NYPD shooting incident data - Analyze / Visualize / Model

Kothandaraman Sikamani

12/02/2022

NYPD Shooting Incident Data (Historic) Analyse, Visualize, and prepare model.

Purpose

I will be focusing on Data Visualization, Model of NYPD Shooting Incident in this project as follows:

1. **NYPD Shooting Incidents.**
   - Data Source
   - Importing the data
   - Summarizing NYPD Shooting Incident historic raw data
2. **Tidying and Transforming Data.**
   - Tidy data and Transform
3. **Visualization and Analysis of NYPD Shooting Incidents data.**
   - Visualizing NYPD Shoot Incidents and Deaths
     - Visualizing NYPD Shoot Incidents by Jurisdiction - Pie/Coxcomb/Bar chart
     - Visualizing NYPD Shoot Incidents by Yearly - Bar/Scatter/Pie/Coxcomb
     - Visualizing NYPD Death by Yearly - Bar/Scatter/Pie/Coxcomb/Multiple-Pie/Interactive chart
   - Top 5 Incidents happened in a day / Maximum Incidents in a day
   - Top 5 Deaths happened in a day / Maximum Deaths in a day
   - Maximum shooting incidents by yearly
   - Maximum death incidents by yearly
4. **Model.**
   - Jurisdiction yearly frequency of Model
5. **Bias.**
   - Explained about bias details for better model performance and model prediction

1. NYPD Shooting Incidents:

1.1 Data Source

I am using the data source from NYPD Shooting Incident Data (Historic), Taken historic data from 2006 to 2020. This data set is in .csv format. https://catalog.data.gov/dataset/nypd-shooting-incident-data-historic

1.2 Import the data

I will read the .csv file using read.csv().

```
shoot_historic <- read.csv("https://data.cityofnewyork.us/api/views/833y-fsy8/rows.csv")
```

1.2.1 Finding Total Incidents

```
paste("The total number of Incidents :", nrow(shoot_historic))
```

```
## [1] "The total number of Incidents : 23585"
```

1.3 Summarizing NYPD Shooting Incident historic raw data

```
summary(shoot_historic)
```

```
##   INCIDENT_KEY        OCCUR_DATE         OCCUR_TIME             BORO
##  Min.   :  9953245   Length:23585       Length:23585       Length:23585
##  1st Qu.: 55322804   Class :character   Class :character   Class :character
##  Median : 83435362   Mode  :character   Mode  :character   Mode  :character
##  Mean   :102280741
##  3rd Qu.:150911774
##  Max.   :230611229
##
##     PRECINCT      JURISDICTION_CODE LOCATION_DESC      STATISTICAL_MURDER_FLAG
##  Min.   :  1.00   Min.   :0.000     Length:23585       Length:23585
##  1st Qu.: 44.00   1st Qu.:0.000     Class :character   Class :character
##  Median : 69.00   Median :0.000     Mode  :character   Mode  :character
##  Mean   : 66.21   Mean   :0.333
##  3rd Qu.: 81.00   3rd Qu.:0.000
##  Max.   :123.00   Max.   :2.000
##                   NA’s   :2
##  PERP_AGE_GROUP       PERP_SEX          PERP_RACE         VIC_AGE_GROUP
##  Length:23585       Length:23585       Length:23585       Length:23585
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    VIC_SEX            VIC_RACE           X_COORD_CD        Y_COORD_CD
##  Length:23585       Length:23585       Min.   : 914928   Min.   :125757
##  Class :character   Class :character   1st Qu.: 999925   1st Qu.:182539
##  Mode  :character   Mode  :character   Median :1007654   Median :193470
##                                        Mean   :1009379   Mean   :207300
##                                        3rd Qu.:1016782   3rd Qu.:239163
##                                        Max.   :1066815   Max.   :271128
##
##     Latitude       Longitude        Lon_Lat
##  Min.   :40.51   Min.   :-74.25   Length:23585
##  1st Qu.:40.67   1st Qu.:-73.94   Class :character
##  Median :40.70   Median :-73.92   Mode  :character
##  Mean   :40.74   Mean   :-73.91
##  3rd Qu.:40.82   3rd Qu.:-73.88
##  Max.   :40.91   Max.   :-73.70
##
```

```
#head(shoot_historic , 2)
#data.table(shoot_historic)
#spec(shoot_historic)
```

2. Tidying and Transforming Data

```
## Adding INCIDENT_COUNT  and DEATH_COUNT(based on STATISTICAL_MURDER_FLAG )

shoot_historic <- shoot_historic %>%
  mutate(OCCUR_DATE = mdy(OCCUR_DATE)) %>%
  mutate(INCIDENT_COUNT = 1 ) %>%
  mutate(
     DEATH_COUNT = case_when(
          STATISTICAL_MURDER_FLAG == "true" ~ 1,
          STATISTICAL_MURDER_FLAG == "false" ~ 0
     )) %>%
  select(-c(INCIDENT_KEY, OCCUR_TIME, LOCATION_DESC, X_COORD_CD, Y_COORD_CD,
            Latitude, Longitude, Lon_Lat))

shoot_historic_by_JURISDICTION_CODE <- shoot_historic %>%
  group_by (JURISDICTION_CODE)  %>%
  summarize( INCIDENT_COUNT = sum(INCIDENT_COUNT) ) %>%
  select (JURISDICTION_CODE, INCIDENT_COUNT )

shoot_historic_by_JURISDICTION_CODE
```

```
## # A tibble: 4 x 2
##   JURISDICTION_CODE INCIDENT_COUNT
##               <int>          <dbl>
## 1                 0          19629
## 2                 1             54
## 3                 2           3900
## 4                NA              2
```

```
# Cleaning NA data.
shoot_historic_by_JURISDICTION_CODE <- shoot_historic_by_JURISDICTION_CODE %>% filter(!is.na(JURISDICTI

#After removing NA data.
shoot_historic_by_JURISDICTION_CODE
```

```
## # A tibble: 3 x 2
##   JURISDICTION_CODE INCIDENT_COUNT
##                <int>          <dbl>
## 1                  0          19629
## 2                  1             54
## 3                  2           3900
```

```
shoot_historic_By_Date <- shoot_historic %>%
  group_by ( OCCUR_DATE )  %>%
  summarize( INCIDENT_COUNT = sum(INCIDENT_COUNT),
             DEATH_COUNT = sum(DEATH_COUNT)) %>%
  select (OCCUR_DATE , INCIDENT_COUNT , DEATH_COUNT)

shoot_historic_By_Date
```

```
## # A tibble: 5,054 x 3
##    OCCUR_DATE INCIDENT_COUNT DEATH_COUNT
##    <date>              <dbl>       <dbl>
##  1 2006-01-01              8           4
##  2 2006-01-02              4           1
##  3 2006-01-03              4           1
##  4 2006-01-04              4           0
##  5 2006-01-05              4           0
##  6 2006-01-06              4           0
##  7 2006-01-07              2           1
##  8 2006-01-08              4           1
##  9 2006-01-09              9           5
## 10 2006-01-10              5           0
## # ... with 5,044 more rows
```

```
shoot_historic_By_Date_Year <- shoot_historic_By_Date %>%
   mutate(OCCUR_DATE_YEAR = year(OCCUR_DATE))  %>%
   group_by(OCCUR_DATE_YEAR)  %>%
   summarize(INCIDENT_COUNT = sum(INCIDENT_COUNT),
             DEATH_COUNT = sum(DEATH_COUNT) ) %>%
   ungroup()

shoot_historic_By_Date_Year
```

```
## # A tibble: 15 x 3
##    OCCUR_DATE_YEAR INCIDENT_COUNT DEATH_COUNT
##              <dbl>          <dbl>       <dbl>
##  1            2006           2055         445
##  2            2007           1887         373
##  3            2008           1959         362
##  4            2009           1828         348
##  5            2010           1912         405
##  6            2011           1939         373
##  7            2012           1717         288
##  8            2013           1339         223
##  9            2014           1464         249
## 10            2015           1434         283
## 11            2016           1208         223
## 12            2017            970         174
## 13            2018            958         204
## 14            2019            967          184
## 15            2020           1948          366
```

```
shoot_historic_by_VIC_RACE_YEAR <- shoot_historic %>%
   mutate(OCCUR_DATE_YEAR = year(OCCUR_DATE))  %>%
  group_by (VIC_RACE, OCCUR_DATE_YEAR)  %>%
  summarize( INCIDENT_COUNT = sum(INCIDENT_COUNT),
             DEATH_COUNT = sum(DEATH_COUNT) ) %>%
  select (VIC_RACE, OCCUR_DATE_YEAR, INCIDENT_COUNT, DEATH_COUNT)
```

```
## ‘summarise()‘ has grouped output by ’VIC_RACE’. You can override using the
## ‘.groups‘ argument.
```

```
shoot_historic_by_VIC_RACE_YEAR
```

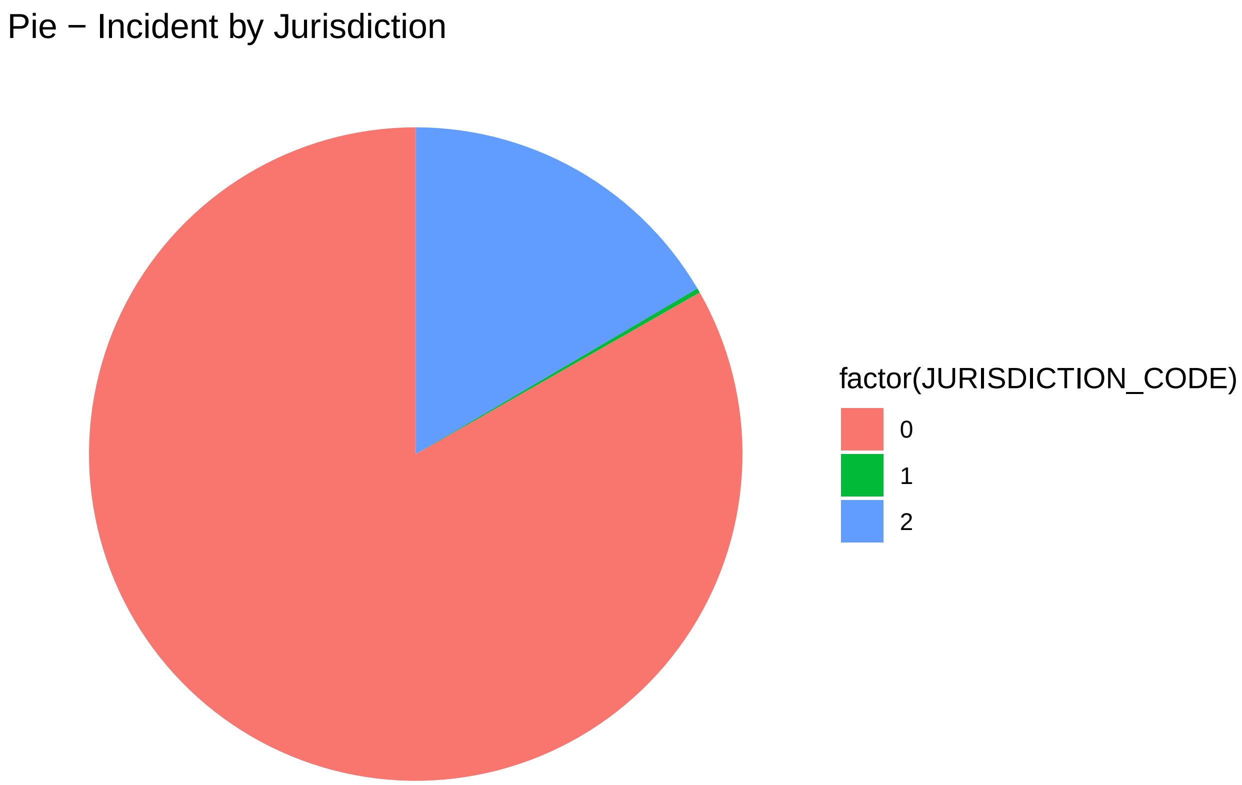
```
## # A tibble: 96 x 4
## # Groups:   VIC_RACE [7]
##    VIC_RACE                       OCCUR_DATE_YEAR INCIDENT_COUNT DEATH_COUNT
##    <chr>                                    <dbl>          <dbl>       <dbl>
##  1 AMERICAN INDIAN/ALASKAN NATIVE            2007              1           0
##  2 AMERICAN INDIAN/ALASKAN NATIVE            2009              2           0
##  3 AMERICAN INDIAN/ALASKAN NATIVE            2010              1           0
##  4 AMERICAN INDIAN/ALASKAN NATIVE            2011              2           0
##  5 AMERICAN INDIAN/ALASKAN NATIVE            2012              1           0
##  6 AMERICAN INDIAN/ALASKAN NATIVE            2016              1           0
##  7 AMERICAN INDIAN/ALASKAN NATIVE            2018              1           0
##  8 ASIAN / PACIFIC ISLANDER                  2006             26           7
##  9 ASIAN / PACIFIC ISLANDER                  2007             18           3
## 10 ASIAN / PACIFIC ISLANDER                  2008             32           5
## # ... with 86 more rows
```

```
# Add tool tip column for plots
shoot_historic_by_VIC_RACE_YEAR <- shoot_historic_by_VIC_RACE_YEAR %>%
   unite("TOOL_TIP_COLS",
          c(OCCUR_DATE_YEAR, VIC_RACE, INCIDENT_COUNT, DEATH_COUNT),
          sep = ", ",
          na.rm = TRUE,
          remove = FALSE)
```

3. Visualization and Analysis of NYPD Shooting Incidents Data:

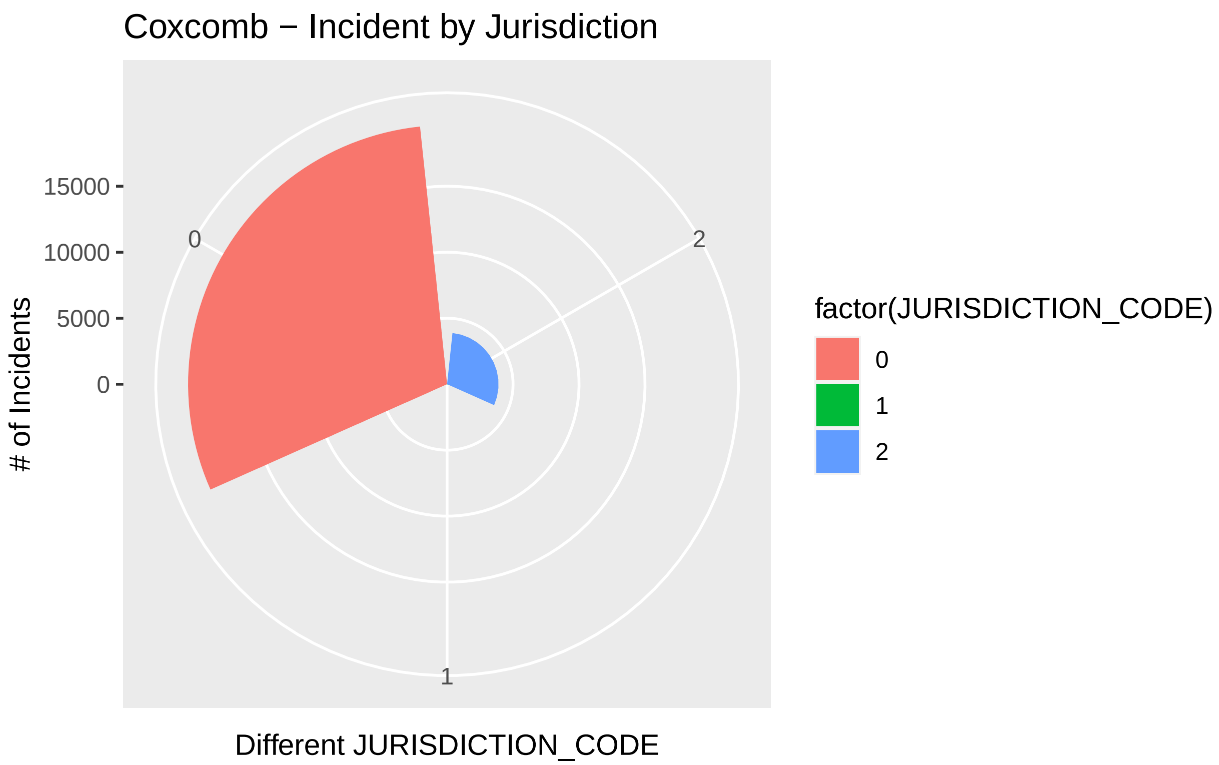
3.1 Visualizing NYPD Shoot Incidents by Jurisdiction: Pie chart

```
# Pie chart
ggplot(shoot_historic_by_JURISDICTION_CODE, aes(x="", y=INCIDENT_COUNT, fill=factor(JURISDICTION_CODE)
  coord_polar("y", start=0) +theme_void() +
  labs(title = "Pie - Incident by Jurisdiction" )
```



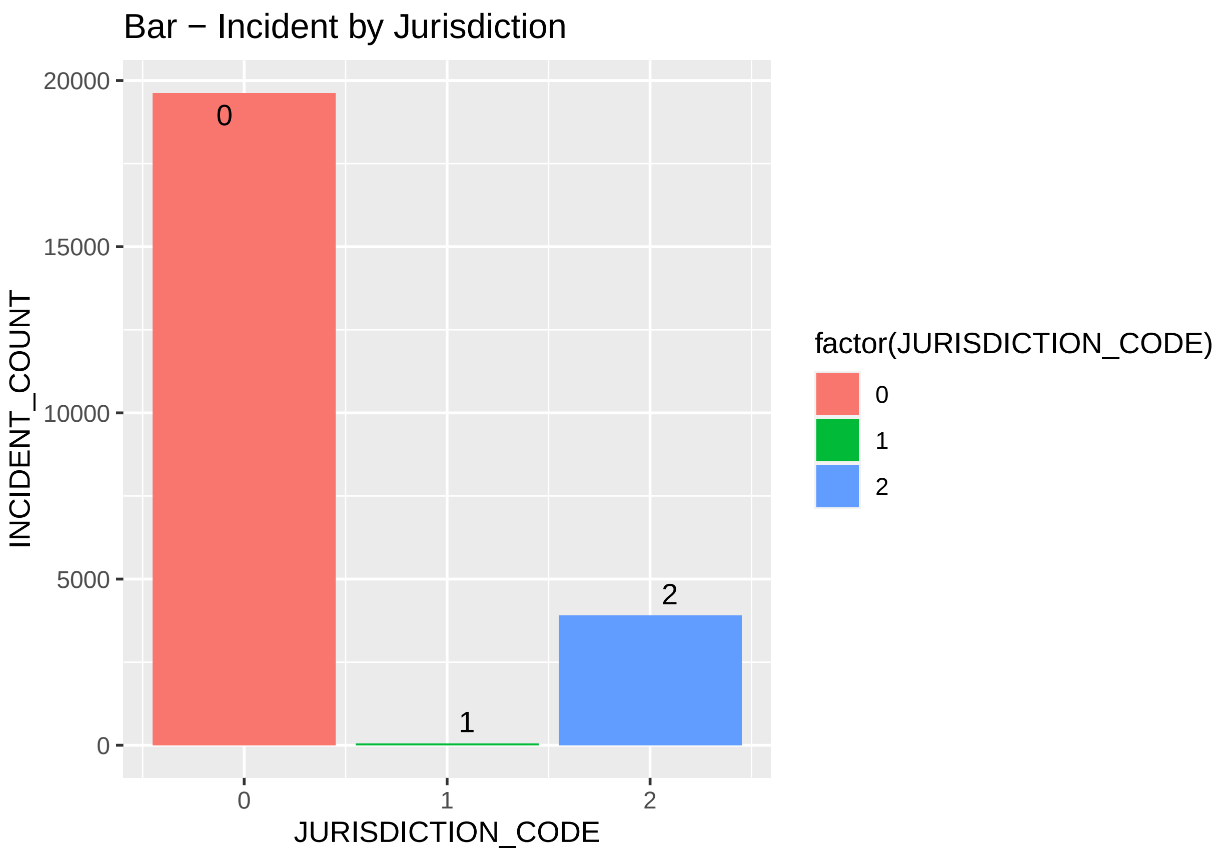
3.2 Visualizing NYPD Shoot Incidents by Jurisdiction: Coxcomb chart

```
# Coxcomb chart
ggplot(shoot_historic_by_JURISDICTION_CODE,
       aes(factor(JURISDICTION_CODE),INCIDENT_COUNT,
           fill=factor(JURISDICTION_CODE))) +
  geom_bar(stat="identity") +
  coord_polar("x", start=0,direction = -1)+
  xlab("Different JURISDICTION_CODE") +
  ylab("# of Incidents") + labs(title = "Coxcomb - Incident by Jurisdiction" )
```



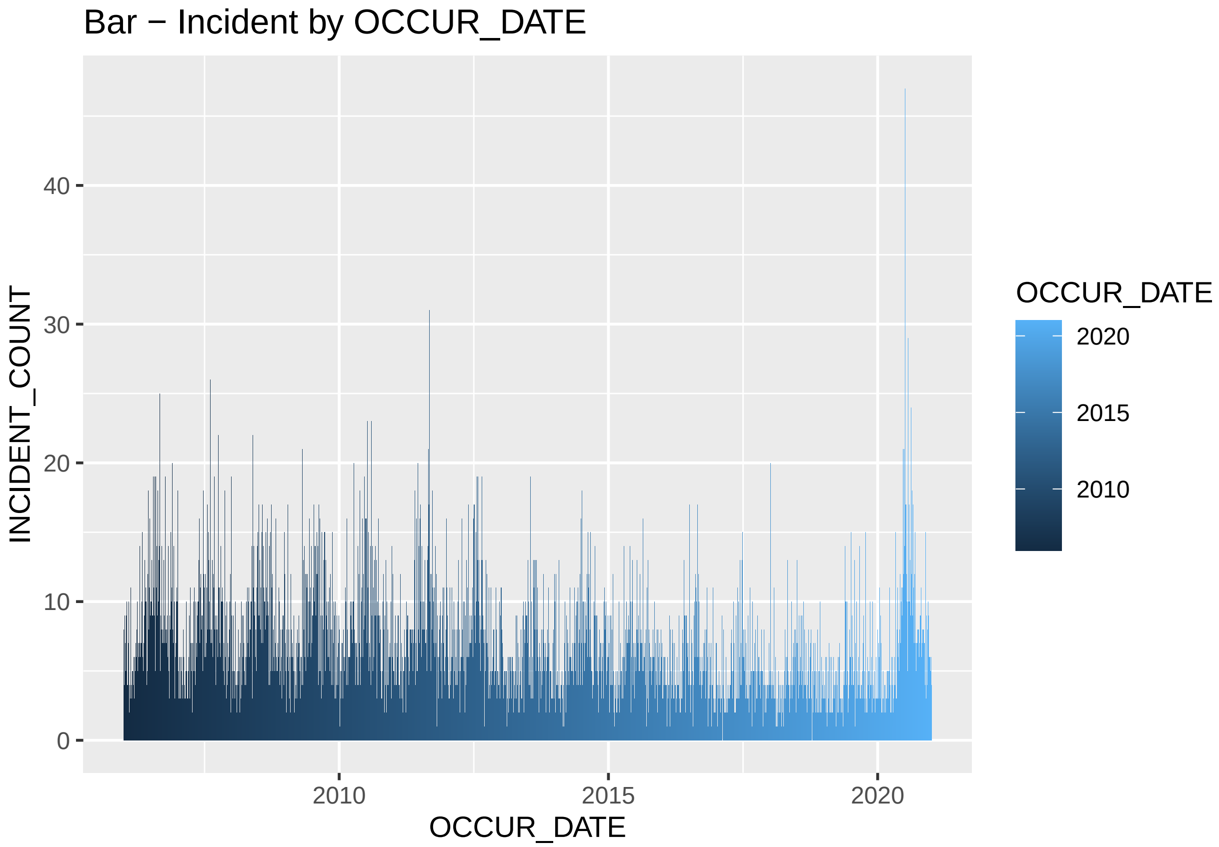
3.2.1 Visualizing NYPD Shoot Incidents by Jurisdiction: Bar chart

```
# Bar chart
ggplot(shoot_historic_by_JURISDICTION_CODE,
       aes(x=JURISDICTION_CODE, y=INCIDENT_COUNT,
           fill=factor(JURISDICTION_CODE))) + geom_bar(stat="identity") + geom_text_repel(data=shoot_his
                                                                                          aes(label=fac
  labs(title = "Bar - Incident by Jurisdiction" )
```



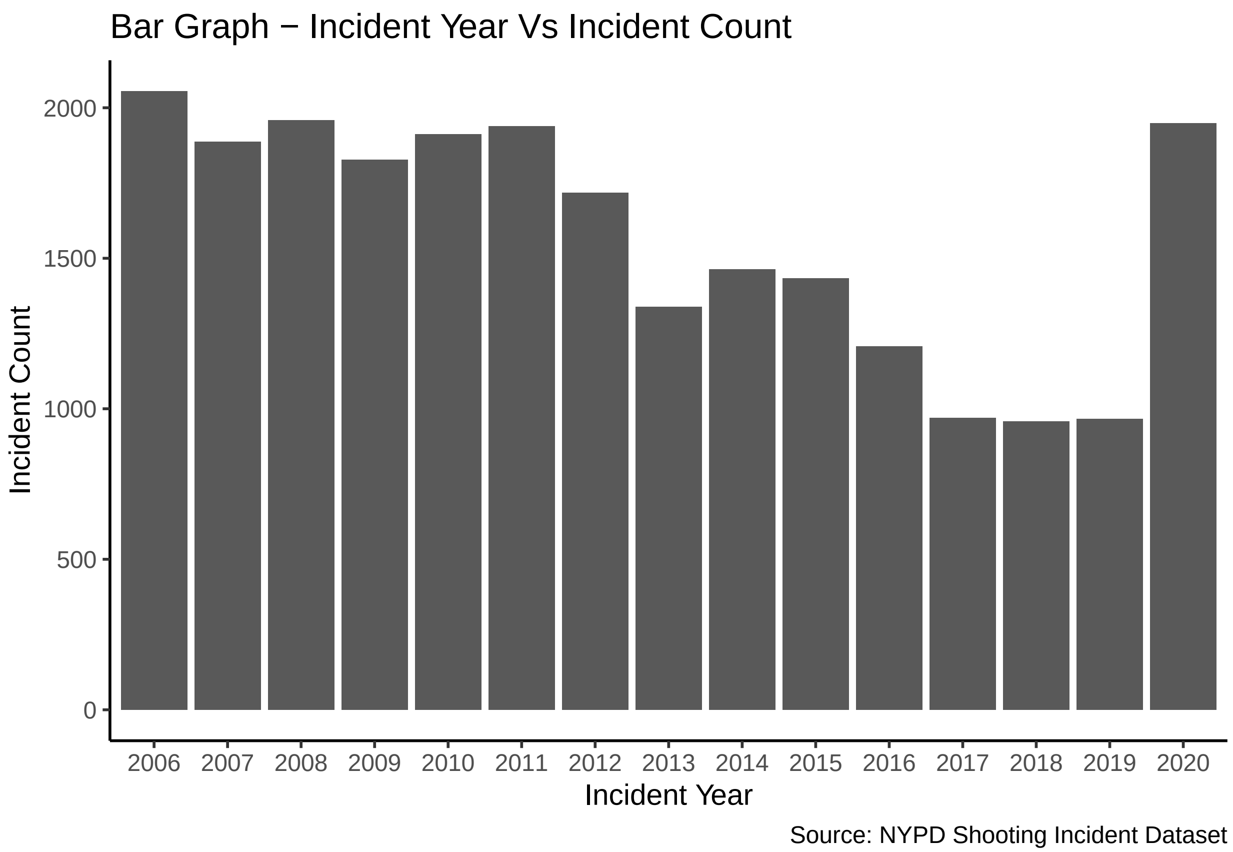
3.3 Visualizing NYPD Shoot Incidents by Date: Bar chart

```
# Bar - Cluttered
ggplot(shoot_historic_By_Date,
       aes(x=OCCUR_DATE, y=INCIDENT_COUNT, fill=OCCUR_DATE )) +
  geom_bar(stat="identity") +
  geom_text_repel(data=shoot_historic_By_Date, aes(label=OCCUR_DATE)) +
  labs(title = "Bar - Incident by OCCUR_DATE" )
```



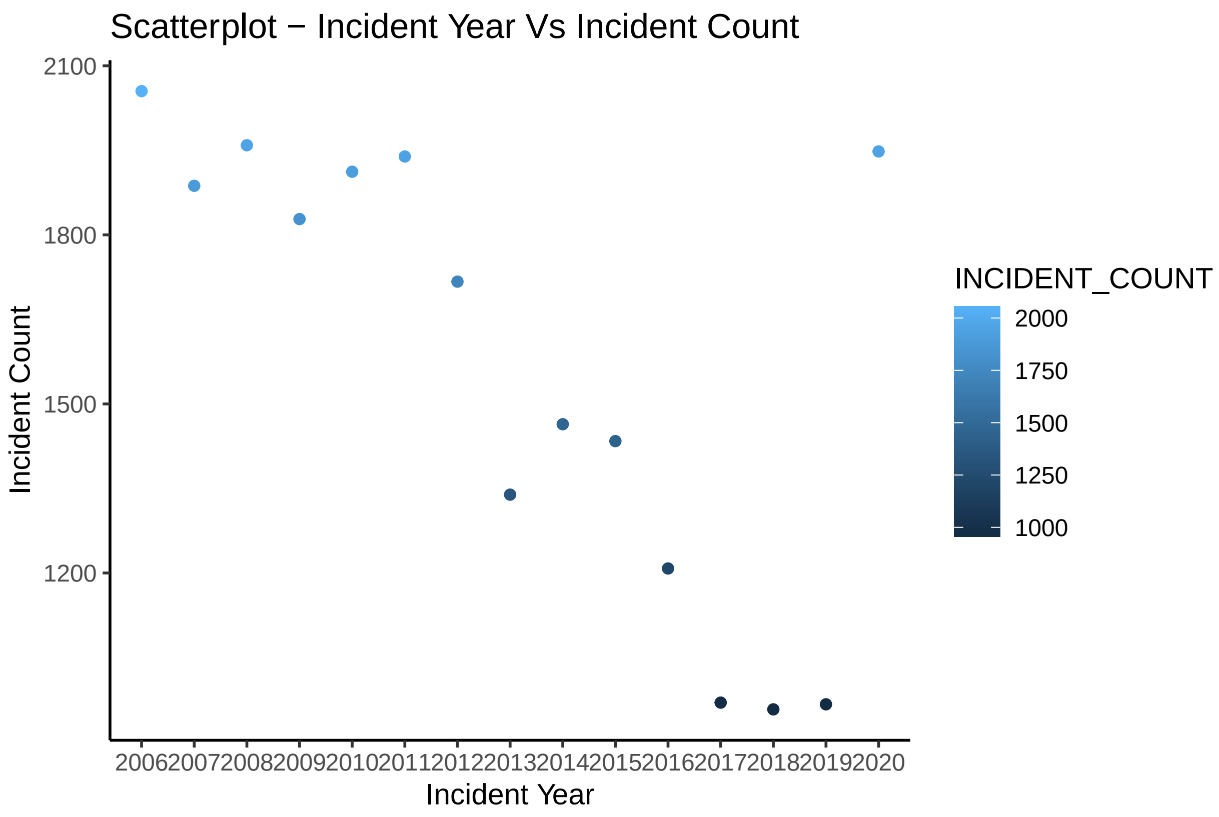
3.4 Visualizing NYPD Shoot Incidents by Yearly: Bar/Scatter/Pie/Coxcomb

```
# Bar Graph
ggplot(data = shoot_historic_By_Date_Year,
       aes(x = factor(OCCUR_DATE_YEAR), y = INCIDENT_COUNT)) +
  geom_bar(stat="identity") + theme_classic() +
labs(title = "Bar Graph - Incident Year Vs Incident Count",
x = "Incident Year",
y = "Incident Count",
caption = "Source: NYPD Shooting Incident Dataset")
```


Bar Graph − Incident Year Vs Incident Count

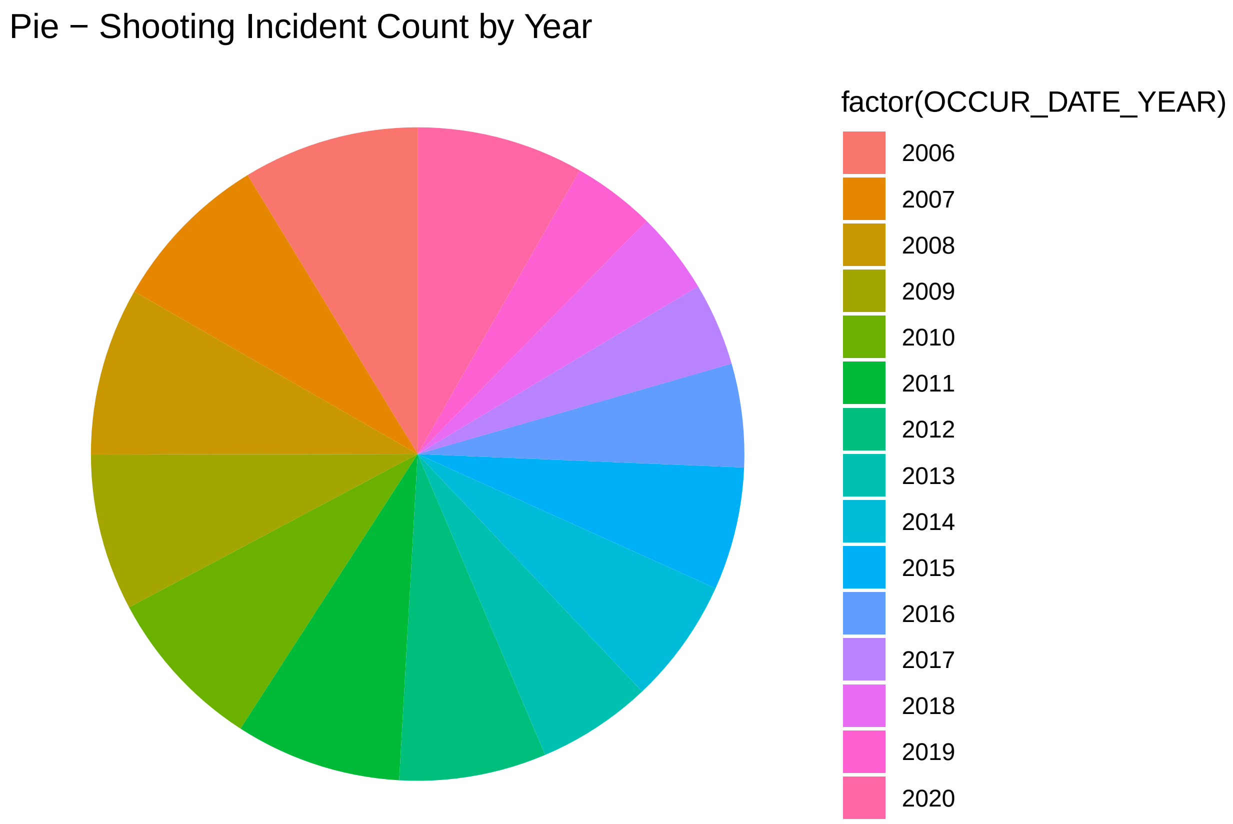
Source: NYPD Shooting Incident Dataset

```
#Scatterplot
ggplot(data = shoot_historic_By_Date_Year,
       aes(x = factor(OCCUR_DATE_YEAR), y = INCIDENT_COUNT)) +
  geom_point(aes(color=INCIDENT_COUNT )) + theme_classic() +
labs(title = "Scatterplot - Incident Year Vs Incident Count",
x = "Incident Year",
y = "Incident Count",
caption = "Source: NYPD Shooting Incident Dataset")
```

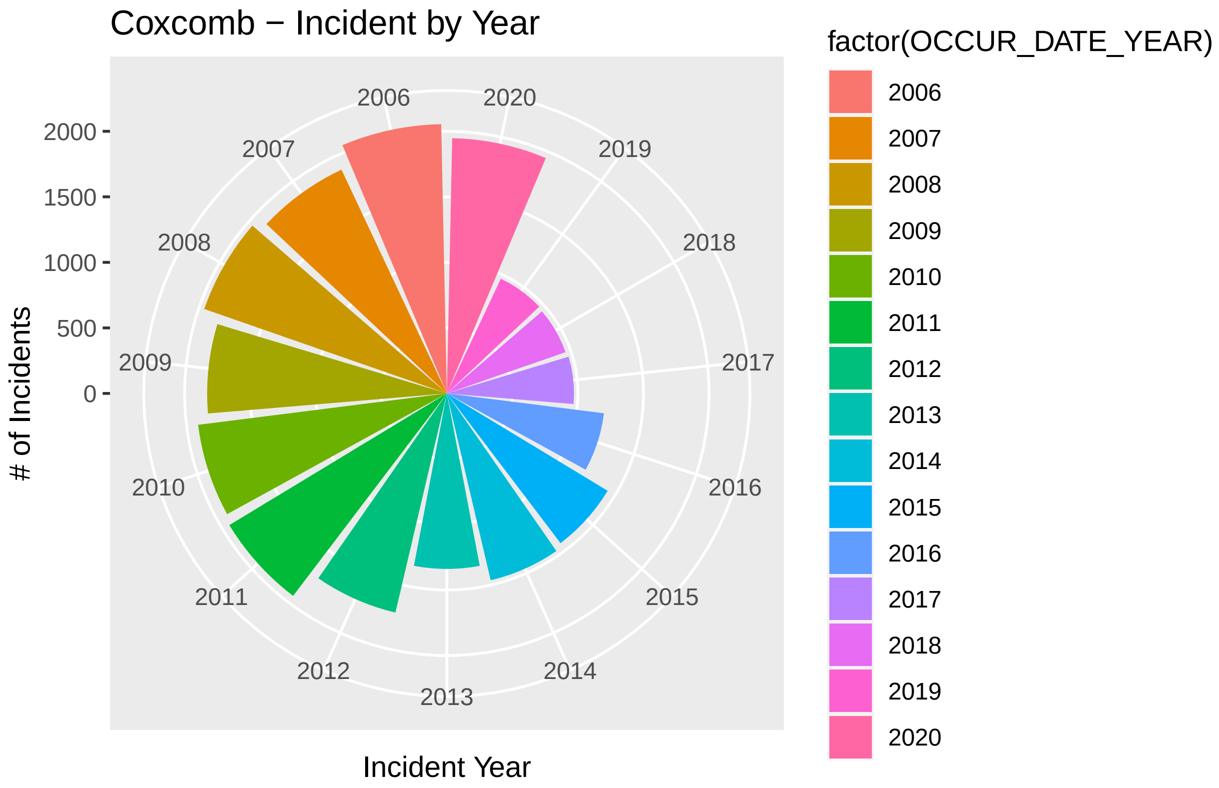


Source: NYPD Shooting Incident Dataset

```
# Pie chart
ggplot(shoot_historic_By_Date_Year,
       aes(x="", y=INCIDENT_COUNT, fill=factor(OCCUR_DATE_YEAR))) +
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y", start=0) +theme_void() +
  labs(title = "Pie - Shooting Incident Count by Year" )
```

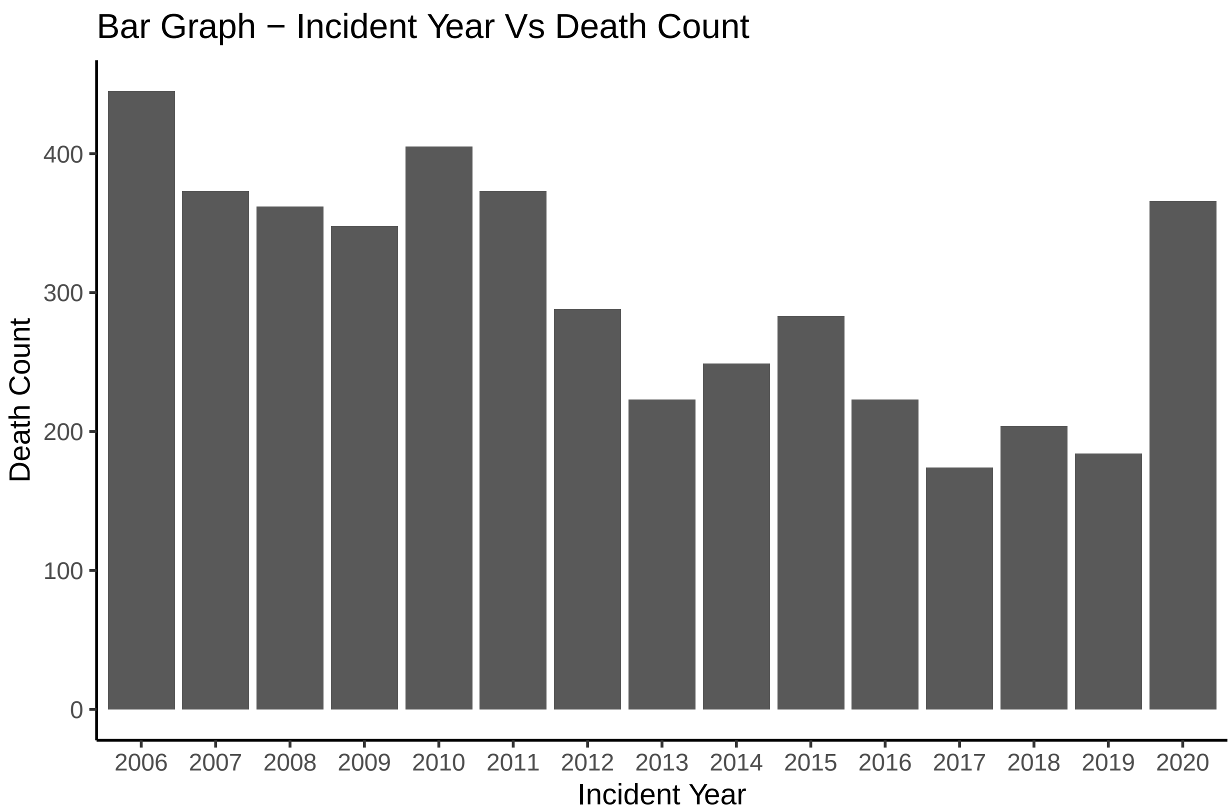


```
# Coxcomb chart
ggplot(shoot_historic_By_Date_Year, aes(factor(OCCUR_DATE_YEAR),INCIDENT_COUNT, fill=factor(OCCUR_DATE_
  geom_bar(stat="identity") +
  coord_polar("x", start=0,direction = -1)+
  xlab("Incident Year") +
  ylab("# of Incidents") + labs(title = "Coxcomb - Incident by Year" )
```



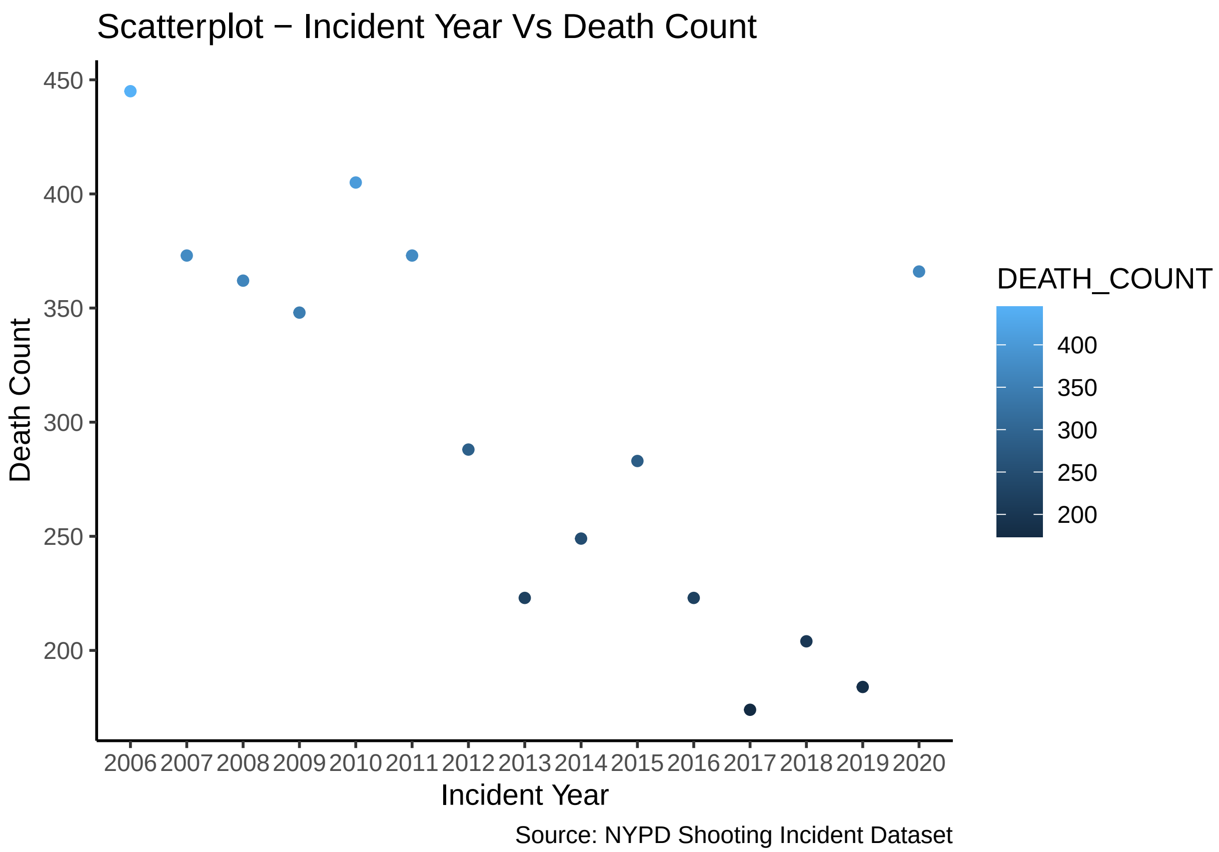
3.5 Visualizing NYPD Shoot Incidents by Yearly Death count: Bar/Scatter/Pie /Coxcomb/Multiple Pie Chart/Interactive chart

```
# Bar chart
ggplot(data = shoot_historic_By_Date_Year,
       aes(x = factor(OCCUR_DATE_YEAR), y = DEATH_COUNT)) +
  geom_bar(stat="identity") + theme_classic() +
labs(title = "Bar Graph - Incident Year Vs Death Count",
x = "Incident Year",
y = "Death Count",
caption = "Source: NYPD Shooting Incident Dataset")
```

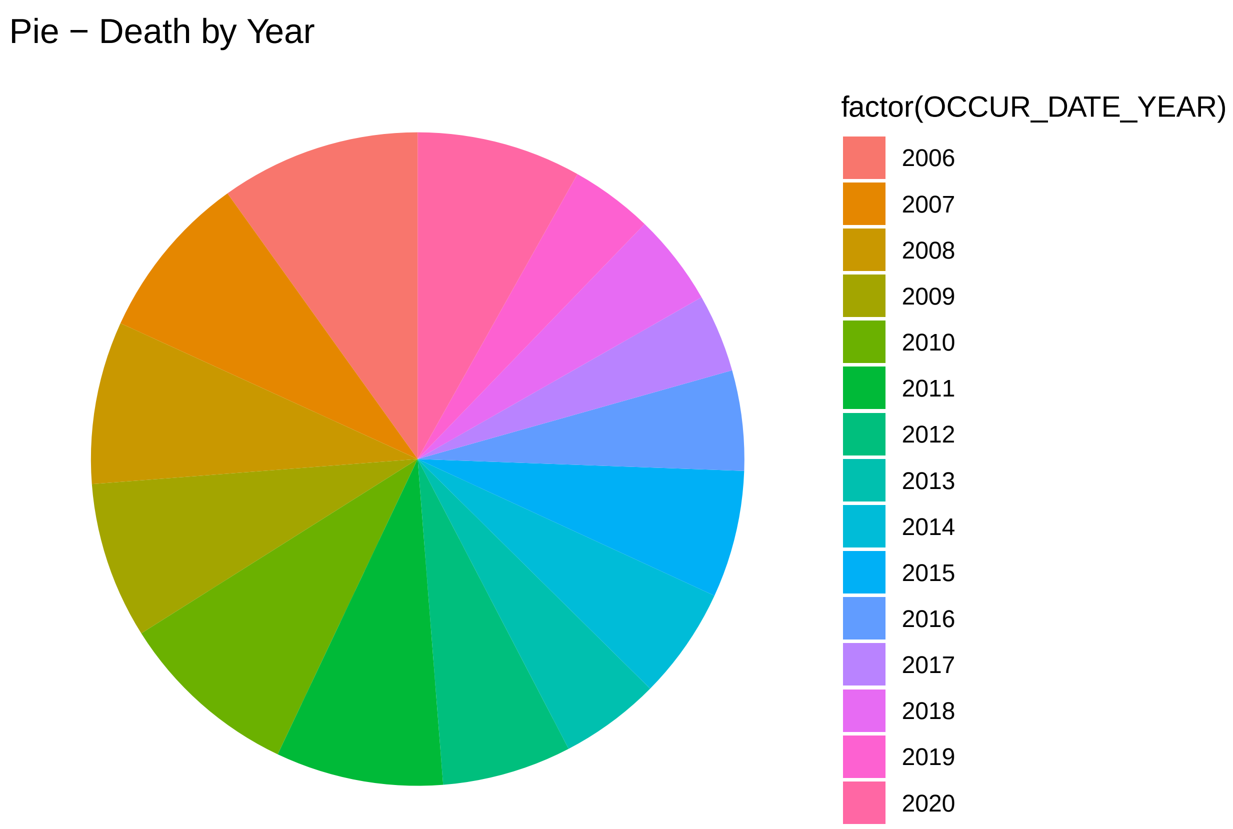


Source: NYPD Shooting Incident Dataset

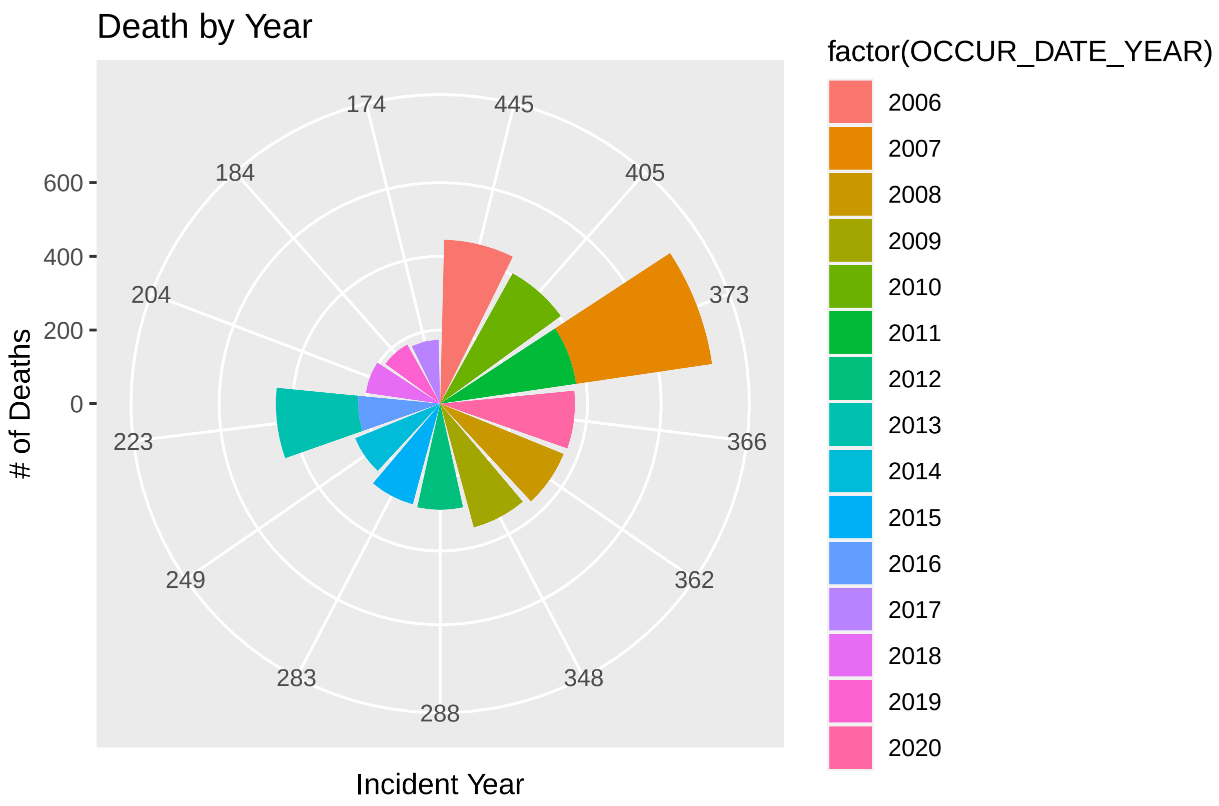
```
# Scatetr Plot
ggplot(data = shoot_historic_By_Date_Year,
       aes(x = factor(OCCUR_DATE_YEAR), y = DEATH_COUNT)) +
  geom_point(aes(color=DEATH_COUNT )) + theme_classic() +
labs(title = "Scatterplot - Incident Year Vs Death Count",
x = "Incident Year",
y = "Death Count",
caption = "Source: NYPD Shooting Incident Dataset")
```



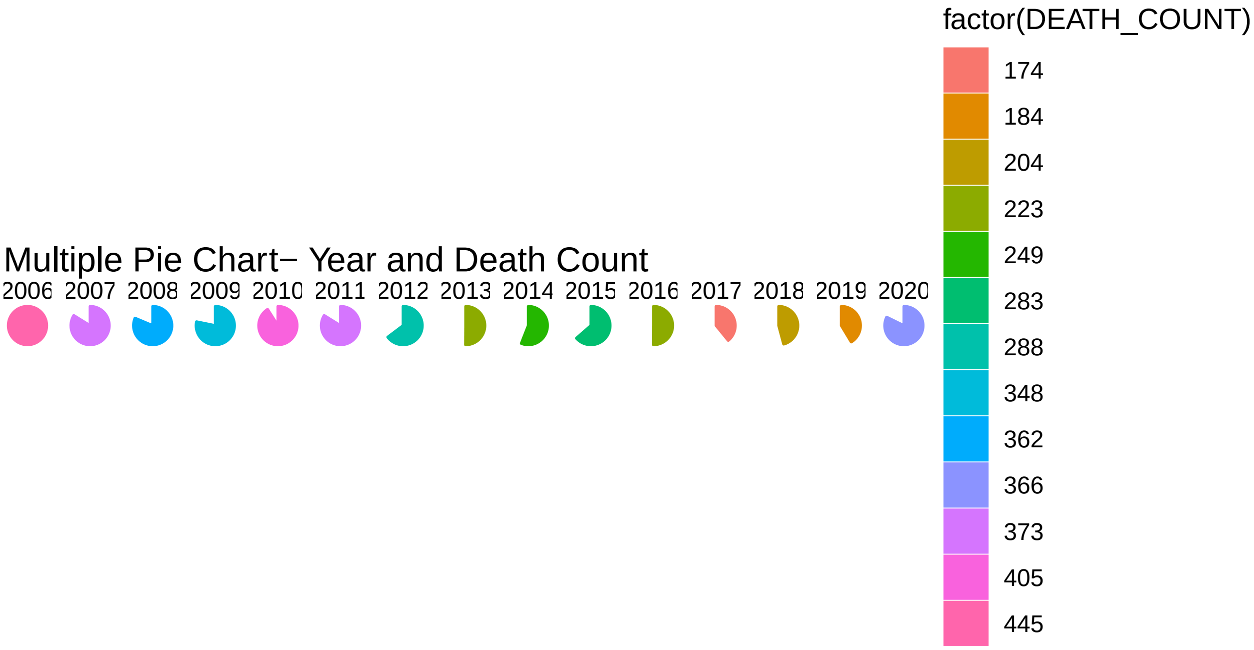
```
# Pie chart
ggplot(shoot_historic_By_Date_Year,
       aes(x="", y=DEATH_COUNT, fill=factor(OCCUR_DATE_YEAR))) +
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y", start=0) +theme_void() + labs(title = "Pie - Death by Year" )
```



```
# Coxcomb chart
ggplot(shoot_historic_By_Date_Year, aes(factor(DEATH_COUNT),DEATH_COUNT, fill=factor(OCCUR_DATE_YEAR)))
  geom_bar(stat="identity") +
  coord_polar("x", start=0,direction = -1)+
  xlab("Incident Year") +
  ylab("# of Deaths") + labs(title = "Death by Year" )
```



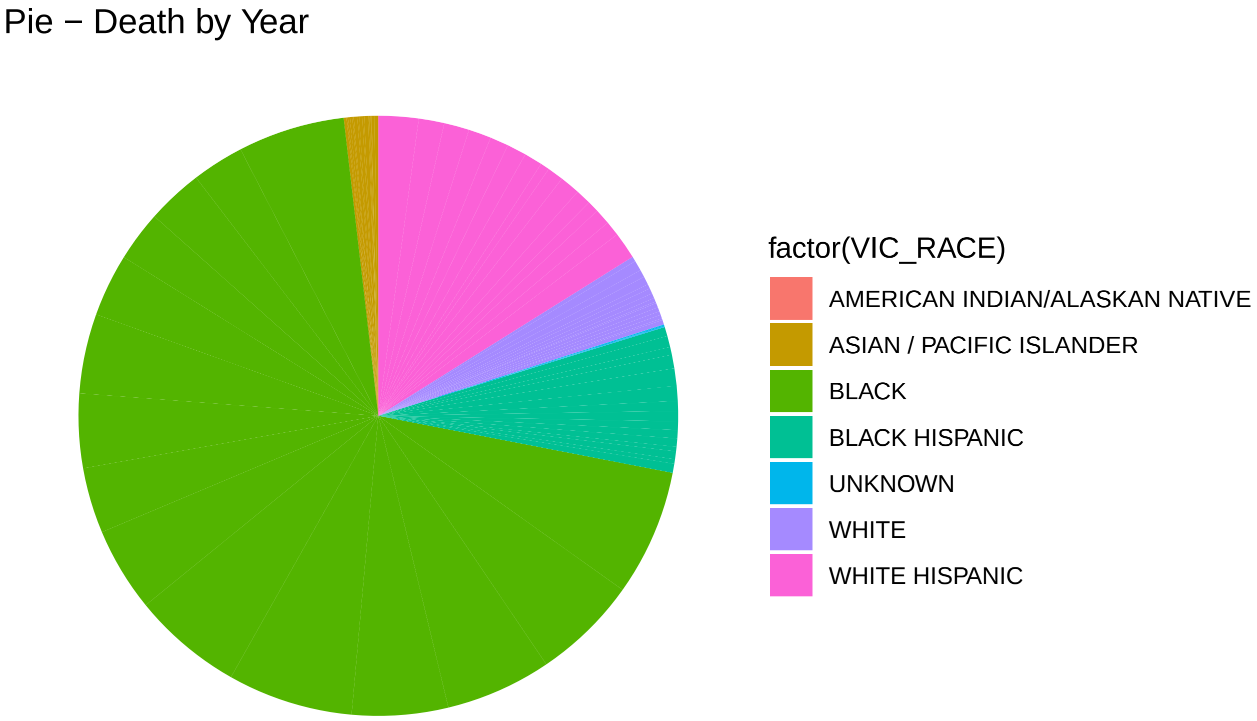
```
# Multiple Pie Chart  - Exploring Multiple for Practise
ggplot(data=shoot_historic_By_Date_Year,
       aes(x=" ", y=DEATH_COUNT, group=factor(DEATH_COUNT),
           colour=factor(DEATH_COUNT), fill=factor(DEATH_COUNT))) +
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y", start=0) +
  facet_grid(.~ factor(OCCUR_DATE_YEAR)) +theme_void() +
  labs(title = "Multiple Pie Chart- Year and Death Count" )
```



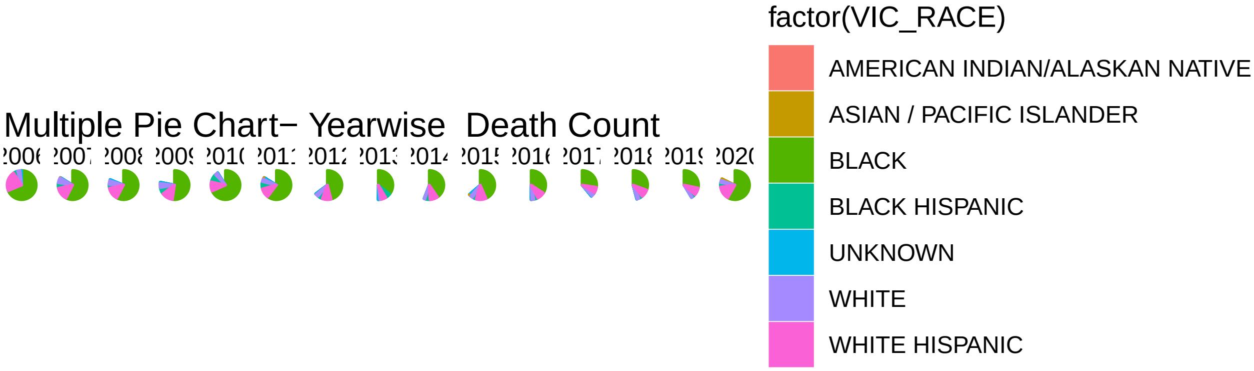
```
shoot_historic_by_VIC_RACE_YEAR
```

```
## # A tibble: 96 x 5
## # Groups:   VIC_RACE [7]
##    TOOL_TIP_COLS               VIC_RACE OCCUR_DATE_YEAR INCIDENT_COUNT DEATH_COUNT
##    <chr>                       <chr>              <dbl>          <dbl>       <dbl>
##  1 2007, AMERICAN INDIAN/AL~ AMERICA~            2007              1           0
##  2 2009, AMERICAN INDIAN/AL~ AMERICA~            2009              2           0
##  3 2010, AMERICAN INDIAN/AL~ AMERICA~            2010              1           0
##  4 2011, AMERICAN INDIAN/AL~ AMERICA~            2011              2           0
##  5 2012, AMERICAN INDIAN/AL~ AMERICA~            2012              1           0
##  6 2016, AMERICAN INDIAN/AL~ AMERICA~            2016              1           0
##  7 2018, AMERICAN INDIAN/AL~ AMERICA~            2018              1           0
##  8 2006, ASIAN / PACIFIC IS~ ASIAN /~            2006             26           7
##  9 2007, ASIAN / PACIFIC IS~ ASIAN /~            2007             18           3
## 10 2008, ASIAN / PACIFIC IS~ ASIAN /~            2008             32           5
## # ... with 86 more rows
```

```
# Pie chart
ggplot(shoot_historic_by_VIC_RACE_YEAR,
       aes(x="", y=DEATH_COUNT, fill=factor(VIC_RACE) )) +
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y", start=0) +theme_void() + labs(title = "Pie - Death by Year" )
```



```r
# Multiple Pie Chart - DeathCount by Vic Race and Year - Exploring Multiple for Practise
ggplot(data=shoot_historic_by_VIC_RACE_YEAR,
       aes(x=" ", y=DEATH_COUNT, group=factor(DEATH_COUNT),
           colour=factor(VIC_RACE), fill=factor(VIC_RACE))) +
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y", start=0) +
  facet_grid(.~ factor(OCCUR_DATE_YEAR)) +theme_void() +
  labs(title = "Multiple Pie Chart- Yearwise  Death Count" )
```



```
# Interactive Tooltip(Year,Race,IncidentCount,DeathCount) using GGplot and giraph for better visualizat
gg_point_year = ggplot(data = shoot_historic_by_VIC_RACE_YEAR) +
    geom_point_interactive(aes(x = factor(OCCUR_DATE_YEAR), y = DEATH_COUNT, colour=factor(DEATH_COUNT)
    , tooltip = TOOL_TIP_COLS, data_id = factor(OCCUR_DATE_YEAR))) +
  labs(title = "Interactive- Yearwise  Death Count" )

girafe(ggobj = gg_point_year  , width_svg = 10, height_svg = 5)
```

```
## PhantomJS not found. You can install it with webshot::install_phantomjs(). If it is installed, please
```

```
# Interactive Tooltip(Year,Race,IncidentCount,DeathCount) using GGplot and giraph for better visualizat
gg_point_vic_race = ggplot(data = shoot_historic_by_VIC_RACE_YEAR) +
    geom_point_interactive(aes(x = factor(VIC_RACE), y = DEATH_COUNT,
                               colour=factor(DEATH_COUNT),
    , tooltip = TOOL_TIP_COLS , data_id = factor(VIC_RACE))) +
  labs(title = "Interactive- Victim Race Death Count" )

girafe(ggobj = gg_point_vic_race , width_svg = 10, height_svg = 5)
```

```
#summary(shoot_historic)
#head(shoot_historic , 5)
#data.table(shoot_historic)
#spec(shoot_historic)
```

Top 5 Incidents happened in a day / Maximum shooting incidents in a day :

```
#shoot_historic_By_Date
# sort dataframe by column in r
# select top N results
shoot_historic_By_Date_TOP_5_Incidents <- shoot_historic_By_Date[order(-shoot_historic_By_Date$INCIDENT_

shoot_historic_By_Date_TOP_5_Incidents
```

```
## # A tibble: 5 x 3
##   OCCUR_DATE INCIDENT_COUNT DEATH_COUNT
##   <date>              <dbl>       <dbl>
## 1 2020-07-05             47          11
## 2 2011-09-04             31           4
## 3 2020-07-26             29          12
## 4 2007-08-11             26           6
## 5 2006-09-04             25           8
```

```
shoot_historic_By_Date_TOP_Incidents <- shoot_historic_By_Date_TOP_5_Incidents[1,]

shoot_historic_By_Date_TOP_Incidents
```

```
## # A tibble: 1 x 3
##   OCCUR_DATE INCIDENT_COUNT DEATH_COUNT
##   <date>              <dbl>       <dbl>
## 1 2020-07-05             47          11
```

```
# Maximum Incidents in a day
max(shoot_historic_By_Date_TOP_Incidents$INCIDENT_COUNT)
```

```
## [1] 47
```

```
# Maximum Incidents day - date
max(shoot_historic_By_Date_TOP_Incidents$OCCUR_DATE)
```

```
## [1] "2020-07-05"
```

Top 5 Deaths happened in a day / Maximum Deaths in a day :

```
# sort dataframe by column in r
# select top N results
shoot_historic_By_Date_TOP_5_Deaths <- shoot_historic_By_Date[order(-shoot_historic_By_Date$DEATH_COUNT

shoot_historic_By_Date_TOP_5_Deaths
```

```
## # A tibble: 5 x 3
##   OCCUR_DATE INCIDENT_COUNT DEATH_COUNT
##   <date>              <dbl>       <dbl>
## 1 2020-07-26             29          12
## 2 2020-07-05             47          11
## 3 2011-12-12             11          10
## 4 2007-10-06             22           9
## 5 2007-11-18             18           9
```

```r
shoot_historic_By_Date_TOP_Deaths_In_Day <- shoot_historic_By_Date_TOP_5_Deaths[1,]

shoot_historic_By_Date_TOP_Deaths_In_Day
```

```
## # A tibble: 1 x 3
##   OCCUR_DATE INCIDENT_COUNT DEATH_COUNT
##   <date>              <dbl>       <dbl>
## 1 2020-07-26             29          12
```

```r
# Maximum Deaths in a day
max(shoot_historic_By_Date_TOP_Deaths_In_Day$DEATH_COUNT)
```

```
## [1] 12
```

```r
# Maximum Deaths day - date
max(shoot_historic_By_Date_TOP_Deaths_In_Day$OCCUR_DATE)
```

```
## [1] "2020-07-26"
```

```r
max(shoot_historic_By_Date$DEATH_COUNT)
```

```
## [1] 12
```

Maximum shooting incidents in a year:

```r
max(shoot_historic_By_Date_Year$INCIDENT_COUNT)
```

```
## [1] 2055
```

Maximum death incidents in a year:

```r
max(shoot_historic_By_Date_Year$DEATH_COUNT)
```

```
## [1] 445
```

```r
summary(shoot_historic_By_Date)
```

```
##    OCCUR_DATE         INCIDENT_COUNT    DEATH_COUNT
##  Min.   :2006-01-01   Min.   : 1.000   Min.   : 0.0000
##  1st Qu.:2009-08-11   1st Qu.: 2.000   1st Qu.: 0.0000
##  Median :2013-04-03   Median : 4.000   Median : 0.0000
##  Mean   :2013-05-08   Mean   : 4.667   Mean   : 0.8904
##  3rd Qu.:2017-01-05   3rd Qu.: 6.000   3rd Qu.: 1.0000
##  Max.   :2020-12-31   Max.   :47.000   Max.   :12.0000
```

```
#head(shoot_historic_By_Date , 5)
#data.table(shoot_historic_By_Date)
#spec(shoot_historic_By_Date)

shoot_historic_By_Date_all <- shoot_historic %>%
    group_by (JURISDICTION_CODE)  %>%
    summarize( INCIDENT_COUNT = sum(INCIDENT_COUNT) ) %>%
    select (JURISDICTION_CODE, INCIDENT_COUNT )

shoot_historic_By_JURISDICTION_CODE_Date_All <- shoot_historic %>%
   group_by(JURISDICTION_CODE,OCCUR_DATE)  %>%
   summarize(INCIDENT_COUNT = sum(INCIDENT_COUNT),
             DEATH_COUNT = sum(DEATH_COUNT) ) %>%
   select (OCCUR_DATE, JURISDICTION_CODE, INCIDENT_COUNT, DEATH_COUNT)  %>%
   ungroup()
```

```
## `summarise()` has grouped output by 'JURISDICTION_CODE'. You can override using
## the `.groups` argument.
```

```
shoot_historic_By_BORO_Date_All <- shoot_historic %>%
   group_by(BORO,OCCUR_DATE)  %>%
   summarize(INCIDENT_COUNT = sum(INCIDENT_COUNT),
             DEATH_COUNT = sum(DEATH_COUNT) ) %>%
   select (OCCUR_DATE, BORO, INCIDENT_COUNT, DEATH_COUNT)  %>%
   ungroup()
```

```
## `summarise()` has grouped output by 'BORO'. You can override using the
## `.groups` argument.
```

4. Model:

I am using Linear Regression model. This model(lm) command takes the dataset in the following format: lm([target variable] ~ [predictor variables], data = [data-source])

```
## Modeling Data:

mod <- lm(DEATH_COUNT ~ INCIDENT_COUNT, data = shoot_historic_By_BORO_Date_All)

summary(mod)
```

```
##
## Call:
## lm(formula = DEATH_COUNT ~ INCIDENT_COUNT, data = shoot_historic_By_BORO_Date_All)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4594 -0.3760 -0.1190  0.1101  5.3393
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.137959   0.009545  -14.45   <2e-16 ***
## INCIDENT_COUNT  0.256956   0.003535   72.68   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.6448 on 11308 degrees of freedom
## Multiple R-squared:  0.3184, Adjusted R-squared:  0.3183
## F-statistic:  5283 on 1 and 11308 DF,  p-value: < 2.2e-16
```

```
shoot_historic_By_BORO_Date_All %>% slice_min(INCIDENT_COUNT)
```

```
## # A tibble: 5,811 x 4
##    OCCUR_DATE BORO  INCIDENT_COUNT DEATH_COUNT
##    <date>     <chr>          <dbl>       <dbl>
##  1 2006-01-04 BRONX              1           0
##  2 2006-01-10 BRONX              1           0
##  3 2006-01-19 BRONX              1           0
##  4 2006-01-23 BRONX              1           1
##  5 2006-01-24 BRONX              1           0
##  6 2006-02-05 BRONX              1           0
##  7 2006-02-16 BRONX              1           0
##  8 2006-02-18 BRONX              1           0
##  9 2006-02-19 BRONX              1           1
## 10 2006-02-21 BRONX              1           1
## # ... with 5,801 more rows
```

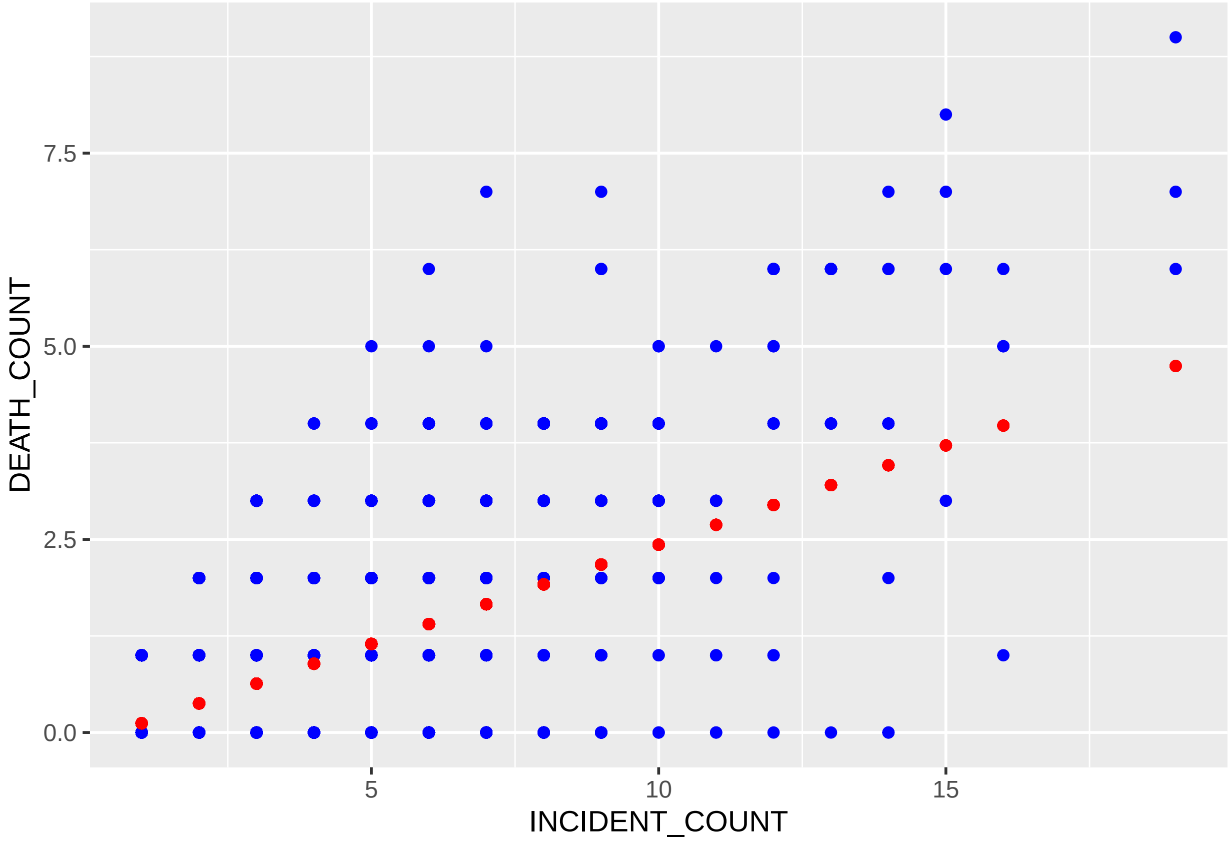
```
NYPD_tot_w_pred <- shoot_historic_By_BORO_Date_All %>%
  mutate(pred = predict(mod))

NYPD_tot_w_pred
```

```
## # A tibble: 11,310 x 5
##    OCCUR_DATE BORO  INCIDENT_COUNT DEATH_COUNT  pred
##    <date>     <chr>          <dbl>       <dbl> <dbl>
##  1 2006-01-01 BRONX              2           0 0.376
##  2 2006-01-04 BRONX              1           0 0.119
##  3 2006-01-05 BRONX              2           0 0.376
##  4 2006-01-06 BRONX              3           0 0.633
##  5 2006-01-09 BRONX              4           2 0.890
##  6 2006-01-10 BRONX              1           0 0.119
##  7 2006-01-13 BRONX              2           0 0.376
##  8 2006-01-14 BRONX              2           2 0.376
##  9 2006-01-15 BRONX              2           1 0.376
## 10 2006-01-16 BRONX              2           1 0.376
## # ... with 11,300 more rows
```

```
summary(NYPD_tot_w_pred)
```

```
##    OCCUR_DATE             BORO           INCIDENT_COUNT    DEATH_COUNT
##  Min.   :2006-01-01   Length:11310       Min.   : 1.000   Min.   :0.0000
##  1st Qu.:2009-05-06   Class :character   1st Qu.: 1.000   1st Qu.:0.0000
##  Median :2012-08-29   Mode  :character   Median : 1.000   Median :0.0000
##  Mean   :2013-01-23                        Mean   : 2.085   Mean   :0.3979
##  3rd Qu.:2016-08-11                        3rd Qu.: 2.000   3rd Qu.:1.0000
##  Max.   :2020-12-31                        Max.   :19.000   Max.   :9.0000
##       pred
##  Min.   :0.1190
##  1st Qu.:0.1190
##  Median :0.1190
##  Mean   :0.3979
##  3rd Qu.:0.3760
##  Max.   :4.7442
```

```
NYPD_tot_w_pred %>% ggplot() +
 geom_point(aes(x = INCIDENT_COUNT, y = DEATH_COUNT), color = "blue") +
 geom_point(aes(x = INCIDENT_COUNT, y = pred), color = "red")
```



Model Performance and Coefficients:
From the model performance above, we can see the values of the intercept ("a" value) and the slope ("b" value) for the year. These "a" and "b" values plot a line between all the points of the data. So in this case, if there is a incident count is 100, a is -0.137959 and b is 0.256956, the model predicts (on average) around (-0.137959 + (0.256956 * 100)) = ~26 deaths can happen. It might be possible to get better model performance by considering other features inflation, job market, financial market, political, wealth-related information, and many more related to these geographical areas. In this way, we can predict a much better crime/death rate and improve model performance much better as well.

5. Bias:

There is a possibility of some types of biases in the NYPD Shooting dataset. By removing or reducing them it's highly possible to predict better test results close to training data and the model can eventually perform better.

With that said, it is important to monitor the data preparation processes closely to make sure the datasets are as bias-free as possible before they are used in the training phase.

Selection Bias: This seems like not an issue as this data is from NYPD
Overfitting and Underfitting: When a model gets trained with large amounts of data, it also starts learning from the noise and inaccurate data entries in the dataset. Consequently, the model does not categorize the data correctly, because of too many details and noise. In this data set, lat lang or many other features can cause noise but can be reduced.

Exclusion Bias: It's possible excluding some features can cause higher bias and this can be reduced including some features that can reduce bias like climate and economic situations and political situations, and inflation and seasons can be included to get more accurate model performance.

Conclusion:

To conclude, I have done the Visualizations, Data analyzing and Modeling using the NYPD shooting incident dataset. I have provided summary below:

1. **Shooting Incidents by Jurisdiction.**
    - Summarizing shoot historic

2. **Visualization and Analysis of NYPD Shooting Incidents data.**
    - Visualizing NYPD Shoot Incidents and Deaths
        - Visualizing NYPD Shoot Incidents by Jurisdiction - Pie/Coxcomb/Bar chart
        - Visualizing NYPD Shoot Incidents by Yearly - Bar/Scatter/Pie/Coxcomb
        - Visualizing NYPD Death by Yearly - Bar/Scatter/Pie/Coxcomb/Multiple-Pie/Interactive chart
    - Maximum shooting incidents by date wise : 47 on 2020-07-05
    - Maximum death incidents by date wise : 12 on 2020-07-26
    - Maximum shooting incidents by yearly : 2055
    - Maximum death incidents by yearly : 445
3. **Linear Regression Model prediction**
    - Model predicts ~26 deaths for 100 shooting incidents.
    - LM model prediction plot for visually understand better.
4. **Bias information**
    - By adding much other information and features like inflation, job market, financial market, wealth-related information and many more related to these geographical areas we can predict a much better crime/death rate and improve model performance much better.