6 hot spots for location that we identified earlier,

```
nypd_5yr_model = nypd_5yr %>%
  mutate(day = wday(occur_date), hour = hour(hms(as.character(occur_time)))) %>%
  mutate(hour = factor(hour, level = 0:23, label = hr), day = factor(day)) %>%
  filter(location_desc %in% location)
```

Is there a relationship between victim's race and perpetrator's race?

```
levels(nypd_5yr_model$vic_race) = c("American Indian/AN", "Asian/PI", "Black", "Black Hispanic", "U", "
levels(nypd_5yr_model$perp_race) = c("American Indian/AN", "Asian/PI", "Black", "Black Hispanic", "U",
table_5yr = as.matrix(table(nypd_5yr_model$vic_race, nypd_5yr_model$perp_race))

corrplot::corrplot(table_5yr, is.corr=FALSE, method="shade", addCoef.col="black", tl.col = "black", cl.
```



From this contingency table, we observed that most shooting incidents involved Black victim with Unknown perpetrator and Black perpetrator. Incidents involving Black victim and White Hispanic perpetrator is pretty high. The reverse is also true, many White Hispanic victim vs Black perpetrator.

```
chisq.test(nypd_5yr_model$vic_race, nypd_5yr_model$perp_race, simulate.p.value=TRUE)
```

```
##
##  Pearson's Chi-squared test with simulated p-value (based on 2000
##  replicates)
##
## data:  nypd_5yr_model$vic_race and nypd_5yr_model$perp_race
## X-squared = 601.02, df = NA, p-value = 0.0004998
```

The p value from Chi-Square Test is significant (0.0004998) compared to standard alpha = 0.05, suggesting that there is a relationship between victim's race and perpetrator's race.

In the logistic model the log odds of the outcome is modeled as a linear combination of the predictor variables. We will use statistical_murder_flag variable as the outcome and boro, precinct, location_desc, perp_race, per_age_group, perp_sex, hour, latitude and longitude as predictors. We will use stepAIC() from MASS package to pick our best model. We did not load library(MASS) up front since it may conflict with select() from dplyr in tidyverse causing further problem when knitting.

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
full = glm(statistical_murder_flag ~ boro + precinct + location_desc + perp_age_group + day +
             hour + perp_sex + perp_race + latitude + longitude, data=nypd_5yr_model, family=binomial)
empty = glm(statistical_murder_flag ~ 1 , data=nypd_5yr_model, family=binomial)
fit_5yr = stepAIC(object=empty , scope=list(upper = full, lower=empty),
          direction="forward", trace=FALSE)
summary(fit_5yr)
```

```
##
## Call:
## glm(formula = statistical_murder_flag ~ perp_age_group + boro,
##     family = binomial, data = nypd_5yr_model)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2384  -0.7878  -0.6081  -0.4895   2.0886
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -1.14914    0.26757  -4.295 1.75e-05 ***
## perp_age_group18-24  0.18221    0.27882   0.654 0.513429
## perp_age_group25-44  0.44133    0.26970   1.636 0.101765
```

```
## perp_age_group45-64  1.16780    0.35144   3.323 0.000891 ***
## perp_age_group65+   -0.00798    0.84350  -0.009 0.992452
## perp_age_groupU     -0.40081    0.26796  -1.496 0.134707
## boroBROOKLYN        -0.04418    0.12666  -0.349 0.727216
## boroMANHATTAN       -0.51142    0.17505  -2.922 0.003483 **
## boroQUEENS          -0.23256    0.17731  -1.312 0.189656
## boroSTATEN ISLAND    0.12364    0.29690   0.416 0.677093
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2323.0  on 2164  degrees of freedom
## Residual deviance: 2243.9  on 2155  degrees of freedom
## AIC: 2263.9
##
## Number of Fisher Scoring iterations: 4
```

stepAIC picked this model: statistical_murder_flag ~ perp_age_group + boro + longitude

The Coefficients for perpetrator age group 45-64 is significant and positive.

The perpetrator being in the age group 45-64, changes the log odds of the victim getting killed when involved in a shooting incident by 1.168 compare to when the perpetrator is in the age group < 18.

The Coefficients for Manhattan is also significant and it is negative. Interpretation: When an incident occurred in Manhattan, the log odds of the victim getting murdered when involved in a shooting incident decreases by 0.511 compare to if the incident to have occurred in the Bronx.

Conclusion

If we look at the overall data, there had been a general trend of a declining shooting incidents from 2006 to 2017, this is perhaps aided by the City investments in violence prevention efforts. Staten Island stands out as having a generally consistent number of incidents over the years. There was a sharp increase post Covid-19 pandemic, then the rate of increase appears to be slowing down.

Brooklyn (20.7%), Bronx (42.5%) and Manhattan (18.8%) had the highest shooting incidents per million in 2021. These 3 Boroughs accounted for over 82% of the total New York City's shooting incidents.

We identified 6 hotspots for the most at risk group, with the majority of incidents occurred in multi-dwelling public housing. Generally, the demographic of people living in public housing points to lower incomes, lower educational levels, and higher unemployment rates. More study needed if we want to understand why so many people in these most at risk groups are becoming the victim of gun violence.

Black-Male, White Hispanic-Male, and Black Hispanic-Male consistently had the highest number of incidents where they were the victims of shooting incidents. They accounted for over 80% of total murder cases.

We identified days of the week and hours when most shooting occurred. by far the majority of incidents reported took place on a Saturday and Sunday. We had more incidents on a Monday between 10PM to 12AM, and on a Friday between 11PM to 1AM post pandemic.

Information pertaining to time and locations can be used to allocate law enforcement to those areas during peak shooting times.

Identify Bias I wanted to find out if there was a relationship between the victim's race and the perpetrator's race. In order to find the answer to this question we have to be extra cautious since any statistical findings linking a victim's race to the perpetrator's race may cause bias towards a certain race by readers. We have to really check the validity of our data, and lower alpha to prevent false positive findings.

```
sessionInfo()
```

Session Info

```
## R version 4.2.3 (2023-03-15 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] MASS_7.3-58.2    gridExtra_2.3    treemapify_2.5.5 lubridate_1.8.0
##  [5] forcats_0.5.2    stringr_1.4.1    dplyr_1.0.10     purrr_0.3.5
##  [9] readr_2.1.3      tidyr_1.2.1      tibble_3.1.8     ggplot2_3.3.6
## [13] 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_0.17       highr_0.9
## [10] httr_1.4.4          pillar_1.8.1        rlang_1.0.6
## [13] curl_4.3.3          googlesheets4_1.0.1 readxl_1.4.1
## [16] rstudioapi_0.14     rmarkdown_2.17      labeling_0.4.2
## [19] googledrive_2.0.0   bit_4.0.4           munsell_0.5.0
## [22] broom_1.0.1         compiler_4.2.3      modelr_0.1.9
## [25] xfun_0.37           pkgconfig_2.0.3     htmltools_0.5.3
## [28] ggfittext_0.9.1     tidyselect_1.2.0    fansi_1.0.3
## [31] crayon_1.5.2        tzdb_0.3.0          dbplyr_2.2.1
## [34] withr_2.5.0         grid_4.2.3          jsonlite_1.8.2
## [37] gtable_0.3.1        lifecycle_1.0.3     DBI_1.1.3
## [40] magrittr_2.0.3      scales_1.2.1        cli_3.4.1
## [43] stringi_1.7.8       vroom_1.6.0         farver_2.1.1
## [46] fs_1.5.2            xml2_1.3.3          ellipsis_0.3.2
## [49] generics_0.1.3      vctrs_0.4.2         tools_4.2.3
## [52] bit64_4.0.5         glue_1.6.2          hms_1.1.2
## [55] parallel_4.2.3      fastmap_1.1.0       yaml_2.3.5
## [58] colorspace_2.0-3    gargle_1.2.1        corrplot_0.92
## [61] rvest_1.0.3         knitr_1.40          haven_2.5.1
```