NYPD Shooting Incident Data Report

DTSA 5301 FINAL

Data:

NYPD Shooting Incident Data (Historic)

This is a breakdown of every shooting incident that occurred in NYC going back to 2006 through the end of the previous calendar year. This data is manually extracted every quarter and reviewed by the Office of Management Analysis and Planning before being posted on the NYPD website. Each record represents a shooting incident in NYC and includes information about the event, the location and time of occurrence. In addition, information related to suspect and victim demographics is also included. This data can be used by the public to explore the nature of shooting/criminal activity. Please refer to NYPD Shooting Incident Data (Historic) - CKAN for additional information about this dataset.

Analysis Deliverables:

1. Day(s) and time(s) when the most shooting incidents occurred
2. Location(s) where the most shooting incidents occurred
3. Age, sex, and race of the perpetrators
4. Age, sex, and race of the victims
5. Best predictor(s) of shooting incidents

Load/Install Packages

```
require('pacman', quietly=T)

pacman::p_load(tidyverse, janitor, install = T)

if(!'relaimpo' %in% installed.packages()){
  install.packages('relaimpo', dependencies=T, quiet=T)
  }else{suppressMessages(library(relaimpo, include.only='calc.relimp'))}
```

Read Data

```
### Read in CSV-formatted dataset from URL
data<-read.csv('https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD')
```

```
### Preview dataset
# remove empty cells
# shorten number of previewed elements to 3 per variable
# shorten number of characters shoum for character strings to 20
str(data[apply(data !='', 1, all),], vec.len=3, nchar.max=20)
```

```
## 'data.frame':    7244 obs. of  19 variables:
##  $ INCIDENT_KEY           : int  22795| __truncated__ ...
##  $ OCCUR_DATE             : chr  "05/09/2021" "03/10/2021" "08/13/2021" ...
##  $ OCCUR_TIME             : chr  "02:50:00" "07:30:00" "01:00:00" ...
##  $ BORO                   : chr  "BRONX" "MANHATTAN" "QUEENS" ...
##  $ PRECINCT               : int  41 28| __truncated__ ...
##  $ JURISDICTION_CODE      : int  2 0 0 2 2 0 2 0 ...
##  $ LOCATION_DESC          : chr  "MUL"| __truncated__ "MUL"| __truncated__ "BAR/NIGHT CLUB" ...
##  $ STATISTICAL_MURDER_FLAG: chr  "true" "false" "true" ...
##  $ PERP_AGE_GROUP         : chr  "25-44" "25-44" "18-24" ...
##  $ PERP_SEX               : chr  "M" "M" "M" ...
##  $ PERP_RACE              : chr  "BLACK" "BLACK" "BLACK" ...
##  $ VIC_AGE_GROUP          : chr  "25-44" "18-24" "18-24" ...
##  $ VIC_SEX                : chr  "M" "M" "F" ...
##  $ VIC_RACE               : chr  "BLACK HISPANIC" "BLACK" "BLACK" ...
##  $ X_COORD_CD             : num  10146| __truncated__ ...
##  $ Y_COORD_CD             : num  24006| __truncated__ ...
##  $ Latitude               : num  40.8 40.8 40.8 40.8 ...
##  $ Longitude              : num  -73.9| __truncated__ ...
##  $ Lon_Lat                : chr  "POI"| __truncated__ "POI"| __truncated__ "POI"| __truncated__ ...
```

Tidy and Transform Data

ANALYSIS VARIABLES:

- 'INCIDENT_KEY' = Randomly generated persistent ID for each arrest
- 'OCCUR_DATE' = Exact date of the shooting incident
- 'OCCUR_TIME' = Exact time of the shooting incident
- 'BORO' = Borough where the shooting incident occurred
- 'LOCATION_DESC' = Location of the shooting incident
- 'PERP_AGE_GROUP' = Perpetrator's age within a category
- 'PERP_SEX' = Perpetrator's sex description
- 'PERP_RACE' = Perpetrator's race description
- 'VIC_AGE_GROUP' = Victim's age within a category
- 'VIC_SEX' = Victim's sex description
- 'VIC_RACE' = Victim's race description

```
### Select relevant variables for analysis and set blanks in data to NA
data=data %>% select(INCIDENT_KEY, OCCUR_DATE, OCCUR_TIME, BORO, LOCATION_DESC,
                     starts_with('perp'), starts_with('vic')) %>% # select relevant
                     ↪  variables
  na_if('') # set blanks to NA

head(data, 10) # preview dataset
```

```
##    INCIDENT_KEY OCCUR_DATE OCCUR_TIME      BORO             LOCATION_DESC
## 1      24050482 08/27/2006   05:35:00     BRONX                      <NA>
## 2      77673979 03/11/2011   12:03:00    QUEENS                      <NA>
## 3     226950018 04/14/2021   21:08:00     BRONX           COMMERCIAL BLDG
## 4     237710987 12/10/2021   19:30:00     BRONX                      <NA>
## 5     224701998 02/22/2021   00:18:00 MANHATTAN                      <NA>
## 6     225295736 03/07/2021   06:15:00  BROOKLYN                      <NA>
## 7     231190175 07/21/2021   00:40:00 MANHATTAN                      <NA>
## 8     233429421 09/11/2021   20:20:00 MANHATTAN MULTI DWELL - PUBLIC HOUS
## 9     227950661 05/09/2021   02:50:00     BRONX MULTI DWELL - PUBLIC HOUS
## 10    227344198 04/23/2021   13:25:00  BROOKLYN                      <NA>
##    PERP_AGE_GROUP PERP_SEX      PERP_RACE VIC_AGE_GROUP VIC_SEX       VIC_RACE
## 1            <NA>     <NA>           <NA>         25-44       F BLACK HISPANIC
## 2            <NA>     <NA>           <NA>           65+       M          WHITE
## 3            <NA>     <NA>           <NA>         18-24       M          BLACK
## 4            <NA>     <NA>           <NA>         25-44       M          BLACK
## 5            <NA>     <NA>           <NA>         25-44       M BLACK HISPANIC
## 6           25-44        M BLACK HISPANIC         25-44       M WHITE HISPANIC
## 7           25-44        M          BLACK         25-44       M          BLACK
## 8            <NA>     <NA>           <NA>         18-24       M          BLACK
## 9           25-44        M          BLACK         25-44       M BLACK HISPANIC
## 10           <NA>     <NA>           <NA>         18-24       M          BLACK
```

```
### See total number of NAs in each variable in data
sapply(data, function(x) sum(is.na(x)))
```

```
##   INCIDENT_KEY     OCCUR_DATE     OCCUR_TIME           BORO  LOCATION_DESC
##              0              0              0              0          14977
## PERP_AGE_GROUP       PERP_SEX      PERP_RACE  VIC_AGE_GROUP        VIC_SEX
##           9344           9310           9310              0              0
##       VIC_RACE
##              0
```

```
### In order to avoid/minimize bias, convert NA values to 'UNKNOWN' so they can still be
↪  reported
data=data %>% replace(is.na(.), 'UNKNOWN')

### Make sure no more NA values in dataset
any(is.na(data)) # if no NA values in data output is 'FALSE'
```

```
## [1] FALSE
```

```
### View unique elements in each analysis variable
lapply(select(data, BORO:VIC_RACE), unique)
```

```
## $BORO
## [1] "BRONX"         "QUEENS"        "MANHATTAN"     "BROOKLYN"
## [5] "STATEN ISLAND"
##
## $LOCATION_DESC
##  [1] "UNKNOWN"                    "COMMERCIAL BLDG"
##  [3] "MULTI DWELL - PUBLIC HOUS" "GROCERY/BODEGA"
##  [5] "MULTI DWELL - APT BUILD"   "BAR/NIGHT CLUB"
##  [7] "PVT HOUSE"                  "HOSPITAL"
##  [9] "HOTEL/MOTEL"                "GAS STATION"
## [11] "DEPT STORE"                 "BEAUTY/NAIL SALON"
## [13] "RESTAURANT/DINER"           "BANK"
## [15] "FAST FOOD"                  "DRY CLEANER/LAUNDRY"
## [17] "NONE"                       "CLOTHING BOUTIQUE"
## [19] "SOCIAL CLUB/POLICY LOCATI" "SMALL MERCHANT"
## [21] "LIQUOR STORE"               "SUPERMARKET"
## [23] "SHOE STORE"                 "SCHOOL"
## [25] "STORE UNCLASSIFIED"         "CHAIN STORE"
## [27] "DRUG STORE"                 "TELECOMM. STORE"
## [29] "JEWELRY STORE"              "FACTORY/WAREHOUSE"
## [31] "CANDY STORE"                "VARIETY STORE"
## [33] "ATM"                        "GYM/FITNESS FACILITY"
## [35] "VIDEO STORE"                "DOCTOR/DENTIST"
## [37] "LOAN COMPANY"               "PHOTO/COPY STORE"
## [39] "CHECK CASH"                 "STORAGE FACILITY"
##
## $PERP_AGE_GROUP
## [1] "UNKNOWN" "25-44"   "18-24"   "<18"     "45-64"   "65+"     "1020"
## [8] "940"     "224"
##
## $PERP_SEX
## [1] "UNKNOWN" "M"       "F"       "U"
##
## $PERP_RACE
## [1] "UNKNOWN"                         "BLACK HISPANIC"
## [3] "BLACK"                           "WHITE HISPANIC"
## [5] "WHITE"                           "ASIAN / PACIFIC ISLANDER"
## [7] "AMERICAN INDIAN/ALASKAN NATIVE"
##
## $VIC_AGE_GROUP
## [1] "25-44"   "65+"     "18-24"   "<18"     "45-64"   "UNKNOWN"
##
## $VIC_SEX
## [1] "F" "M" "U"
##
## $VIC_RACE
## [1] "BLACK HISPANIC"                 "WHITE"
## [3] "BLACK"                          "WHITE HISPANIC"
## [5] "ASIAN / PACIFIC ISLANDER"       "AMERICAN INDIAN/ALASKAN NATIVE"
## [7] "UNKNOWN"
```

```r
### Remove extraneous/outlier values from dataset
data=data %>% filter(PERP_AGE_GROUP!='1020' & PERP_AGE_GROUP!='940' &
↪  PERP_AGE_GROUP!='224')

### Recode 'U' to 'UNKNOWN' in 'PERP_SEX' & 'VIC_SEX' variables
### Recode 'NONE' to 'UNKNOWN' in 'LOCATION_DESC' variable
data=data %>% mutate(across(c(PERP_SEX, VIC_SEX), ~recode(., 'U'='UNKNOWN')),
                     LOCATION_DESC=recode(LOCATION_DESC, 'NONE'='UNKNOWN'))

### View unique elements in each analysis variable to make sure everything is correct
lapply(select(data, BORO:VIC_RACE), unique)
```

```
## $BORO
## [1] "BRONX"         "QUEENS"        "MANHATTAN"     "BROOKLYN"
## [5] "STATEN ISLAND"
##
## $LOCATION_DESC
##  [1] "UNKNOWN"                    "COMMERCIAL BLDG"
##  [3] "MULTI DWELL - PUBLIC HOUS" "GROCERY/BODEGA"
##  [5] "MULTI DWELL - APT BUILD"   "BAR/NIGHT CLUB"
##  [7] "PVT HOUSE"                  "HOSPITAL"
##  [9] "HOTEL/MOTEL"                "GAS STATION"
## [11] "DEPT STORE"                 "BEAUTY/NAIL SALON"
## [13] "RESTAURANT/DINER"           "BANK"
## [15] "FAST FOOD"                  "DRY CLEANER/LAUNDRY"
## [17] "CLOTHING BOUTIQUE"          "SOCIAL CLUB/POLICY LOCATI"
## [19] "SMALL MERCHANT"             "LIQUOR STORE"
## [21] "SUPERMARKET"                "SHOE STORE"
## [23] "SCHOOL"                     "STORE UNCLASSIFIED"
## [25] "CHAIN STORE"                "DRUG STORE"
## [27] "TELECOMM. STORE"            "JEWELRY STORE"
## [29] "FACTORY/WAREHOUSE"          "CANDY STORE"
## [31] "VARIETY STORE"              "ATM"
## [33] "GYM/FITNESS FACILITY"       "VIDEO STORE"
## [35] "DOCTOR/DENTIST"             "LOAN COMPANY"
## [37] "PHOTO/COPY STORE"           "CHECK CASH"
## [39] "STORAGE FACILITY"
##
## $PERP_AGE_GROUP
## [1] "UNKNOWN" "25-44"   "18-24"   "<18"     "45-64"   "65+"
##
## $PERP_SEX
## [1] "UNKNOWN" "M"       "F"
##
## $PERP_RACE
## [1] "UNKNOWN"                         "BLACK HISPANIC"
## [3] "BLACK"                           "WHITE HISPANIC"
## [5] "WHITE"                           "ASIAN / PACIFIC ISLANDER"
## [7] "AMERICAN INDIAN/ALASKAN NATIVE"
##
## $VIC_AGE_GROUP
## [1] "25-44"   "65+"     "18-24"   "<18"     "45-64"   "UNKNOWN"
##
```

```
## $VIC_SEX
## [1] "F"       "M"       "UNKNOWN"
##
## $VIC_RACE
## [1] "BLACK HISPANIC"                 "WHITE"
## [3] "BLACK"                          "WHITE HISPANIC"
## [5] "ASIAN / PACIFIC ISLANDER"       "AMERICAN INDIAN/ALASKAN NATIVE"
## [7] "UNKNOWN"
```

Format Analysis Variables

```
### Create 'OCCUR_DAY' variable => convert 'OCCUR_DATE' variable to day of the week
↪  factor variable
### Create 'OCCUR_YEAR' variable => extract year from 'OCCUR_DATE' variable
data=data %>% mutate(OCCUR_DAY=factor(weekdays(as.Date(OCCUR_DATE, format='%m/%d/%Y'),
↪  abbreviate=T),
                                      levels=c('Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat',
                                      ↪  'Sun')),
                     .after='OCCUR_TIME') %>%
  mutate(OCCUR_YEAR=as.integer(format(as.Date(OCCUR_DATE, format='%m/%d/%Y'), '%Y')),
  ↪  .after='INCIDENT_KEY') %>%
  select(-OCCUR_DATE)

### Convert 'OCCUR_TIME' to 24-hour format => extract HOUR of day as factor variable
data=data %>% mutate(OCCUR_TIME=as.factor(format(strptime(OCCUR_TIME,'%H:%M:%S'),'%H')))

### Create 'YR_TOTAL_INCIDENTS' variable => total number of incidents per year
data=data %>% select(INCIDENT_KEY, OCCUR_YEAR) %>% distinct(INCIDENT_KEY, .keep_all=T)
↪  %>%
  group_by(OCCUR_YEAR) %>% add_count(OCCUR_YEAR, name='YR_TOTAL_INCIDENTS') %>% ungroup()
  ↪  %>%
  left_join(data)

### Convert variables to factors
data=data %>% mutate(across(c(BORO:VIC_RACE), ~as.factor(.)))

str(data, vec.len=3, nchar.max=20) # preview dataset
```

```
## tibble [25,593 x 13] (S3: tbl_df/tbl/data.frame)
##  $ INCIDENT_KEY      : int [1:25593] 24050| __truncated__ ...
##  $ OCCUR_YEAR        : int [1:25593] 2006 | __truncated__ ...
##  $ YR_TOTAL_INCIDENTS: int [1:25593] 1566 | __truncated__ ...
##  $ OCCUR_TIME        : Factor w/ 24 levels "00","01","02",..: 6 6 6 6 13 22 20 1 ...
##  $ OCCUR_DAY         : Factor w/ 7 levels "Mon","Tue","Wed",..: 7 7 7 7 5 3 5 1 ...
##  $ BORO              : Factor w/ 5 levels "BRONX","BROOKLYN",..: 1 1 1 1 4 1 1 3 ...
##  $ LOCATION_DESC     : Factor w/ 39 levels "ATM","BANK","BAR/NIGHT CLUB",..: 37 37| __truncated__ ..
##  $ PERP_AGE_GROUP    : Factor w/ 6 levels "<18","18-24",..: 6 6 6 6 6 6 6 6 ...
##  $ PERP_SEX          : Factor w/ 3 levels "F","M","UNKNOWN": 3 3 3 3 3 3 3 3 ...
##  $ PERP_RACE         : Factor w/ 7 levels "AME"| __truncated__,..: 5 5 5 5 5 5 5 5 ...
##  $ VIC_AGE_GROUP     : Factor w/ 6 levels "<18","18-24",..: 3 3 3 3 5 2 3 3 ...
##  $ VIC_SEX           : Factor w/ 3 levels "F","M","UNKNOWN": 1 2 1 2 2 2 2 2 ...
##  $ VIC_RACE          : Factor w/ 7 levels "AME"| __truncated__,..: 4 4 4 4 6 3 3 4 ...
```

```
summary(select(data, BORO:VIC_RACE)) # view summary of dataset variables
```

```
##             BORO                          LOCATION_DESC   PERP_AGE_GROUP
##  BRONX        : 7400   UNKNOWN                   :15152   <18    : 1463
##  BROOKLYN     :10364   MULTI DWELL - PUBLIC HOUS: 4558   18-24  : 5844
##  MANHATTAN    : 3265   MULTI DWELL - APT BUILD  : 2664   25-44  : 5202
##  QUEENS       : 3828   PVT HOUSE                :  893   45-64  :  535
##  STATEN ISLAND:  736   GROCERY/BODEGA           :  622   65+    :   57
##                        BAR/NIGHT CLUB           :  587   UNKNOWN:12492
##                        (Other)                  : 1117
##     PERP_SEX                                PERP_RACE     VIC_AGE_GROUP
##  F      :  371   AMERICAN INDIAN/ALASKAN NATIVE:    2   <18    : 2681
##  M      :14413   ASIAN / PACIFIC ISLANDER      :  141   18-24  : 9603
##  UNKNOWN:10809   BLACK                         :10667   25-44  :11384
##                  BLACK HISPANIC                : 1203   45-64  : 1698
##                  UNKNOWN                       :11146   65+    :  167
##                  WHITE                         :  272   UNKNOWN:   60
##                  WHITE HISPANIC                : 2162
##     VIC_SEX                              VIC_RACE
##  F      : 2403   AMERICAN INDIAN/ALASKAN NATIVE:    9
##  M      :23179   ASIAN / PACIFIC ISLANDER      :  354
##  UNKNOWN:   11   BLACK                         :18280
##                  BLACK HISPANIC                : 2485
##                  UNKNOWN                       :   65
##                  WHITE                         :  660
##                  WHITE HISPANIC                : 3740
```

```
data %>% distinct(INCIDENT_KEY) %>% count() # total number of shooting incidents
```

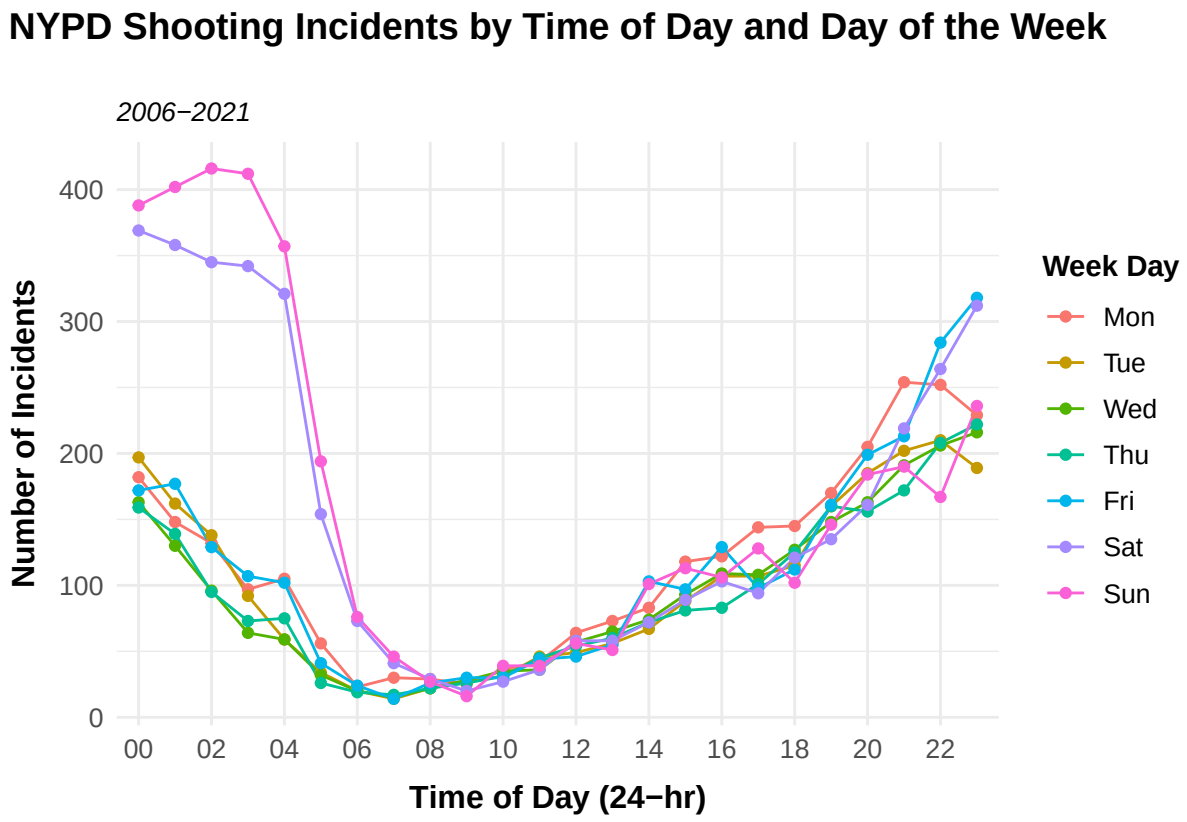
```
## # A tibble: 1 x 1
##       n
##   <int>
## 1 20124
```

Visualizations and Analysis

1. Day(s) and time(s) when the most shooting incidents occurred

```
#### PLOT: Number of Incidents x Time of Day, Grouped by Days of the Week
data %>% distinct(INCIDENT_KEY, .keep_all=T) %>% # remove duplicate incidents
  group_by(OCCUR_DAY) %>% count(OCCUR_TIME) %>% # count total number of incidents by time
  ↪  of day
  ggplot(aes(x=OCCUR_TIME, y=n, group=OCCUR_DAY)) + # plot
  geom_line(aes(color=OCCUR_DAY)) +
  geom_point(size=1.5, aes(color=OCCUR_DAY)) +
  scale_fill_hue(c=90) +
  scale_x_discrete(breaks=sort(unique(data$OCCUR_TIME))[c(TRUE, FALSE)]) +
  labs(title='NYPD Shooting Incidents by Time of Day and Day of the Week\n',
       x='Time of Day (24-hr)', y='Number of Incidents',
```

```
      color='Week Day',
      subtitle=paste(min(data$OCCUR_YEAR), max(data$OCCUR_YEAR), sep='-')) +
  theme_minimal() +
  theme(plot.title=element_text(hjust=0.5, face='bold', size=14),
        axis.title.x=element_text(vjust=-1, face='bold', size=12),
        axis.title.y=element_text(vjust=2.5, face='bold', size=12),
        axis.text.x=element_text(size=10), axis.text.y=element_text(size=10),
        legend.text=element_text(size=10), legend.title=element_text(size=11,
        face='bold'),
        plot.subtitle=element_text(hjust=0, face='italic', size=10),
        plot.margin=margin(0.5,0.5,0.5,0.5, 'cm'))
```



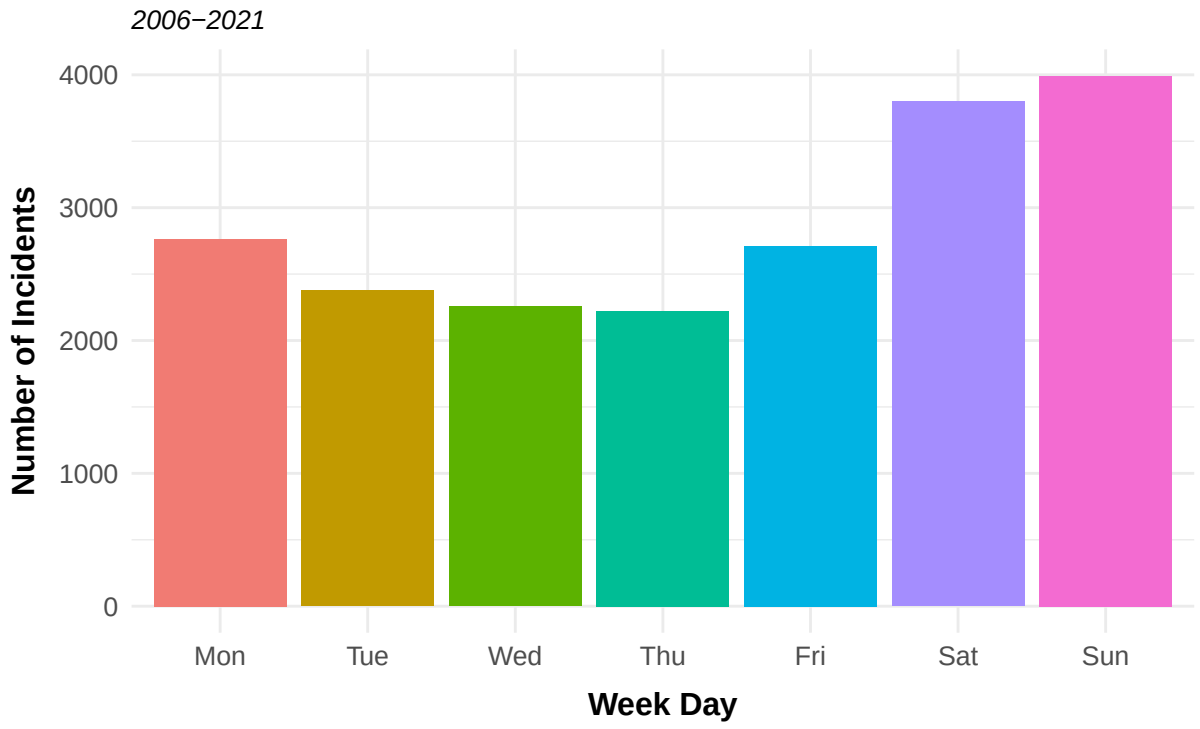

```
#### ANALYZE: Number of Incidents x Day of the Week
days=data %>% distinct(INCIDENT_KEY, .keep_all=T) %>% # remove duplicate incidents so day
  and time of occurrence for same incident isn't counted multiple times
  count(OCCUR_DAY) %>% # count total number of incidents by day
  mutate(PERCENT=round(n/sum(n), 3)*100, # relative percent of total number of incidents
    per day
         QUANT=ntile(n, 4)) %>% # split data into quantiles (0%, 25%, 50%, 75%, 100%)
  arrange(across(c(PERCENT, QUANT), desc)) # arrange by quantile and relative percent of
    total number of incidents per day (descending)

days # view dataset
```

```
## # A tibble: 7 x 4
```

```
##   OCCUR_DAY     n PERCENT QUANT
##   <fct>     <int>   <dbl> <int>
## 1 Sun        3992    19.8     4
## 2 Sat        3801    18.9     3
## 3 Mon        2764    13.7     3
## 4 Fri        2711    13.5     2
## 5 Tue        2378    11.8     2
## 6 Wed        2257    11.2     1
## 7 Thu        2221    11       1
```

```
#### PLOT: Number of Incidents x Day of the Week
days %>%
  ggplot(aes(x=OCCUR_DAY, y=n, fill=OCCUR_DAY)) + # plot
  geom_bar(stat='identity') +
  scale_fill_hue(c=90) +
  labs(title='NYPD Shooting Incidents by Day of the Week\n',
       x='Week Day', y='Number of Incidents',
       subtitle=paste(min(data$OCCUR_YEAR), max(data$OCCUR_YEAR), sep='-')) +
  theme_minimal() +
  theme(plot.title=element_text(hjust=0.5, face='bold', size=14),
        axis.title.x=element_text(vjust=-1, face='bold', size=12),
        ↪  axis.title.y=element_text(vjust=2.5, face='bold', size=12),
        axis.text.x=element_text(size=10), axis.text.y=element_text(size=10),
        legend.position='none',
        plot.subtitle=element_text(hjust=0, face='italic', size=10),
        plot.margin=margin(0.5,0.5,0.5,0.5, 'cm'))
```




```
#### ANALYZE: Number of Incidents x Time of Day
times=data %>% distinct(INCIDENT_KEY, .keep_all=T) %>% # remove duplicate incidents so
↪  day and time of occurrence for same incident isn't counted multiple times
  count(OCCUR_TIME) %>% # count total number of incidents by time of day
  mutate(PERCENT=round(n/sum(n), 3)*100, # relative percent of total number of incidents
  ↪  per day
         QUANT=ntile(n, 4)) %>% # split data into quantiles (0%, 25%, 50%, 75%, 100%)
  arrange(across(c(PERCENT, QUANT), desc)) # arrange by quantile and relative percent of
  ↪  total number of incidents per day (descending)

times # view dataset
```

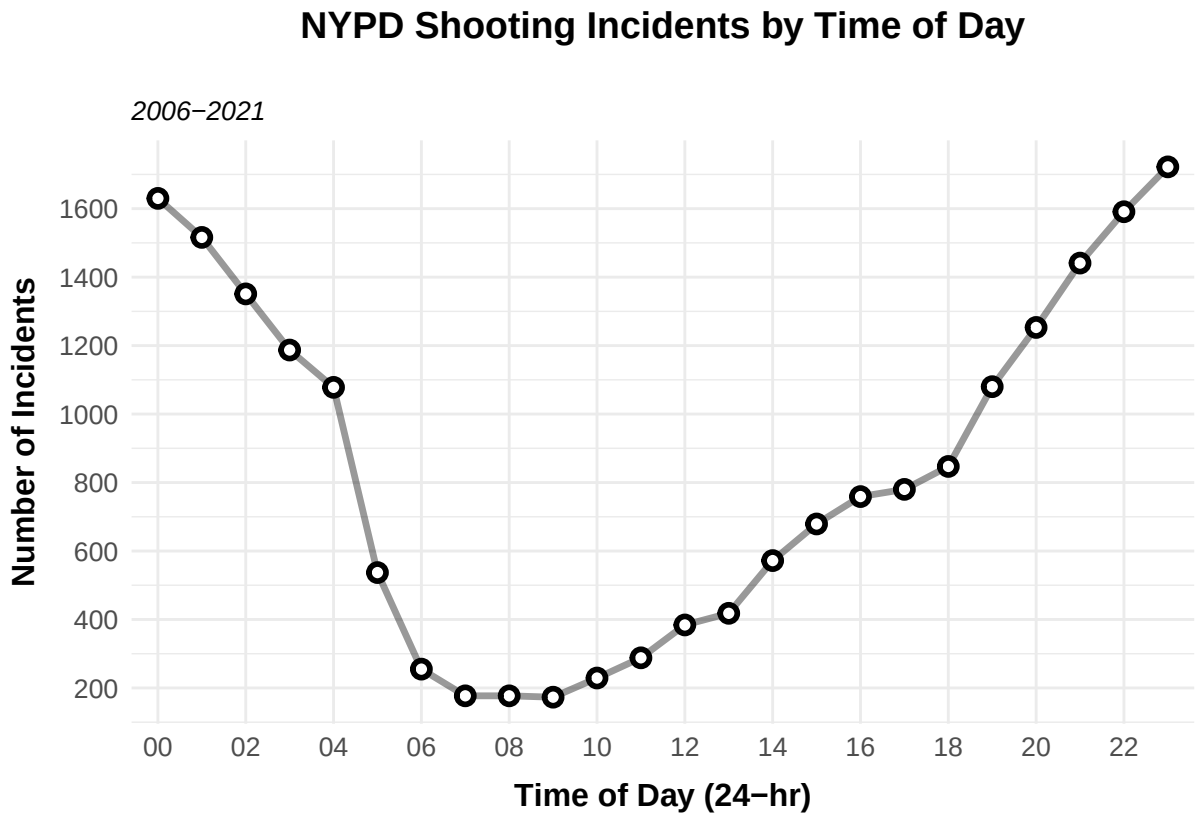
```
## # A tibble: 24 x 4
##    OCCUR_TIME     n PERCENT QUANT
##    <fct>      <int>   <dbl> <int>
##  1 23          1722     8.6     4
##  2 00          1630     8.1     4
##  3 22          1591     7.9     4
##  4 01          1516     7.5     4
##  5 21          1441     7.2     4
##  6 02          1351     6.7     4
##  7 20          1253     6.2     3
##  8 03          1187     5.9     3
##  9 04          1078     5.4     3
## 10 19          1080     5.4     3
```

```
## # ... with 14 more rows
## # i Use `print(n = ...)` to see more rows
```

```
#### ANALYZE: Percent of Crimes by Time Range
times %>%
  mutate(OCCUR_TIME_12=str_replace(format(strptime(OCCUR_TIME,'%H'),'%I'),' ', ''), #
  ↪  12-hour time variable
         OCCUR_TIME_AP=str_sub(str_replace(format(strptime(OCCUR_TIME,'%H'),'%I %p'),' ',
         ↪  ''), -2)) %>% # AM/PM time variable
  group_by(QUANT) %>% arrange(desc(OCCUR_TIME_AP), OCCUR_TIME) %>% # order by times of
  ↪  day
  mutate(OCCUR_TIME_RANGE=paste0(OCCUR_TIME_12, OCCUR_TIME_AP, collapse = ", ")) %>% #
  ↪  list all 12-hour AM/PM time variables in quantile in column
  group_by(OCCUR_TIME_RANGE) %>% summarise(TOTAL_PERCENT=sum(PERCENT)) %>% # sum relative
  ↪  percentages within groups to create total relative percentage of shootings that
  ↪  occur during that time range
  distinct(OCCUR_TIME_RANGE, TOTAL_PERCENT) %>% bind_rows() %>%
  ↪  arrange(desc(TOTAL_PERCENT)) # bind split data back into one dataset and arrange by
  ↪  relative total percent (descending)
```

```
## # A tibble: 4 x 2
##   OCCUR_TIME_RANGE                    TOTAL_PERCENT
##   <chr>                                       <dbl>
## 1 09PM, 10PM, 11PM, 12AM, 01AM, 02AM          46
## 2 05PM, 06PM, 07PM, 08PM, 03AM, 04AM          31
## 3 12PM, 01PM, 02PM, 03PM, 04PM, 05AM          16.7
## 4 06AM, 07AM, 08AM, 09AM, 10AM, 11AM           6.5
```

```
#### PLOT: Number of Incidents x Time of Day
times %>%
  ggplot(aes(x=OCCUR_TIME, y=n, group=1)) + # plot
  geom_line(size=1.2, alpha=0.4) +
  geom_point(fill='white', size=2, stroke=1.5, shape=21) +
  scale_x_discrete(breaks=sort(unique(times$OCCUR_TIME))[c(TRUE, FALSE)]) +
  scale_y_continuous(breaks=seq(0, 1800, by=200)) +
  labs(title='NYPD Shooting Incidents by Time of Day\n',
       x='Time of Day (24-hr)', y='Number of Incidents',
       subtitle=paste(min(data$OCCUR_YEAR), max(data$OCCUR_YEAR), sep='-')) +
  theme_minimal() +
  theme(plot.title=element_text(hjust=0.5, face='bold', size=14),
        axis.title.x=element_text(vjust=-1, face='bold', size=12),
        ↪  axis.title.y=element_text(vjust=2.5, face='bold', size=12),
        axis.text.x=element_text(size=10), axis.text.y=element_text(size=10),
        plot.subtitle=element_text(hjust=0, face='italic', size=10),
        plot.margin=margin(0.5,0.5,0.5,0.5, 'cm'))
```



2. Location(s) where the most shooting incidents occurred

```
#### ANALYZE: Number of Incidents x Borough and Location
locations=data %>% distinct(INCIDENT_KEY, .keep_all=T) %>% # remove duplicate incidents
↪  so location of occurrence for same incident isn't counted multiple times
  dplyr::filter(LOCATION_DESC!='UNKNOWN' & LOCATION_DESC!='NONE') %>% # filter out
↪  unknown and 'NONE' locations
  count(BORO, LOCATION_DESC) %>% # count total number of incidents by borough and
  ↪  location
  mutate(PER_RANK=round(percent_rank(n), 3)*100, # percent rank total number of incidents
  ↪  by borough and location
         PERCENT=round(n/sum(n), 3)*100) %>% #percentage of shootings that occur at that
          ↪  borough and location (relative to total shootings)
  arrange(desc(PER_RANK)) # arrange by percentile rank (descending)

locations %>% dplyr::filter(PER_RANK>=75) %>%
  select(-PER_RANK) # view locations with a percentile rank of 75% or greater (where
  ↪  incidents occurred most frequently)
```

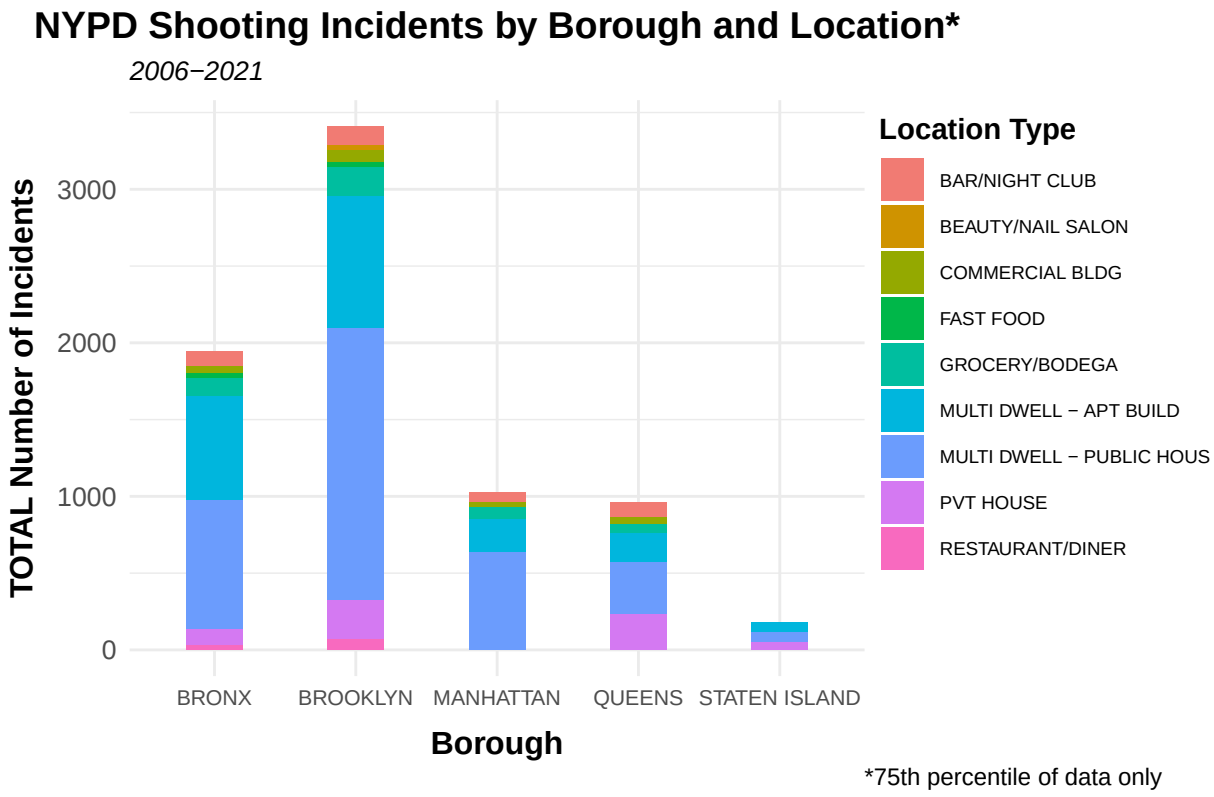
```
## # A tibble: 31 x 4
##    BORO      LOCATION_DESC                n PERCENT
##    <fct>     <fct>                    <int>   <dbl>
##  1 BROOKLYN  MULTI DWELL - PUBLIC HOUS  1770    22.1
```

```
##  2 BROOKLYN  MULTI DWELL - APT BUILD     859    10.7
##  3 BRONX     MULTI DWELL - PUBLIC HOUS   844    10.5
##  4 BRONX     MULTI DWELL - APT BUILD     681     8.5
##  5 MANHATTAN MULTI DWELL - PUBLIC HOUS   637     7.9
##  6 QUEENS    MULTI DWELL - PUBLIC HOUS   345     4.3
##  7 BROOKLYN  PVT HOUSE                   260     3.2
##  8 QUEENS    PVT HOUSE                   230     2.9
##  9 MANHATTAN MULTI DWELL - APT BUILD     217     2.7
## 10 BROOKLYN  GROCERY/BODEGA              193     2.4
## # ... with 21 more rows
## # i Use `print(n = ...)` to see more rows
```

```
### Number of incidents by borough only
data %>% distinct(INCIDENT_KEY, .keep_all=T) %>% # remove duplicate incidents so location
↪  of occurrence for same incident isn't counted multiple times
  dplyr::filter(LOCATION_DESC!='UNKNOWN' & LOCATION_DESC!='NONE') %>% # filter out
↪  unknown and 'NONE' locations
  count(BORO) %>% mutate(PERCENT=round(n/sum(n), 3)*100) %>% # frequency counts and
  ↪  percentages of shootings by borough
  arrange(desc(PERCENT)) # arrange by percentage (descending)
```

```
## # A tibble: 5 x 3
##   BORO              n PERCENT
##   <fct>         <int>   <dbl>
## 1 BROOKLYN       3563    44.4
## 2 BRONX          2040    25.4
## 3 MANHATTAN      1110    13.8
## 4 QUEENS         1076    13.4
## 5 STATEN ISLAND   231     2.9
```

```
#### PLOT: Number of Incidents x Borough and Location
locations %>% dplyr::filter(PER_RANK>=75) %>% # only keep locations with a percentile
↪  rank of 75% or greater
  ggplot(aes(x=BORO, y=n, fill=LOCATION_DESC)) + # plot
  geom_bar(position='stack', stat='identity', width=0.4) +
  scale_fill_hue(c=90) +
  labs(title='NYPD Shooting Incidents by Borough and Location*',
       x='Borough', y='TOTAL Number of Incidents',
       fill='Location Type',
       subtitle=paste(min(data$OCCUR_YEAR), max(data$OCCUR_YEAR), sep='-'),
       tag='*75th percentile of data only') +
  theme_minimal() +
  theme(plot.title=element_text(hjust=0.5, face='bold', size=14),
        axis.title.x=element_text(vjust=-1, face='bold', size=12),
        ↪  axis.title.y=element_text(vjust=2.5, face='bold', size=12),
        axis.text.x=element_text(size=8), axis.text.y=element_text(size=10),
        plot.subtitle=element_text(hjust=0, face='italic', size=10),
        legend.text=element_text(size=7), legend.title=element_text(size=11,
        ↪  face='bold'),
        plot.tag=element_text(size=9, vjust=-3), plot.tag.position='bottomright',
        legend.position=c(1,1), legend.justification=c(0, 1),
        plot.margin=margin(0.7,0.7,0.7,0.7, 'cm'))
```



3. Age, sex, and race of the perpetrators

```
#### ANALYZE: Age, Sex, and Race of Perpetrators
perps_dems=data %>% select(contains('perp')) %>% # select relevant variables
  filter_at(vars(contains('perp')), all_vars(.!='UNKNOWN')) # exclude rows where
  ↪  perpetrator information is unknown

### Demographics separately across age, sex, and race
perps_dems %>%
  lapply(tabyl) %>%
  map(., ~.x %>% mutate(percent=round(percent, 3)*100) %>% rename('GROUP'=1,
  ↪  'PERCENT'='percent') %>% # frequency table for each category (age, sex, race)
       arrange(desc(PERCENT))) # arrange by relative percent (descending)
```

```
## $PERP_AGE_GROUP
##    GROUP    n PERCENT
##    18-24 5771    44.6
##    25-44 5140    39.7
##      <18 1446    11.2
##    45-64  530     4.1
##      65+   56     0.4
##  UNKNOWN    0     0.0
##
```

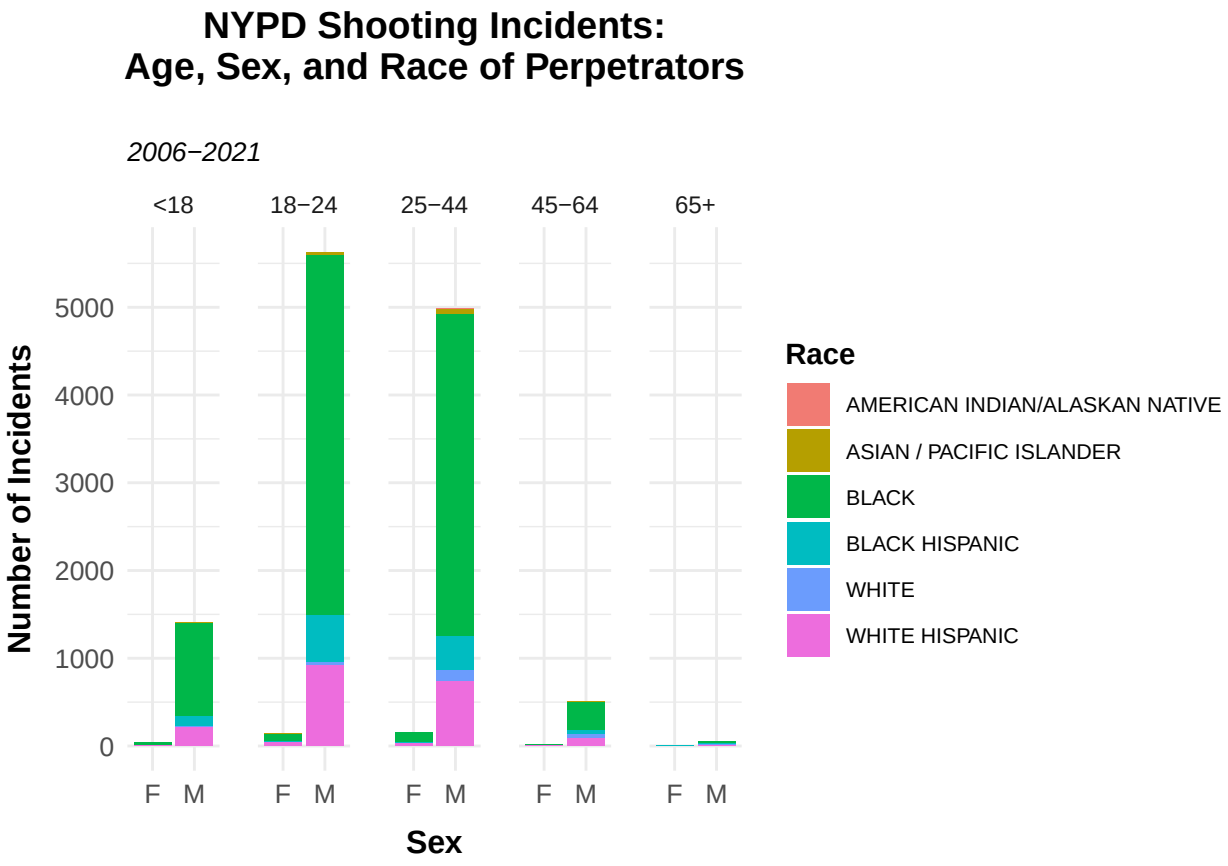
```
## $PERP_SEX
##    GROUP     n PERCENT
##        M 12591    97.3
##        F   352     2.7
##  UNKNOWN     0     0.0
##
## $PERP_RACE
##                           GROUP    n PERCENT
##                           BLACK 9407    72.7
##                  WHITE HISPANIC 2035    15.7
##                  BLACK HISPANIC 1108     8.6
##                           WHITE  261     2.0
##        ASIAN / PACIFIC ISLANDER  130     1.0
##  AMERICAN INDIAN/ALASKAN NATIVE    2     0.0
##                         UNKNOWN    0     0.0
```

```
### Demographics together across age, sex, and race
perps_dems %>%
  count(across(everything())) %>% # count all combinations of demographics variables
  mutate(PERCENT=round(n/sum(n), 3)*100) %>% distinct() %>% arrange(desc(PERCENT)) #
  ↪  remove duplicate rows and arrange by relative percent (descending)
```

```
## # A tibble: 44 x 5
##    PERP_AGE_GROUP PERP_SEX PERP_RACE          n PERCENT
##    <fct>          <fct>    <fct>          <int>   <dbl>
##  1 18-24          M        BLACK           4101    31.7
##  2 25-44          M        BLACK           3665    28.3
##  3 <18            M        BLACK           1053     8.1
##  4 18-24          M        WHITE HISPANIC   914     7.1
##  5 25-44          M        WHITE HISPANIC   740     5.7
##  6 18-24          M        BLACK HISPANIC   534     4.1
##  7 25-44          M        BLACK HISPANIC   387     3
##  8 45-64          M        BLACK            329     2.5
##  9 <18            M        WHITE HISPANIC   216     1.7
## 10 25-44          M        WHITE            126     1
## # ... with 34 more rows
## # i Use `print(n = ...)` to see more rows
```

```
#### PLOT: Age, Sex, and Race of Perpetrators
perps_dems %>% count(across(everything())) %>% # count total number of incidents across
↪  demographics
  ggplot(aes(x=PERP_SEX, y=n, fill=PERP_RACE)) + # plot
  geom_bar(position='stack', stat='identity') +
  scale_fill_hue(c=90) +
  scale_y_continuous(breaks=seq(0, 13000, by=1000)) +
  labs(title='NYPD Shooting Incidents:\nAge, Sex, and Race of Perpetrators\n',
       x='Sex', y='Number of Incidents',
       fill='Race',
       subtitle=paste(min(data$OCCUR_YEAR), max(data$OCCUR_YEAR), sep='-')) +
  facet_wrap(~PERP_AGE_GROUP, nrow=1) +
  theme_minimal() +
  theme(panel.spacing=unit(1, 'lines'),
        plot.title=element_text(hjust=0.5, face='bold', size=14),
```

```
        axis.title.x=element_text(vjust=-1, face='bold', size=12),
        ↪  axis.title.y=element_text(vjust=2.5, face='bold', size=12),
        axis.text.x=element_text(size=10), axis.text.y=element_text(size=10),
        plot.subtitle=element_text(hjust=0, face='italic', size=10),
        legend.text=element_text(size=8), legend.title=element_text(size=11,
        ↪  face='bold'))
```



4. Age, sex, and race of the victims

```
#### ANALYZE: Age, Sex, and Race of Victims
vics_dems=data %>% select(contains('vic')) %>% # select relevant variables
  filter_at(vars(contains('vic')), all_vars(.!='UNKNOWN')) # exclude rows where victim
  ↪  information is unknown

### Demographics separately across age, sex, and race
vics_dems %>% lapply(tabyl) %>% map(., ~.x %>% mutate(percent=round(percent, 3)*100) %>%
↪  rename('group'=1) %>% # frequency table for each category (age, sex, race)
                         arrange(desc(percent))) # arrange by relative percent
                          ↪  (descending)
```

```
## $VIC_AGE_GROUP
##    group     n percent
```

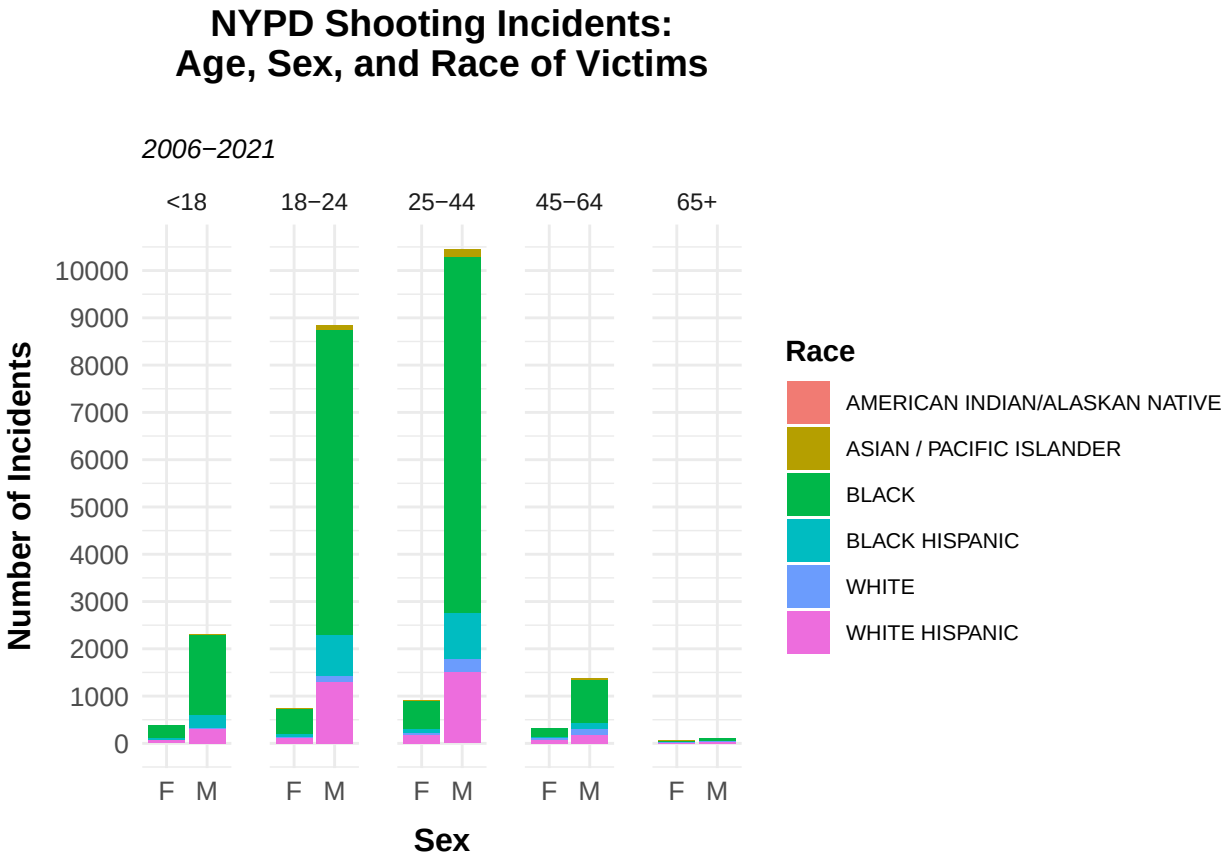
```
##    25-44 11363    44.6
##    18-24  9579    37.6
##      <18  2677    10.5
##    45-64  1693     6.6
##      65+   167     0.7
##  UNKNOWN     0     0.0
##
## $VIC_SEX
##    group     n percent
##        M 23082    90.6
##        F  2397     9.4
##  UNKNOWN     0     0.0
##
## $VIC_RACE
##                           group     n percent
##                           BLACK 18258    71.7
##                  WHITE HISPANIC  3732    14.6
##                  BLACK HISPANIC  2481     9.7
##                           WHITE   646     2.5
##        ASIAN / PACIFIC ISLANDER   353     1.4
##  AMERICAN INDIAN/ALASKAN NATIVE     9     0.0
##                         UNKNOWN     0     0.0
```

```
### Demographics together across age, sex, and race
vics_dems %>%
  count(across(everything())) %>% # count all combinations of demographics variables
  mutate(PERCENT=round(n/sum(n), 3)*100) %>% distinct() %>% arrange(desc(PERCENT)) #
  ↪  remove duplicate rows and arrange by relative percent (descending)
```

```
## # A tibble: 54 x 5
##    VIC_AGE_GROUP VIC_SEX VIC_RACE           n PERCENT
##    <fct>         <fct>   <fct>          <int>   <dbl>
##  1 25-44         M       BLACK           7527    29.5
##  2 18-24         M       BLACK           6456    25.3
##  3 <18           M       BLACK           1704     6.7
##  4 25-44         M       WHITE HISPANIC  1500     5.9
##  5 18-24         M       WHITE HISPANIC  1300     5.1
##  6 25-44         M       BLACK HISPANIC   987     3.9
##  7 45-64         M       BLACK            892     3.5
##  8 18-24         M       BLACK HISPANIC   868     3.4
##  9 25-44         F       BLACK            605     2.4
## 10 18-24         F       BLACK            528     2.1
## # ... with 44 more rows
## # i Use `print(n = ...)` to see more rows
```

```
#### PLOT: Age, Sex, and Race of Victims
vics_dems %>% count(across(everything())) %>% # count total number of incidents across
↪  demographics
  ggplot(aes(x=VIC_SEX, y=n, fill=VIC_RACE)) +
  geom_bar(position='stack', stat='identity') +
  scale_fill_hue(c=90) +
  scale_y_continuous(breaks=seq(0, 15000, by=1000)) +
  labs(title='NYPD Shooting Incidents:\nAge, Sex, and Race of Victims\n',
```

```
      x='Sex', y='Number of Incidents',
      fill='Race',
      subtitle=paste(min(data$OCCUR_YEAR), max(data$OCCUR_YEAR), sep='-')) +
facet_wrap(~VIC_AGE_GROUP, nrow=1) +
theme_minimal() +
theme(panel.spacing=unit(1, 'lines'),
      plot.title=element_text(hjust=0.5, face='bold', size=14),
      axis.title.x=element_text(vjust=-1, face='bold', size=12),
      ↪  axis.title.y=element_text(vjust=2.5, face='bold', size=12),
      axis.text.x=element_text(size=10), axis.text.y=element_text(size=10),
      plot.subtitle=element_text(hjust=0, face='italic', size=10),
      legend.text=element_text(size=8), legend.title=element_text(size=11,
      ↪  face='bold'))
```



Model Data

5. Best predictor(s) of shooting incidents

Calculate Relative Importance (RI) metrics for a multivariate linear model, by regressing the variables listed below onto the total number of shooting incidents by year, and calculating the R^2 contribution, averaged over orderings among regressor variables, to see which variables are the relatively most important, and best, indicators of shooting incidents overall.

REGRESSOR VARIABLES:

- 'OCCUR_TIME'
- 'OCCUR_DAY'
- 'BORO'
- 'LOCATION_DESC'
- 'PERP_AGE_GROUP'
- 'PERP_SEX'
- 'PERP_RACE'
- 'VIC_AGE_GROUP'
- 'VIC_SEX'
- 'VIC_RACE'

```
### Prep dataset for calculating Relative Importance (RI)
RI_data=data %>% select(-c(INCIDENT_KEY, OCCUR_YEAR)) # select relevant variables

all(sapply(RI_data[,-1], is.factor)) # make sure analysis variables are factor types
↪  (except for 'YR_TOTAL_INCIDENTS')
```

```
## [1] TRUE
```

```
RI_data # view dataset
```

```
## # A tibble: 25,593 x 11
##    YR_TO~1 OCCUR~2 OCCUR~3 BORO  LOCAT~4 PERP_~5 PERP_~6 PERP_~7 VIC_A~8 VIC_SEX
##      <int> <fct>   <fct>   <fct> <fct>   <fct>   <fct>   <fct>   <fct>   <fct>
##  1    1566 05      Sun     BRONX UNKNOWN UNKNOWN UNKNOWN UNKNOWN 25-44   F
##  2    1566 05      Sun     BRONX UNKNOWN UNKNOWN UNKNOWN UNKNOWN 25-44   M
##  3    1566 05      Sun     BRONX UNKNOWN UNKNOWN UNKNOWN UNKNOWN 25-44   F
##  4    1566 05      Sun     BRONX UNKNOWN UNKNOWN UNKNOWN UNKNOWN 25-44   M
##  5    1509 12      Fri     QUEE~ UNKNOWN UNKNOWN UNKNOWN UNKNOWN 65+     M
##  6    1562 21      Wed     BRONX COMMER~ UNKNOWN UNKNOWN UNKNOWN 18-24   M
##  7    1562 19      Fri     BRONX UNKNOWN UNKNOWN UNKNOWN UNKNOWN 25-44   M
##  8    1562 00      Mon     MANH~ UNKNOWN UNKNOWN UNKNOWN UNKNOWN 25-44   M
##  9    1562 06      Sun     BROO~ UNKNOWN 25-44   M       BLACK ~ 25-44   M
## 10    1562 06      Sun     BROO~ UNKNOWN 25-44   M       BLACK ~ 25-44   M
## # ... with 25,583 more rows, 1 more variable: VIC_RACE <fct>, and abbreviated
## #   variable names 1: YR_TOTAL_INCIDENTS, 2: OCCUR_TIME, 3: OCCUR_DAY,
## #   4: LOCATION_DESC, 5: PERP_AGE_GROUP, 6: PERP_SEX, 7: PERP_RACE,
## #   8: VIC_AGE_GROUP
## # i Use `print(n = ...)` to see more rows, and `colnames()` to see all variable names
```

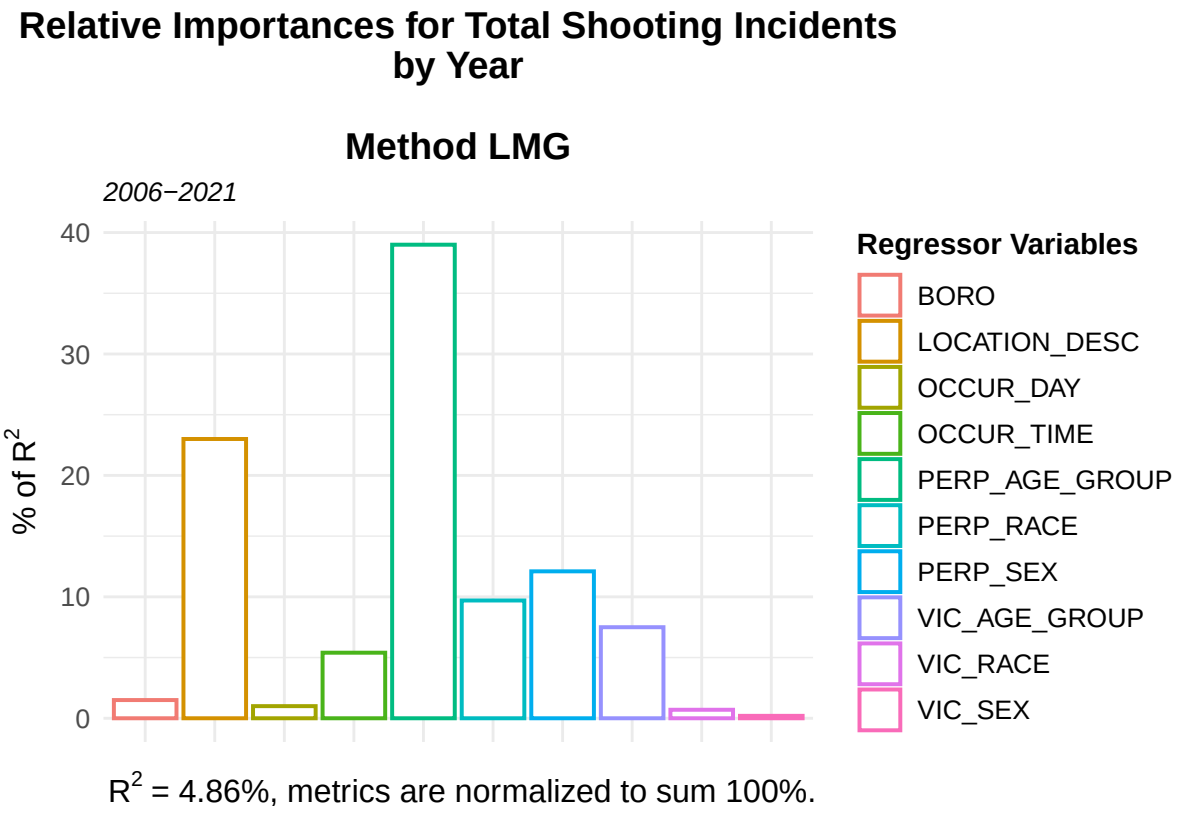
```
#### ANALYZE: Calculate Relative Importance (RI) using Fitted Regression Model
RI=data.frame(calc.relimp(RI_data, type='lmg', rela=T)$lmg)

### Rank relative contributions/importance of regressor variables as percentages, from
↪  most important to least important
RI=RI %>%
  rename('RI_percent'=1) %>% # 'RI_percent' => relative contributions, as percentages,
  ↪  obtained from the regression method
  rownames_to_column('REGRESS_VAR') %>% # name regressor variables column ('REGRESS_VAR')
  mutate(RI_percent=round(RI_percent, 3)*100) %>% arrange(desc(RI_percent)) # format
  ↪  'RI_percent' as percent and arrange by decreasing relative importance

RI # view dataset
```

```
##       REGRESS_VAR RI_percent
## 1  PERP_AGE_GROUP       39.0
## 2   LOCATION_DESC       23.0
## 3        PERP_SEX       12.1
## 4       PERP_RACE        9.7
## 5   VIC_AGE_GROUP        7.5
## 6      OCCUR_TIME        5.4
## 7            BORO        1.5
## 8       OCCUR_DAY        1.0
## 9        VIC_RACE        0.7
## 10        VIC_SEX        0.2
```

```
#### PLOT: Relative Importance (RI) Metrics
RI %>%
  ggplot(aes(x=REGRESS_VAR, y=RI_percent, color=REGRESS_VAR)) + # plot
  geom_bar(stat='identity', fill='white', size=0.7) +
  scale_color_hue(c=90) +
  labs(title='Relative Importances for Total Shooting Incidents\nby Year\n\nMethod LMG',
       subtitle=paste(min(data$OCCUR_YEAR), max(data$OCCUR_YEAR), sep='-'),
       colour='Regressor Variables',
       x=c(as.expression(bquote(~ R^2 ~ '= 4.86%, metrics are normalized to sum
       ↪  100%.'))),
       y=c(as.expression(bquote('% of' ~ R^2)))) +
  theme_minimal() +
  theme(plot.title=element_text(hjust=0.5, face='bold', size=14),
        axis.title.x=element_text(vjust=-1, face='bold', size=12),
        ↪  axis.title.y=element_text(vjust=2.5, face='bold', size=12),
        axis.text.x=element_blank(), axis.text.y=element_text(size=10),
        plot.subtitle=element_text(hjust=0, face='italic', size=10),
        legend.text=element_text(size=10), legend.title=element_text(size=11,
        ↪  face='bold'),
        plot.margin=margin(0.5,0.5,0.5,0.5, 'cm'))
```



Conclusion

Data analyses run on run on the NYPD shooting incident data, from 2006 to 2021, answered the following objectives:

1. Day(s) and time(s) when the most shooting incidents occurred

The greatest number of shootings occurred on Sundays (19.8%), followed by Saturdays (18.9%).

46% of shootings occurred from 9pm-2am, peaking with 8.6% of all shootings at 11pm. The second greatest number of shootings (31%) occurred from 5pm-8pm and 3am-4am, peaking at 8pm with 6.2% of all shootings happening at that hour.

2. Location(s) where the most shooting incidents occurred

The most shootings occurred in Brooklyn, which saw 44.4% of shootings, followed by the Bronx with 25.4% of shootings.

Out of all shooting incidents, 22.1% occurred specifically at Brooklyn public housing dwellings and 10.7% occurred at Brooklyn apartment buildings.

The Bronx saw similar rankings, with 10.5% of overall shootings occurring at Bronx public housing dwellings and 8.5% of overall shootings occurring at Bronx apartment buildings.

3. Age, sex, and race of the perpetrators

In 44.6% of shooting incidents, perpetrators were between the ages of 18-24. Males were the perpetrators in 97.3% of shooting incidents, and 72.7% of perpetrators were Black.

In 31.7% of shooting incidents, perpetrators were Black men aged 18-24, followed by Black men 25-44 years of age (28.3% of incidents).

4. Age, sex, and race of the victims

In 44.6% of shooting incidents, victims were between the ages of 25-44. Males were the victims in 90.6% of shooting incidents, and 71.7% of victims were Black.

In 29.5% of shooting incidents, victims were Black men aged 25-44, followed by Black men 18-24 years of age (25.3% of incidents).

5. Best predictor(s) of shooting incidents

The best predictors of shooting incidents (based on their relative importance (RI) predicting shootings), in decreasing order, were perpetrator age, location, perpetrator sex, perpetrator race, victim age, time of day, borough, day of the week, victim race, and victim sex (RI=39%, 23%, 12.1%, 9.7%, 7.5%, 5.4%, 1.5%, 1%, 0.7%, 0.2%).

Bias Identification

Possible external sources of bias for this data may be under-reported shootings. There may be shootings that are not reported to the New York Police Department, and, thus, excluded from the data. There may also be shootings that are reported but not thoroughly investigated by the NYPD, resulting in copious amounts of 'unknown' data from these reported shootings.

Additionally, shootings from lower socioeconomic areas may be disproportionately reported and appear in the data more, skewing the data towards those areas and the individuals who live there.

Also, the data does not take population sizes of the boroughs into account, which may skew the data to make boroughs with smaller population sizes more dangerous, because shooting incidents are not reported per capita, and population data is not included in the dataset.

Possible personal sources of bias for this analysis may be implicit bias and pre-conceived notions about how the data will look. For example, I believed that the Bronx would be the borough with the most shootings, Fridays and Saturdays would have the most shootings, and women would make up the majority of victims. However, upon analyzing the data, I was surprised to see that Brooklyn was the borough with the most shootings, Sundays and Mondays had the more shootings than Fridays, and men made up the overwhelming majority of shooting victims.

That's why it's important to put any biases aside when analyzing data so that you can make data-driven and data-backed conclusions, rather than basing them on personal beliefs or a priori conceptions and drawing incorrect, and possibly harmful, conclusions.