NYPD Shooting Incident Data Report

Venkata Kasireddy

9/13/2021

R Markdown

#install tidyverse Package – library(tidyverse)

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble  3.1.3     v purrr   0.3.4
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.0     v forcats 0.5.1

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

#Read the data from DATA.Gov, NYPD Shooting Incident Data Report into uid_ny object

```
uid_ny <- read_csv("https://data.cityofnewyork.us/api/views/5ucz-vwe8/rows.csv?accessType=DOWNLOAD")
```

```
## Rows: 902 Columns: 19

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (10): OCCUR_DATE, BORO, LOCATION_DESC, PERP_AGE_GROUP, PERP_SEX, PERP_R...
## dbl   (7): INCIDENT_KEY, PRECINCT, JURISDICTION_CODE, X_COORD_CD, Y_COORD_CD...
## lgl   (1): STATISTICAL_MURDER_FLAG
## time  (1): OCCUR_TIME

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

#removing some columns from data that is not needed for the analysis

```
uid_ny <- uid_ny %>% select (-c(OCCUR_TIME,PRECINCT,JURISDICTION_CODE,LOCATION_DESC,STATISTICAL_MURDER_
```

#filtering rows where data is not avialable

```
uid_ny <- uid_ny %>%  filter_at(vars(starts_with("PERP_")), any_vars(! is.na(.)))
```

#library(lubridate) – to convert data char column to date object use the lubridate function

```
library(lubridate)
```

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

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

#conevrt OCCUR_DATE (char) to occur_data (date object)

```
uid_ny <- uid_ny %>% mutate( occur_date = mdy( OCCUR_DATE ) )
```

#remove chr OCCUR_DATE from uid_ny object

```
uid_ny <- uid_ny %>% select (-c(OCCUR_DATE))
```

Analysis -1

#daily cases

```
daily_cases <- uid_ny %>% count(occur_date)
```

#linear model (prediction/estimation number of cases per day based on history)

```
mod1 <- lm(n ~ occur_date,data = daily_cases)
```

#dispaly statstical details of model

```
summary(mod1)
```

```
##
## Call:
## lm(formula = n ~ occur_date, data = daily_cases)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.3808 -1.4432 -0.8723  1.1231  6.9370
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.295e+02  6.392e+01  -2.025   0.0449 *
## occur_date   7.063e-03  3.414e-03   2.069   0.0405 *
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.139 on 132 degrees of freedom
## Multiple R-squared:  0.03141,    Adjusted R-squared:  0.02407
## F-statistic:  4.28 on 1 and 132 DF,  p-value: 0.04051
```

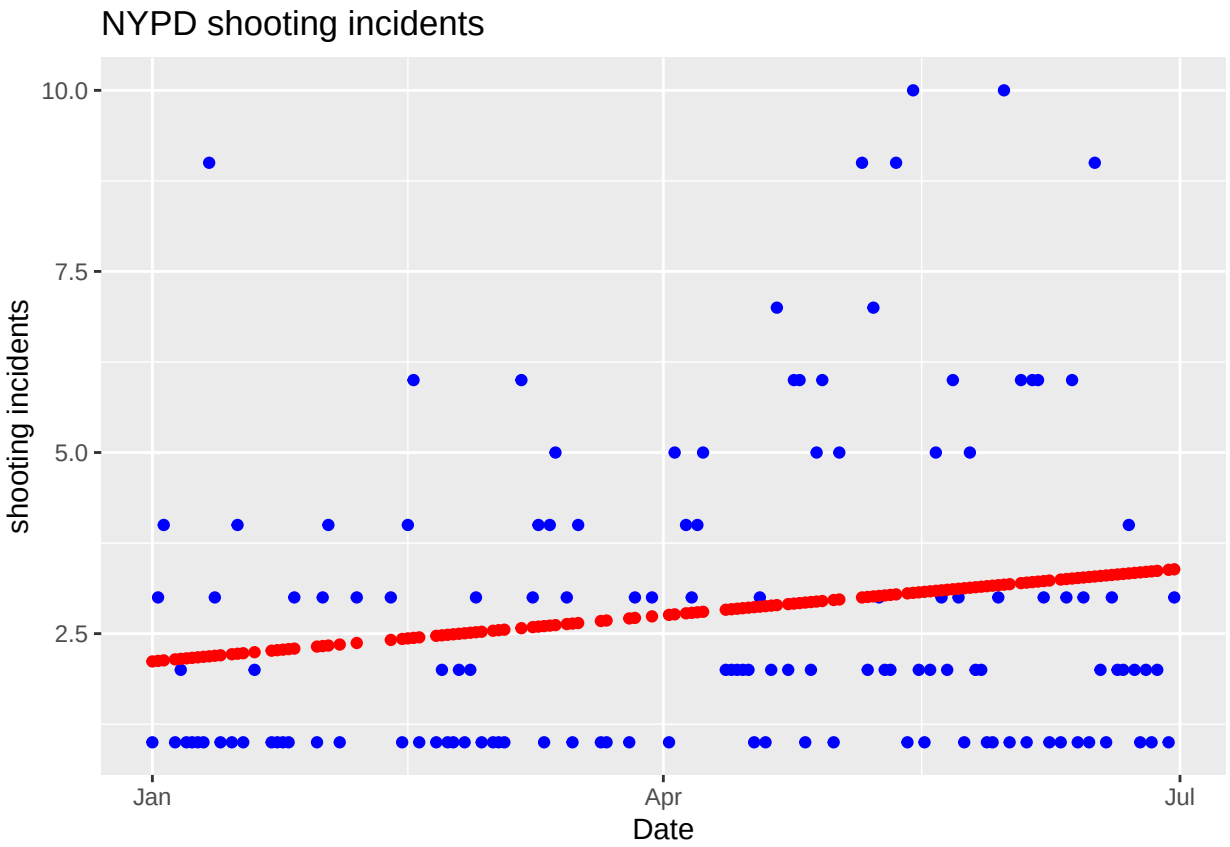
#add new coloumn of est (estimate/prediction)

```
daily_cases %>% mutate(est = predict(mod1))
```

```
## # A tibble: 134 x 3
##    occur_date     n   est
##    <date>     <int> <dbl>
##  1 2021-01-01     1  2.12
##  2 2021-01-02     3  2.12
##  3 2021-01-03     4  2.13
##  4 2021-01-05     1  2.14
##  5 2021-01-06     2  2.15
##  6 2021-01-07     1  2.16
##  7 2021-01-08     1  2.17
##  8 2021-01-09     1  2.17
##  9 2021-01-10     1  2.18
## 10 2021-01-11     9  2.19
## # ... with 124 more rows
```

#plot x axis as 'time', y axis number of shooting incidents, blue color as history, red color as estimate of linear #model

```
daily_cases %>% ggplot() + geom_point(aes(x = occur_date, y = n),color = "blue") + geom_point(aes(x = o
```



#Complete code to run above plot — #daily_cases %>% ggplot() + geom_point(aes(x = occur_date, y = n),color = "blue") + geom_point(aes(x = occur_date, y = predict(mod1)),color = "red") + labs(x = "Date",y = "shooting incidents",title = "NYPD shooting incidents",)

Analysis - 2

#count by location (area wise with in new york)

```
Location_count <- uid_ny %>% count(BORO)

Location_count
```

```
## # A tibble: 5 x 2
##   BORO              n
##   <chr>         <int>
## 1 BRONX           144
## 2 BROOKLYN        104
## 3 MANHATTAN        67
## 4 QUEENS           48
## 5 STATEN ISLAND    12
```

#Based on above data, Bronx and Brooklyn have highest number of incidents hence top two unsafe places where as #Staten Island is lowest cases hence safe place.