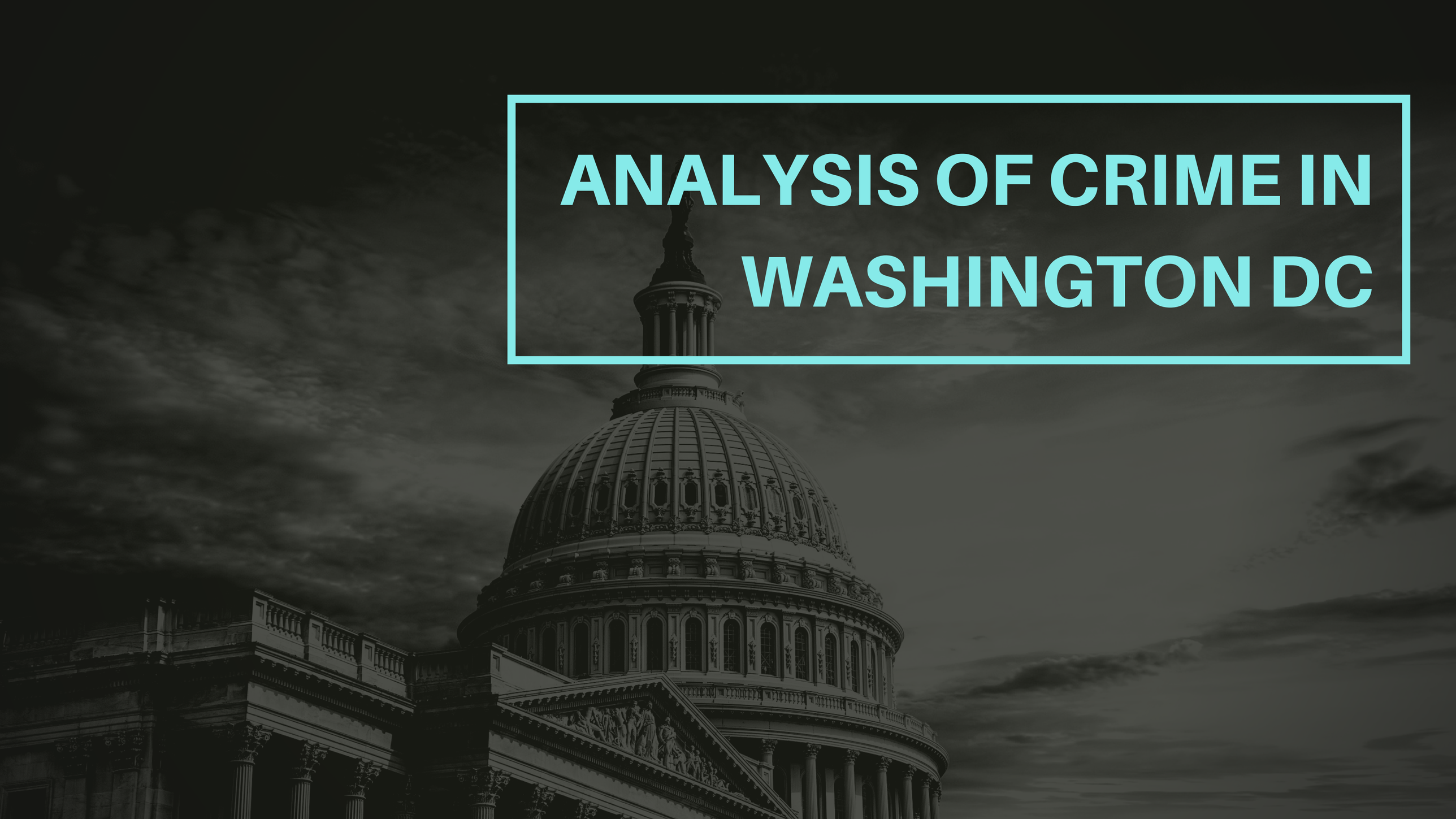


ANALYSIS OF CRIME IN WASHINGTON DC

OUR OBJECTIVES

DATA SOURCE

Open Data DC

Crime Incidents

https://opendata.dc.gov/

WHAT

Finding patterns and trends to understand the safety level and types of crimes that occur in different areas in Washington D.C.

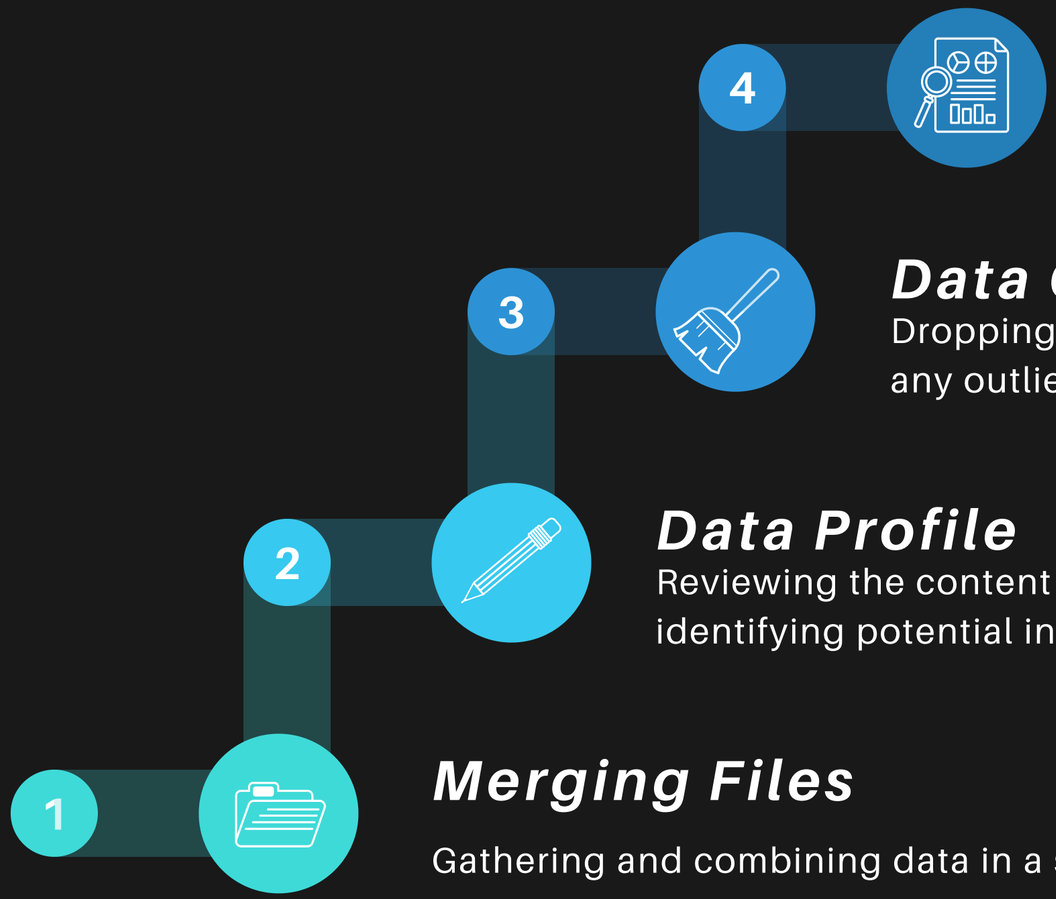
WHY

To shine a light on criminal activity and provide insight into the safety of the country's capital city.

HOW

Using crime data compiled over the past 10 years until October 2019.

OUR PROCESS



Data Analysis

Finding trends and forming conclusions and observations on the data

Data Cleansing

Dropping incomplete rows, duplicates and identifying any outliers in the dataset

Data Profile

Reviewing the content and structure of the data, and identifying potential interrelationships

Merging Files

Gathering and combining data in a single data frame

Read the csv files and convert them into DataFrames

```
# Files to Load
#File for 2019
csv_2019_df=pd.read_csv("../Project1_DC_Crime_Analysis/Resources/Crime_Incidents_in_2019.csv")
#File for 2018
csv_2018_df=pd.read_csv("../Project1_DC_Crime_Analysis/Resources/Crime_Incidents_in_2018.csv")
#File for 2017
csv_2017_df=pd.read_csv("../Project1_DC_Crime_Analysis/Resources/Crime_Incidents_in_2017.csv")
#File for 2016
csv_2016_df=pd.read_csv("../Project1_DC_Crime_Analysis/Resources/Crime_Incidents_in_2016.csv")
#File for 2015
csv_2015_df=pd.read_csv("../Project1_DC_Crime_Analysis/Resources/Crime_Incidents_in_2015.csv")
#File for 2014
csv_2014_df=pd.read_csv("../Project1_DC_Crime_Analysis/Resources/Crime_Incidents_in_2014.csv")
#File for 2013
csv_2013_df=pd.read_csv("../Project1_DC_Crime_Analysis/Resources/Crime_Incidents_in_2013.csv")
#File for 2012
csv_2012_df=pd.read_csv("../Project1_DC_Crime_Analysis/Resources/Crime_Incidents_in_2012.csv")
#File for 2011
csv_2011_df=pd.read_csv("../Project1_DC_Crime_Analysis/Resources/Crime_Incidents_in_2011.csv")
#File for 2010
csv_2010_df=pd.read_csv("../Project1_DC_Crime_Analysis/Resources/Crime_Incidents_in_2010.csv")
#File for 2009
csv_2009_df=pd.read_csv("../Project1_DC_Crime_Analysis/Resources/Crime_Incidents_in_2009.csv")
```

Merge all the csv files from 2009-2019 into a single file and a single DataFrame

```
#Use glob to match the pattern 'csv'
#Match the pattern ('csv') and save the list of file names in the 'all_files' variable.
#extension = 'csv'
#Use pandas to concatenate all files in the list and export as CSV.
#The output file is named "dc_crimes.csv" located in your working directory.
#combine all files in the list
path = '../Project1_DC_Crime_Analysis/Resources/'
all_files = glob.glob(path + "/*.csv")
all_files
row = []

for filename in all_files:
   df = pd.read_csv(filename, index_col=None, header=0)
   row.append(df)

dc_crimes = pd.concat(row, axis=0, ignore_index=True)

#Convert to DataFrame
dc_crimes_df=pd.DataFrame(dc_crimes)
dc_crimes_df.head(2)
```

	X	Y	CCN	REPORT_DAT	SHIFT	METHOD	OFFENSE	BLOCK	XBLOCK	YBLOCK	...	BLOCK_GROUP
0	-77.039673	38.903151	8005297	2009-01-12T10:30:00.000Z	DAY	OTHERS	BURGLARY	1000 - 1099 BLOCK OF CONNECTICUT AVENUE NW	396559.0	137255.0	...	010700 1
1	-76.999518	38.904689	162302	2009-11-12T18:00:00.000Z	EVENING	OTHERS	MOTOR VEHICLE THEFT	1100 - 1199 BLOCK OF 5TH STREET NE	400042.0	137425.0	...	010600 2

2 rows × 25 columns

Merging and reading files

```
#Counts - Data Profiling
#Saving a backup copy
dc_crimes_copy=dc_crimes_df.copy()
dc_crimes_df.count()
```

```
X                        374403
Y                        374403
CCN                      374403
REPORT_DAT               374403
SHIFT                    374403
METHOD                   374403
OFFENSE                  374403
BLOCK                    374402
XBLOCK                   374403
YBLOCK                   374403
WARD                     374392
ANC                      374403
DISTRICT                 374201
PSA                      374173
NEIGHBORHOOD_CLUSTER     369548
BLOCK_GROUP              373335
CENSUS_TRACT             373335
VOTING_PRECINCT          374339
LATITUDE                 374403
LONGITUDE                374403
BID                       62208
START_DATE               374393
END_DATE                 352236
OBJECTID                 374403
OCTO_RECORD_ID           374403
dtype: int64
```

```
#Split the data in START_DATE column to get Date and Time information
dc_crimes_df[['START_DATE','Start_Time']] = dc_crimes_df['START_DATE'].str.split('T',expand=True)
dc_crimes_df[['REPORT_DAT','Report_Time']] = dc_crimes_df['REPORT_DAT'].str.split('T',expand=True)
dc_crimes_df[['END_DATE','End_Time']] = dc_crimes_df['END_DATE'].str.split('T',expand=True)
#Display the data
dc_crimes_df.head(2)
```

	X	Y	CCN	REPORT_DAT	SHIFT	METHOD	OFFENSE	BLOCK	XBLOCK	YBLOCK	...	LATITUDE	LONGITUDE	
0	-77.039673	38.903151	8005297	2009-01-12	DAY	OTHERS	BURGLARY	1000 - 1099 BLOCK OF CONNECTICUT AVENUE NW	396559.0	137255.0	...	38.903143	-77.039670	GOL TRIAN
1	-76.999518	38.904689	162302	2009-11-12	EVENING	OTHERS	MOTOR VEHICLE THEFT	1100 - 1199 BLOCK OF 5TH STREET NE	400042.0	137425.0	...	38.904682	-76.999516	

2 rows × 28 columns

- Identifying null values in the data set
- As part of cleansing, splitting the date and time from the START_DATE, REPORT_DATE, END_DATE columns
- Determining the day of the week, months and year for respective records
- Converting data types

Data Profiling and Cleansing

01 Dropping columns

```
# Drop the columns that are not required for analysis
dc_crimes_df.drop(['XBLOCK', 'YBLOCK',
                   'X','Y','ANC','BLOCK_GROUP',
                   'CENSUS_TRACT','VOTING_PRECINCT',
                   'BID','OBJECTID','OCTO_RECORD_ID','BID'],axis=1,inplace=True)
```

02 Reorganizing columns

```
#Reorganizing the columns in the DataFrame
dc_crimes_df=dc_crimes_df[["CCN","START_DATE","Start_Time","END_DATE","End_Time",
                           "DAY_OF_WEEK","WEEK_NUMBER","DAY","MONTH","YEAR",
                            "LATITUDE","LONGITUDE","BLOCK","NEIGHBORHOOD_CLUSTER",
                            "METHOD","OFFENSE","PSA",
                            "SHIFT","WARD","DISTRICT",
                            "REPORT_DAT","Report_Time"]]
```

03 Deleting null values

```
#Deleting rows that have null values in them - Retain END_DATE ( to analyze open cases if required)
dc_crimes_df = dc_crimes_df.dropna(axis=0, subset=['START_DATE','NEIGHBORHOOD_CLUSTER','BLOCK',
                                                   'PSA','WARD','DISTRICT'])
dc_crimes_df.isna().sum()
```

04 Identifying and deleting duplicates

```
#Any Duplicate incidents?
dc_crimes_df['Dup_CCN'] = dc_crimes_df.groupby('CCN')['CCN'].transform('size')
dc_crimes_df['Dup_CCN']
dc_crimes_df=dc_crimes_df.sort_values("Dup_CCN",ascending=False)
dc_crimes_df['Dup_CCN']
```

```
#Delete duplicate Incident Data
dc_crimes_df.drop_duplicates(subset ="CCN",keep = False, inplace = True)
dc_crimes_df['Dup_CCN']
dc_crimes_df.count()
```

Data Profiling and Cleansing

```
arr_year=dc_crimes_df["YEAR"].unique()
#Find the median
mid=median(arr_year)
mid
#Use numpy to create quartiles
q1=np.percentile(arr_year,25)
q3=np.percentile(arr_year,75)
#Calculate interquartile range
iqr=(q3-q1)
#Finding the boundaries
lower_boundary=q1-(1.5*iqr)
upper_boundary=q3+(1.5*iqr)
#Check for any lower outliers
print("Lower Outliers:", arr_year[arr_year<=lower_boundary])
#Check for any upper outliers
print("Upper Outliers:", arr_year[arr_year>=upper_boundary])

#Creating box plot for outliers
plt.boxplot(arr_year,showmeans=True)
plt.title("Boxplot to identify the Outliers(Plot based on Years)")
plt.grid()
plt.savefig("Images/Outliers.png")
plt.show()
```

```
Lower Outliers: [1800 1912 1914 1915 1916]
Upper Outliers: []
```

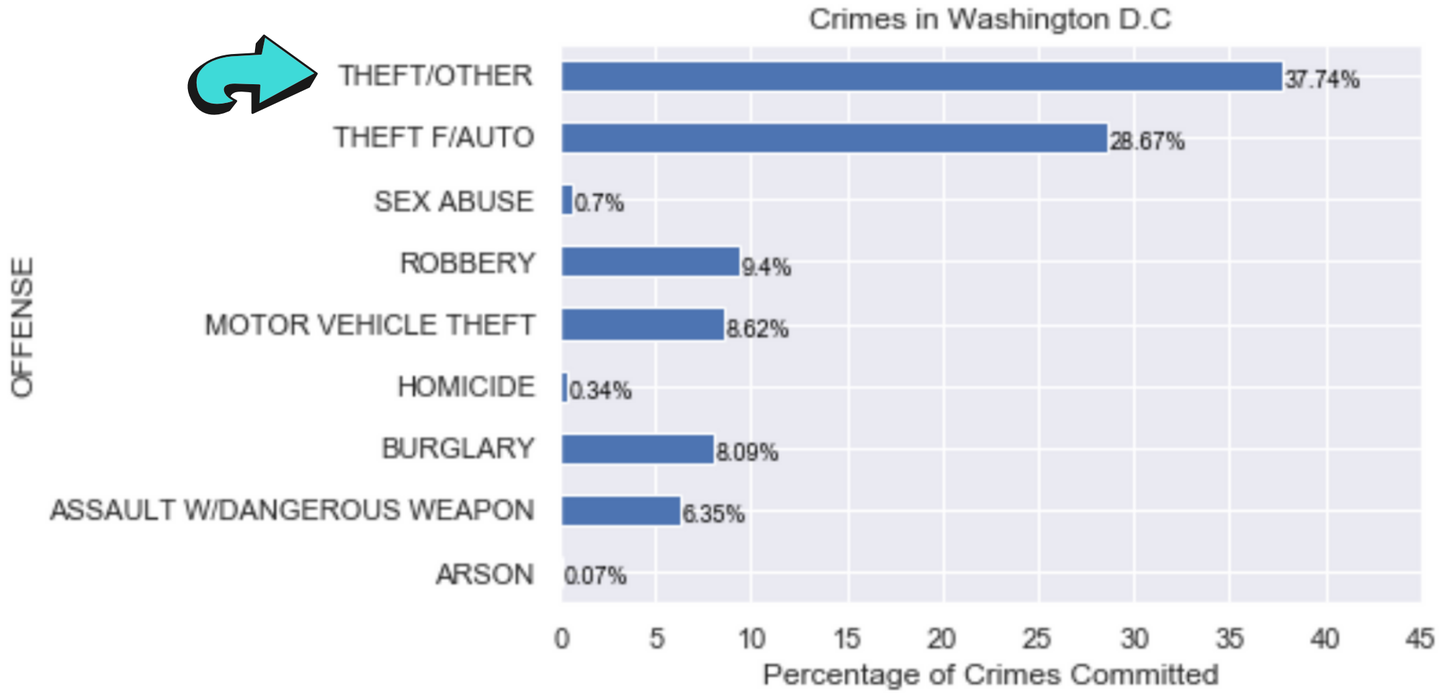
Identifying Outliers:

Incidents that occurred before 2009 but were reported after 2009

- Are these dates an error or were these crimes unsolved until recently through new forensic technology?
- Are they truly 100-year-old cases?
- The crimes under the Category "Homicide" could be cold cases

For our analysis, we will be deleting these

Data Analysis

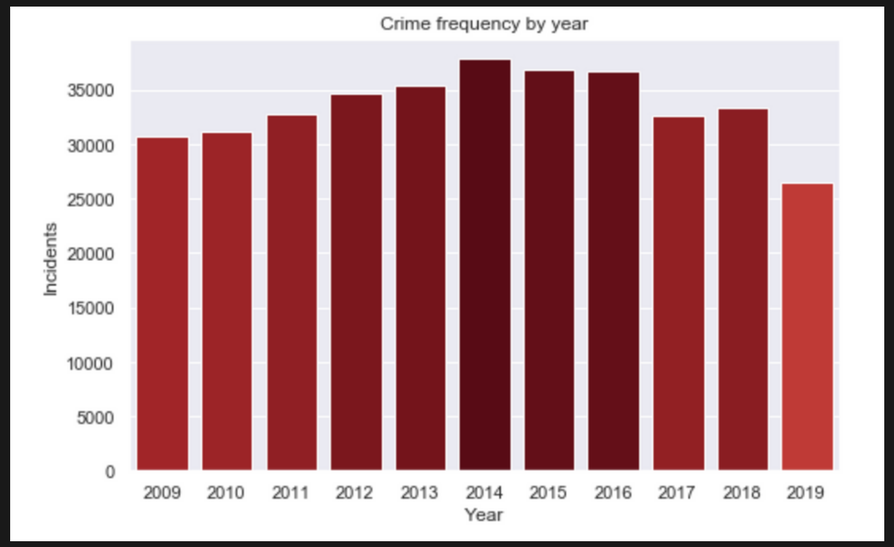


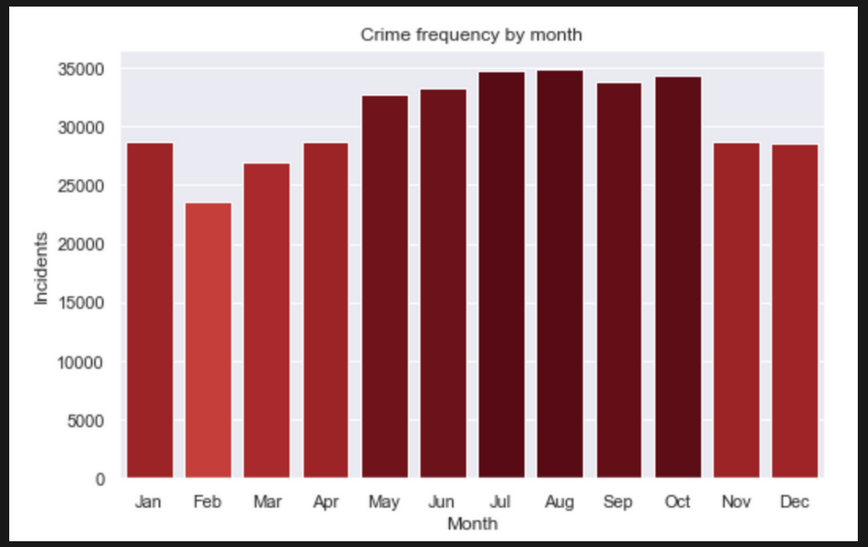
Percentage of crimes for each category of offense

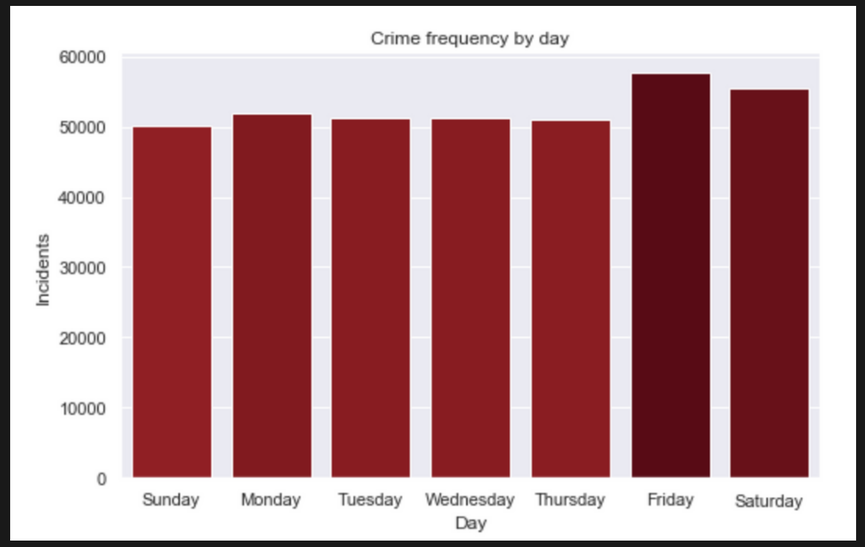
```
OFFENSE
ARSON                             273
ASSAULT W/DANGEROUS WEAPON      23409
BURGLARY                        29863
HOMICIDE                         1243
MOTOR VEHICLE THEFT             31789
ROBBERY                         34671
SEX ABUSE                        2595
THEFT F/AUTO                   105710
THEFT/OTHER                    139172
Name: CCN, dtype: int64
```

Total number of crimes for each category of offense

Data Analysis



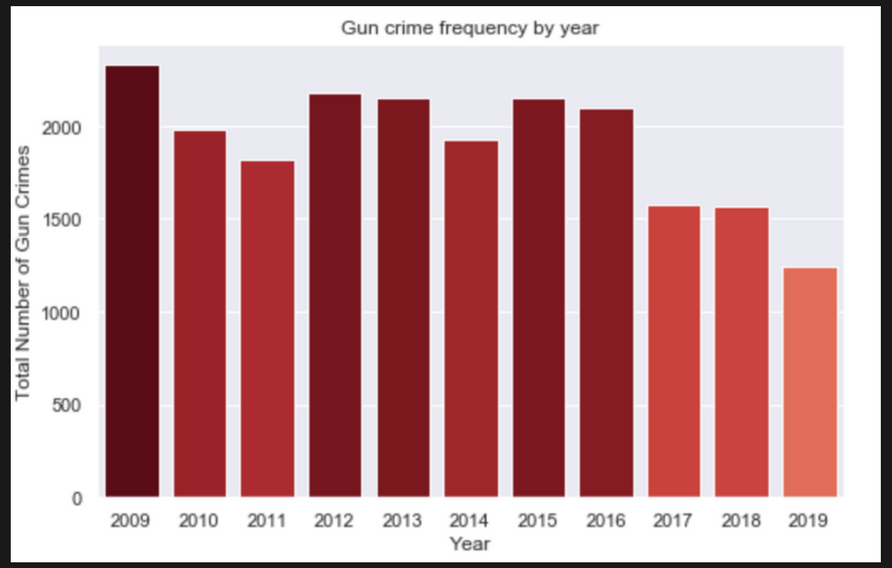


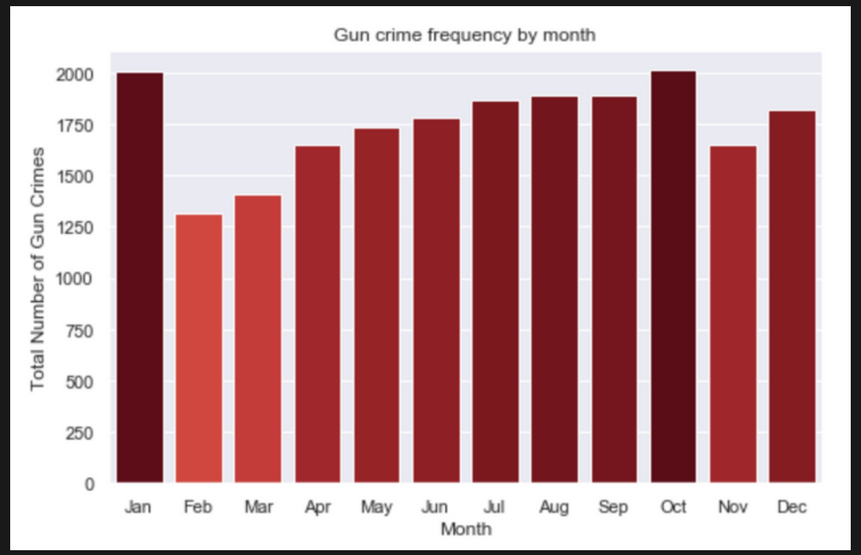


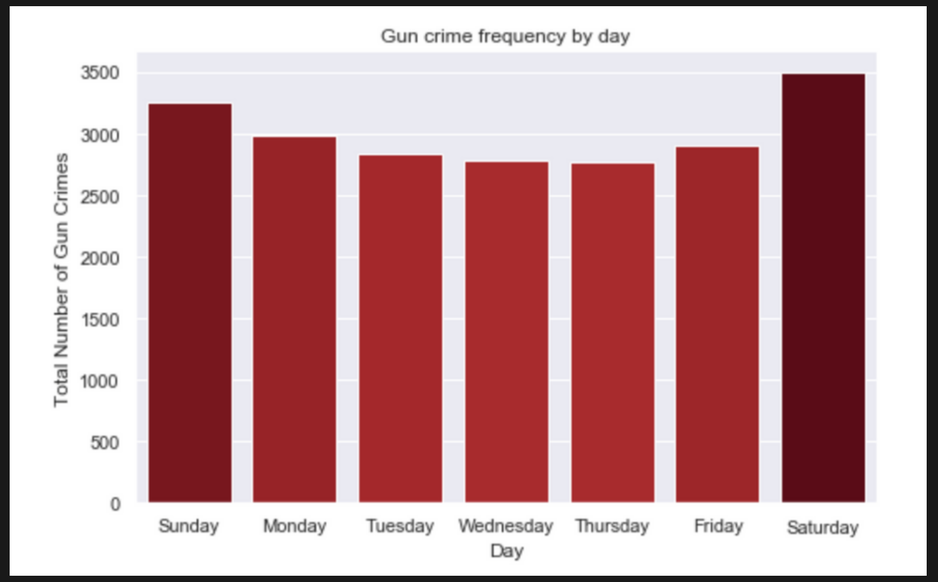
Number of Incidents vs. Time

1. Total Number of Crimes by Year
2. Total Number of Crimes by Month
3. Total Number of Crimes by Day of the Week

Data Analysis



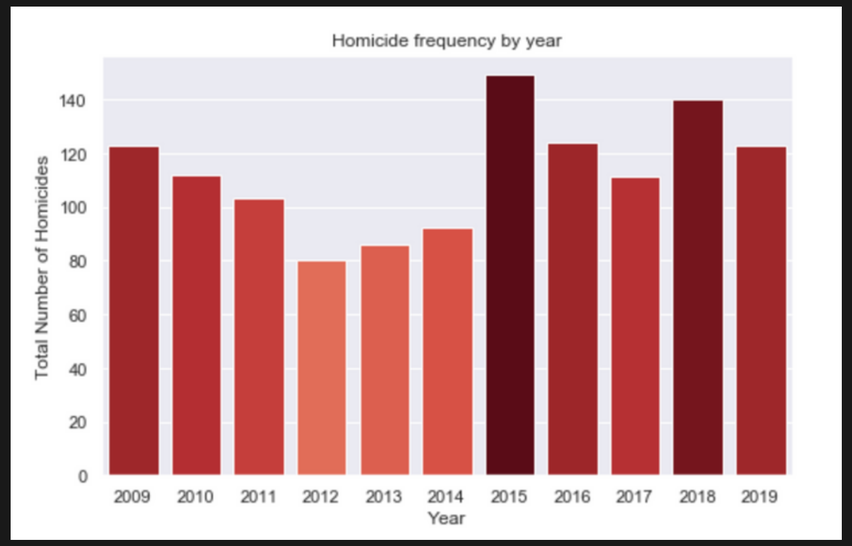


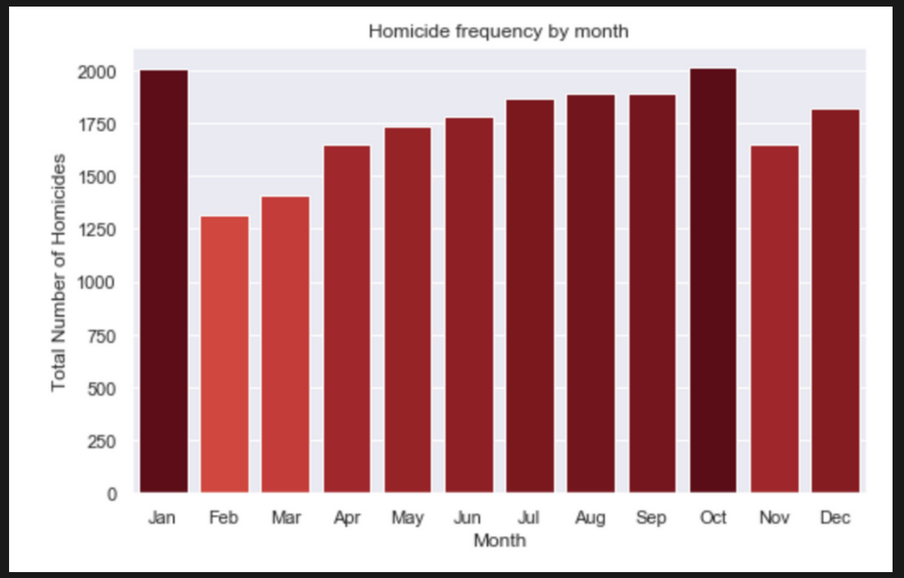


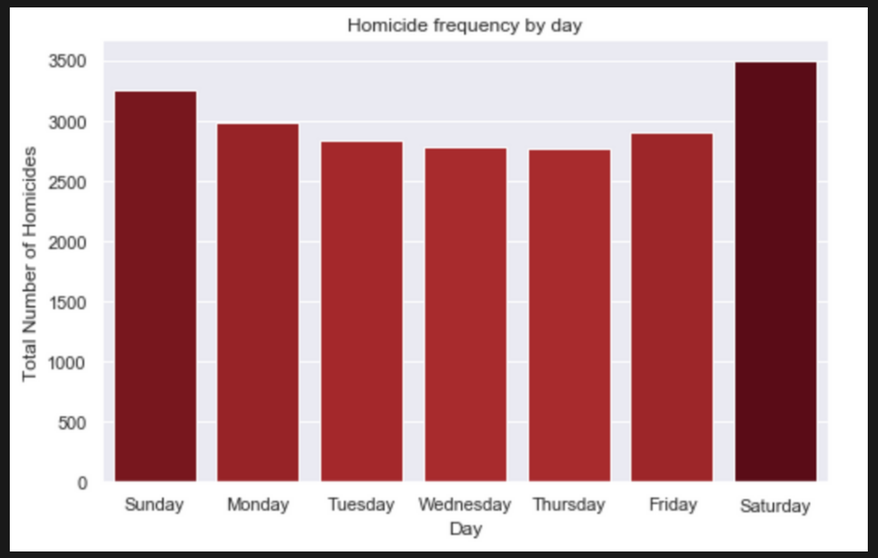
Number of Incidents vs. Time

1. Total Number of Gun Crimes by Year
2. Total Number of Gun Crimes by Month
3. Total Number of Gun Crimes by Day of the Week

Data Analysis





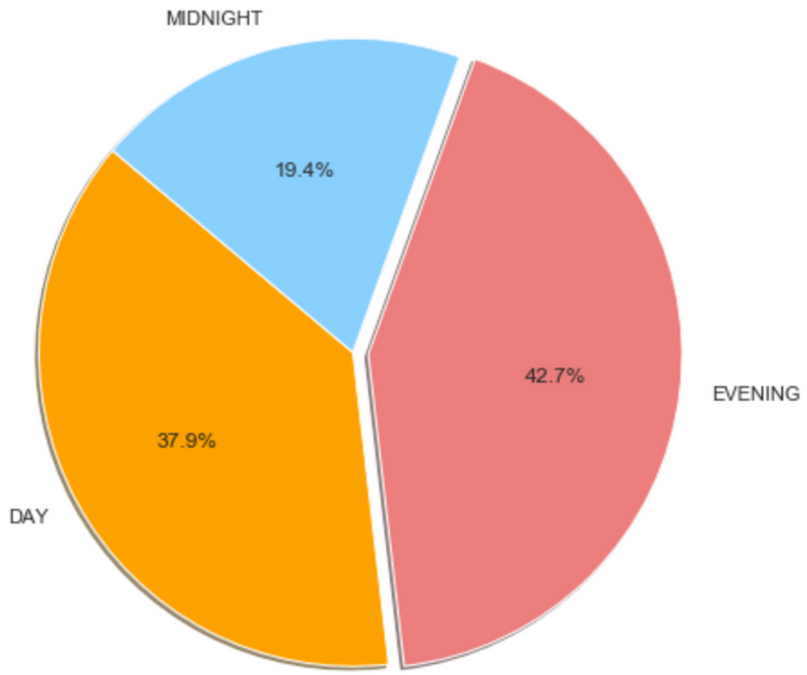


Number of Incidents vs. Time

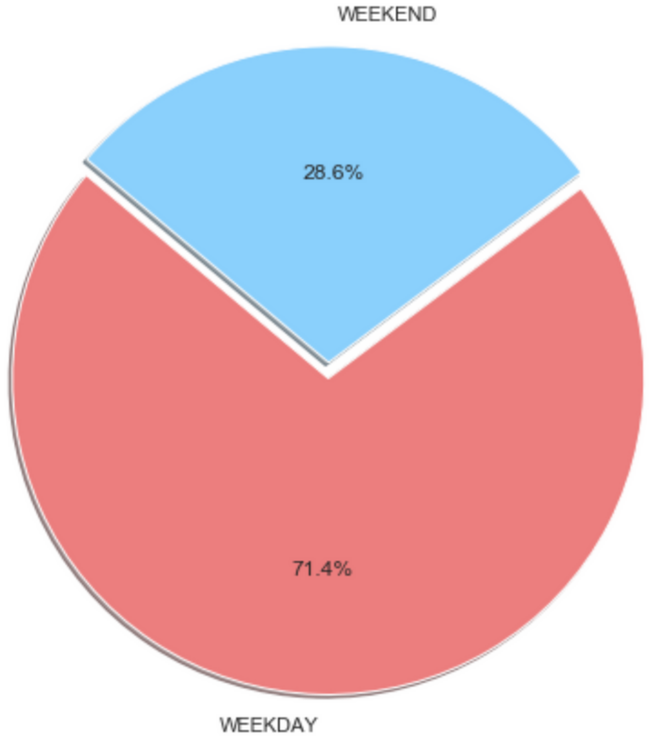
1. Total Number of Homicides by Year
2. Total Number of Homicides by Month
3. Total Number of Homicides by Day of the Week

Data Analysis

Distribution of crime

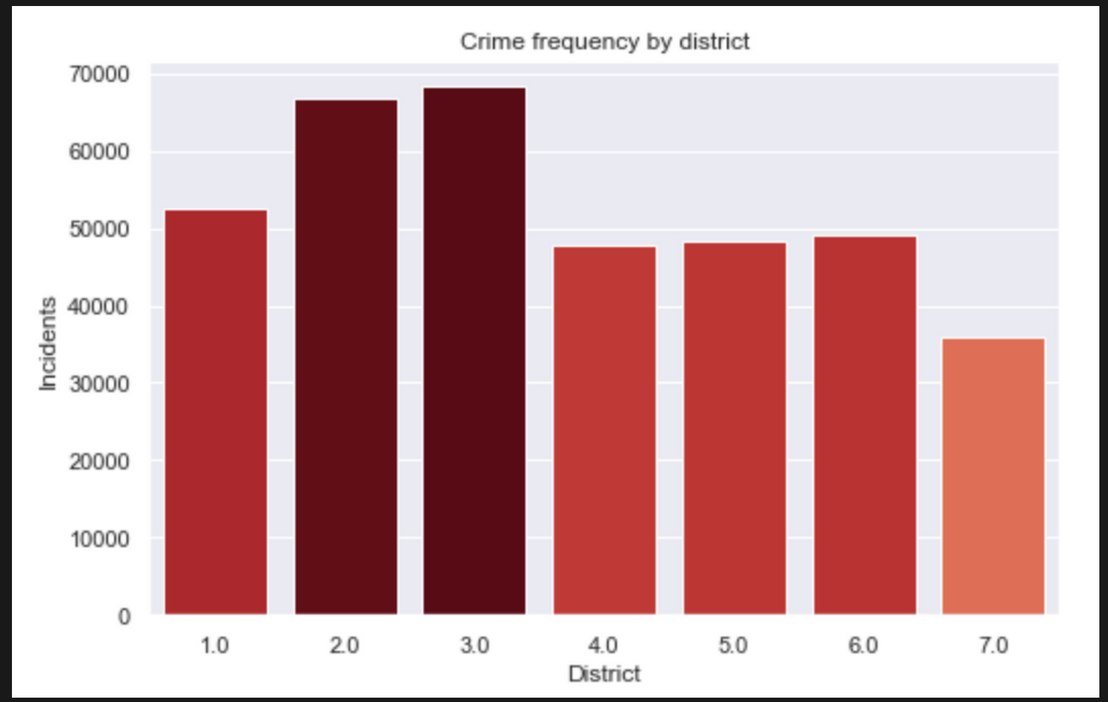


Work Shift

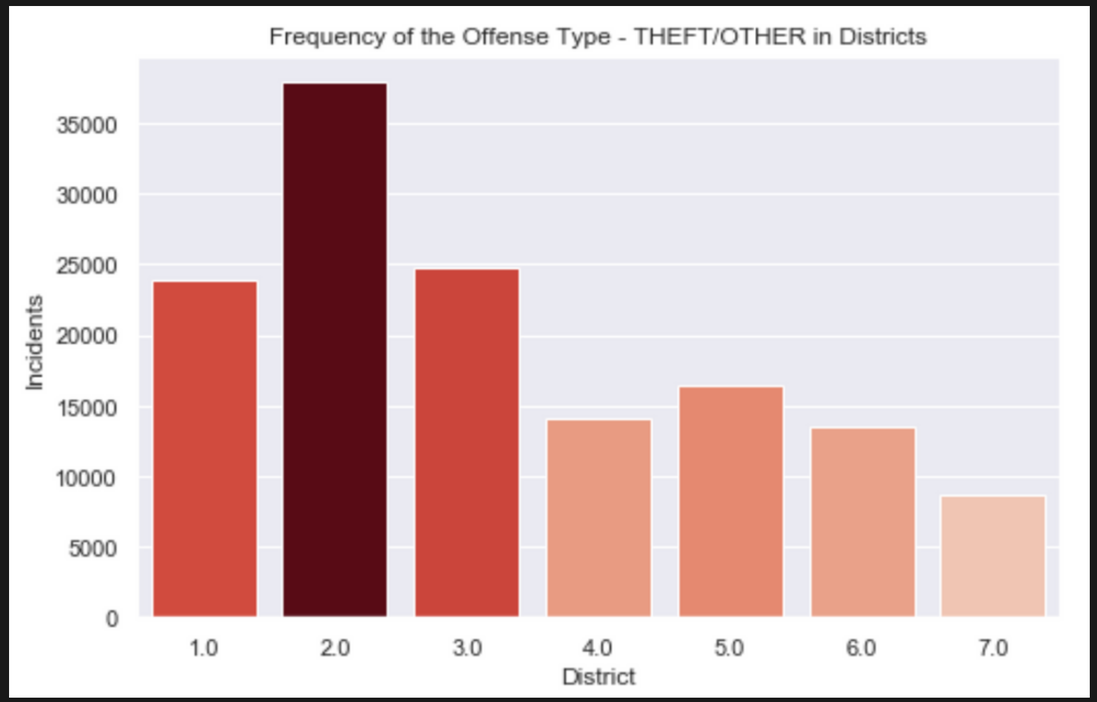


Weekday/Weekend

Data Analysis

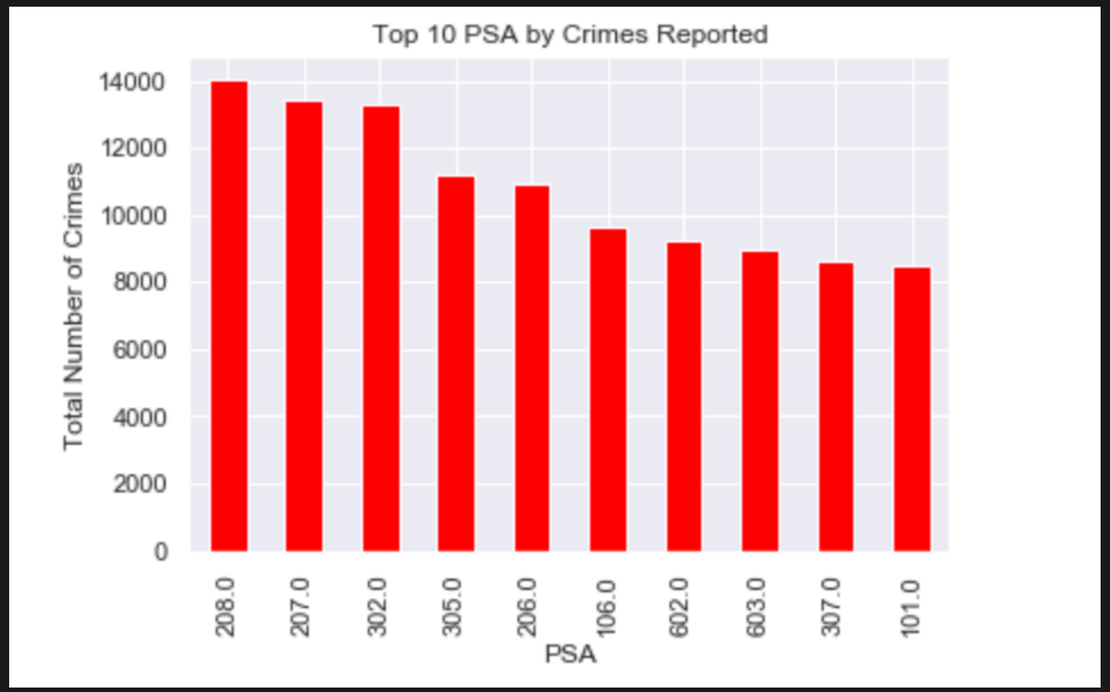


Crime Frequency by District

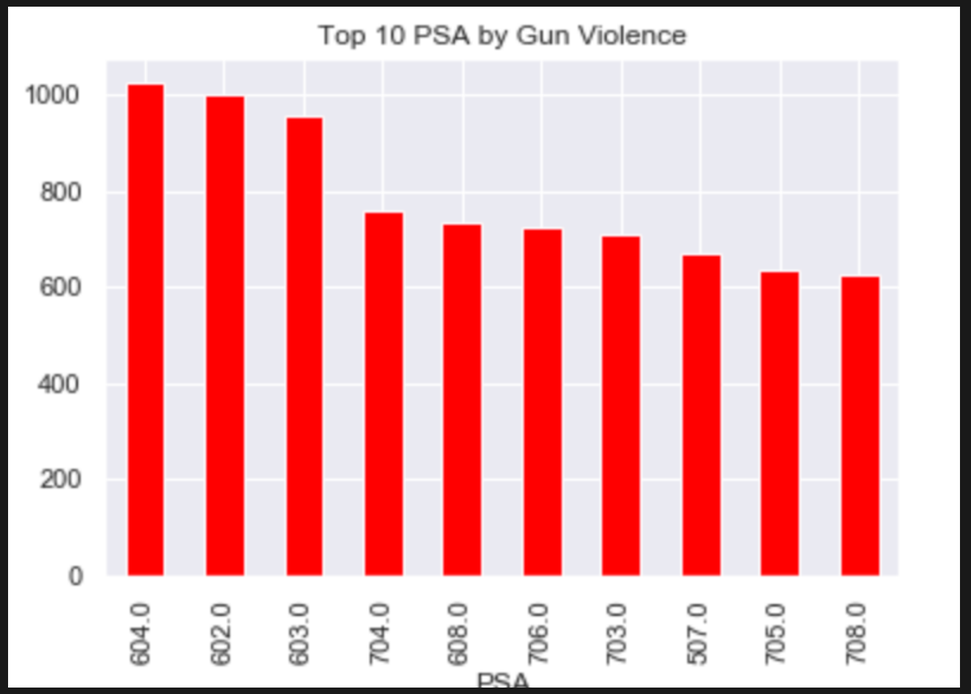


Theft frequency by District

Data Analysis

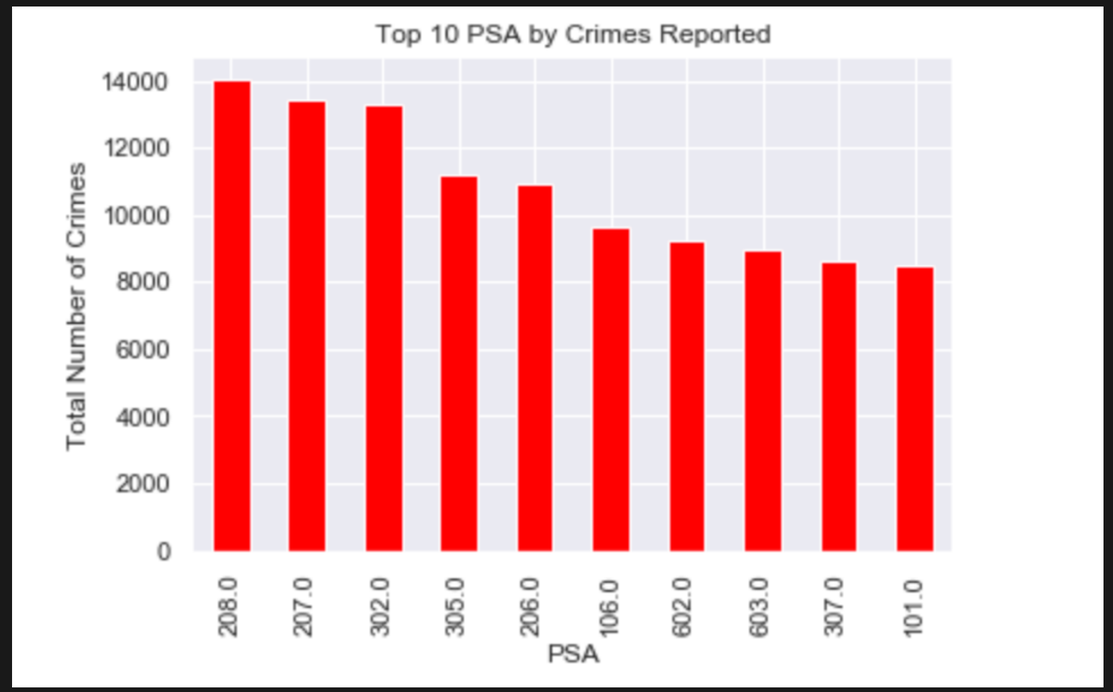


Crime Frequency by PSA

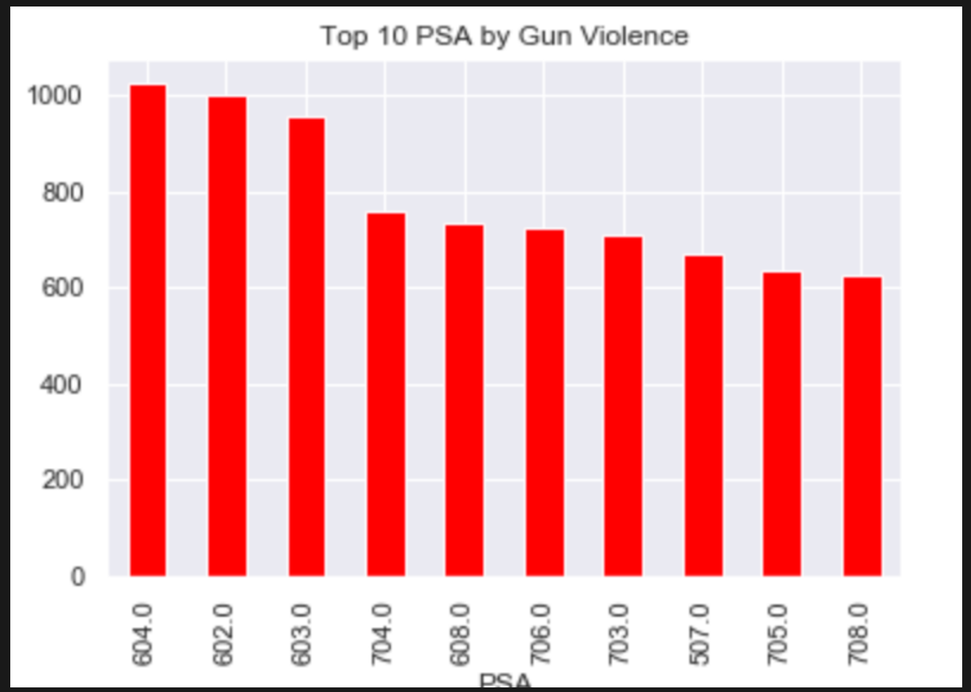


Gun violence by PSA

Data Analysis

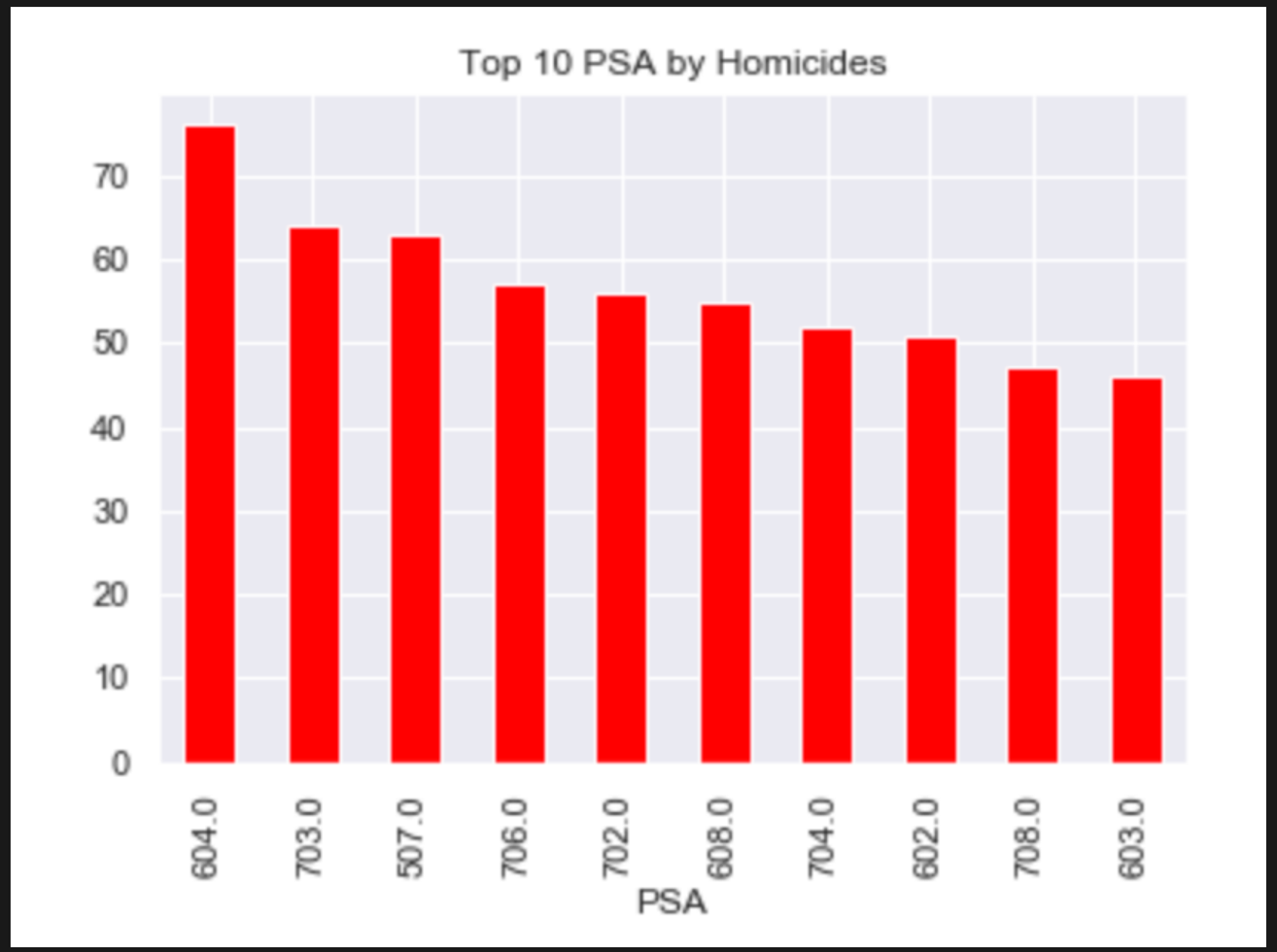


Crime Frequency by PSA



Gun violence by PSA

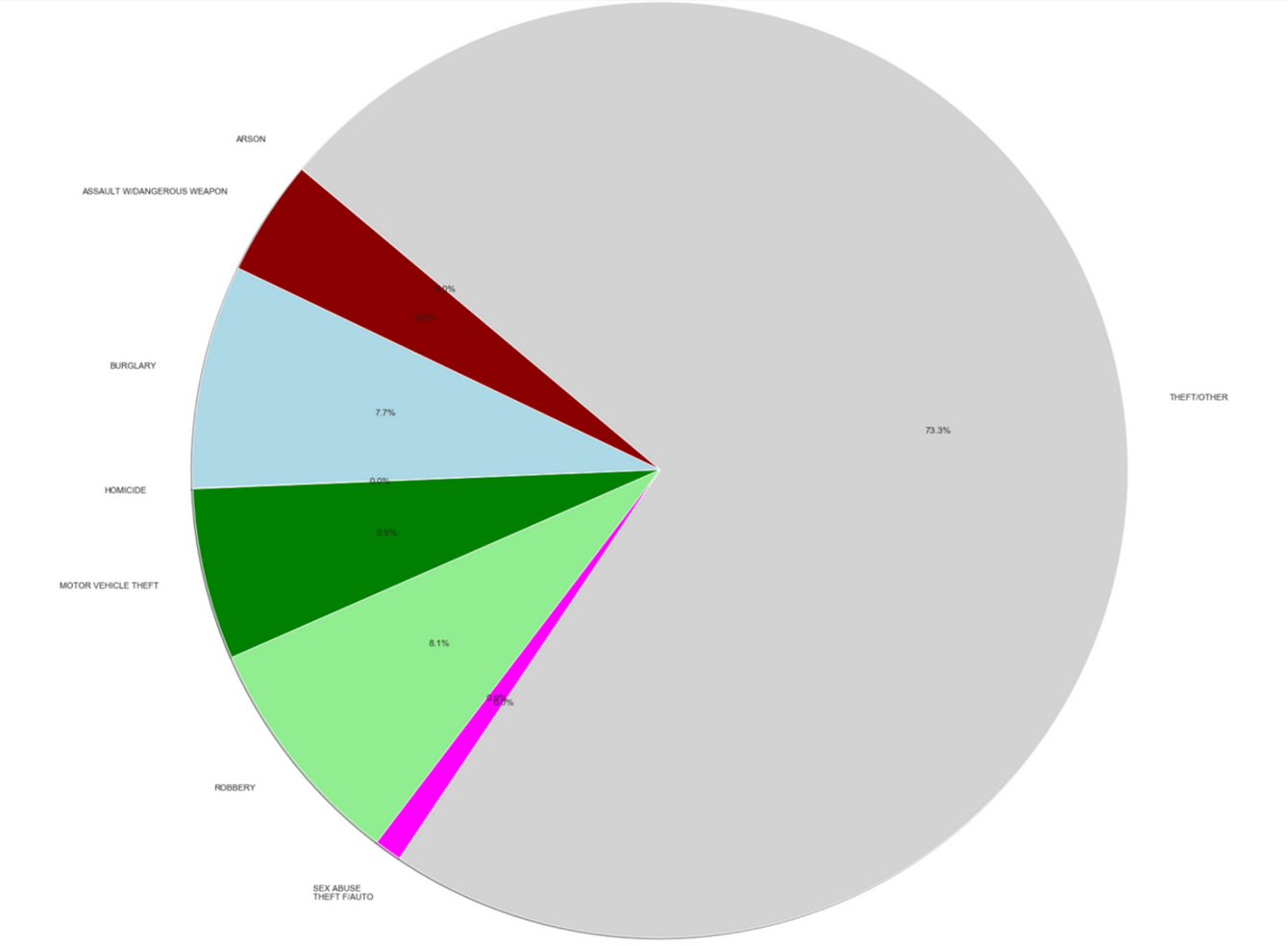
Data Analysis



Homicides by PSA

- The PSAs with the most crime reported were usually not the most dangerous PSAs
- Gun violence and homicide is much more highly correlated than total crime rates
- Crime is likely underreported in high violence areas.

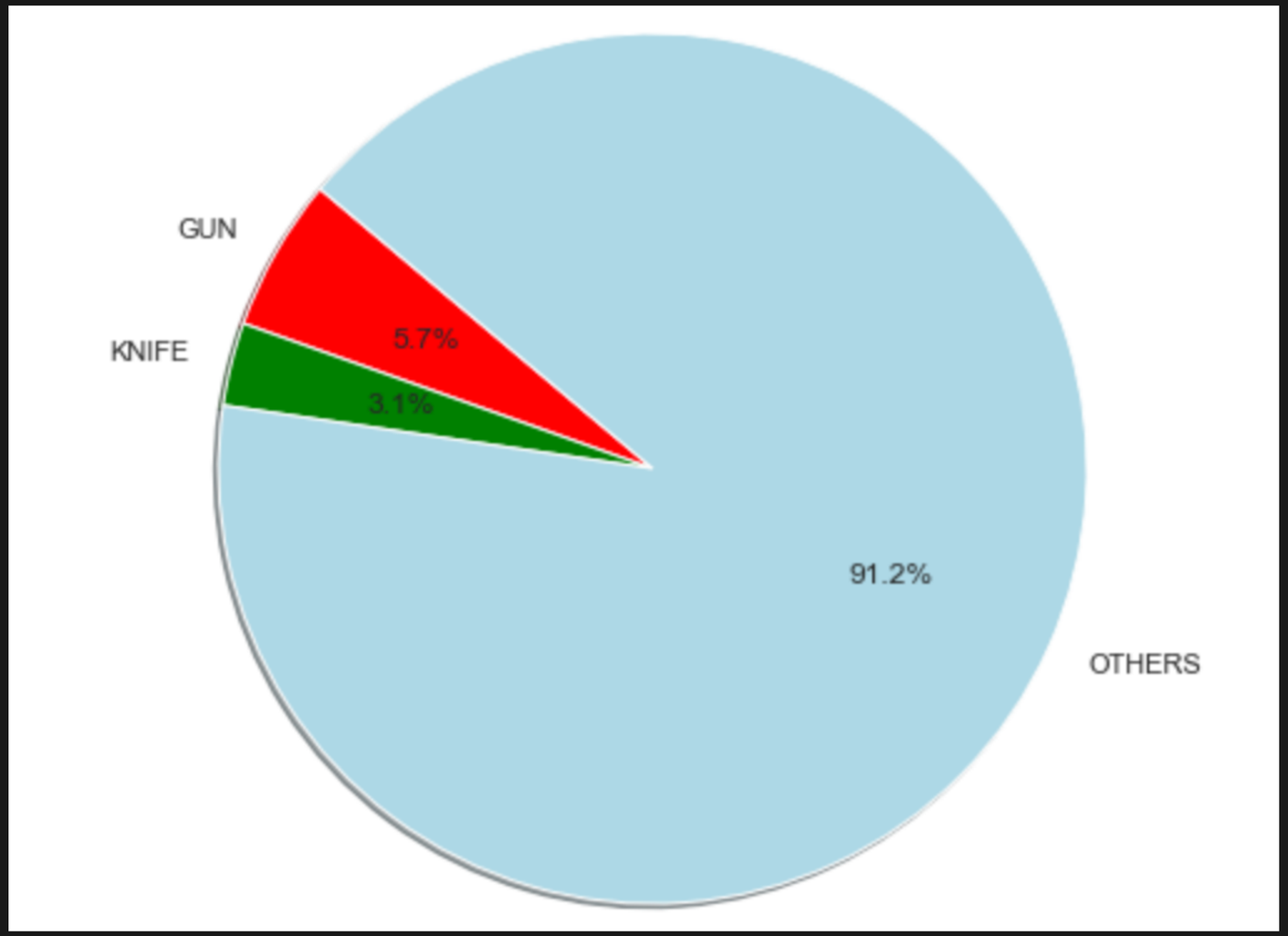
Data Analysis



Crime breakdown in PSA 208

- PSA 208 is the highest crime PSA in DC
- Vast majority of crimes are theft related
- The crimes under the Category "Homicide" could be cold cases
- Only 2 homicides reported over 10 years
- Could be that crimes are most likely to be reported in PSA 208

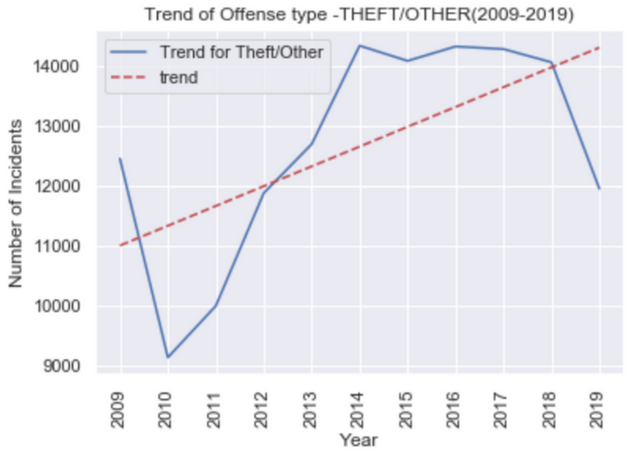
Data Analysis

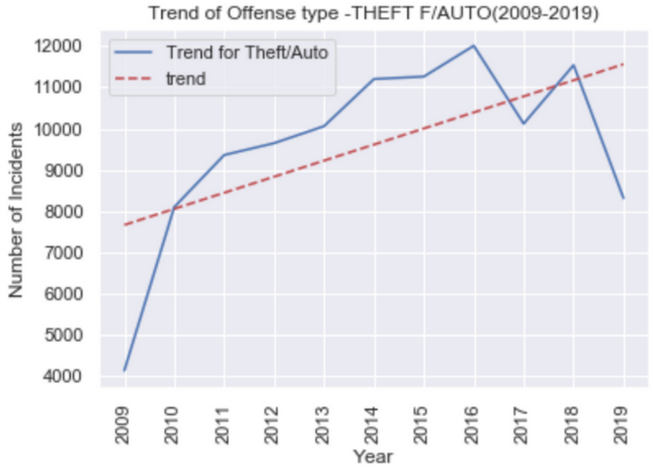


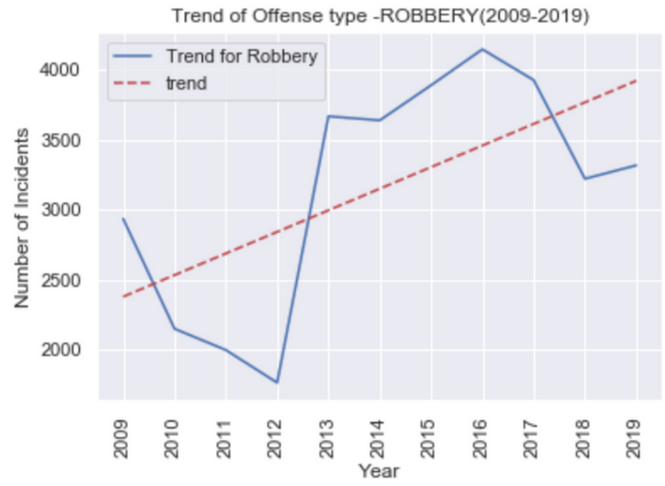
Crimes by Method

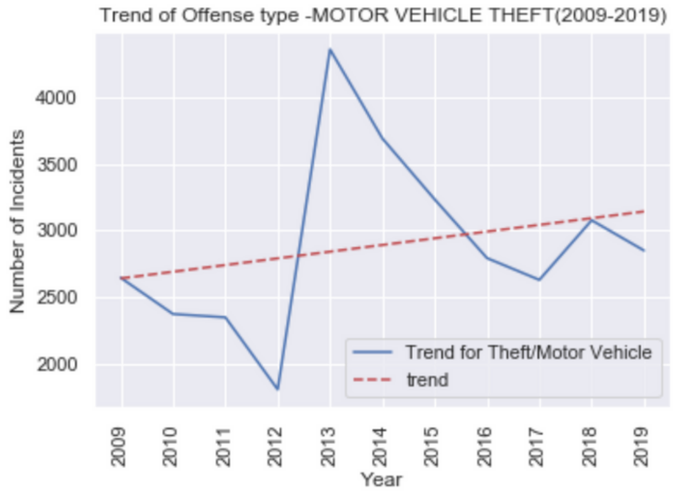
- Vast majority of crimes did not involve a weapon or something other than a gun or knife
- Others also includes crimes in which the weapon used was unknown

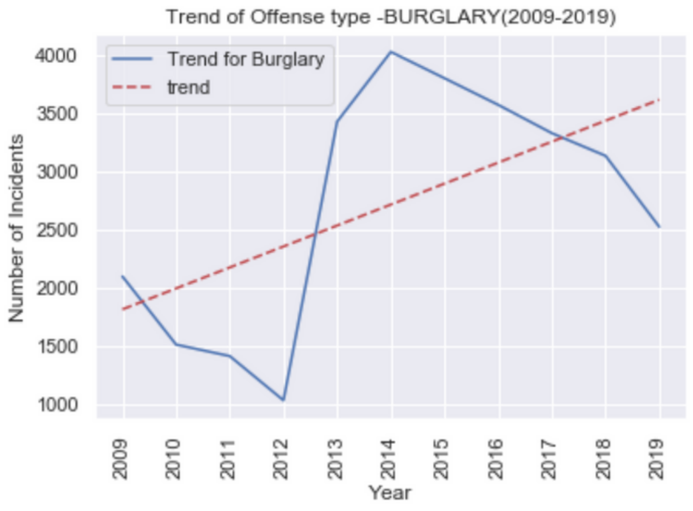
Trend Analysis

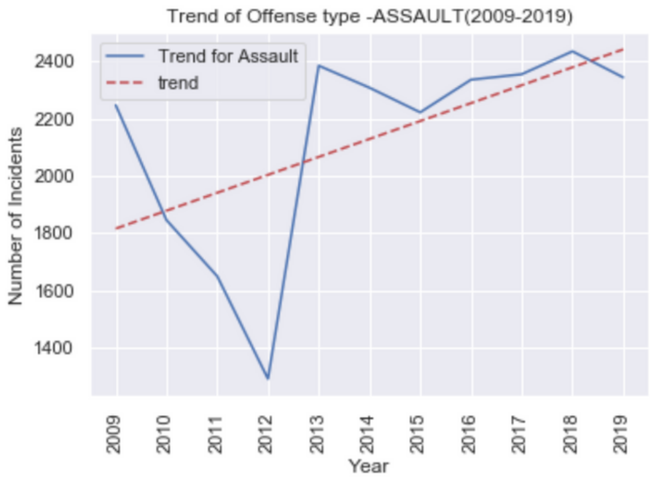




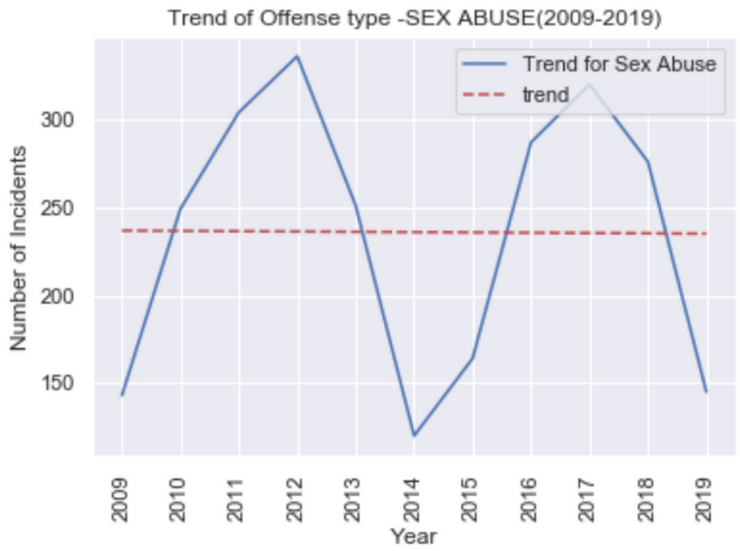


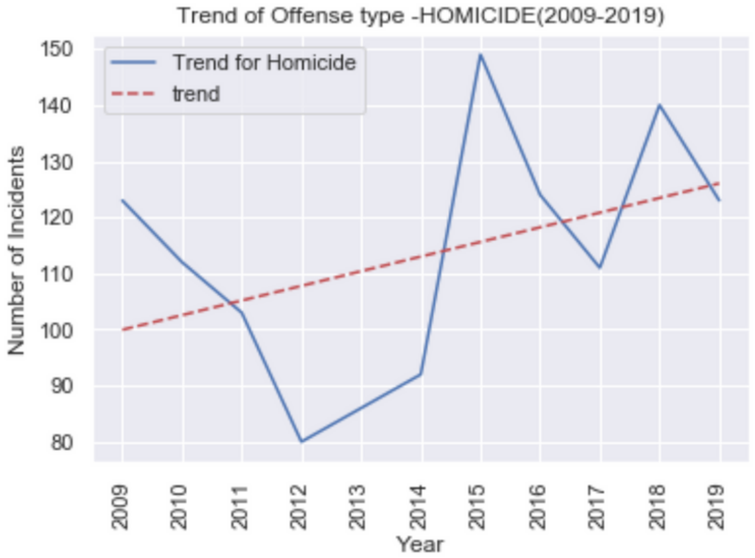


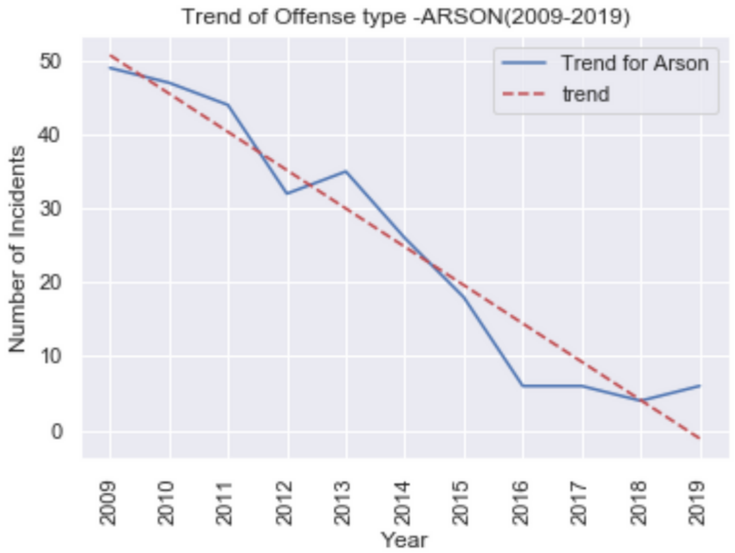


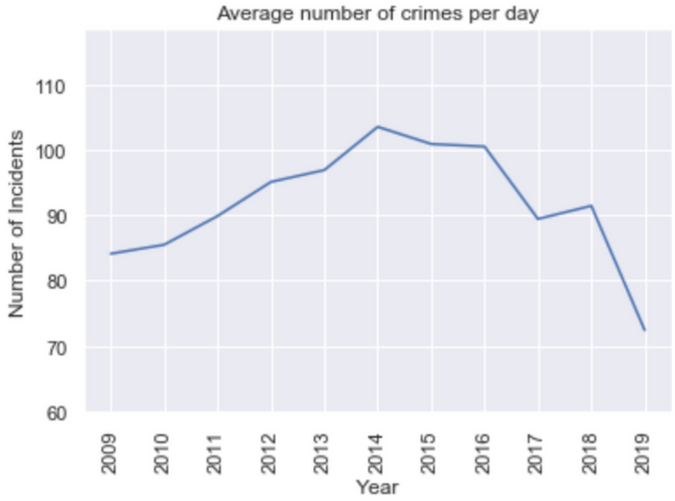


Trend Analysis









Conclusions & Observations

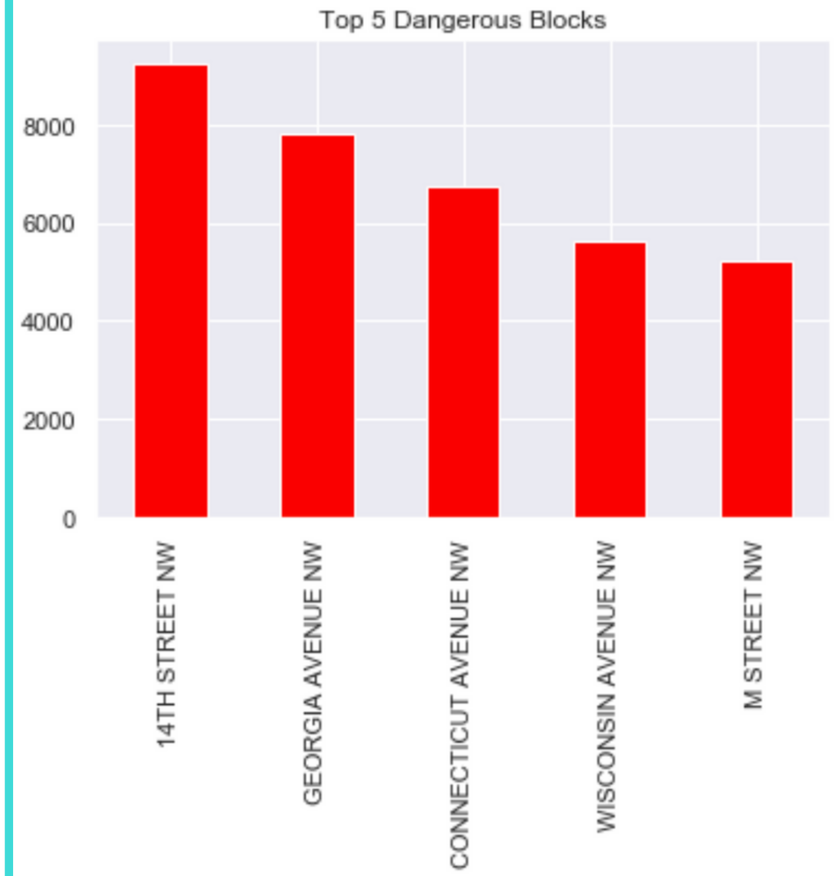
2 days

Average time to report a crime

Most frequent type of crime overall is

THEFT/OTHER

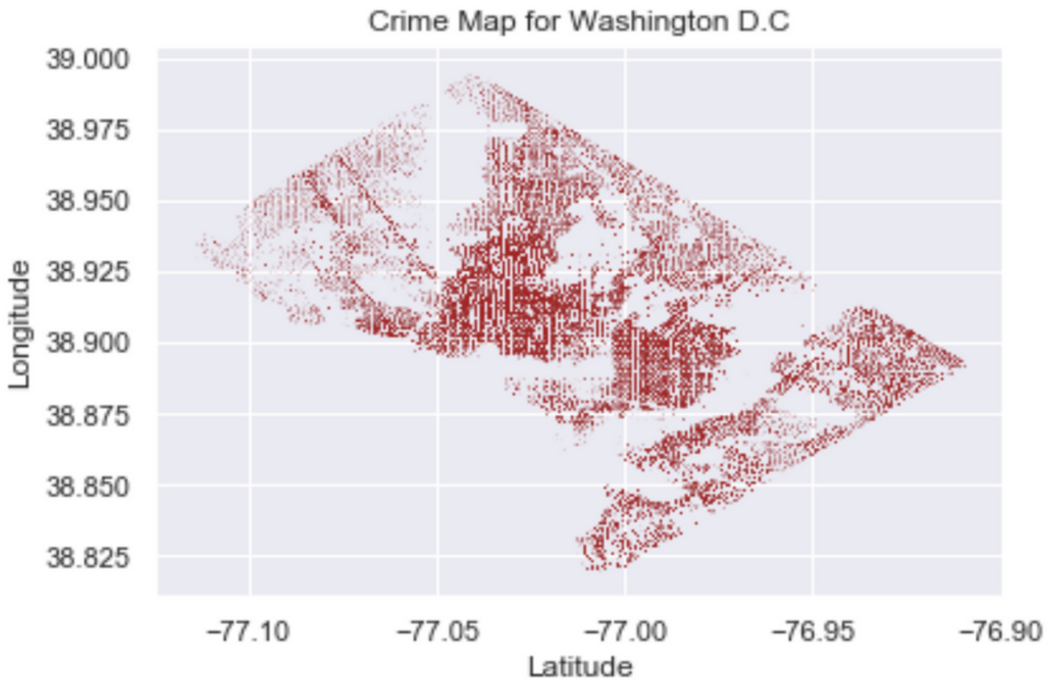
1. *Most crimes were committed during the **day***
2. *Most frequent type of crime during the **Night** is THEFT F/AUTO*
3. *Most frequent type of crime during the **Day** is THEFT/OTHER*



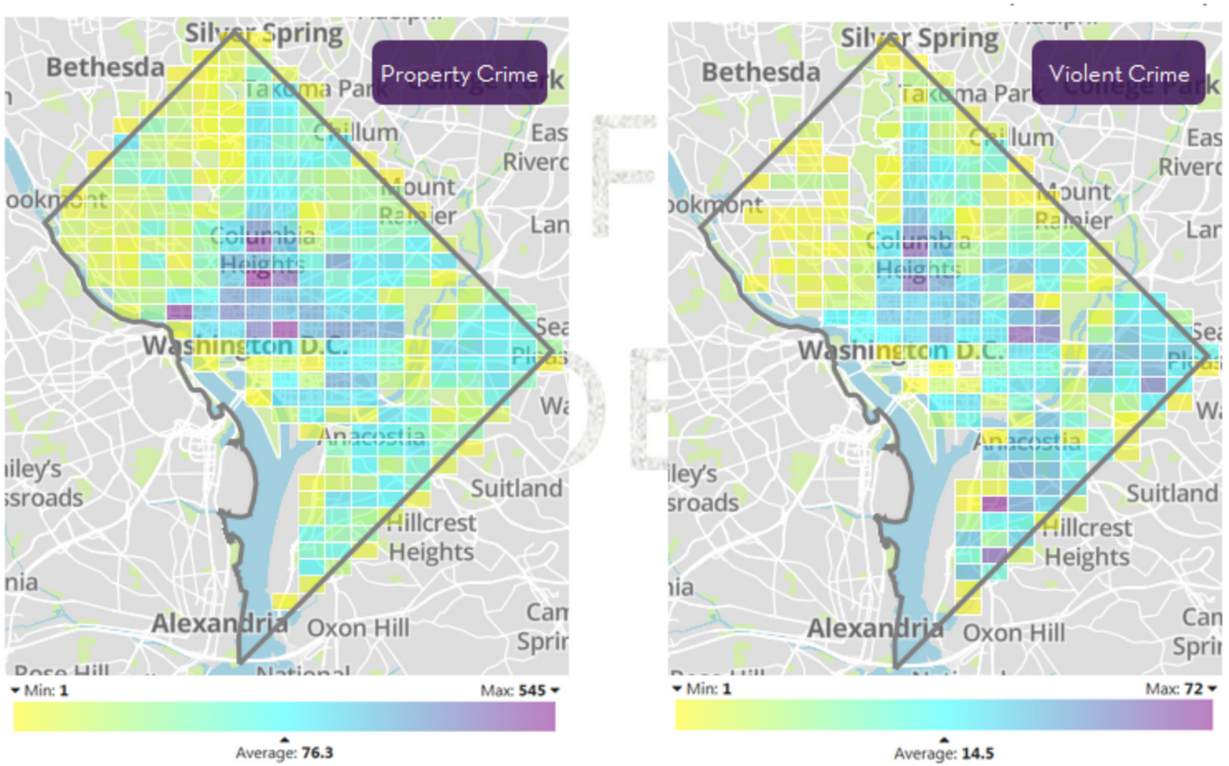
Based on the analysis, we are able to determine that the most dangerous block in D.C. is

14th St. NW

Conclusions & Observations



DC Crime Card Maps, 2017-2018



Source: Metropolitan Police Department Crime Map

Columbia Heights

The neighborhood where most of the crime happens

https://edscape.dc.gov/page/neighborhood-factors-crime-and-safety


Thanks!
Any Questions?

Sources

- *https://opendata.dc.gov/datasets/crime-incidents-in-2019?orderBy=REPORT_DAT*
- *https://edscape.dc.gov/page/neighborhood-factors-crime-and-safety*
- *Metadata: https://www.arcgis.com/sharing/rest/content/items/fdacfbdda7654e06a161352247d3a2f0/info/metadata/metadata.xml?format=default&output=html*