NYPD Shooting Incident Data Report

5/21/2021

Question to answer

In this report, we will look at whether you are statistically more likely to die at any one block of time in the day after being shot. Another way to say this is are there any trends or blocks of time where shooting incidents are more fatal.

Description and import of data

The data used in this report is the list of NYC shooting incidents ranging from 2006 to the end of the previous year (2020) provided by the NYPD. The data was obtained from the data catalog archive.

```
data_url <- "https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv?accessType=DOWNLOAD"
nypd_shooting_incidents <- read_csv(data_url)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   INCIDENT_KEY = col_double(),
##   OCCUR_DATE = col_character(),
##   OCCUR_TIME = col_time(format = ""),
##   BORO = col_character(),
##   PRECINCT = col_double(),
##   JURISDICTION_CODE = col_double(),
##   LOCATION_DESC = col_character(),
##   STATISTICAL_MURDER_FLAG = col_logical(),
##   PERP_AGE_GROUP = col_character(),
##   PERP_SEX = col_character(),
##   PERP_RACE = col_character(),
##   VIC_AGE_GROUP = col_character(),
##   VIC_SEX = col_character(),
##   VIC_RACE = col_character(),
##   X_COORD_CD = col_number(),
##   Y_COORD_CD = col_number(),
##   Latitude = col_double(),
##   Longitude = col_double(),
##   Lon_Lat = col_character()
## )
```

Tidying the data

After importing the data, we will tidy and parse it such that we are left with a tibble that contains data with the following characteristics:

- Blocks of time
- Numbers of shooting incidents that occurred in that block
- Number of shooting incidents in that block that were fatal
- Ratio of fatal shooting incidents to all shooting incidents in that block

Method for generating this data as well as the first 10 rows are show below.

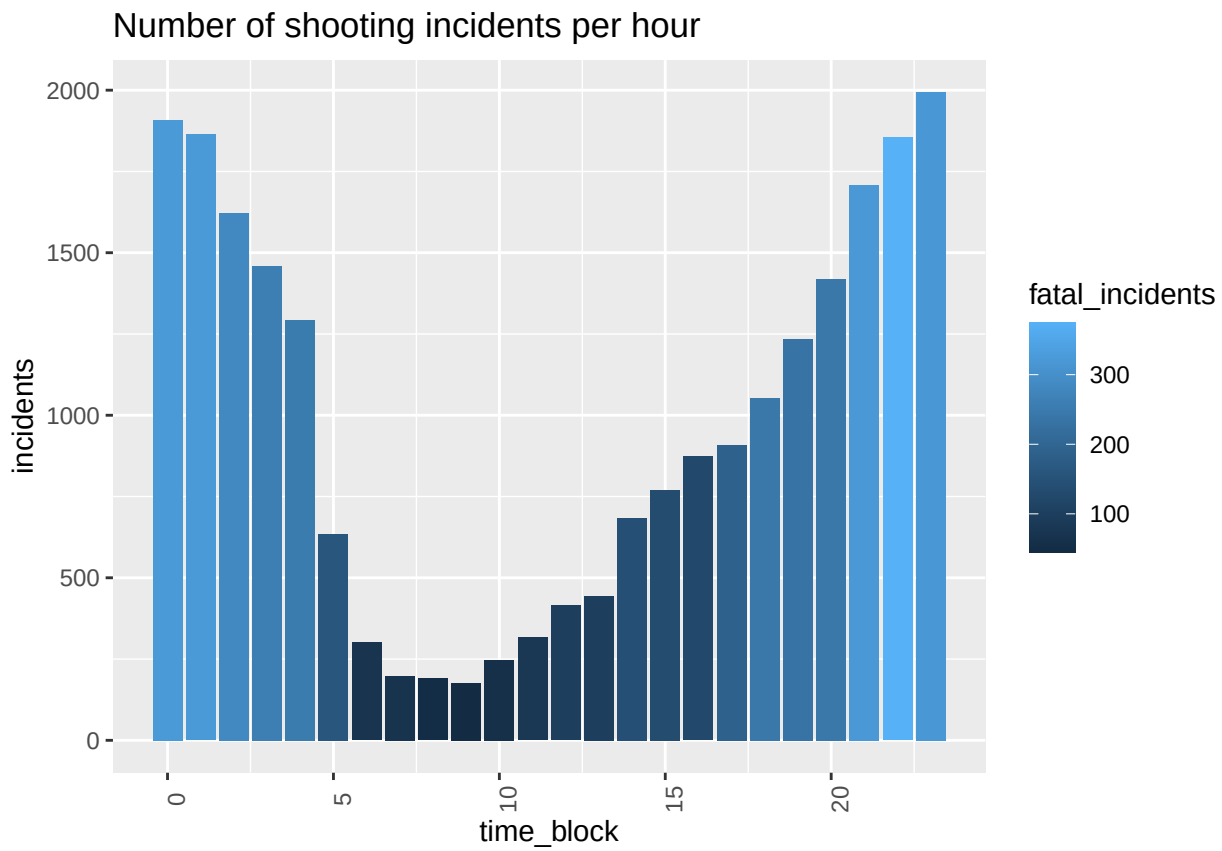
```
filtered_npyd_shooting_incidents <- nypd_shooting_incidents %>%
  mutate(time_block = hour(OCCUR_TIME))%>%
  select(time_block, STATISTICAL_MURDER_FLAG) %>%
  group_by(time_block) %>%
  summarise(incidents = n(),
            fatal_incidents = sum(STATISTICAL_MURDER_FLAG)) %>%
  mutate(per_fatal = fatal_incidents / incidents)

# First 5 rows of data frame shown below

head(filtered_npyd_shooting_incidents, n=5L)
```

```
## # A tibble: 5 x 4
##   time_block incidents fatal_incidents per_fatal
##        <int>     <int>           <int>     <dbl>
## 1          0      1908             322     0.169
## 2          1      1864             323     0.173
## 3          2      1622             284     0.175
## 4          3      1459             259     0.178
## 5          4      1293             252     0.195
```

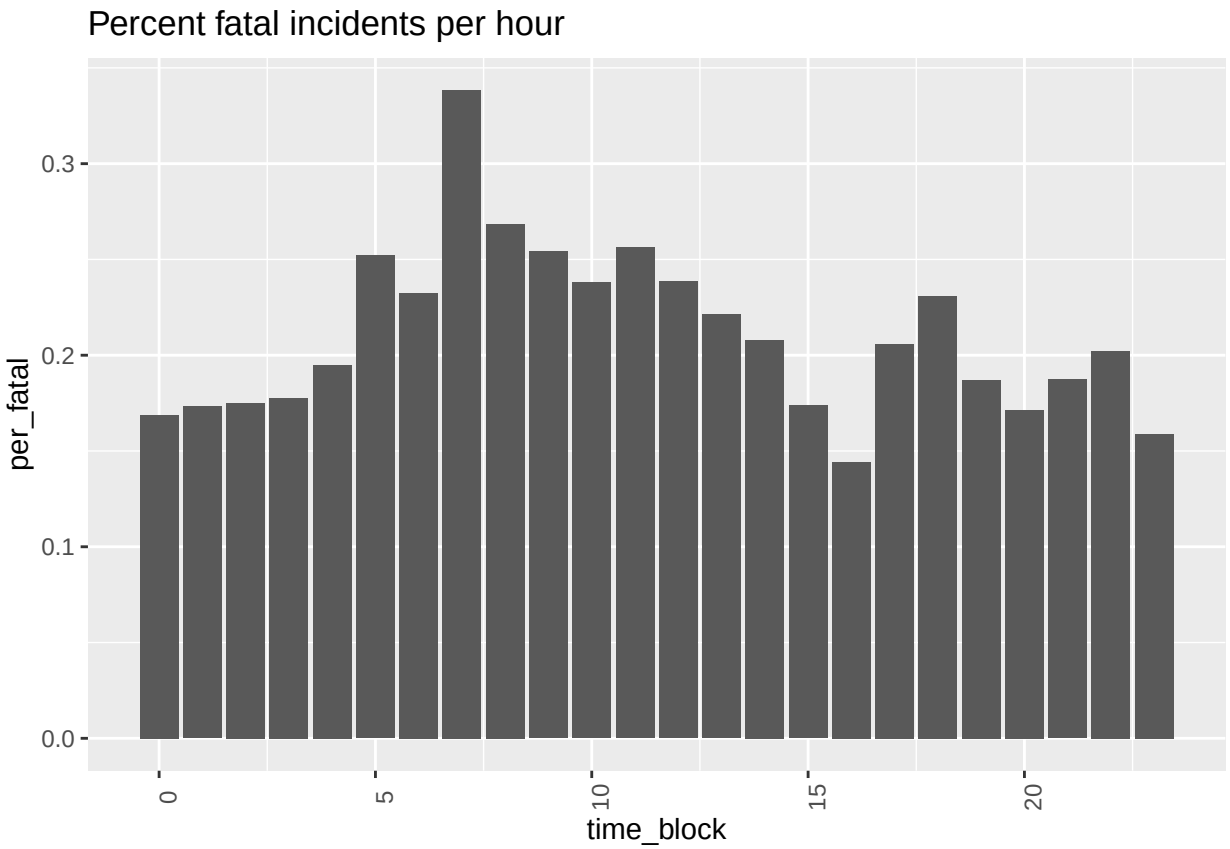
Visualizing and modeling the data

To visualize the data, we will look at two bar plots. The first will show the number of incidents per time block, color coded to show those that resulted in fatalities. The second will show the percent fatality per time block.

```
filtered_npyd_shooting_incidents %>%
  ggplot(aes(fill=fatal_incidents, x=time_block, y=incidents)) +
  ggtitle("Number of shooting incidents per hour") +
  geom_bar(stat="identity", position="stack") +
  theme(axis.text.x = element_text(angle = 90))
```



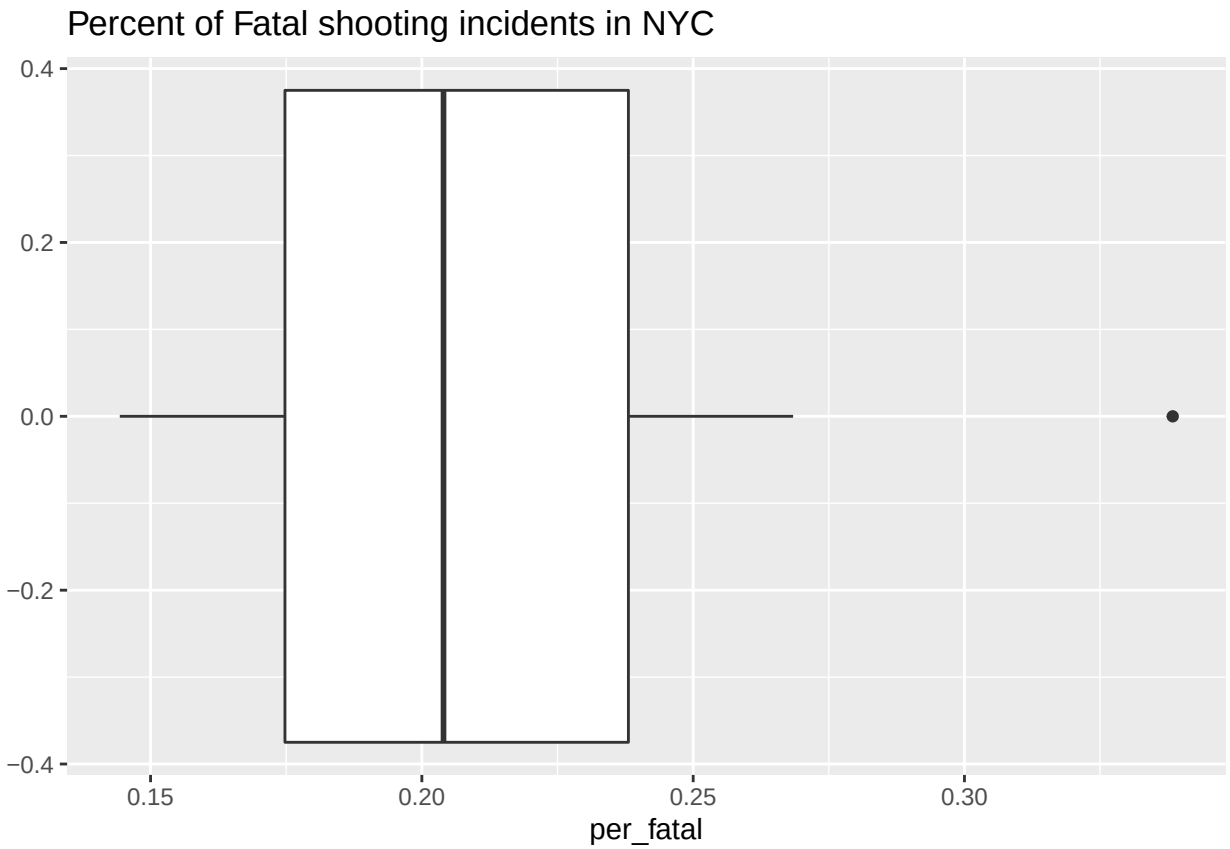

```
filtered_npyd_shooting_incidents %>%
  ggplot(aes(x=time_block, y=per_fatal)) +
  ggtitle("Percent fatal incidents per hour") +
  geom_bar(stat="identity") +
  theme(axis.text.x = element_text(angle = 90))
```



Analysis

To gain insight into the data, we will generate a box plot of the percent fatalities and look at common statistical measures

```
filtered_npyd_shooting_incidents %>%
  ggplot(aes(x = per_fatal)) +
  geom_boxplot() +
  ggtitle("Percent of Fatal shooting incidents in NYC")
```




```
print(summary(filtered_npyd_shooting_incidents$per_fatal))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.1443  0.1748  0.2040  0.2108  0.2381  0.3384
```

Conclusion and potential biases

Looking at the visualizations and analysis from this data, we can say that there exists a definitive uptick in percent of shooting incidents which were fatal in the morning hours. This reaches a peak in the 7-8 am time block where we see an outlier indicating an 11% higher fatality rate.

While this is speculative without doing a deeper dive on how the data was gathered, it is possible that the data is biased in a few ways. Two that come to mind immediately is that it is possible that data gathered during off hours could have possibly have been entered later when more individuals were around leading to incorrect times. This could certainly explain why we see a spike in the 7-8 am time slot. Another area of bias is that areas with lower economic activity could have less resources aimed at accurately recording data which could leave out potentially chucks of incidents.

Session info

Session info block shown below

```
sessionInfo()
```

```
## R version 4.1.0 (2021-05-18)
## 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] lubridate_1.7.10 forcats_0.5.1    stringr_1.4.0    dplyr_1.0.6
##  [5] purrr_0.3.4      readr_1.4.0      tidyr_1.1.3      tibble_3.1.2
##  [9] ggplot2_3.3.3    tidyverse_1.3.1
##
## loaded via a namespace (and not attached):
##  [1] tidyselect_1.1.1  xfun_0.23         haven_2.4.1       colorspace_2.0-1
##  [5] vctrs_0.3.8       generics_0.1.0    htmltools_0.5.1.1 yaml_2.2.1
##  [9] utf8_1.2.1        rlang_0.4.11      pillar_1.6.1      glue_1.4.2
## [13] withr_2.4.2       DBI_1.1.1         dbplyr_2.1.1      modelr_0.1.8
## [17] readxl_1.3.1      lifecycle_1.0.0   munsell_0.5.0     gtable_0.3.0
## [21] cellranger_1.1.0  rvest_1.0.0       evaluate_0.14     labeling_0.4.2
## [25] knitr_1.33        curl_4.3.1        fansi_0.4.2       highr_0.9
## [29] broom_0.7.6       Rcpp_1.0.6        scales_1.1.1      backports_1.2.1
## [33] jsonlite_1.7.2    farver_2.1.0      fs_1.5.0          hms_1.1.0
## [37] digest_0.6.27     stringi_1.6.1     grid_4.1.0        cli_2.5.0
## [41] tools_4.1.0       magrittr_2.0.1    crayon_1.4.1      pkgconfig_2.0.3
## [45] ellipsis_0.3.2    xml2_1.3.2        reprex_2.0.0      assertthat_0.2.1
## [49] rmarkdown_2.8     httr_1.4.2        rstudioapi_0.13   R6_2.5.0
## [53] compiler_4.1.0
```