NYPD SHOOTINGS DATA ANALYSIS

2023-06-24

###IMPORT DATASETS

```
url_in <- "https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD"
url_pop <- "https://data.cityofnewyork.us/api/views/xywu-7bv9/rows.csv?accessType=DOWNLOAD"
```

###IMPORT LIBRARY

```
library(readr)
library(lubridate)
library(dplyr)
library(ggplot2)
library(stringr)
library(tidyverse)
nypd_shooting_main <- read_csv(url_in, show_col_types = FALSE)
nypd_shooting_main
```

```
## # A tibble: 27,312 x 21
##    INCIDENT_KEY OCCUR_DATE OCCUR_TIME BORO     LOC_OF_OCCUR_DESC PRECINCT
##           <dbl> <chr>      <time>     <chr>    <chr>                <dbl>
##  1    228798151 05/27/2021 21:30      QUEENS   <NA>                   105
##  2    137471050 06/27/2014 17:40      BRONX    <NA>                    40
##  3    147998800 11/21/2015 03:56      QUEENS   <NA>                   108
##  4    146837977 10/09/2015 18:30      BRONX    <NA>                    44
##  5     58921844 02/19/2009 22:58      BRONX    <NA>                    47
##  6    219559682 10/21/2020 21:36      BROOKLYN <NA>                    81
##  7     85295722 06/17/2012 22:47      QUEENS   <NA>                   114
##  8     71662474 03/08/2010 19:41      BROOKLYN <NA>                    81
##  9     83002139 02/05/2012 05:45      QUEENS   <NA>                   105
## 10     86437261 08/26/2012 01:10      QUEENS   <NA>                   101
## # i 27,302 more rows
## # i 15 more variables: JURISDICTION_CODE <dbl>, LOC_CLASSFCTN_DESC <chr>,
## #   LOCATION_DESC <chr>, STATISTICAL_MURDER_FLAG <lgl>, PERP_AGE_GROUP <chr>,
## #   PERP_SEX <chr>, PERP_RACE <chr>, VIC_AGE_GROUP <chr>, VIC_SEX <chr>,
## #   VIC_RACE <chr>, X_COORD_CD <dbl>, Y_COORD_CD <dbl>, Latitude <dbl>,
## #   Longitude <dbl>, Lon_Lat <chr>
```

```
ny_population <- read_csv(url_pop, show_col_types = FALSE)
ny_population
```

```
## # A tibble: 6 x 22
##   `Age Group`      Borough        `1950` `1950 - Boro share of NYC total` `1960`
##   <chr>            <chr>           <dbl>                            <dbl>  <dbl>
```

```
## 1 Total Population NYC Total     7891957                            100    7.78e6
## 2 Total Population Bronx         1451277                             18.4  1.42e6
## 3 Total Population Brooklyn      2738175                             34.7  2.63e6
## 4 Total Population Manhattan     1960101                             24.8  1.70e6
## 5 Total Population Queens        1550849                             19.6  1.81e6
## 6 Total Population Staten Island  191555                              2.43 2.22e5
## # i 17 more variables: `1960 - Boro share of NYC total` <dbl>, `1970` <dbl>,
## #   `1970 - Boro share of NYC total` <dbl>, `1980` <dbl>,
## #   `1980 - Boro share of NYC total` <dbl>, `1990` <dbl>,
## #   `1990 - Boro share of NYC total` <dbl>, `2000` <dbl>,
## #   `2000 - Boro share of NYC total` <dbl>, `2010` <dbl>,
## #   `2010 - Boro share of NYC total` <dbl>, `2020` <dbl>,
## #   `2020 - Boro share of NYC total` <dbl>, `2030` <dbl>, ...
```

###CLEAN DATA

```
nypd_shooting <- nypd_shooting_main %>%
  rename(DATE = OCCUR_DATE,
         TIME = OCCUR_TIME)
nypd_shooting <- nypd_shooting %>%
  mutate(across(-TIME, ~ifelse(is.na(.), "N/A", .)))
nypd_shooting$DATE <- as.Date(nypd_shooting$DATE, format = "%m/%d/%Y")
nypd_shooting <- subset(nypd_shooting, select = c("DATE","TIME", "BORO", "LOC_OF_OCCUR_DESC","LOC_CLASSI
```

```
ny_population_2020 <- subset(ny_population, select = c("Borough","2020"))
ny_population_2020$Borough = toupper(ny_population_2020$Borough)
ny_population_2020 <- ny_population_2020 %>%
  rename(BORO = Borough, POPULATION = "2020")

nypd_shooting <- nypd_shooting %>%
  left_join(ny_population_2020, by = c("BORO"))

nypd_shooting %>%
select (DATE, TIME, BORO, POPULATION)
```

```
## # A tibble: 27,312 x 4
##    DATE       TIME   BORO     POPULATION
##    <date>     <time> <chr>         <dbl>
##  1 2021-05-27 21:30  QUEENS      2330295
##  2 2014-06-27 17:40  BRONX       1446788
##  3 2015-11-21 03:56  QUEENS      2330295
##  4 2015-10-09 18:30  BRONX       1446788
##  5 2009-02-19 22:58  BRONX       1446788
##  6 2020-10-21 21:36  BROOKLYN    2648452
##  7 2012-06-17 22:47  QUEENS      2330295
##  8 2010-03-08 19:41  BROOKLYN    2648452
##  9 2012-02-05 05:45  QUEENS      2330295
## 10 2012-08-26 01:10  QUEENS      2330295
## # i 27,302 more rows
```

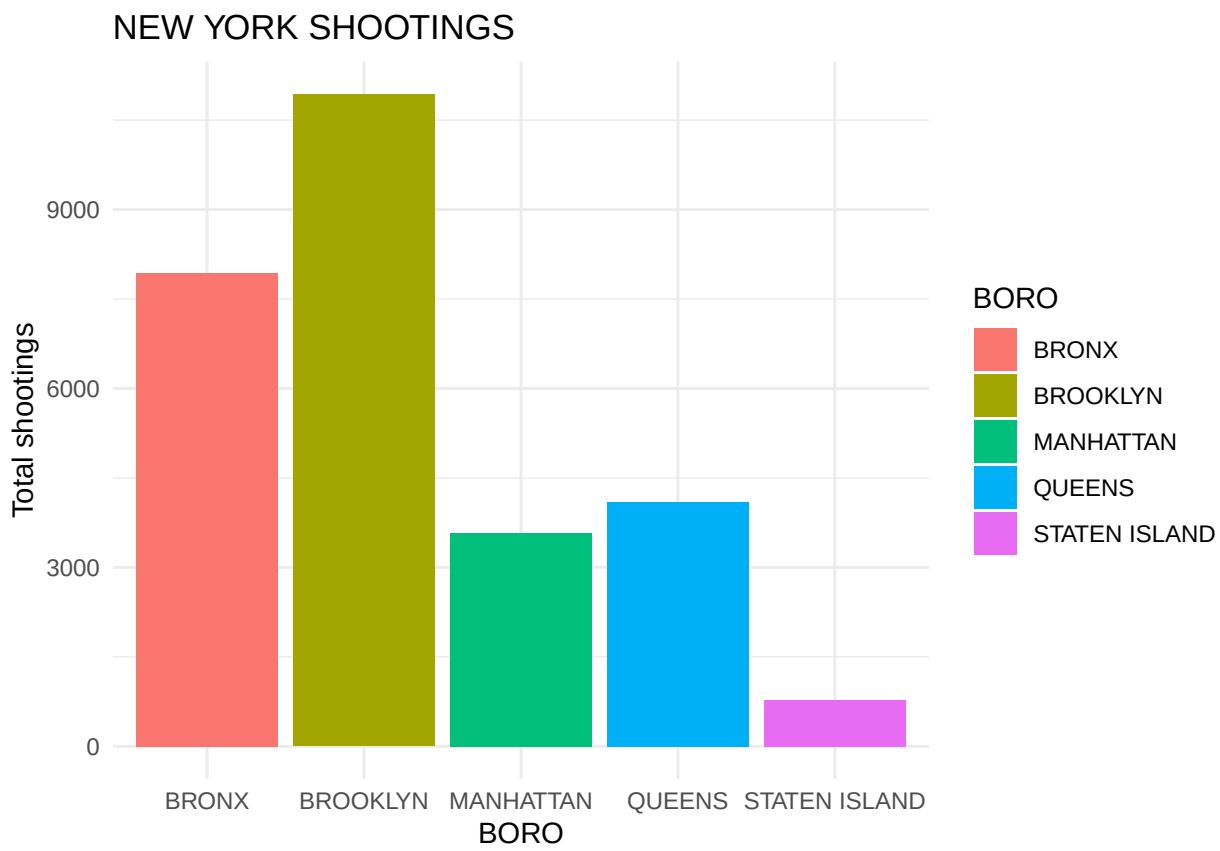
##ANALYZE DATA ##Calculate shootings per thousand

```
shootings_by_boro <- nypd_shooting %>%
  group_by(BORO) %>%
  summarise(Total = n()) %>%
  arrange(desc(Total))

shootings_by_boro <- shootings_by_boro %>% left_join(ny_population_2020, by = c("BORO"))
data_new <-  shootings_by_boro
shootings_by_boro$SHOOTINGS_PER_THOUSAND <- data_new$Total * 1000/ data_new$POPULATION

shootings_by_boro
```

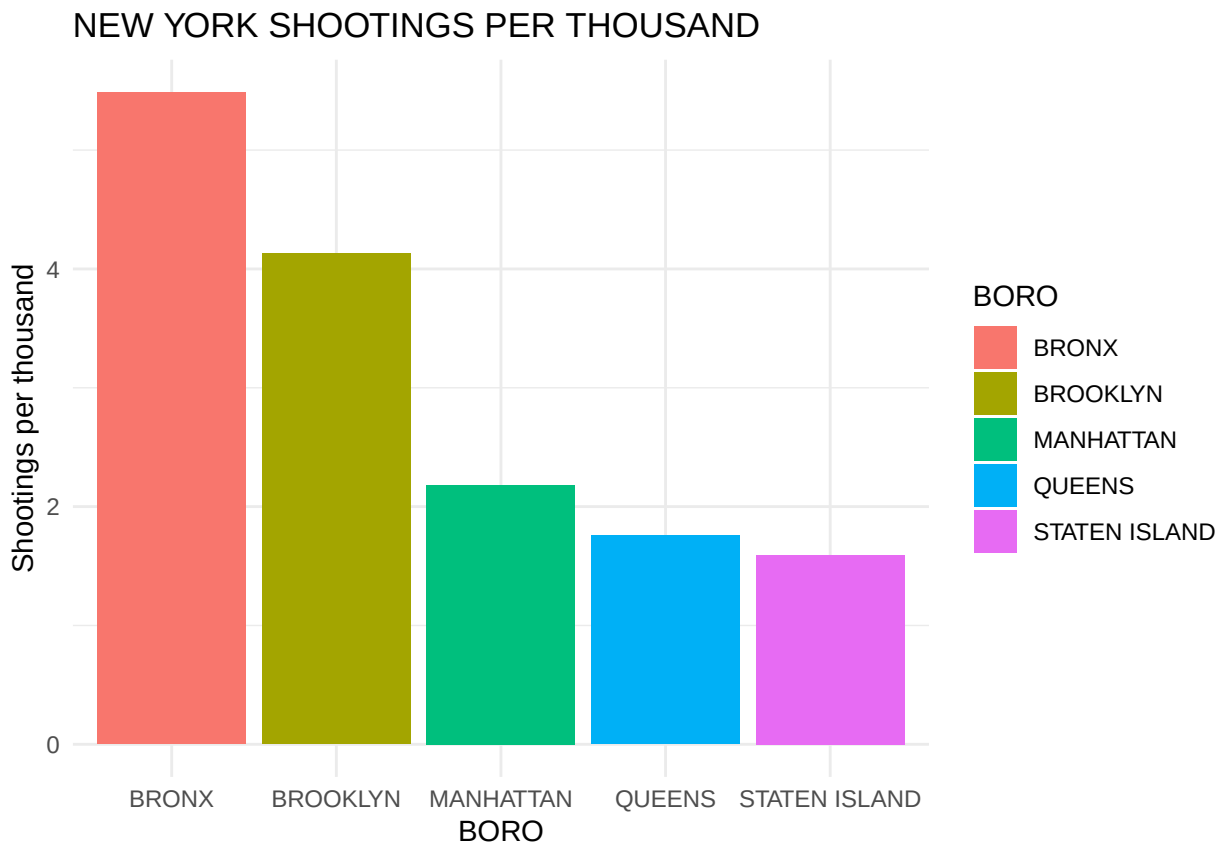
```
## # A tibble: 5 x 4
##   BORO          Total POPULATION SHOOTINGS_PER_THOUSAND
##   <chr>         <int>      <dbl>                  <dbl>
## 1 BROOKLYN      10933    2648452                   4.13
## 2 BRONX          7937    1446788                   5.49
## 3 QUEENS         4094    2330295                   1.76
## 4 MANHATTAN      3572    1638281                   2.18
## 5 STATEN ISLAND   776     487155                   1.59
```

##Visualization of shootings by BORO. Calcuate shootings per thousand population

```
ggplot(shootings_by_boro, aes(x=BORO, y=Total, fill=BORO)) +
  geom_bar(stat="identity") +
  xlab("BORO") + ylab("Total shootings") +
  ggtitle("NEW YORK SHOOTINGS") +
  theme_minimal()
```




```
ggplot(shootings_by_boro, aes(x=BORO, y=SHOOTINGS_PER_THOUSAND, fill=BORO)) +
  geom_bar(stat="identity") +
  xlab("BORO") + ylab("Shootings per thousand") +
  ggtitle("NEW YORK SHOOTINGS PER THOUSAND") +
  theme_minimal()
```



###BRONX boro has the highest shootings per thousand

##Visualization of shootings by YEAR

```
shootings_by_year <- nypd_shooting %>%
mutate(year = year(DATE)) %>%
group_by(year, BORO) %>%
  summarise(Total = n()) %>%
  arrange((year))

pivot_data <- shootings_by_year %>% pivot_wider(names_from = BORO, values_from = Total)
pivot_data
```

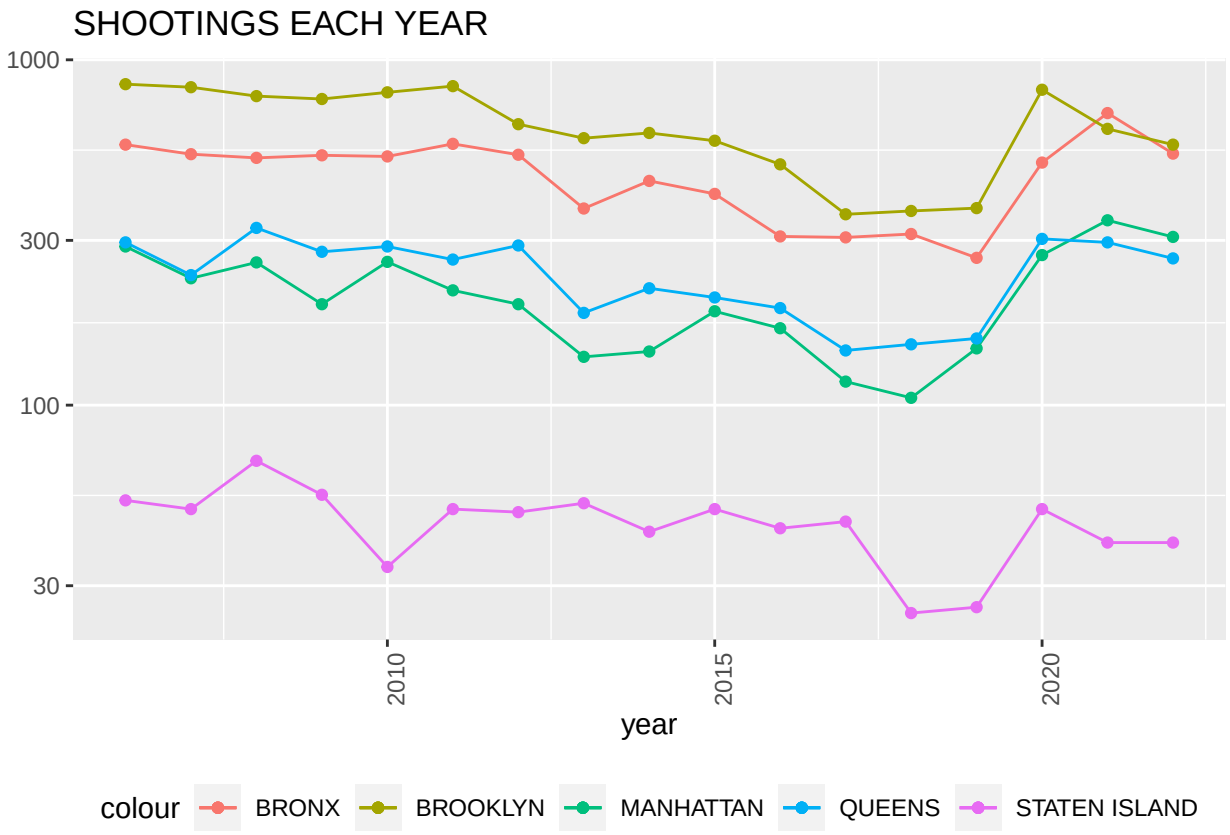
```
## # A tibble: 17 x 6
## # Groups:   year [17]
##     year BRONX BROOKLYN MANHATTAN QUEENS `STATEN ISLAND`
##    <dbl> <int>    <int>     <int>  <int>           <int>
##  1  2006   568      850       288    296              53
##  2  2007   533      833       233    238              50
##  3  2008   520      785       259    326              69
##  4  2009   529      770       196    278              55
##  5  2010   525      805       260    288              34
##  6  2011   571      839       215    264              50
##  7  2012   531      651       196    290              49
##  8  2013   371      593       138    185              52
##  9  2014   446      614       143    218              43
## 10  2015   409      583       187    205              50
```

```
## 11  2016   308      498       167    191             44
## 12  2017   306      357       117    144             46
## 13  2018   313      365       105    150             25
## 14  2019   267      372       146    156             26
## 15  2020   504      819       272    303             50
## 16  2021   701      631       343    296             40
## 17  2022   535      568       307    266             40
```

```
shootings_by_year
```

```
## # A tibble: 85 x 3
## # Groups:   year [17]
##     year BORO          Total
##    <dbl> <chr>         <int>
##  1  2006 BRONX           568
##  2  2006 BROOKLYN        850
##  3  2006 MANHATTAN       288
##  4  2006 QUEENS          296
##  5  2006 STATEN ISLAND    53
##  6  2007 BRONX           533
##  7  2007 BROOKLYN        833
##  8  2007 MANHATTAN       233
##  9  2007 QUEENS          238
## 10  2007 STATEN ISLAND    50
## # i 75 more rows
```

```
pivot_data %>%

  ggplot(aes(x = year, y = BRONX)) +
  geom_line(aes(color = "BRONX")) +
  geom_point(aes(color = "BRONX")) +
  geom_line(aes(y = MANHATTAN, color = "MANHATTAN")) +
  geom_point(aes(y = MANHATTAN, color = "MANHATTAN")) +
  geom_line(aes(y = BROOKLYN, color = "BROOKLYN")) +
  geom_point(aes(y = BROOKLYN, color = "BROOKLYN")) +
    geom_line(aes(y = QUEENS, color = "QUEENS")) +
  geom_point(aes(y = QUEENS, color = "QUEENS")) +
    geom_line(aes(y = `STATEN ISLAND`, color = "STATEN ISLAND")) +
  geom_point(aes(y = `STATEN ISLAND`, color = "STATEN ISLAND")) +
  scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = str_c("SHOOTINGS EACH YEAR"), y=NULL)
```



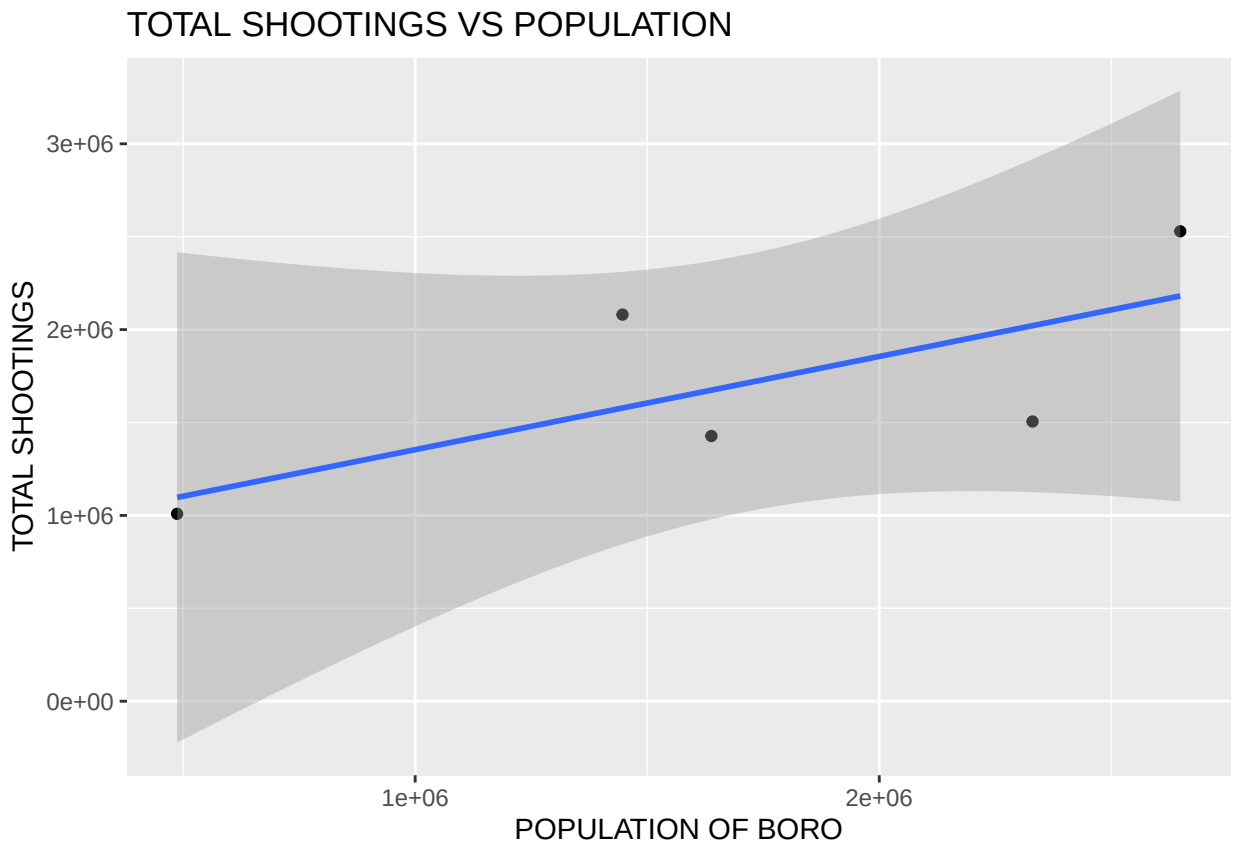
##LINEAR MODEL

```
mod <- lm(POPULATION ~ Total, data = shootings_by_boro)
summary(mod)
```

```
##
## Call:
## lm(formula = POPULATION ~ Total, data = shootings_by_boro)
##
## Residuals:
##       1       2       3       4       5
##  119295 -633860  824954  211084 -521473
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 892459.33  561844.04   1.588    0.210
## Total          149.70      86.15   1.738    0.181
##
## Residual standard error: 686300 on 3 degrees of freedom
## Multiple R-squared:  0.5016, Adjusted R-squared:  0.3355
## F-statistic: 3.019 on 1 and 3 DF,  p-value: 0.1807
```

```
shoot_pred <- shootings_by_boro %>% mutate(pred = predict(mod))
ggplot(shoot_pred, aes(x = POPULATION, y = pred)) +
  geom_point() +
```

```
  geom_smooth(method = "lm") +
  xlab("POPULATION OF BORO") +
  ylab("TOTAL SHOOTINGS") +
  ggtitle("TOTAL SHOOTINGS VS POPULATION")
```

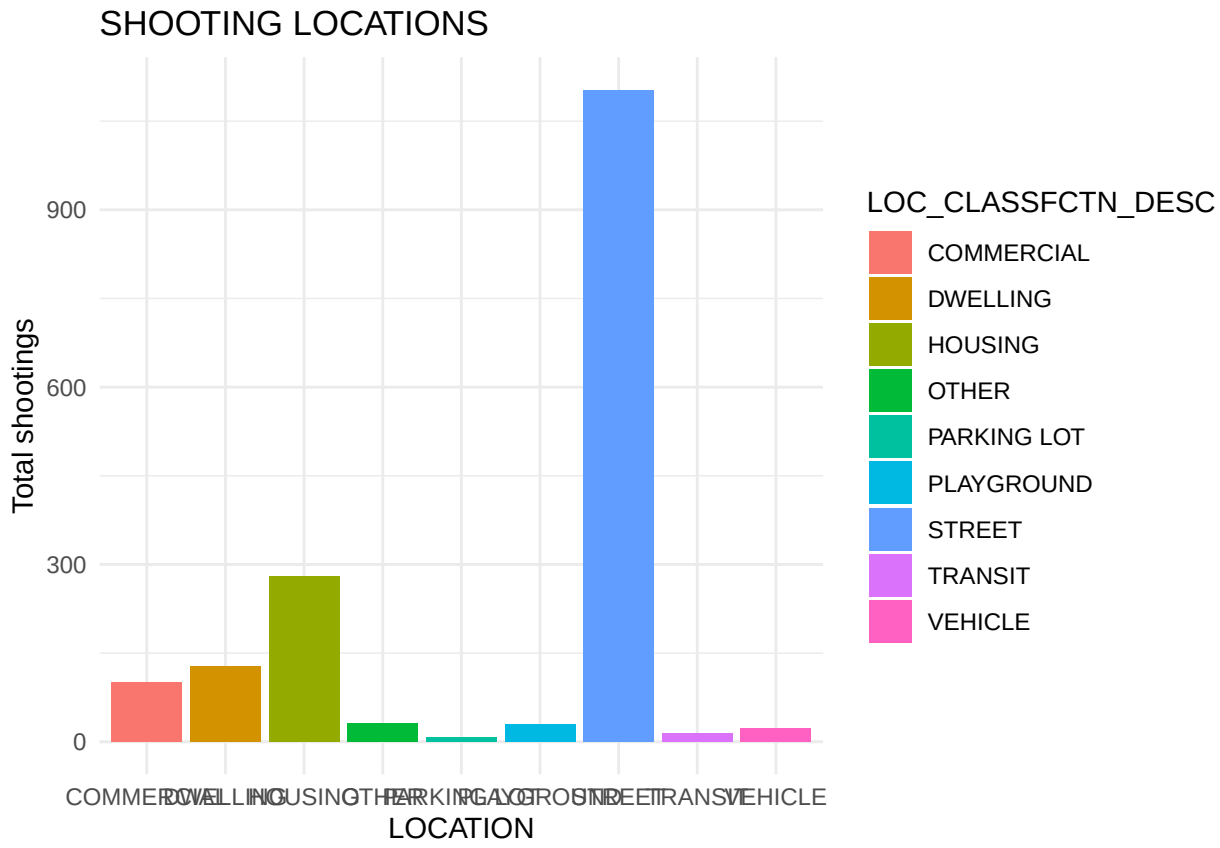


##BIAS IDENTIFICATION

```
shootings_by_loc <- nypd_shooting %>%
 filter(LOC_CLASSFCTN_DESC != 'N/A') %>%
group_by(LOC_CLASSFCTN_DESC) %>%
  summarise(Total = n())
shootings_by_loc
```

```
## # A tibble: 9 x 2
##   LOC_CLASSFCTN_DESC Total
##   <chr>              <int>
## 1 COMMERCIAL           100
## 2 DWELLING             127
## 3 HOUSING              280
## 4 OTHER                 31
## 5 PARKING LOT            7
## 6 PLAYGROUND            30
## 7 STREET              1103
## 8 TRANSIT               15
## 9 VEHICLE               23
```

```
ggplot(shootings_by_loc, aes(x=LOC_CLASSFCTN_DESC, y=Total, fill=LOC_CLASSFCTN_DESC)) +
  geom_bar(stat="identity") +
  xlab("LOCATION") + ylab("Total shootings") +
  ggtitle("SHOOTING LOCATIONS") +
  theme_minimal()
```



- Before conducting this data analysis, I had a personal bias that the most number of shootings occurs either on housing or commerical buildings either due to disagreement, fights, robbery/attack at home etc.
- This data analysis is a real eye opener for me and it clearly shows that highest number of shootings take place on STREETS. This data helps in clearing out my personal bias. This is a real value add and data speaks above any bias.

###session info

```
## R version 4.3.0 (2023-04-21)
## Platform: x86_64-apple-darwin20 (64-bit)
## Running under: macOS Monterey 12.6.7
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.3-x86_64/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.3-x86_64/Resources/lib/libRlapack.dylib;  LAPACK
##
## locale:
```

```
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## time zone: America/Chicago
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] forcats_1.0.0   purrr_1.0.1     tidyr_1.3.0     tibble_3.2.1
##  [5] tidyverse_2.0.0 stringr_1.5.0   ggplot2_3.4.2   dplyr_1.1.2
##  [9] lubridate_1.9.2 readr_2.1.4
##
## loaded via a namespace (and not attached):
##  [1] utf8_1.2.3       generics_0.1.3   lattice_0.21-8   stringi_1.7.12
##  [5] hms_1.1.3        digest_0.6.31    magrittr_2.0.3   evaluate_0.21
##  [9] grid_4.3.0       timechange_0.2.0 fastmap_1.1.1    Matrix_1.5-4
## [13] mgcv_1.8-42      fansi_1.0.4      scales_1.2.1     cli_3.6.1
## [17] rlang_1.1.1      crayon_1.5.2     splines_4.3.0    bit64_4.0.5
## [21] munsell_0.5.0    withr_2.5.0      yaml_2.3.7       tools_4.3.0
## [25] parallel_4.3.0   tzdb_0.4.0       colorspace_2.1-0 curl_5.0.1
## [29] vctrs_0.6.3      R6_2.5.1         lifecycle_1.0.3  bit_4.0.5
## [33] vroom_1.6.3      pkgconfig_2.0.3  pillar_1.9.0     gtable_0.3.3
## [37] glue_1.6.2       xfun_0.39        tidyselect_1.2.0 highr_0.10
## [41] rstudioapi_0.14  knitr_1.43       farver_2.1.1     nlme_3.1-162
## [45] htmltools_0.5.5  rmarkdown_2.22   labeling_0.4.2   compiler_4.3.0
```