# NY Collision Data Analysis

February 21, 2018

## 1 NYPD Motor Vehicle Collision Analysis

I'm trying to analyze data and traffic collision patterns from this NYPD Motor Vehicle Collision Dataset

loading the libraries

```
In [1]: library(tidyverse)
```

```
Warning message:
package tidyverse was built under R version 3.4.2 Attaching packages  tidyverse 1.2.1
 ggplot2 2.2.1      purrr   0.2.4
 tibble  1.4.2      dplyr   0.7.4
 tidyr   0.8.0      stringr 1.2.0
 readr   1.1.1      forcats 0.3.0
Warning message:
package tibble was built under R version 3.4.3Warning message:
package tidyr was built under R version 3.4.3Warning message:
package purrr was built under R version 3.4.2Warning message:
package dplyr was built under R version 3.4.2Warning message:
package forcats was built under R version 3.4.3 Conflicts  tidyverse_conflicts()
 dplyr::filter() masks stats::filter()
 dplyr::lag()    masks stats::lag()
```

```
In [2]: library(plotly)
```

```
Attaching package: plotly

The following object is masked from package:ggplot2:

    last_plot

The following object is masked from package:stats:

    filter
```

```
The following object is masked from package:graphics:

    layout
```

```
In [3]: library(ggplot2)
```

reading the data

```
In [4]: collision<-read_csv("NYPD_Motor_Vehicle_Collisions.csv")
```

```
Parsed with column specification:
cols(
  .default = col_character(),
  TIME = col_time(format = ""),
  `ZIP CODE` = col_integer(),
  LATITUDE = col_double(),
  LONGITUDE = col_double(),
  `NUMBER OF PERSONS INJURED` = col_integer(),
  `NUMBER OF PERSONS KILLED` = col_integer(),
  `NUMBER OF PEDESTRIANS INJURED` = col_integer(),
  `NUMBER OF PEDESTRIANS KILLED` = col_integer(),
  `NUMBER OF CYCLIST INJURED` = col_integer(),
  `NUMBER OF CYCLIST KILLED` = col_integer(),
  `NUMBER OF MOTORIST INJURED` = col_integer(),
  `NUMBER OF MOTORIST KILLED` = col_integer(),
  `UNIQUE KEY` = col_integer()
)
See spec(...) for full column specifications.
```

```
In [5]: glimpse(collision)
```

```
Observations: 1,209,947
Variables: 29
$ DATE                           <chr> "02/13/2018", "02/13/2018", "02/13/...
$ TIME                           <time> 00:00:00, 00:00:00, 00:00:00, 00:0...
$ BOROUGH                        <chr> "BRONX", "BRONX", NA, NA, NA, NA, "...
$ `ZIP CODE`                     <int> 10451, 10466, NA, NA, NA, NA, 11218...
$ LATITUDE                       <dbl> NA, 40.90144, 40.73070, NA, 40.6679...
$ LONGITUDE                      <dbl> NA, -73.84129, -73.79085, NA, -73.9...
$ LOCATION                       <chr> NA, "(40.90144, -73.841286)", "(40...
$ `ON STREET NAME`               <chr> "EXTERIOR STREET", NA, "179 STREET"...
$ `CROSS STREET NAME`            <chr> "EAST 138 STREET", NA, NA, "OCEAN P...
$ `OFF STREET NAME`              <chr> NA, "4445      SETON AVENUE", NA, N...
$ `NUMBER OF PERSONS INJURED`    <int> 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0,...
$ `NUMBER OF PERSONS KILLED`     <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
$ `NUMBER OF PEDESTRIANS INJURED` <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
```

```
$ `NUMBER OF PEDESTRIANS KILLED`  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
$ `NUMBER OF CYCLIST INJURED`     <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
$ `NUMBER OF CYCLIST KILLED`      <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
$ `NUMBER OF MOTORIST INJURED`    <int> 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0,...
$ `NUMBER OF MOTORIST KILLED`     <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
$ `CONTRIBUTING FACTOR VEHICLE 1` <chr> "Unspecified", "Unspecified", "Fail...
$ `CONTRIBUTING FACTOR VEHICLE 2` <chr> "Unspecified", NA, "Unspecified", "...
$ `CONTRIBUTING FACTOR VEHICLE 3` <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA,...
$ `CONTRIBUTING FACTOR VEHICLE 4` <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA,...
$ `CONTRIBUTING FACTOR VEHICLE 5` <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA,...
$ `UNIQUE KEY`                    <int> 3845795, 3845844, 3845388, 3845406,...
$ `VEHICLE TYPE CODE 1`           <chr> "PASSENGER VEHICLE", "PASSENGER VEH...
$ `VEHICLE TYPE CODE 2`           <chr> "SPORT UTILITY / STATION WAGON", NA...
$ `VEHICLE TYPE CODE 3`           <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA,...
$ `VEHICLE TYPE CODE 4`           <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA,...
$ `VEHICLE TYPE CODE 5`           <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA,...
```

```
In [6]: collision=separate(collision,DATE,c("MONTH","DAY","YEAR"),sep="/")
```

```
In [7]: collision=separate(collision,TIME,c("HourOfCollision","Min","Sec"),sep=":")
```

```
In [8]: collision = subset(collision, select = -c(Min,Sec) )
```

```
In [9]: collision$YEAR<-as.factor(collision$YEAR)
```

```
In [10]: collision$MONTH<-as.factor(collision$MONTH)
```

```
In [11]: collision$DAY<-as.factor(collision$DAY)
```

```
In [12]: collision$HourOfCollision<-as.factor(collision$HourOfCollision)
```

```
In [13]: collision$BOROUGH<-as.factor(collision$BOROUGH)
```

```
In [14]: collision$LOCATION<-as.factor(collision$LOCATION)
```

```
In [15]: summary(collision)
```

```
     MONTH              DAY            YEAR        HourOfCollision
 10     :112321   13     : 41053   2012:100537   16     : 90325
 07     :110152   18     : 40761   2013:203721   17     : 87684
 12     :109366   07     : 40712   2014:206020   14     : 82798
 09     :108721   06     : 40621   2015:217658   18     : 76953
 08     :108583   17     : 40585   2016:227808   15     : 75263
 11     :107471   21     : 40498   2017:229233   13     : 71839
 (Other):553333   (Other):965717   2018: 24970   (Other):725085
          BOROUGH          ZIP CODE         LATITUDE        LONGITUDE
 BRONX         :115547   Min.   :10000    Min.   : 0.00    Min.   :-201.36
 BROOKLYN      :266311   1st Qu.:10128    1st Qu.:40.67    1st Qu.: -73.98
 MANHATTAN     :217704   Median :11205    Median :40.72    Median : -73.93
```

```
QUEENS       :226469   Mean   :10814    Mean   :40.72    Mean   : -73.92
STATEN ISLAND: 39760   3rd Qu.:11236    3rd Qu.:40.77    3rd Qu.: -73.87
NA's         :344156   Max.   :11697    Max.   :41.13    Max.   :   0.00
                       NA's   :344292   NA's   :214958   NA's   :214958
                     LOCATION      ON STREET NAME     CROSS STREET NAME
(40.6960346, -73.9845292):   673   Length:1209947     Length:1209947
(40.7606005, -73.9643142):   540   Class :character   Class :character
(40.7572323, -73.9897922):   485   Mode  :character   Mode  :character
(40.6757357, -73.8968533):   479
(40.6585778, -73.8906229):   464
(Other)                  :992348
NA's                     :214958
OFF STREET NAME    NUMBER OF PERSONS INJURED NUMBER OF PERSONS KILLED
Length:1209947     Min.   : 0.0000           Min.   :0.000000
Class :character   1st Qu.: 0.0000           1st Qu.:0.000000
Mode  :character   Median : 0.0000           Median :0.000000
                   Mean   : 0.2564           Mean   :0.001193
                   3rd Qu.: 0.0000           3rd Qu.:0.000000
                   Max.   :43.0000           Max.   :8.000000

NUMBER OF PEDESTRIANS INJURED NUMBER OF PEDESTRIANS KILLED
Min.   : 0.0000               Min.   :0.000000
1st Qu.: 0.0000               1st Qu.:0.000000
Median : 0.0000               Median :0.000000
Mean   : 0.0522               Mean   :0.000658
3rd Qu.: 0.0000               3rd Qu.:0.000000
Max.   :27.0000               Max.   :8.000000

NUMBER OF CYCLIST INJURED NUMBER OF CYCLIST KILLED NUMBER OF MOTORIST INJURED
Min.   :0.00000           Min.   :0.00e+00         Min.   : 0.0000
1st Qu.:0.00000           1st Qu.:0.00e+00         1st Qu.: 0.0000
Median :0.00000           Median :0.00e+00         Median : 0.0000
Mean   :0.02036           Mean   :8.35e-05         Mean   : 0.1852
3rd Qu.:0.00000           3rd Qu.:0.00e+00         3rd Qu.: 0.0000
Max.   :4.00000           Max.   :2.00e+00         Max.   :43.0000

NUMBER OF MOTORIST KILLED CONTRIBUTING FACTOR VEHICLE 1
Min.   :0.000000          Length:1209947
1st Qu.:0.000000          Class :character
Median :0.000000          Mode  :character
Mean   :0.000454
3rd Qu.:0.000000
Max.   :5.000000

CONTRIBUTING FACTOR VEHICLE 2 CONTRIBUTING FACTOR VEHICLE 3
Length:1209947                Length:1209947
Class :character              Class :character
Mode  :character              Mode  :character
```

```
 CONTRIBUTING FACTOR VEHICLE 4 CONTRIBUTING FACTOR VEHICLE 5   UNIQUE KEY
 Length:1209947                Length:1209947                Min.   :     22
 Class :character              Class :character              1st Qu.: 304716
 Mode  :character              Mode  :character              Median :3241094
                                                             Mean   :2358810
                                                             3rd Qu.:3543638
                                                             Max.   :3847223

 VEHICLE TYPE CODE 1 VEHICLE TYPE CODE 2 VEHICLE TYPE CODE 3
 Length:1209947      Length:1209947      Length:1209947
 Class :character    Class :character    Class :character
 Mode  :character    Mode  :character    Mode  :character




 VEHICLE TYPE CODE 4 VEHICLE TYPE CODE 5
 Length:1209947      Length:1209947
 Class :character    Class :character
 Mode  :character    Mode  :character
```

### 1.1 Comparing Collision Occourances Through the Years

```
In [16]: table(collision$YEAR)/nrow(collision)
```

```
      2012       2013       2014       2015       2016       2017       2018
0.08309207 0.16837184 0.17027192 0.17989052 0.18827932 0.18945706 0.02063727
```

```
In [17]: CollisionByYear = collision %>% count(YEAR)
```

```
In [18]: YEARplot = CollisionByYear %>%
             plot_ly(labels=~YEAR, values=~n) %>%
             add_pie(hole = 0.4) %>%
             layout(title = "Donut Chart for Collisions per Year",  showlegend = T,
                    xaxis = list(showgrid = FALSE, zeroline = FALSE, showticklabels = FALSE),
                    yaxis = list(showgrid = FALSE, zeroline = FALSE, showticklabels = FALSE))
```

```
In [19]: YEARplot
```

```
HTML widgets cannot be represented in plain text (need html)
```

2017 saw the highest collision rate of 18.94%. We have lesser collision in 2018 at 2%. However, this may be because we have just covered two months yet.

### 1.2 Number of persons killed vs Number of people Injured

```
In [20]: collision$"NUMBER OF PERSONS KILLED"<-as.numeric(collision$"NUMBER OF PERSONS KILLED")
```

```
In [21]: table(collision$"NUMBER OF PERSONS KILLED",collision$BOROUGH)
```

```
   BRONX BROOKLYN MANHATTAN QUEENS STATEN ISLAND
0 115416   266011    217530 226210         39707
1    130      294       173    246            51
2      1        4         0      9             2
3      0        2         0      2             0
4      0        0         0      1             0
5      0        0         0      1             0
8      0        0         1      0             0
```

```
In [22]: which.max(collision$"NUMBER OF PERSONS KILLED")
```

64321

```
In [23]: collision[64321,]
```

| MONTH | DAY | YEAR | HourOfCollision | BOROUGH | ZIP CODE | LATITUDE | LONGITUDE |
|---|---|---|---|---|---|---|---|
| 10 | 31 | 2017 | 15 | MANHATTAN | 10014 | 40.72905 | -74.01073 |

**The deadliest collision (based on the number of persons killed) took place on 31st October 2017 in Manhattan.**

**This collision also took place in the evening hours.**

```
In [24]: #table(collision$LOCATION,collision$"NUMBER OF PERSONS KILLED")
```

```
In [25]: casualties = collision %>%
             group_by(BOROUGH) %>%
             summarise(TotalNumberOfPedestriansKilled=sum(`NUMBER OF PEDESTRIANS KILLED`),
                       TotalNumberOfCyclistKilled=sum(`NUMBER OF CYCLIST KILLED`),
                       TotalNumberOfMotoristKilled=sum(`NUMBER OF MOTORIST KILLED`))
```

```
In [26]: casualties
```

| BOROUGH | TotalNumberOfPedestriansKilled | TotalNumberOfCyclistKilled | TotalNumberOfMo |
|---|---|---|---|
| BRONX | 77 | 9 | 46 |
| BROOKLYN | 177 | 30 | 101 |
| MANHATTAN | 140 | 20 | 23 |
| QUEENS | 160 | 20 | 99 |
| STATEN ISLAND | 23 | 2 | 31 |
| NA | 219 | 20 | 249 |

```
In [27]: plotTwo = plot_ly(casualties,x=~BOROUGH,y=~TotalNumberOfPedestriansKilled,type = 'bar
                           name='Pedestrians Killed') %>%
                           add_trace(y=~TotalNumberOfCyclistKilled,name='Cyclist Killed')%>%
                          add_trace(y=~TotalNumberOfMotoristKilled,name='Motorist Killed')%>%
           layout(yaxis = list(title = 'Count'), barmode = 'stack')
```

```
In [28]: plotTwo
```

```
Warning message:
Ignoring 1 observationsWarning message:
Ignoring 1 observationsWarning message:
Ignoring 1 observations
```

```
HTML widgets cannot be represented in plain text (need html)
```

As per the data, brooklyn turns out to be the area where the most collisions occour. Staten Island turns out to be a borough not many collisions happen as compared to the rest.

Also, looking at the graph, I see that cyclists are the safest on the road. *However, we cannot rule out the fact that there are less number of cyclists on the road and hence less collisions(maybe).*

**It is safer to be a cyclist in Staten Island than it is to be in any other borough.**

However, it is pedestrians who are the least safe in case of collisions.

```
In [29]: injured = collision %>%
             group_by(BOROUGH) %>%
             summarise(TotalNumberOfPedestriansInjured=sum(`NUMBER OF PEDESTRIANS INJURED`),
                       TotalNumberOfCyclistInjured=sum(`NUMBER OF CYCLIST INJURED`),
                       TotalNumberOfMotoristInjured=sum(`NUMBER OF MOTORIST INJURED`))
```

```
In [30]: injured
```

| BOROUGH | TotalNumberOfPedestriansInjured | TotalNumberOfCyclistInjured | TotalNumberOfM |
|---|---|---|---|
| BRONX | 8227 | 1871 | 23042 |
| BROOKLYN | 17918 | 8175 | 51755 |
| MANHATTAN | 13233 | 6199 | 18016 |
| QUEENS | 12138 | 4071 | 42766 |
| STATEN ISLAND | 1499 | 224 | 7959 |
| NA | 10149 | 4088 | 80532 |

```
In [31]: plotThree = plot_ly(injured,x=~BOROUGH,y=~TotalNumberOfPedestriansInjured,type = 'bar
                           name='Pedestrians Injured') %>%
                           add_trace(y=~TotalNumberOfCyclistInjured,name='Cyclist Injured')%>%
                          add_trace(y=~TotalNumberOfMotoristInjured,name='Motorist Injured')%>%
           layout(yaxis = list(title = 'Count'), barmode = 'stack')
```

```
In [41]: print(plotThree)
```

```
Warning message:
Ignoring 1 observationsWarning message:
Ignoring 1 observationsWarning message:
Ignoring 1 observations
```

**Motorists are the highest injured, even though do not ket killed in collisions as compared to the pedestrians.Also, it is the least safe to drive a motorbike in Brooklyn.**

### 1.3 Analysis 3: Finding hotspots vs the hour of collision

```
In [33]: sort(table(collision$HourOfCollision))
```

```
   03    02    04    05    01    06    23    00    07    22    21    20    10
12431 14531 14533 15844 19017 24841 31334 32519 33923 38620 42809 51983 62199
   19    11    09    12    08    13    15    18    14    17    16
62676 64204 67397 67965 68259 71839 75263 76953 82798 87684 90325
```

We observer that the maximum collisions take place during the evening hours and the least during late nights.

```
In [34]: which.max(collision$LOCATION)
```

125837

```
In [35]: collision[125837, ]
```

| MONTH | DAY | YEAR | HourOfCollision | BOROUGH | ZIP CODE | LATITUDE | LONGITUDE | LO |
|---|---|---|---|---|---|---|---|---|
| 07 | 26 | 2017 | 16 | BROOKLYN | 11239 | 41.12615 | -73.71353 | (4 |

```
In [36]: table(collision$HourOfCollision,collision$BOROUGH)/nrow(collision)
```

```
          BRONX     BROOKLYN    MANHATTAN       QUEENS STATEN ISLAND
00 0.0024455617 0.0054465196 0.0057465327 0.0042654761  0.0006107706
01 0.0014322941 0.0031869165 0.0034753588 0.0024513470  0.0003578669
02 0.0010917834 0.0024306850 0.0026513558 0.0018802476  0.0003305930
03 0.0009421900 0.0019992611 0.0021893521 0.0017827227  0.0002752187
04 0.0011917877 0.0023463838 0.0021711695 0.0022447264  0.0002520772
05 0.0012116233 0.0025042419 0.0020662062 0.0025472190  0.0003190222
06 0.0018480148 0.0040224902 0.0029844282 0.0038315728  0.0005620081
07 0.0028935152 0.0057523181 0.0038381846 0.0053192413  0.0008950805
08 0.0062283720 0.0122385526 0.0080408481 0.0120468087  0.0020190967
09 0.0055142911 0.0125418717 0.0096243885 0.0107277426  0.0017513164
10 0.0048093016 0.0117806813 0.0100938306 0.0093847086  0.0016504855
11 0.0048729407 0.0120616853 0.0106409620 0.0097756348  0.0017323073
12 0.0051035293 0.0128567615 0.0110194909 0.0103764876  0.0019587635
13 0.0054911496 0.0137386183 0.0113319013 0.0111632989  0.0020595943
14 0.0064424309 0.0153981951 0.0124782325 0.0131146240  0.0024852328
15 0.0056324781 0.0142452521 0.0103525196 0.0117682841  0.0023348130
16 0.0073755297 0.0167098228 0.0124666618 0.0138055634  0.0027711958
17 0.0069879094 0.0160692989 0.0118467999 0.0137741570  0.0026075522
18 0.0061002672 0.0140171429 0.0108889067 0.0122691324  0.0022067082
19 0.0050241870 0.0111616459 0.0094185944 0.0098467123  0.0017463575
20 0.0040968737 0.0095343019 0.0081185374 0.0081251493  0.0013240249
21 0.0034629616 0.0078598484 0.0067796358 0.0064821021  0.0010694683
22 0.0030282318 0.0068664165 0.0063159791 0.0058134778  0.0008521034
23 0.0022703474 0.0053324650 0.0053886658 0.0043762247  0.0006892864
```

Brooklyn observes highest relative collision rate at 0.0167098228 at 4:00 PM in the evening.

```
In [37]: tapply(collision$"NUMBER OF PEDESTRIANS KILLED",collision$BOROUGH,mean)
```

```
ERROR while rich displaying an object: Error in dn[[2L]]: subscript out of bounds

Traceback:
1. FUN(X[[i]], ...)
2. tryCatch(withCallingHandlers({
 .     rpr <- mime2repr[[mime]](obj)
 .     if (is.null(rpr))
 .         return(NULL)
 .     prepare_content(is.raw(rpr), rpr)
 . }, error = error_handler), error = outer_handler)
3. tryCatchList(expr, classes, parentenv, handlers)
4. tryCatchOne(expr, names, parentenv, handlers[[1L]])
5. doTryCatch(return(expr), name, parentenv, handler)
6. withCallingHandlers({
 .     rpr <- mime2repr[[mime]](obj)
 .     if (is.null(rpr))
 .         return(NULL)
 .     prepare_content(is.raw(rpr), rpr)
 . }, error = error_handler)
7. mime2repr[[mime]](obj)
8. repr_markdown.numeric(obj)
9. repr_vector_generic(html_escape_names(obj), "%s. %s\n", "%s\n:   %s",
 .     "**%s:** %s", "%s\n\n", item_uses_numbers = TRUE, escape_fun = html_escape)
10. html_escape_names(obj)
11. .escape_names(obj, "html")
12. colnames(obj)
ERROR while rich displaying an object: Error in dn[[2L]]: subscript out of bounds

Traceback:
1. FUN(X[[i]], ...)
2. tryCatch(withCallingHandlers({
 .     rpr <- mime2repr[[mime]](obj)
 .     if (is.null(rpr))
 .         return(NULL)
 .     prepare_content(is.raw(rpr), rpr)
 . }, error = error_handler), error = outer_handler)
3. tryCatchList(expr, classes, parentenv, handlers)
4. tryCatchOne(expr, names, parentenv, handlers[[1L]])
5. doTryCatch(return(expr), name, parentenv, handler)
6. withCallingHandlers({
 .     rpr <- mime2repr[[mime]](obj)
 .     if (is.null(rpr))
 .         return(NULL)
 .     prepare_content(is.raw(rpr), rpr)
```

```
 . }, error = error_handler)
7. mime2repr[[mime]](obj)
8. repr_latex.numeric(obj)
9. repr_vector_generic(latex_escape_names(obj), "\\item %s\n", "\\item[%s] %s\n",
 .     "\\textbf{%s:} %s", enum_wrap = "\\begin{enumerate*}\n%s\\end{enumerate*}\n",
 .     named_wrap = "\\begin{description*}\n%s\\end{description*}\n",
 .     only_named_item = "\\textbf{%s:} %s", escape_fun = latex_escape)
10. latex_escape_names(obj)
11. .escape_names(obj, "latex")
12. colnames(obj)
```

```
       BRONX      BROOKLYN     MANHATTAN        QUEENS STATEN ISLAND
0.0006663955  0.0006646365  0.0006430750  0.0007064985  0.0005784708
```

```
In [38]: BoroughVCollision = collision %>%
             count(HourOfCollision,BOROUGH)

In [42]: plotFour = plot_ly(BoroughVCollision, x=~BOROUGH,y=~HourOfCollision,z=~n,
                 colors = colorRamp(c("green", "red")),type="heatmap")

In [43]: embed_notebook(plotFour)
```