DTSE 5301: NYPD Shooting Incident Project

ZK

2021-07-27

Introduction

In this project, we use the NYPD shootings dataset to do some exploratory data analysis. In the data, we see how gun incidents have changed over time in NY, and we use that as the starting point to ask questions for further and future analysis.

```
knitr::opts_chunk$set(echo = TRUE)
# let's load some libraries
library(tidyverse) # tidying and plotting
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate) # dates
```

```
##
## Attaching package: 'lubridate'
```

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

```
library(ggmap) # for mapping, see citation
```

```
## Warning: package 'ggmap' was built under R version 4.0.5
```

```
## Google's Terms of Service: https://cloud.google.com/maps-platform/terms/.
```

```
## Please cite ggmap if you use it! See citation("ggmap") for details.
```

Data

First, we want to get our data which is provided by the city of New York. This dataset contains NYPD shootings data from the last 15 years including information regarding locations, time of day, jursidiction, perp info, and victim info. Full variable descriptions and also be found at their website: https://data.cityofnewyork.us/Public-Safety/NYPD-Shooting-Incident-Data-Historic-/833y-fsy8

```
# on the LaTeX output some of these strings are too long to fit on the page... sorry!
raw_data <- read.csv("https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD",
                     na.strings = "")
```

Next, we want to clean our data. Looking at a subset of the data, one can see that some of the variables are the wrong format (i.e, dates). We will clean up the data by converting categorical data into factors and changing strings into dates where applicable. There are also a number of variables that or redundant or do not appear to be useful which will get dropped.

```
head(raw_data) # subset of data
```

```
##   INCIDENT_KEY OCCUR_DATE OCCUR_TIME          BORO PRECINCT JURISDICTION_CODE
## 1    201575314 08/23/2019   22:10:00        QUEENS      103                 0
## 2    205748546 11/27/2019   15:54:00         BRONX       40                 0
## 3    193118596 02/02/2019   19:40:00     MANHATTAN       23                 0
## 4    204192600 10/24/2019   00:52:00 STATEN ISLAND      121                 0
## 5    201483468 08/22/2019   18:03:00         BRONX       46                 0
## 6    198255460 06/07/2019   17:50:00      BROOKLYN       73                 0
##   LOCATION_DESC STATISTICAL_MURDER_FLAG PERP_AGE_GROUP PERP_SEX      PERP_RACE
## 1          <NA>                  false           <NA>     <NA>           <NA>
## 2          <NA>                  false            <18        M          BLACK
## 3          <NA>                  false          18-24        M WHITE HISPANIC
## 4     PVT HOUSE                   true          25-44        M          BLACK
## 5          <NA>                  false          25-44        M BLACK HISPANIC
## 6          <NA>                  false          45-64        M WHITE HISPANIC
##   VIC_AGE_GROUP VIC_SEX       VIC_RACE X_COORD_CD Y_COORD_CD Latitude Longitude
## 1         25-44       M          BLACK    1037451     193561 40.69781 -73.80814
## 2         25-44       F          BLACK    1006789     237559 40.81870 -73.91857
## 3         18-24       M BLACK HISPANIC     999347     227795 40.79192 -73.94548
## 4         25-44       F          BLACK     938149     171781 40.63806 -74.16611
## 5         18-24       M          BLACK    1008224     250621 40.85455 -73.91334
## 6         25-44       M          BLACK    1009650     186966 40.67983 -73.90843
##                                    Lon_Lat
## 1 POINT (-73.80814071699996 40.697805308000056)
## 2  POINT (-73.91857061799993 40.81869973000005)
## 3 POINT (-73.94547965999999 40.791916091000076)
## 4  POINT (-74.16610830199996 40.63806398200006)
## 5  POINT (-73.91333944399999 40.85454734900003)
## 6  POINT (-73.90842523899994 40.67982701600005)
```

```
# fixing data types
clean_data <- raw_data %>%
  # convert categoricla data to factors
  mutate(across(.cols = c("BORO", "PRECINCT", "JURISDICTION_CODE", "LOCATION_DESC",
                "PERP_AGE_GROUP", "PERP_SEX", "PERP_RACE", "VIC_AGE_GROUP", "VIC_SEX", "VIC_RACE"),
                as.factor)) %>%
```

```
  # clean up other data types
  mutate(OCCUR_DATE = lubridate::mdy(OCCUR_DATE),
         OCCUR_TIME = lubridate::hms(OCCUR_TIME),
         STATISTICAL_MURDER_FLAG = as.logical(STATISTICAL_MURDER_FLAG)) %>%
  # drop uncessary vars
  select(-c("INCIDENT_KEY", "Lon_Lat", "X_COORD_CD", "Y_COORD_CD")) # we will use latitude and longitud
```

There are 16725 rows containing missing data of some kind – more than half the data! It seems like much of the missing data is either perp information or building information. To deal with the missing data, we will completely drop the rows that are missing PERP information instead of trying impute values. For missing location data, we will retain the missing rows and fill them with the value: "UNKNOWN". Excluding the data with missing perp data may introduce bias into our data set.

```
# This probably introduces some bias.
clean_data <- clean_data  %>%
  drop_na(PERP_AGE_GROUP, PERP_SEX, PERP_RACE, JURISDICTION_CODE) %>%
  mutate(LOCATION_DESC = ifelse(is.na(LOCATION_DESC), "UNKNOWN", LOCATION_DESC))
```

Finally, let's take a look at the summary() output for our cleaned up data.

```
summary(clean_data)
```

```
##    OCCUR_DATE           OCCUR_TIME                                    BORO
##  Min.   :2006-01-01   Min.   :0S                          BRONX        :4497
##  1st Qu.:2008-04-02   1st Qu.:3H 39M 0S                   BROOKLYN     :5744
##  Median :2010-07-10   Median :15H 15M 0S                  MANHATTAN    :1993
##  Mean   :2011-09-26   Mean   :12H 47M 8.59071953398961S   QUEENS       :2307
##  3rd Qu.:2015-01-03   3rd Qu.:20H 35M 0S                  STATEN ISLAND: 566
##  Max.   :2020-12-29   Max.   :23H 59M 0S
##
##     PRECINCT     JURISDICTION_CODE LOCATION_DESC      STATISTICAL_MURDER_FLAG
##  75     :  856   0:12680           Length:15107       Mode :logical
##  73     :  750   1:   43           Class :character   FALSE:12231
##  47     :  589   2: 2384           Mode  :character   TRUE :2876
##  46     :  563
##  44     :  561
##  67     :  530
##  (Other):11258
##  PERP_AGE_GROUP PERP_SEX                              PERP_RACE    VIC_AGE_GROUP
##  18-24  :5448   F:  334   AMERICAN INDIAN/ALASKAN NATIVE:   2   <18    :1788
##  25-44  :4613   M:13303   ASIAN / PACIFIC ISLANDER      : 120   18-24  :5713
##  UNKNOWN:3155   U: 1470   BLACK                         :9854   25-44  :6399
##  <18    :1353             BLACK HISPANIC                :1081   45-64  :1033
##  45-64  : 481             UNKNOWN                       :1835   65+    : 117
##  65+    :  54             WHITE                         : 255   UNKNOWN:  57
##  (Other):   3             WHITE HISPANIC                :1960
##  VIC_SEX                              VIC_RACE        Latitude
##  F: 1576   AMERICAN INDIAN/ALASKAN NATIVE:    7   Min.   :40.52
##  M:13519   ASIAN / PACIFIC ISLANDER      :  235   1st Qu.:40.67
##  U:   12   BLACK                         :10324   Median :40.70
##            BLACK HISPANIC                : 1490   Mean   :40.74
##            UNKNOWN                       :   68   3rd Qu.:40.83
```

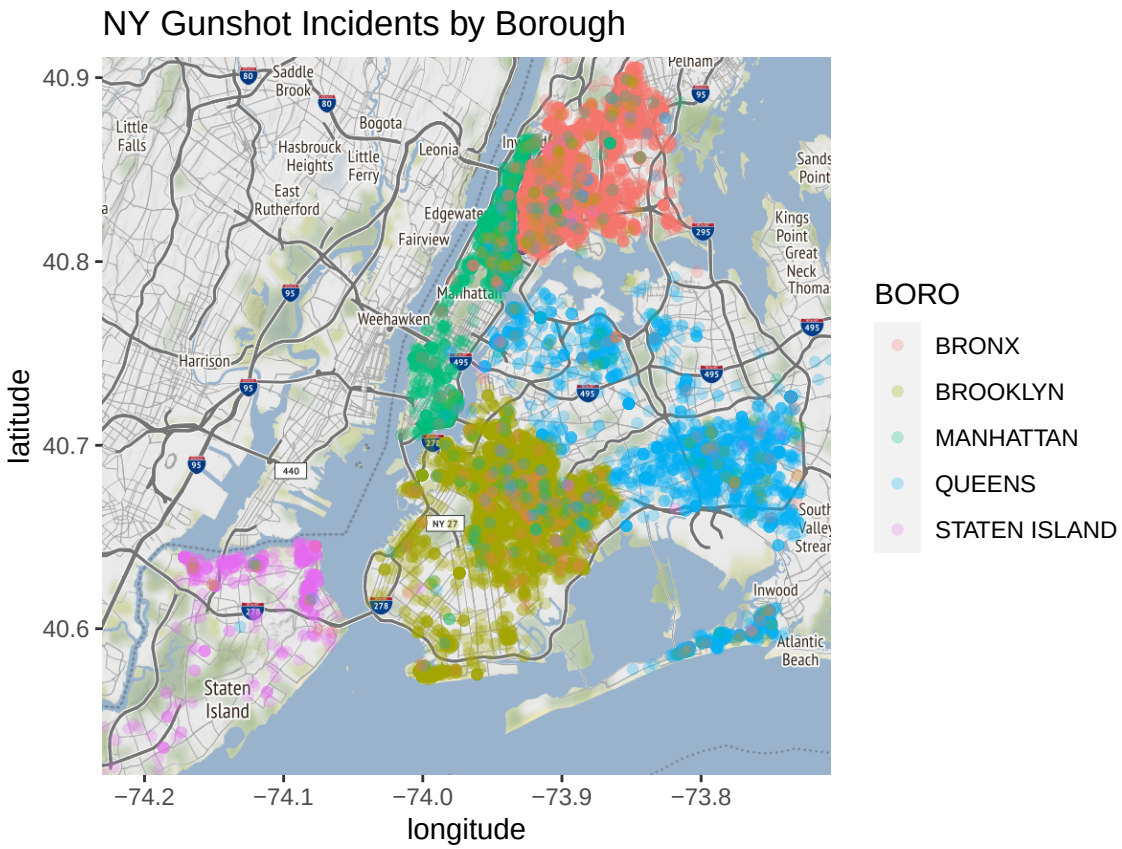
```
##            WHITE                          :  477   Max.   :40.91
##            WHITE HISPANIC                 : 2506
##    Longitude
##  Min.   :-74.23
##  1st Qu.:-73.94
##  Median :-73.91
##  Mean   :-73.91
##  3rd Qu.:-73.88
##  Max.   :-73.71
##
```

Visualizations/Analysis

Now that the data has been cleaned up, we can start to exploring it and develop questions about it. First, let's plot the data by latitude, longitude and borough as a gut check on the data to make sure that it makes sense. One can easily see that the data appears to align with a map of NY and the various boroughs.

```
# see ggmap cheat sheet:
# https://www.nceas.ucsb.edu/sites/default/files/2020-04/ggmapCheatsheet.pdf

# make bounding box
myLocation <- c(min(clean_data$Longitude),
                min(clean_data$Latitude),
                max(clean_data$Longitude),
                max(clean_data$Latitude))
# specify map type
myMap <- get_map(location=myLocation,
                 source="google",
                 maptype="roadmap")
# make plot
ggmap(myMap) +
  geom_point(aes(x=Longitude, y=Latitude, color = BORO),
             data = clean_data,
             alpha = 0.25) +
  labs(title = "NY Gunshot Incidents by Borough",
       x = "longitude",
       y = "latitude")
```



We will look at how volume of incidents have changed over time by looking at the monthly number of incidents and murders. In the chart below, it is clear that the overall number of gun incidents has decreased in New York over time. The negative sloping linear regression line in both cases indicates that the number of gun incidents and murders have been decreasing.

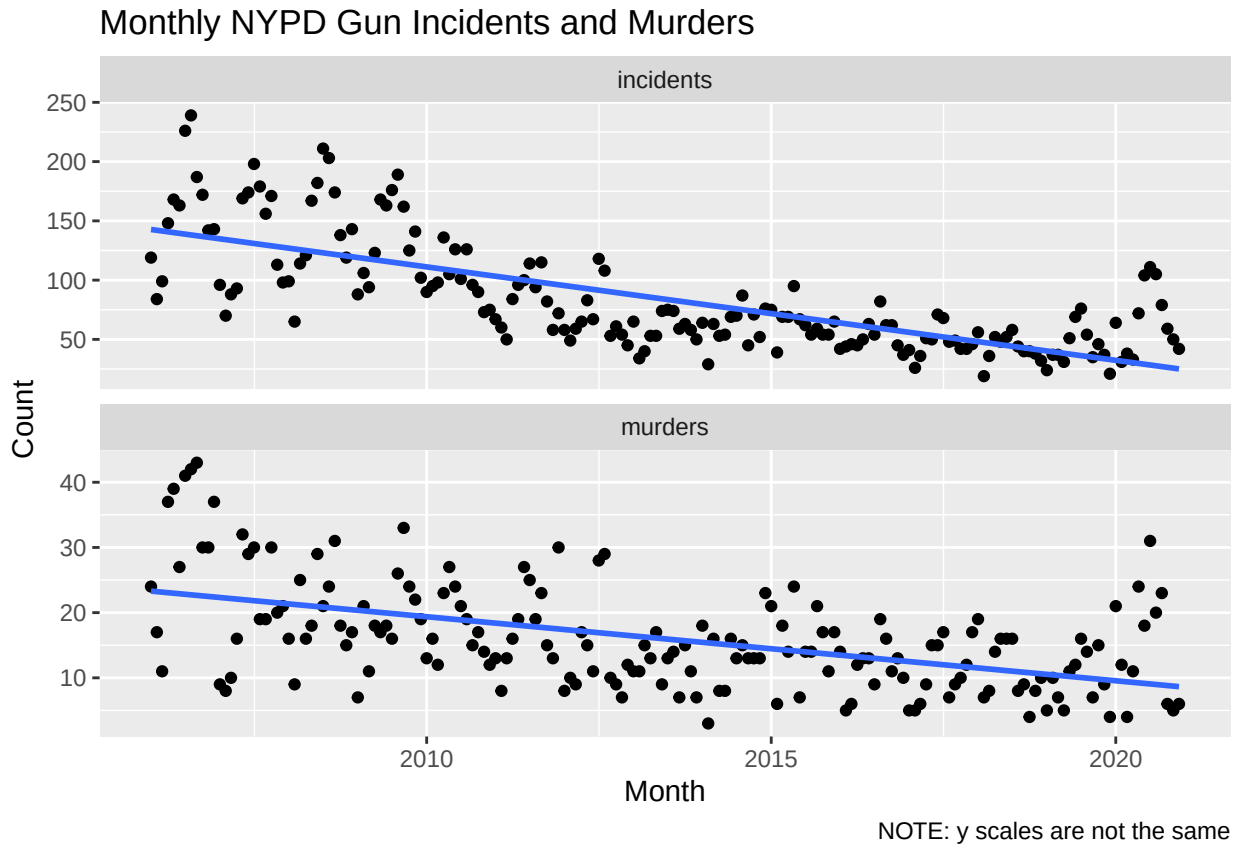
```
# first, let's add a variable to the data and create some summary data:
monthly_data <- clean_data %>%
  mutate(OCCUR_MONTH = floor_date(x = OCCUR_DATE, unit = "month")) %>%
  group_by(OCCUR_MONTH) %>%
  summarize(incidents = n(),
            murders = sum(STATISTICAL_MURDER_FLAG),
            ratio = murders / incidents)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
# plot
monthly_data %>%
  pivot_longer(cols = c("incidents", "murders"),
               names_to = "type",
               values_to = "count") %>%
  ggplot(aes(OCCUR_MONTH, count)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) + # let's include a linear model
  facet_wrap(vars(type), nrow = 2,
             scales = "free_y") +
  labs(title = "Monthly NYPD Gun Incidents and Murders",
```

```
       x = "Month", y = "Count",
       caption = "NOTE: y scales are not the same")
```

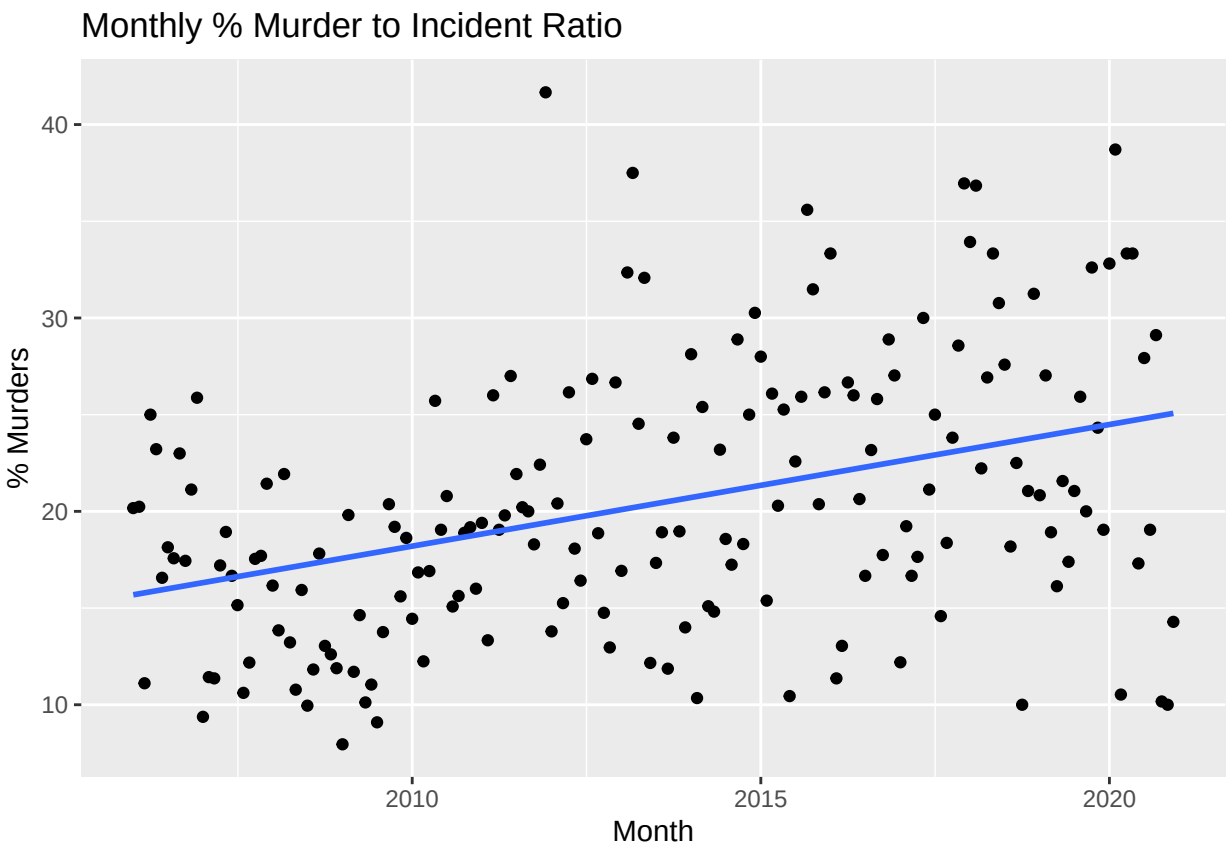
```
## `geom_smooth()` using formula 'y ~ x'
```



Digging a little bit deeper, we can try to determine whether or not incidents have been decreasing proportionally to each other by plotting the ratio of murders to incidents over time. If they have been decreasing together, we will expect to see a relatively flat sloping regression line. Unfortunately, that is not what the plot below shows. The positively sloping regression line suggests that gun murders are becoming relatively more frequent in comparison to gun incidents as reported to NYPD.

```
monthly_data %>%
  ggplot(aes(OCCUR_MONTH, ratio*100)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  labs(title = "Monthly % Murder to Incident Ratio",
       x = "Month",
       y = "% Murders")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

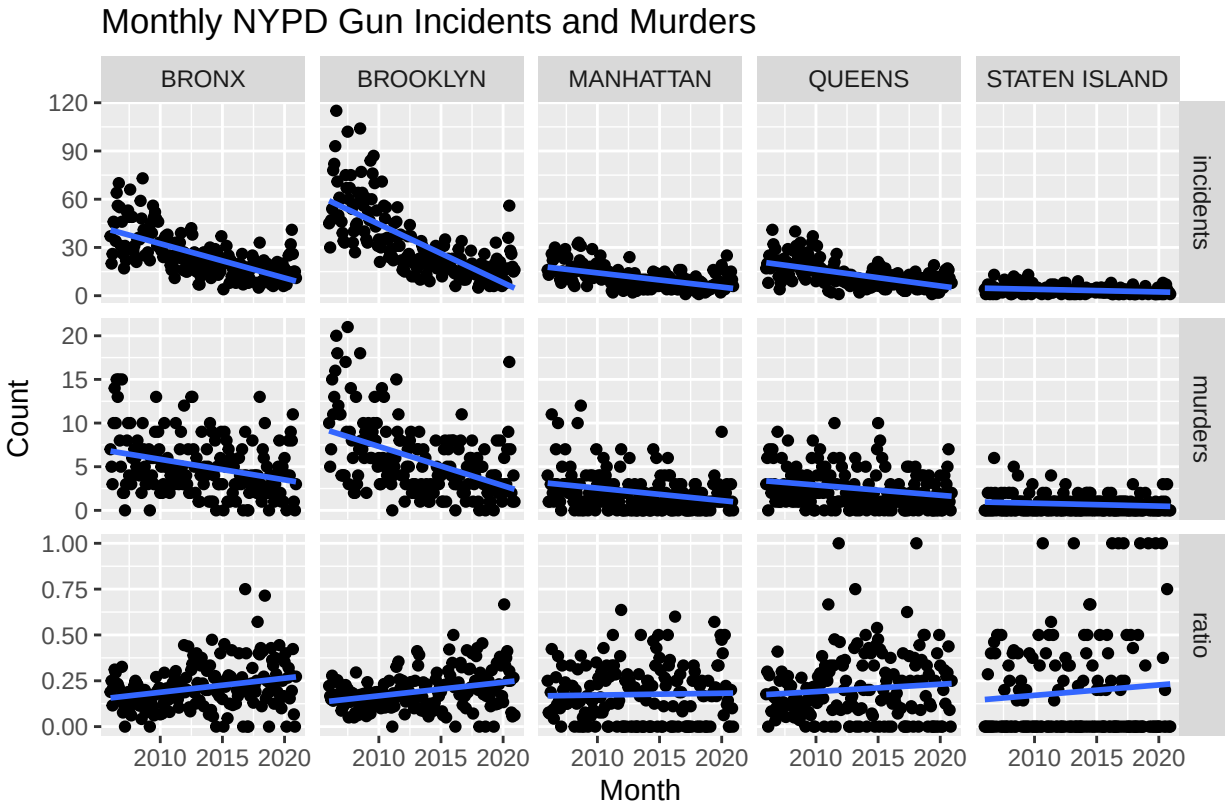


Digging once again, we can ask if that relationship holds across different subsets of the city. Below the data is stratified by borough and we can see the same relationship in every borough. Though it is apparent that certain boroughs such as Brooklyn are much more dramatic than others.

```
# sorry about the big blob of dplyr and ggplot
clean_data %>%
  mutate(OCCUR_MONTH = floor_date(x = OCCUR_DATE, unit = "month")) %>%
  group_by(OCCUR_MONTH, BORO) %>%
  summarize(incidents = n(),
            murders = sum(STATISTICAL_MURDER_FLAG),
            ratio = murders / incidents) %>%
  pivot_longer(cols = c("incidents", "murders", "ratio"),
               names_to = "type",
               values_to = "count") %>%
  ggplot(aes(OCCUR_MONTH, count)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) + # let's include a linear model
  facet_grid(rows = vars(type), cols = vars(BORO),
             scales = "free_y") +
  labs(title = "Monthly NYPD Gun Incidents and Murders",
       x = "Month", y = "Count",
       caption = "NOTE: y scales are not the same")
```

```
## `summarise()` regrouping output by 'OCCUR_MONTH' (override with `.groups` argument)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



This raises several other questions about our data set that might be worth exploring further to understand why lethal incidents are becoming relatively more common over time despite the overall reduciton of incidents. For example,

1. Would the effect go away if we added back in the rows of data that we removed earlier?
2. Do these relationships change if we stratify the data by perp or victim traits such as age?
3. Does the time of day or day of the week have any effect on what sort of incidents are lethal?
4. Is there an under reporting or understaffing issue in the NYPD that has caused them to ignore no-lethal gun incidents in more recent history?
5. How will this relationship change in the future? What limitations does our linear model have as we approach 0 incidents per month?

Bias and conclusion

There are a couple of sources of bias in this project.

1. **Data cleaning:** We likely introduced bias into our data, when we dropped the data that was missing PERP data. The dropped data may have had a different distribution from the rest of our data. However, in hindsight, the analysis that we performed did not actually uses the variables that we were missing data is, so we could have included all of the observations and mitigated this issue.
2. **Personal bias:** My personal bias manifests itself in the form of problem selection. I am sure that there are many interesting ways to look at and analyze this data, but I decided to focus on discrepancy between decreasing incident rates alongside rising lethal cases.

Exploring this dataset gives us some insight into the nature of shooting incidents in New York over the last 15 years. Our analysis raises questions regarding the relationship between lethal and non-lethal incidents and how the relationship varies over time and across different subsets of the data.

Citations

D. Kahle and H. Wickham. ggmap: Spatial Visualization with ggplot2. The R Journal, 5(1), 144-161. URL http://journal.r-project.org/archive/2013-1/kahle-wickham.pdf

```
sessionInfo()
```

```
## R version 4.0.3 (2020-10-10)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19041)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.1252
## [2] LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] ggmap_3.0.0       lubridate_1.7.9.2 forcats_0.5.0     stringr_1.4.0
##  [5] dplyr_1.0.2       purrr_0.3.4       readr_1.4.0       tidyr_1.1.2
##  [9] tibble_3.0.4      ggplot2_3.3.2     tidyverse_1.3.0
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.5          lattice_0.20-41     png_0.1-7
##  [4] assertthat_0.2.1    digest_0.6.27       R6_2.5.0
##  [7] cellranger_1.1.0    plyr_1.8.6          backports_1.2.0
## [10] reprex_0.3.0        evaluate_0.14       httr_1.4.2
## [13] pillar_1.4.7        RgoogleMaps_1.4.5.3 rlang_0.4.9
## [16] curl_4.3            readxl_1.3.1        rstudioapi_0.13
## [19] Matrix_1.2-18       rmarkdown_2.5       splines_4.0.3
## [22] labeling_0.4.2      munsell_0.5.0       broom_0.7.2
## [25] compiler_4.0.3      modelr_0.1.8        xfun_0.24
## [28] pkgconfig_2.0.3     mgcv_1.8-33         htmltools_0.5.1.1
## [31] tidyselect_1.1.0    fansi_0.4.1         crayon_1.3.4
## [34] dbplyr_2.0.0        withr_2.3.0         bitops_1.0-6
## [37] grid_4.0.3          nlme_3.1-149        jsonlite_1.7.1
## [40] gtable_0.3.0        lifecycle_0.2.0     DBI_1.1.0
## [43] magrittr_2.0.1      scales_1.1.1        cli_2.2.0
## [46] stringi_1.5.3       farver_2.0.3        fs_1.5.0
## [49] sp_1.4-5            xml2_1.3.2          ellipsis_0.3.1
## [52] generics_0.1.0      vctrs_0.3.5         rjson_0.2.20
## [55] tools_4.0.3         glue_1.4.2          hms_0.5.3
## [58] jpeg_0.1-8.1        yaml_2.2.1          colorspace_2.0-0
```

```
## [61] rvest_0.3.6         knitr_1.30          haven_2.3.1
```