NYPD Shooting Incident Analysis

R. Rhoads

03 Dec 2022

Purpose

This report is created to analyze the historical data from NYPD shootings going back to 2006 through the end of the previous calendar year (2021). The data is comprised of shooting incidents that include information about the event, location, and the given time. Additionally, information about the perpetrators and victims demographics is included in the data set. This data set is publicly available and for any additional information about the data visit www.data.cityofnewyork.us and review the data footnotes.

The setup of this report will include all details necessary for anyone to quickly and easily reproduce the results starting with important libraries used, the exact data set used and where it was downloaded from, how the data was altered for easy processing, visualizations and analysis, and finally a discussion on any bias that may have occurred while generating this reports.

Given the wealth and type of information in the data set, this report will focus on answering the following questions.

1. What is the overall distribution of the incidents by borough? Which borough has the highest and lowest incident count? Of the incidents, what percentage were murders?
2. When are shooting incidents the highest? Are there times during the day that are safer than others? Are there days of the week that are safer than others?
3. What factors are statistically significant in predicting a murder given the perpetrators profile, time of day, weekday, and location?

Libraries

The `tidyverse` and `lubridate` libraries are necessary to reproduce any work that has been done in this report. The `sessioninfo` library is not necessary for any analysis since the output is used to provide the information of the session for this report.

```
# Import necessary libraries
library(tidyverse)
library(lubridate)
# Library used to provide session information
library(sessioninfo)
```

Data Source and Structure

The data in its entirety can be found in at the following URL. Since the data is in a Comma-Separated Values (CSV) format, the `read.csv()` function is used to read in the data. Any empty character data fields will be turned into `NA` data types which will make data manipulation easier in the *Data Processing* section.

```r
# Set URL address
url_address <-
  "https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD"
# Load incident data from URL address and turn empty data fields into NA
inc_data <- read.csv(url_address, na.strings = (""), fill = FALSE)
# Turn data into a tibble data format
inc_data <- as_tibble(inc_data)
```

The table below shows the column names of the data, including the data type of each column and a brief description of them.

Column Name	Data Type	Description
INCIDENT_KEY	int	Randomly generated persistent ID for each arrest
OCCUR_DATE	chr	Exact date of the shooting incident
OCCUR_TIME	chr	Exact time of the shooting incident
BORO	chr	Borough where the shooting incident occurred
PRECINCT	int	Precinct where the shooting incident occurred
JURISDICTION_CODE	int	Jurisdiction where the shooting incident occurred
LOCATION_DESC	chr	Location of the shooting incident
STATISTICAL_MURDER_FLAG	chr	Shooting resulted in the victim's death which would be counted as a murder
PERP_AGE_GROUP	chr	Perpetrator's age within a category
PERP_SEX	chr	Perpetrator's sex description
PERP_RACE	chr	Perpetrator's race description
VIC_AGE_GROUP	chr	Victim's age within a category
VIC_SEX	chr	Victim's sex description
VIC_RACE	chr	Victim's race description
X_COORD_CD	dbl	Midblock X-coordinate for New York State Plane Coordinate System
Y_COORD_CD	dbl	Midblock Y-coordinate for New York State Plane Coordinate System
Latitude	dbl	Latitude coordinate for Global Coordinate System
Longitude	dbl	Longitude coordinate for Global Coordinate System
Lon_Lat	chr	Longitude and Latitude Coordinates for mapping

Data Processing

The data set has a lot of information available but not all data is necessary for this report. To answer the questions stated in the *Purpose* section, the data is modified to only include the following columns:

- OCCUR_DATE
- OCCUR_TIME
- BORO
- STATISTICAL_MURDER_FLAG
- PERP_AGE_GROUP
- PERP_SEX
- PERP_RACE
- Latitude
- Longitude

```
colnames(inc_data)
```

```
##  [1] "INCIDENT_KEY"            "OCCUR_DATE"
##  [3] "OCCUR_TIME"              "BORO"
##  [5] "PRECINCT"                "JURISDICTION_CODE"
##  [7] "LOCATION_DESC"           "STATISTICAL_MURDER_FLAG"
##  [9] "PERP_AGE_GROUP"          "PERP_SEX"
## [11] "PERP_RACE"               "VIC_AGE_GROUP"
## [13] "VIC_SEX"                 "VIC_RACE"
## [15] "X_COORD_CD"              "Y_COORD_CD"
## [17] "Latitude"                "Longitude"
## [19] "Lon_Lat"
```

```
# Choose the columns necessary for the analysis
inc_data <- inc_data %>%
  select(OCCUR_DATE:BORO,
         STATISTICAL_MURDER_FLAG:PERP_RACE,
         Latitude:Longitude)
colnames(inc_data)
```

```
## [1] "OCCUR_DATE"              "OCCUR_TIME"
## [3] "BORO"                    "STATISTICAL_MURDER_FLAG"
## [5] "PERP_AGE_GROUP"          "PERP_SEX"
## [7] "PERP_RACE"               "Latitude"
## [9] "Longitude"
```

The next step after selecting the required columns for the analysis, is to verify that all data fields within the newly modified data set contains data. If any fields are missing data, this will be replaced with a "UNKNOWN" or "U" character type. Empty data fields are indicated with the `NA` data type. Missing data could be an indication that the incident has not been closed and is still being investigated or over the years of collecting this data, fields have been forgotten to be updated after a case had been closed. Understanding why a data set looks the way it does is very important when trying to analyze it. The best way to understand the data is to explore it and "play" with it.

```
# Count how many data fields are NA
lapply(inc_data, function(x) sum(is.na(x)))
```

```
## $OCCUR_DATE
## [1] 0
##
## $OCCUR_TIME
## [1] 0
##
## $BORO
## [1] 0
##
## $STATISTICAL_MURDER_FLAG
## [1] 0
##
## $PERP_AGE_GROUP
## [1] 9344
##
```

```
## $PERP_SEX
## [1] 9310
##
## $PERP_RACE
## [1] 9310
##
## $Latitude
## [1] 0
##
## $Longitude
## [1] 0
```

```
unknown_replace <- "UNKNOWN"
u_replace <- "U"
inc_data <- inc_data %>%
  replace_na(list(PERP_AGE_GROUP=unknown_replace,
                  PERP_SEX=u_replace,
                  PERP_RACE=unknown_replace))
```

Now that the entire data frame is updated to have some data in each field, factors should be applied to most columns since they represent categorical data. Factors are not applied to the columns of OCCUR_DATE, OCCUR_TIME, Latitude, and Longitude. When running `as.factor()` on all the data columns, PERP_AGE_GROUP had several miscellaneous data entries that had to be addressed. Since it is unknown what "1020", "224", and "940" represents, those entries are changed to "UNKNOWN".

```
# Update the data frame before factoring can occur
# PERP_AGE_GROUP
inc_data$PERP_AGE_GROUP <- recode(inc_data$PERP_AGE_GROUP, "1020" = unknown_replace)
inc_data$PERP_AGE_GROUP <- recode(inc_data$PERP_AGE_GROUP, "224" = unknown_replace)
inc_data$PERP_AGE_GROUP <- recode(inc_data$PERP_AGE_GROUP, "940" = unknown_replace)

# Factors
inc_data$BORO <- as.factor(inc_data$BORO)
inc_data$STATISTICAL_MURDER_FLAG <- as.factor(as.logical(inc_data$STATISTICAL_MURDER_FLAG))
inc_data$PERP_AGE_GROUP <- as.factor(inc_data$PERP_AGE_GROUP)
inc_data$PERP_SEX <- as.factor(inc_data$PERP_SEX)
inc_data$PERP_RACE <- as.factor(inc_data$PERP_RACE)
```

As a final step in processing and manipulating the data, the OCCUR_DATE and OCCUR_TIME data will be used to create two additional columns for quick and easy analysis. To best represent when during the week a shooting occurred, the OCCUR_DATE is transformed in to a weekday and the OCCUR_TIME is manipulated into only have the hour of the occurrence. The exact minute and seconds are unnecessary details for this analysis.

```
# Add a column with the weekday when the incident occurred
# Add a column with the hour when the incident occurred
inc_data <- inc_data %>%
  mutate(WKDAY = wday(mdy(inc_data$OCCUR_DATE), label=TRUE)) %>%
  mutate(HR = hour(hms(as.character(inc_data$OCCUR_TIME))))

summary(inc_data)
```

```
##   OCCUR_DATE         OCCUR_TIME                    BORO
```

```
##  Length:25596       Length:25596       BRONX        : 7402
##  Class :character   Class :character   BROOKLYN     :10365
##  Mode  :character   Mode  :character   MANHATTAN    : 3265
##                                        QUEENS       : 3828
##                                        STATEN ISLAND:  736
##
##
##  STATISTICAL_MURDER_FLAG PERP_AGE_GROUP  PERP_SEX
##  FALSE:20668             <18    : 1463   F:  371
##  TRUE : 4928             18-24  : 5844   M:14416
##                          25-44  : 5202   U:10809
##                          45-64  :  535
##                          65+    :   57
##                          UNKNOWN:12495
##
##                           PERP_RACE        Latitude       Longitude
##  AMERICAN INDIAN/ALASKAN NATIVE:    2   Min.   :40.51   Min.   :-74.25
##  ASIAN / PACIFIC ISLANDER      :  141   1st Qu.:40.67   1st Qu.:-73.94
##  BLACK                         :10668   Median :40.70   Median :-73.92
##  BLACK HISPANIC                : 1203   Mean   :40.74   Mean   :-73.91
##  UNKNOWN                       :11146   3rd Qu.:40.82   3rd Qu.:-73.88
##  WHITE                         :  272   Max.   :40.91   Max.   :-73.70
##  WHITE HISPANIC                : 2164
##  WKDAY            HR
##  Sun:5156   Min.   : 0.00
##  Mon:3597   1st Qu.: 3.00
##  Tue:2945   Median :15.00
##  Wed:2818   Mean   :12.19
##  Thu:2809   3rd Qu.:20.00
##  Fri:3384   Max.   :23.00
##  Sat:4887
```

Analysis

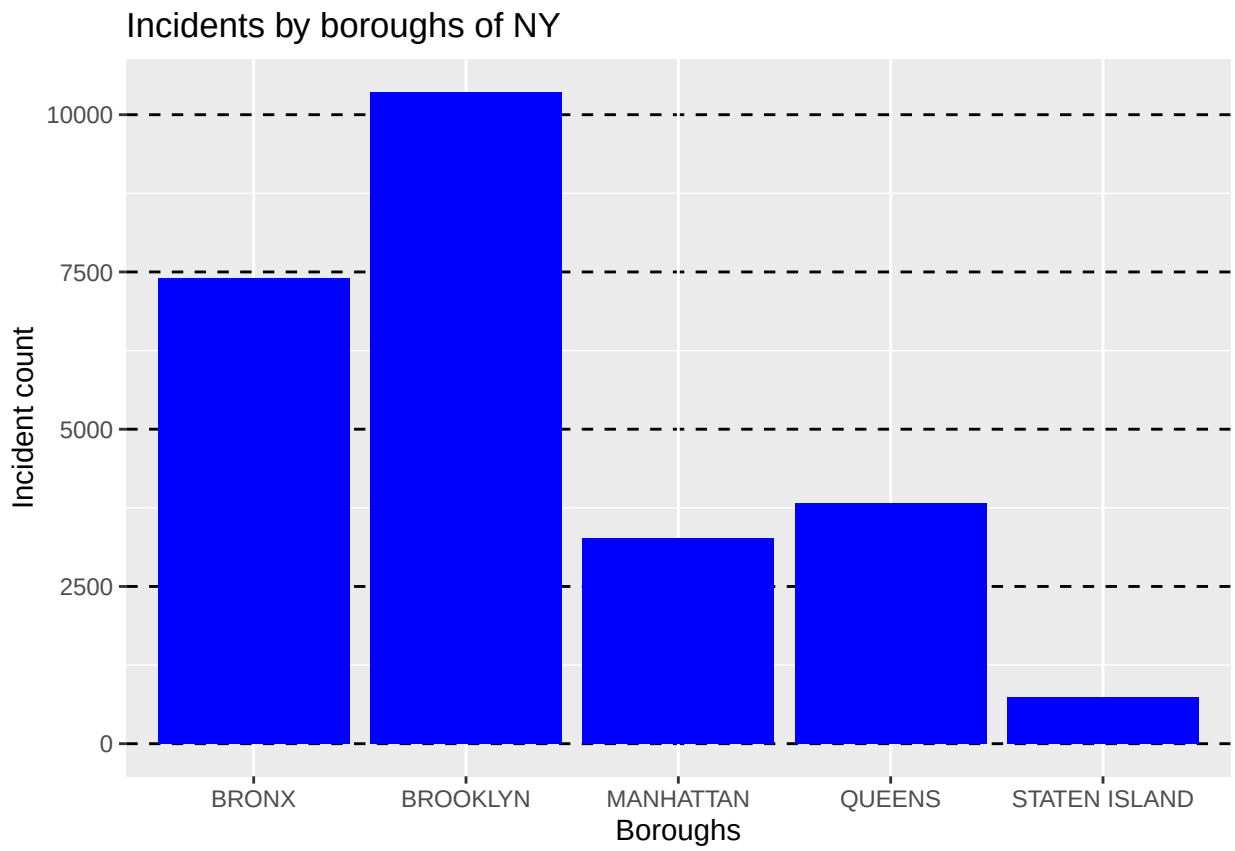
In this section, the processed data is used to answer the questions posed in the *Purpose* section. The analysis has several metrics and graphs in order to get a better understanding of the data.

1. What is the overall distribution of the incidents by borough? Which borough has the highest and lowest incident count? Of the incidents, what percentage were murders?

In the data set, the column header BORO was provided which categorizes the incidents into individual boroughs of New York which include Brooklyn, Queens, Bronx, Manhattan, and Staten Island. The following graph depicts the overall distribution of incidents by boroughs. Overall, 41% of the incidents happen in Brooklyn with 10365 incidents and the lowest incident count happened in Staten Island with 736 which accounts for 3% of the overall total.

```
# Plot the incident distribution by boroughs
boro_graph <- ggplot(data = inc_data, mapping = aes(x = BORO)) +
              geom_bar(fill = "blue") +
              labs(title = "Incidents by boroughs of NY",
```

```
                  x = "Boroughs",
                  y = "Incident count") +
             theme(panel.grid.major.y = element_line(color = "black",
                                                     linewidth = 0.5,
                                                     linetype = 2))
boro_graph
```




```
# Create a table with the STATISTICAL_MURDER_FLAG and the murder percentage
murder_table <- table(inc_data$BORO, inc_data$STATISTICAL_MURDER_FLAG)
row_names <- rownames(murder_table)
true_col <- c(murder_table[,"TRUE"])
false_col <- c(murder_table[,"FALSE"])
# Change the table into a data frame and add the boroughs as the row names
murder_table <- tibble(data.frame("TRUE" = true_col,
                                  "FALSE" = false_col,
                                  "PERC_MURDER" = 100*true_col / (true_col+false_col)))
murder_table <- murder_table %>%
  mutate(BORO = row_names)
# Rearrange the columns
murder_table <- murder_table[, c(4,1,2,3)]
murder_table
```

```
## # A tibble: 5 x 4
##   BORO          TRUE. FALSE. PERC_MURDER
##   <chr>         <int>  <int>       <dbl>
```

```
## 1 BRONX          1417   5985        19.1
## 2 BROOKLYN       2020   8345        19.5
## 3 MANHATTAN       574   2691        17.6
## 4 QUEENS          762   3066        19.9
## 5 STATEN ISLAND   155    581        21.1
```

Given the table above, the percentage of murders can be found for each borough. Interestingly, the highest and lowest incident count falls onto Brooklyn and Staten Island respectively, however, they do not represent the highest and lowest percentage of murders for given shooting incidents. The highest murder percentage falls on Staten Island with 21.1% and the lowest on Manhattan with 17.6%.

2. When are shooting incidents the highest? Are there times during the day that are safer than others? Are there days of the week that are safer than others?

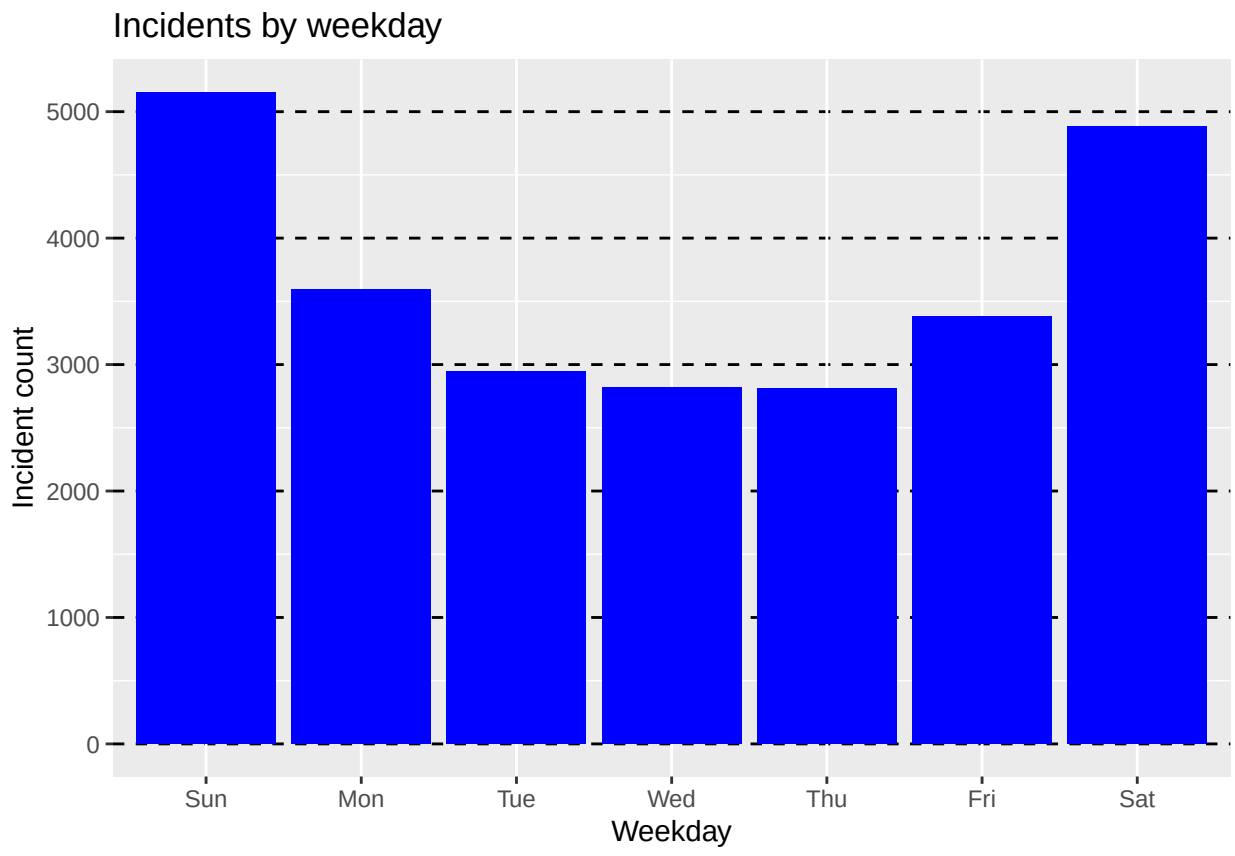
There are two parts in answering the question as to when are incidents the highest. One is the day of the week and the other is the time of the day. The following graphs and table depict both of those statistics. With the data set, it can be determined that the most shooting incidents happen on the weekends. A combined percentage for the weekend (Saturday and Sunday) is 39.2%. If Friday and Monday are included, this increases to 66.5%. Therefore roughly 2/3 of all incidents happen in a 4 day span. The lowest shooting incidents happen between Tuesday and Thursday. When combining and looking at weekdays only, the total incidents account for 60.8%. Over the course of 5 days during the week, the total incident count is less than looking at the weekends with Monday and Friday.

```
# Table of individual counts for the weekdays
total_inc = dim(inc_data)[1]
WKDAY_COUNT <- table(inc_data$WKDAY)
wk_rownames <- rownames(WKDAY_COUNT)
wk_count_table <- tibble(WKDAY_COUNT) %>%
                  mutate(WKDAY = wk_rownames) %>%
                  mutate(PERC = 100*WKDAY_COUNT/total_inc)
wk_count_table$WKDAY_COUNT <- as.integer(wk_count_table$WKDAY_COUNT)
wk_count_table$PERC <- as.double(wk_count_table$PERC)
wk_count_table <- wk_count_table[, c(2,1,3)]
wk_count_table
```

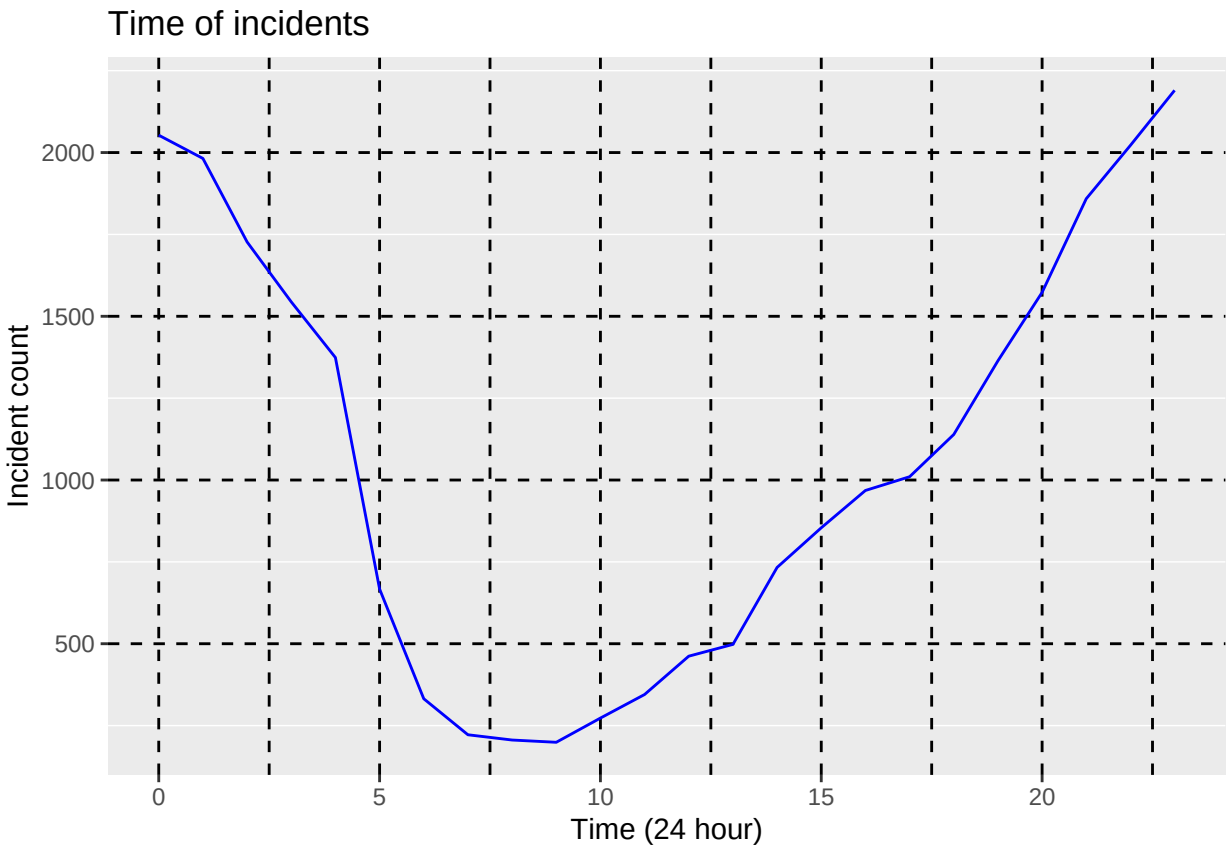
```
## # A tibble: 7 x 3
##   WKDAY WKDAY_COUNT  PERC
##   <chr>       <int> <dbl>
## 1 Sun          5156  20.1
## 2 Mon          3597  14.1
## 3 Tue          2945  11.5
## 4 Wed          2818  11.0
## 5 Thu          2809  11.0
## 6 Fri          3384  13.2
## 7 Sat          4887  19.1
```

```
# Use WKDAY and HR to generate the graphs
wkday_graph <- ggplot(data = inc_data, mapping = aes(x = WKDAY)) +
               geom_bar(fill = "blue") +
               labs(title = "Incidents by weekday",
                    x = "Weekday",
                    y = "Incident count") +
```

```
              theme(panel.grid.major.y = element_line(color = "black",
                                                      linewidth = 0.5,
                                                      linetype = 2))
# Need hour and total count of each hour to draw a line
hr_count_data <- inc_data %>%
                  group_by(HR) %>%
                  count()
hr_graph <- ggplot(data = hr_count_data, mapping = aes(x = HR, y = n)) +
            geom_line(color = "blue") +
            labs(title = "Time of incidents",
                 x = "Time (24 hour)",
                 y = "Incident count") +
            theme(panel.grid.major = element_line(color = "black",
                                                  linewidth = 0.5,
                                                  linetype = 2),
                  panel.grid.minor.x = element_line(color = "black",
                                                    linewidth = 0.5,
                                                    linetype = 2))
wkday_graph
```




```
hr_graph
```



Finally, when looking at when it is the most safe relative to incidents, this is between 05:00 and 10:00 (global minimum). The number of incidents increase almost linearly as the day progresses and starts dropping again shortly after midnight.

3. What factors are statistically significant in predicting a murder given the perpetrators profile, time of day, weekday, and location?

Answering the question on whether there was a murder or not is a binomial answer and is best fit with a logical regression model. Primarily, the focus is on determining which inputs are statistically significant. The data used for the model comprised of PERP_AGE_GROUP, PERP_SEX, PERP_RACE, OCCUR_DATE, OCCUR_TIME, Latitude, and Longitude. Statistically significant for this report is determined by the p-value. Any p-value below 0.05 is considered statistically significant.

```
# Setup a logical regression model
mod <- glm(formula = STATISTICAL_MURDER_FLAG ~ PERP_AGE_GROUP +
                                               PERP_SEX +
                                               PERP_RACE +
                                               WKDAY +
                                               HR +
                                               Latitude +
                                               Longitude,
           family = binomial,
           data = inc_data)
summary(mod)
```

```
##
## Call:
## glm(formula = STATISTICAL_MURDER_FLAG ~ PERP_AGE_GROUP + PERP_SEX +
##     PERP_RACE + WKDAY + HR + Latitude + Longitude, family = binomial,
##     data = inc_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9214  -0.6805  -0.6070  -0.2242   2.9334
##
## Coefficients:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     41.011966  86.674495   0.473 0.636090
## PERP_AGE_GROUP18-24              0.171251   0.075386   2.272 0.023107 *
## PERP_AGE_GROUP25-44              0.505664   0.075080   6.735 1.64e-11 ***
## PERP_AGE_GROUP45-64              0.837460   0.114547   7.311 2.65e-13 ***
## PERP_AGE_GROUP65+                1.008420   0.282806   3.566 0.000363 ***
## PERP_AGE_GROUPUNKNOWN           -2.242106   0.171326 -13.087  < 2e-16 ***
## PERP_SEXM                       -0.190178   0.120994  -1.572 0.115999
## PERP_SEXU                        2.430830   0.268124   9.066  < 2e-16 ***
## PERP_RACEASIAN / PACIFIC ISLANDER  9.970651  84.227839   0.118 0.905769
## PERP_RACEBLACK                   9.522585  84.227640   0.113 0.909985
## PERP_RACEBLACK HISPANIC          9.400811  84.227670   0.112 0.911131
## PERP_RACEUNKNOWN                 9.048935  84.227892   0.107 0.914445
## PERP_RACEWHITE                  10.152451  84.227737   0.121 0.904059
## PERP_RACEWHITE HISPANIC          9.679333  84.227651   0.115 0.908510
## WKDAY.L                         -0.057106   0.039780  -1.436 0.151134
## WKDAY.Q                         -0.085793   0.042774  -2.006 0.044888 *
## WKDAY.C                         -0.056723   0.043019  -1.319 0.187316
## WKDAY^4                         -0.022607   0.043831  -0.516 0.606013
## WKDAY^5                         -0.009544   0.046033  -0.207 0.835754
## WKDAY^6                         -0.059771   0.047397  -1.261 0.207278
## HR                              -0.002729   0.001978  -1.380 0.167540
## Latitude                        -0.434830   0.190204  -2.286 0.022247 *
## Longitude                        0.461700   0.241070   1.915 0.055465 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 25077  on 25595  degrees of freedom
## Residual deviance: 24150  on 25573  degrees of freedom
## AIC: 24196
##
## Number of Fisher Scoring iterations: 9
```

From the table above it can be determined that the most significant contributors in predicting a murder are PERP_AGE_GROUP (all age groups including UNKNOWN), PERP_SEXU (sex was unknown from the perpetrator), Latitude, and Longitude. Interestingly, not knowing the sex of the perpetrator was better for the model than knowing. The perpetrators race, time of day, day of the week had no impact on the prediction considering the distributions discussed in this section.

Bias Considerations

When thinking about New York, and any depiction through TV or movies, you always hear how bad each of the boroughs are, especially Brooklyn and Bronx. Although both of those boroughs have the highest shooting incident count, they do not necessary have the highest statistical murder rate. That falls on Staten Island which by incident count is the lowest. Perception and prior knowledge of any topic that is analyzed can be a bias towards data. It is important to analyze any data with as much neutrality as possible. However, even with that analysis, caution should be exercised. In the data set, there is no data about the total population in each borough. This additional information could be very valuable in painting a fuller picture.

Appendix

Session Information

```
## R version 4.2.2 (2022-10-31 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] sessioninfo_1.2.2 lubridate_1.9.0   timechange_0.1.1  forcats_0.5.2
##  [5] stringr_1.4.1     dplyr_1.0.10      purrr_0.3.5       readr_2.1.3
##  [9] tidyr_1.2.1       tibble_3.1.8      ggplot2_3.4.0     tidyverse_1.3.2
##
## loaded via a namespace (and not attached):
##  [1] tidyselect_1.2.0    xfun_0.34           haven_2.5.1
##  [4] gargle_1.2.1        colorspace_2.0-3    vctrs_0.5.0
##  [7] generics_0.1.3      htmltools_0.5.3     yaml_2.3.6
## [10] utf8_1.2.2          rlang_1.0.6         pillar_1.8.1
## [13] withr_2.5.0         glue_1.6.2          DBI_1.1.3
## [16] dbplyr_2.2.1        modelr_0.1.10       readxl_1.4.1
## [19] lifecycle_1.0.3     munsell_0.5.0       gtable_0.3.1
## [22] cellranger_1.1.0    rvest_1.0.3         evaluate_0.18
## [25] labeling_0.4.2      knitr_1.41          tzdb_0.3.0
## [28] fastmap_1.1.0       fansi_1.0.3         highr_0.9
## [31] broom_1.0.1         scales_1.2.1        backports_1.4.1
## [34] googlesheets4_1.0.1 jsonlite_1.8.3      farver_2.1.1
## [37] fs_1.5.2            hms_1.1.2           digest_0.6.30
## [40] stringi_1.7.8       grid_4.2.2          cli_3.4.1
## [43] tools_4.2.2         magrittr_2.0.3      crayon_1.5.2
## [46] pkgconfig_2.0.3     ellipsis_0.3.2      xml2_1.3.3
```

```
## [49] reprex_2.0.2        googledrive_2.0.0   assertthat_0.2.1
## [52] rmarkdown_2.18      httr_1.4.4          rstudioapi_0.14
## [55] R6_2.5.1            compiler_4.2.2
```