Crime Analysis

Table of contents

01

Introduction

Dataset Overview

The dataset comprises detailed information on various crime incidents reported within the city of Chicago from 2012 to 2017. Each record in the dataset represents a specific crime incident and includes attributes such as the unique identifier (ID) and case number assigned by the Chicago Police Department, the date and location of the incident, the type and description of the crime, arrest details, and geographic identifiers such as beat, district, ward, and community area.

Attributes

- ID: Unique identifier for each crime record.
- Case Number: Unique identifier assigned by the Chicago Police Department for each incident.
- Date: Date when the incident occurred, sometimes estimated.
- Block: Partially redacted address where the incident occurred, indicating the same block as the actual address.
- IUCR: Illinois Uniform Crime Reporting code linked to the Primary Type and Description
- Primary Type: Primary description of the crime.
- Description: Secondary description of the crimes, providing further details.
- Location Description: Description of where the incident occurred.
- Arrest: Indicates whether an arrest was made in connection with the incident.

Attributes

- Domestic: Indicates if the incident was domestic-related according to Illinois Domestic Violence Act.·
- Beat: Indicates the smallest police geographic area where the incident occurred.·
- District: Indicates the police district where the incident occurred.·
- Ward: City Council district where the incident occurred.·
- Community Area: Indicates the community area where the incident occurred.·
- FBI Code: Federal Bureau of Investigation code, is a system used to classify different types of crimes based on their characteristics

Attributes

- X Coordinate: X-coordinate of the incident location in State Plane Illinois East NAD 1983 projection.·
- Y Coordinate: Y-coordinate of the incident location in State Plane Illinois East NAD 1983 projection.·
- Year: Year when the incident occurred.·
- Updated On: Date and time when the record was last updated.·
- Latitude: Latitude of the incident location.·
- Longitude: Longitude of the incident location.·
- Location: Location of the incident in a format suitable for geographic operations and mapping.

02

Purpose

Purpose

Analyzing the Chicago Crime Dataset (2012-2017) helps us understand how crime changes over time and where it happens most often in the city. By looking at when and where crimes occur, we can figure out which areas need more attention from the police. We can also see how well the police are doing in catching criminals, especially in cases of domestic violence. This information helps city leaders make smarter decisions about how to keep people safe and reduce crime in Chicago.

03

Exploratory Data Analysis

Exploratory Data Analysis

1. Import the data
2. Displaying initial data

In [1]:
```
import pandas as pd
import numpy as np
import seaborn as sns
```

In [3]:
```
df = pd.read_csv("CrimeData.csv")
df.head()
```

```
C:\Users\disha\AppData\Local\Temp\ipykernel_21800\3344383920.py:1: DtypeWarning: Columns (9,10) have mixed types. Specify dt
ype option on import or set low_memory=False.
  df = pd.read_csv("CrimeData.csv")
```

Out[3]:

	Unnamed: 0	ID	Case Number	Date	Block	IUCR	Primary Type	Description	Location Description	Arrest	...	Ward	Community Area	FBI Code	X Coordinate	Coo
0	3.0	10508693.0	HZ250496	05-03-2016 23:40	013XX S SAWYER AVE	486	BATTERY	DOMESTIC BATTERY SIMPLE	APARTMENT	True	...	24.0	29.0	08B	1154907.0	189
1	89.0	10508695.0	HZ250409	05-03-2016 21:40	061XX S DREXEL AVE	486	BATTERY	DOMESTIC BATTERY SIMPLE	RESIDENCE	False	...	20.0	42.0	08B	1183066.0	186
2	197.0	10508697.0	HZ250503	05-03-2016 23:31	053XX W CHICAGO AVE	470	PUBLIC PEACE VIOLATION	RECKLESS CONDUCT	STREET	False	...	37.0	25.0	24	1140789.0	190
3	673.0	10508698.0	HZ250424	05-03-2016 22:10	049XX W FULTON ST	460	BATTERY	SIMPLE	SIDEWALK	False	...	28.0	25.0	08B	1143223.0	190
4	911.0	10508699.0	HZ250455	05-03-2016	003XX N LOTUS	820	THEFT	$500 AND UNDER	RESIDENCE	False	...	28.0	25.0	6	1139890.0	190

Exploratory Data Analysis

3. Displaying last data

```
In [3]: df.dropna(inplace=True)
        df.tail()
```

Out[3]:

	Unnamed: 0	ID	Case Number	Date	Block	IUCR	Primary Type	Description	Location Description	Arrest	...	Ward	Community Area	FBI Code	Coord
199994	2533684.0	8624881.0	HV298331	5/22/2012 17:30	016XX W 92ND PL	810	THEFT	OVER $500	RESIDENCE-GARAGE	False	...	21.0	73.0	6	1166
199995	2533685.0	8624882.0	HV298111	5/22/2012 15:40	133XX S LANGLEY AVE	460	BATTERY	SIMPLE	CHA PARKING LOT/GROUNDS	False	...	9.0	54.0	08B	1183
199996	2533686.0	8624884.0	HV297965	5/22/2012 12:45	002XX W 110TH PL	820	THEFT	$500 AND UNDER	RESIDENTIAL YARD (FRONT/BACK)	False	...	34.0	49.0	6	1176
199997	2533687.0	8624885.0	HV298273	5/6/2012 12:00	106XX S HARDING AVE	486	BATTERY	DOMESTIC BATTERY SIMPLE	RESIDENCE-GARAGE	False	...	19.0	74.0	08B	1151
199998	2533688.0	8624886.0	HV298416	5/22/2012 18:16	023XX N LOREL AVE	143A	WEAPONS VIOLATION	UNLAWFUL POSS OF HANDGUN	RESIDENTIAL YARD (FRONT/BACK)	True	...	37.0	19.0	15	1140

5 rows × 23 columns

Exploratory Data Analysis

4. Statistical summary of numerical columns

```
In [4]: df.describe()
```

Out[4]:

	Unnamed: 0	ID	Beat	District	Ward	Community Area	X Coordinate	Y Coordinate	Year	Latitud
count	1.924890e+05	1.924890e+05	192489.000000	192489.000000	192489.000000	192489.000000	1.924890e+05	1.924890e+05	192489.000000	192489.00000
mean	1.873710e+06	9.104666e+06	1155.949857	11.213358	22.647481	37.888456	1.164457e+06	1.884956e+06	2013.196681	41.83991
std	7.951918e+05	8.101823e+05	693.842145	6.898488	13.665015	21.405446	1.784708e+04	3.340824e+04	1.616702	0.09196
min	3.000000e+00	2.085900e+04	111.000000	1.000000	1.000000	0.000000	0.000000e+00	0.000000e+00	2012.000000	36.61944
25%	8.629250e+05	8.502760e+06	613.000000	6.000000	10.000000	23.000000	1.152440e+06	1.858253e+06	2012.000000	41.76628
50%	2.436033e+06	8.581773e+06	1023.000000	10.000000	22.000000	32.000000	1.165915e+06	1.890190e+06	2012.000000	41.85439
75%	2.485087e+06	9.953633e+06	1711.000000	16.000000	33.000000	58.000000	1.176388e+06	1.908404e+06	2015.000000	41.90440
max	2.533688e+06	1.054914e+07	2535.000000	31.000000	50.000000	77.000000	1.205119e+06	1.951523e+06	2016.000000	42.02257

Exploratory Data Analysis

5. Displaying total number of rows and columns

```
In [8]: df.shape

Out[8]: (192489, 23)
```

Exploratory Data Analysis

6. Displaying total number of rows in each column

```
In [9]: df.count()

Out[9]: Unnamed: 0            192489
        ID                    192489
        Case Number           192489
        Date                  192489
        Block                 192489
        IUCR                  192489
        Primary Type          192489
        Description           192489
        Location Description  192489
        Arrest                192489
        Domestic              192489
        Beat                  192489
        District              192489
        Ward                  192489
        Community Area        192489
        FBI Code              192489
        X Coordinate          192489
        Y Coordinate          192489
        Year                  192489
        Updated On            192489
        Latitude              192489
        Longitude             192489
        Location              192489
        dtype: int64
```

Exploratory Data Analysis

7. Displaying Data types of each column and the number of non-null values

```
In [12]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 192489 entries, 0 to 199998
Data columns (total 23 columns):
 #   Column                Non-Null Count   Dtype  
---  ------                --------------   -----  
 0   Unnamed: 0            192489 non-null  float64
 1   ID                    192489 non-null  float64
 2   Case Number           192489 non-null  object 
 3   Date                  192489 non-null  object 
 4   Block                 192489 non-null  object 
 5   IUCR                  192489 non-null  object 
 6   Primary Type          192489 non-null  object 
 7   Description           192489 non-null  object 
 8   Location Description  192489 non-null  object 
 9   Arrest                192489 non-null  object 
 10  Domestic              192489 non-null  object 
 11  Beat                  192489 non-null  float64
 12  District              192489 non-null  float64
 13  Ward                  192489 non-null  float64
 14  Community Area        192489 non-null  float64
 15  FBI Code              192489 non-null  object 
 16  X Coordinate          192489 non-null  float64
 17  Y Coordinate          192489 non-null  float64
 18  Year                  192489 non-null  float64
 19  Updated On            192489 non-null  object 
 20  Latitude              192489 non-null  float64
 21  Longitude             192489 non-null  float64
 22  Location              192489 non-null  object 
dtypes: float64(11), object(12)
memory usage: 35.2+ MB
```

Exploratory Data Analysis

8. Displaying categorical and numerical columns

In [8]:
```
cat_col = [col for col in df.columns if df[col].dtype == 'object']
print('Categorical columns:', cat_col)
num_col = [col for col in df.columns if df[col].dtype != 'object']
print('Numerical columns:', num_col)
```

```
Categorical columns: ['Case Number', 'Date', 'Block', 'IUCR', 'Primary Type', 'Description', 'Location Description', 'Arres
t', 'Domestic', 'FBI Code', 'Updated On', 'Location']
Numerical columns: ['Unnamed: 0', 'ID', 'Beat', 'District', 'Ward', 'Community Area', 'X Coordinate', 'Y Coordinate', 'Yea
r', 'Latitude', 'Longitude']
```

Exploratory Data Analysis

9. Displaying number of unique values in each categorical columns

```
In [9]: df[cat_col].nunique()

Out[9]: Case Number            199993
        Date                    84770
        Block                   26157
        IUCR                      325
        Primary Type               32
        Description               304
        Location Description      107
        Arrest                      2
        Domestic                    2
        FBI Code                   26
        Updated On                246
        Location                110187
        dtype: int64
```

04

Data Cleaning

Data cleaning

1. Calculate the total number of missing values in each column

```
In [19]: df.isnull().sum()
Out[19]: Unnamed: 0              0
         ID                      0
         Case Number             0
         Date                    0
         Block                   0
         IUCR                    0
         Primary Type            0
         Description             0
         Location Description    0
         Arrest                  0
         Domestic                0
         Beat                    0
         District                0
         Ward                    0
         Community Area          0
         FBI Code                0
         X Coordinate            0
         Y Coordinate            0
         Year                    0
         Updated On              0
         Latitude                0
         Longitude               0
         Location                0
         dtype: int64
```

Data cleaning

2. Identify duplicate rows in a dataframe

```
In [10]: df.duplicated()

Out[10]: 0         False
         1         False
         2         False
         3         False
         4         False
                   ...
         199994    False
         199995    False
         199996    False
         199997    False
         199998    False
         Length: 192489, dtype: bool

In [11]: df.duplicated().sum()

Out[11]: 0
```

Data cleaning

3. Removing unwanted columns

```
In [20]: df.drop(columns=['Unnamed: 0'], inplace=True)

In [21]: df.shape

Out[21]: (192488, 22)

In [ ]: df.columns

Out[52]: Index(['ID', 'Case Number', 'Date', 'Block', 'IUCR', 'Primary Type',
       'Description', 'Location Description', 'Arrest', 'Domestic', 'Beat',
       'District', 'Ward', 'Community Area', 'FBI Code', 'X Coordinate',
       'Y Coordinate', 'Year', 'Updated On', 'Latitude', 'Longitude',
       'Location'],
      dtype='object')
```

Analysis of Categorical columns

1. Count of each unique value in the ‘Primary Type’ column of the DataFrame, showing the frequency of Different types of crimes

```
In [ ]: print(df['Primary Type'].value_counts())
```

```
Primary Type
THEFT                                40390
BATTERY                              35503
NARCOTICS                            22137
CRIMINAL DAMAGE                      20588
ASSAULT                              12285
OTHER OFFENSE                        11650
BURGLARY                             10877
MOTOR VEHICLE THEFT                   8849
DECEPTIVE PRACTICE                    7953
ROBBERY                               6876
CRIMINAL TRESPASS                     5079
WEAPONS VIOLATION                     2372
PUBLIC PEACE VIOLATION                1770
OFFENSE INVOLVING CHILDREN            1390
PROSTITUTION                          1349
INTERFERENCE WITH PUBLIC OFFICER       825
CRIM SEXUAL ASSAULT                    760
SEX OFFENSE                            559
LIQUOR LAW VIOLATION                   326
ARSON                                  288
GAMBLING                               217
KIDNAPPING                             129
STALKING                               114
INTIMIDATION                            90
HOMICIDE                                68
OBSCENITY                               20
PUBLIC INDECENCY                         9
NON-CRIMINAL                             6
CONCEALED CARRY LICENSE VIOLATION        4
HUMAN TRAFFICKING                        2
```

Analysis of Categorical columns

1. Count of each unique value in the ‘Location’ column of the DataFrame, showing the frequency of Different types of locations where the crimes occurred

```
In [ ]: print(df['Location Description'].value_counts())
```

```
Location Description
STREET                          45363
RESIDENCE                       30508
APARTMENT                       24691
SIDEWALK                        21018
OTHER                            6878
                                ...
PORCH                               2
NEWSSTAND                           2
YARD                                2
PARKING LOT                         1
BARBER SHOP/BEAUTY SALON            1
Name: count, Length: 107, dtype: int64
```

05

Group By And Cross Tabulation

GroupBy

1. Displaying total number of arrets made for each type of crime

```
In [ ]:  df.groupby('Primary Type')['Arrest'].sum()
```

Out[61]:

Primary Type	
ARSON	33
ASSAULT	2956
BATTERY	8213
BURGLARY	617
CONCEALED CARRY LICENSE VIOLATION	4
CRIM SEXUAL ASSAULT	105
CRIMINAL DAMAGE	1463
CRIMINAL TRESPASS	3784
DECEPTIVE PRACTICE	1263
GAMBLING	217
HOMICIDE	18
HUMAN TRAFFICKING	0
INTERFERENCE WITH PUBLIC OFFICER	783
INTIMIDATION	25
KIDNAPPING	9
LIQUOR LAW VIOLATION	325
MOTOR VEHICLE THEFT	567
NARCOTICS	22093
NON - CRIMINAL	0
NON-CRIMINAL	0
OBSCENITY	17
OFFENSE INVOLVING CHILDREN	267
OTHER NARCOTIC VIOLATION	False
OTHER OFFENSE	2371
PROSTITUTION	1344
PUBLIC INDECENCY	9
PUBLIC PEACE VIOLATION	1342
ROBBERY	733

GroupBy

2. Displaying total number of arrets made in each police district

```
In [ ]: df.groupby('District')['Arrest'].sum()
```

```
Out[63]: District
1.0       2124
2.0       1788
3.0       2620
4.0       3018
5.0       2748
6.0       3646
7.0       3968
8.0       3785
9.0       2735
10.0      2634
11.0      6187
12.0      1967
13.0     False
14.0      1492
15.0      3703
16.0      1192
17.0      1213
18.0      1839
19.0      1686
20.0       790
22.0      1476
24.0      1429
25.0      3617
31.0         1
Name: Arrest, dtype: object
```

GroupBy

3. Displaying total number of domestic related incident in each police district

```
In [ ]: df.groupby('District')['Domestic'].sum()
```

```
Out[64]: District
1.0       420
2.0      1461
3.0      2150
4.0      2239
5.0      1686
6.0      2312
7.0      2289
8.0      2038
9.0      1434
10.0     1575
11.0     2221
12.0      963
13.0     True
14.0      733
15.0     1780
16.0      803
17.0      720
18.0      311
19.0      538
20.0      327
22.0     1038
24.0      784
25.0     1842
31.0        0
Name: Domestic, dtype: object
```

GroupBy

4. Displaying total number of arrest made in each year

```
In [ ]: df.groupby('Year')['Arrest'].sum()
```

```
Out[65]: Year
2012.0    36083
2013.0       34
2014.0       46
2015.0    14791
2016.0     4704
Name: Arrest, dtype: object
```

GroupBy

5. Displaying total no of arrest made at each type of location

```
df.groupby('Location Description')['Arrest'].sum()
```

```
Location Description
ABANDONED BUILDING                                298
AIRCRAFT                                            4
AIRPORT BUILDING NON-TERMINAL - NON-SECURE AREA     14
AIRPORT BUILDING NON-TERMINAL - SECURE AREA          8
AIRPORT EXTERIOR - NON-SECURE AREA                  18
                                                   ...
VEHICLE - OTHER RIDE SERVICE                        0
VEHICLE NON-COMMERCIAL                           1273
VEHICLE-COMMERCIAL                                 32
WAREHOUSE                                          34
YARD                                                0
Name: Arrest, Length: 107, dtype: object
```

GroupBy

6. Total no of arrest for each FBI crime code

```
df.groupby('FBI Code')['Arrest'].sum()
```

```
FBI Code
01A          16
01B           2
04A         951
04B         920
08A        2024
08B        7293
10          117
11         1107
12            4
13           35
14         1463
15         1917
16         1344
17          165
18        21043
19          218
2           118
20          169
22          325
24         2041
26         7412
3           733
5           617
6          5024
7           567
9            33
Name: Arrest, dtype: object
```

Crosstab

1. Displaying the frequency of arrest and non-arrest for each year

```
pd.crosstab(df['Year'], df['Arrest'])
```

Arrest	False	True
Year		
2012.0	86735	36083
2013.0	183	34
2014.0	487	46
2015.0	31824	14791
2016.0	17601	4704

Crosstab

2. Displaying the count of Arrest and non-arrest for each type Location and then select the 10 rows with the largest counts of arrest.

```
pd.crosstab(df['Location Description'], df['Arrest']).nlargest(10, columns=True)
```

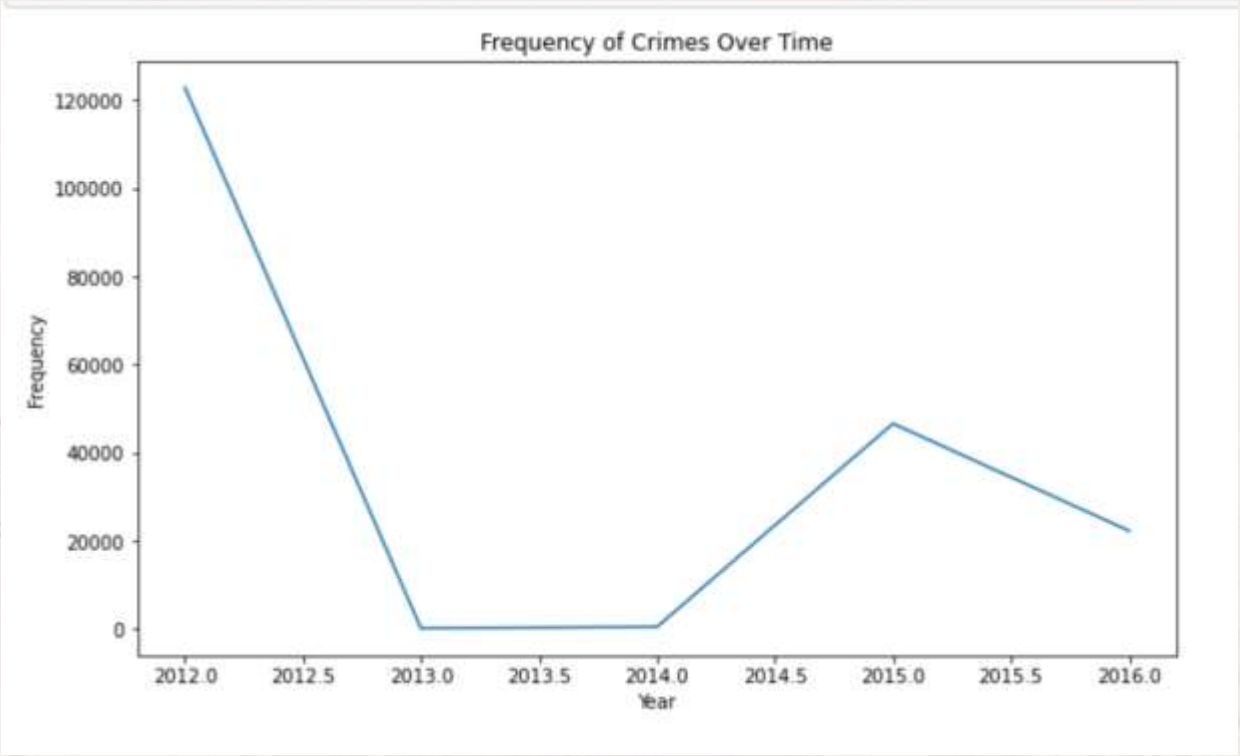
Arrest	False	True
Location Description		
STREET	32225	13138
SIDEWALK	9720	11298
APARTMENT	20273	4418
RESIDENCE	26401	4107
ALLEY	2064	2164
SCHOOL, PUBLIC, BUILDING	2998	1602
DEPARTMENT STORE	1033	1590
PARKING LOT/GARAGE(NON.RESID.)	4099	1523
VEHICLE NON-COMMERCIAL	1955	1273
GROCERY FOOD STORE	960	1270

06

Visualization

Visualization

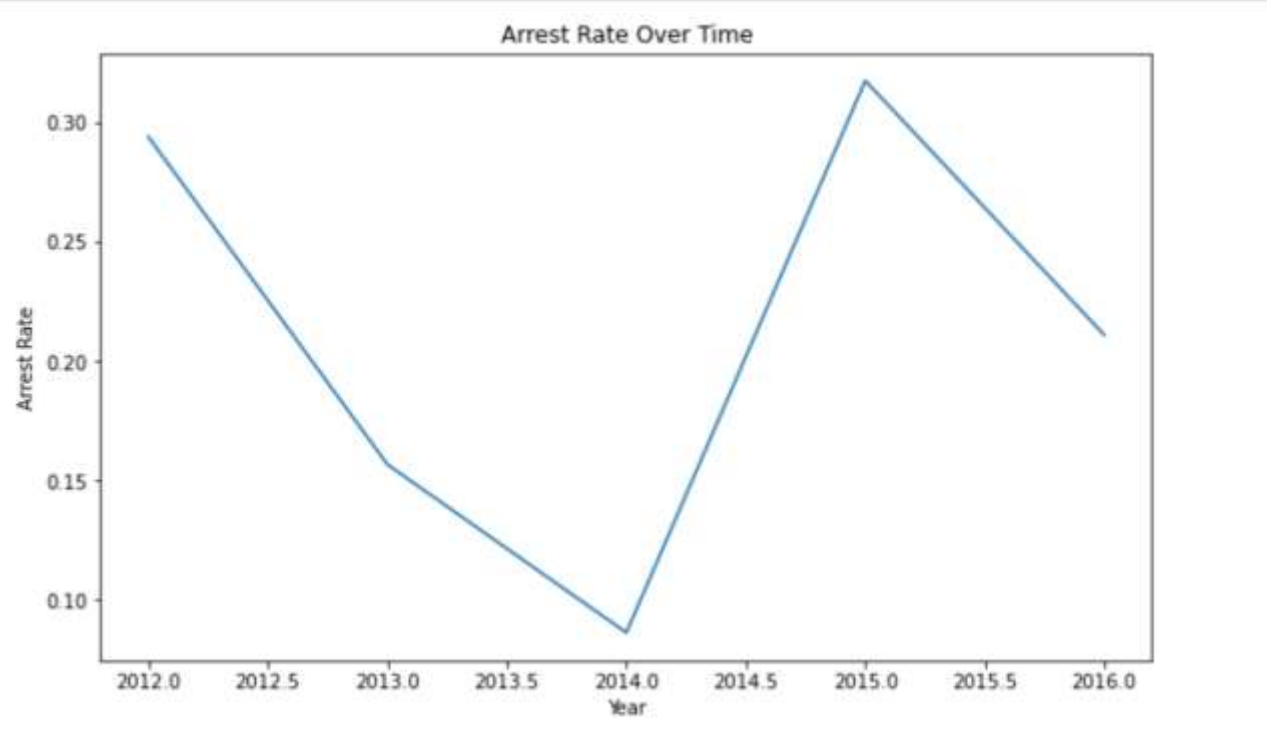
1. Frequency of Crimes Over Times:



The graph shows that crime was highest in 2012, but it dropped significantly in 2013 and 2014, reaching its lowest point during that time.

Visualization

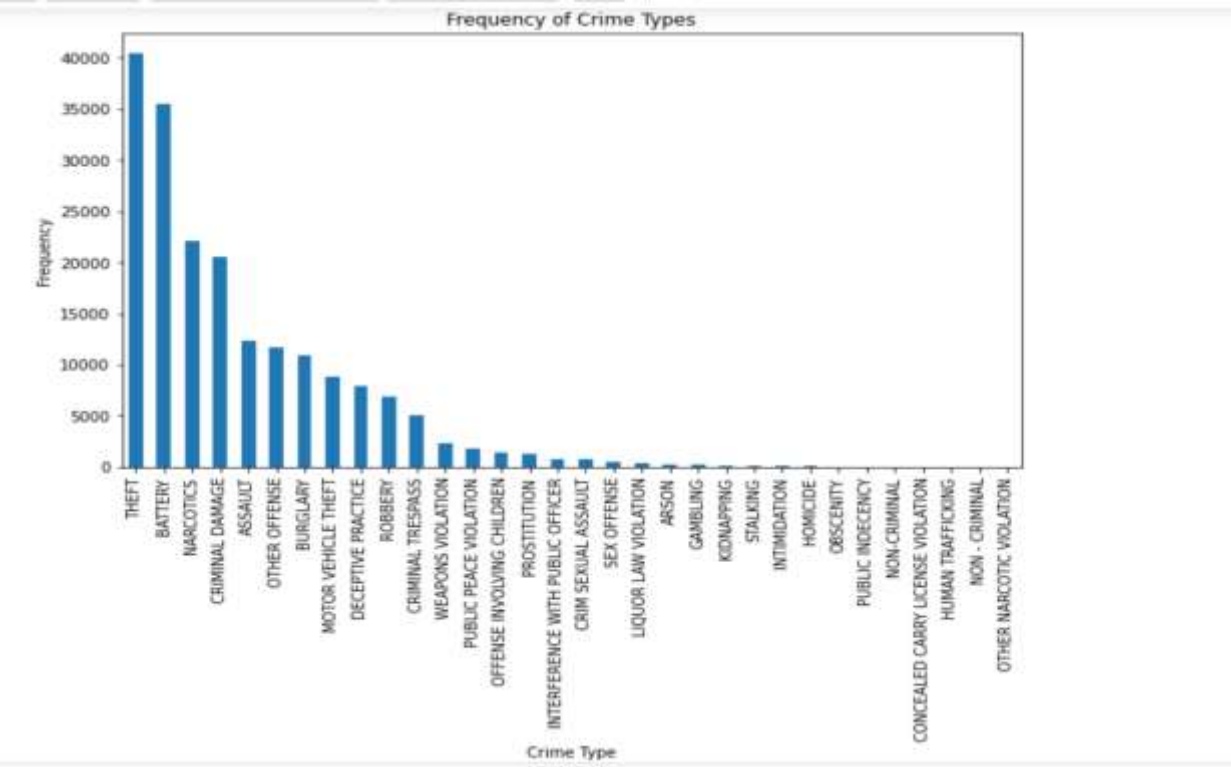
2. Arrest Rate Over Time



From this graph, we can see that the number of arrests was highest in the year 2015. On the other hand, the year 2014 had the fewest arrests compared to the other years shown in the graph.

Visualization

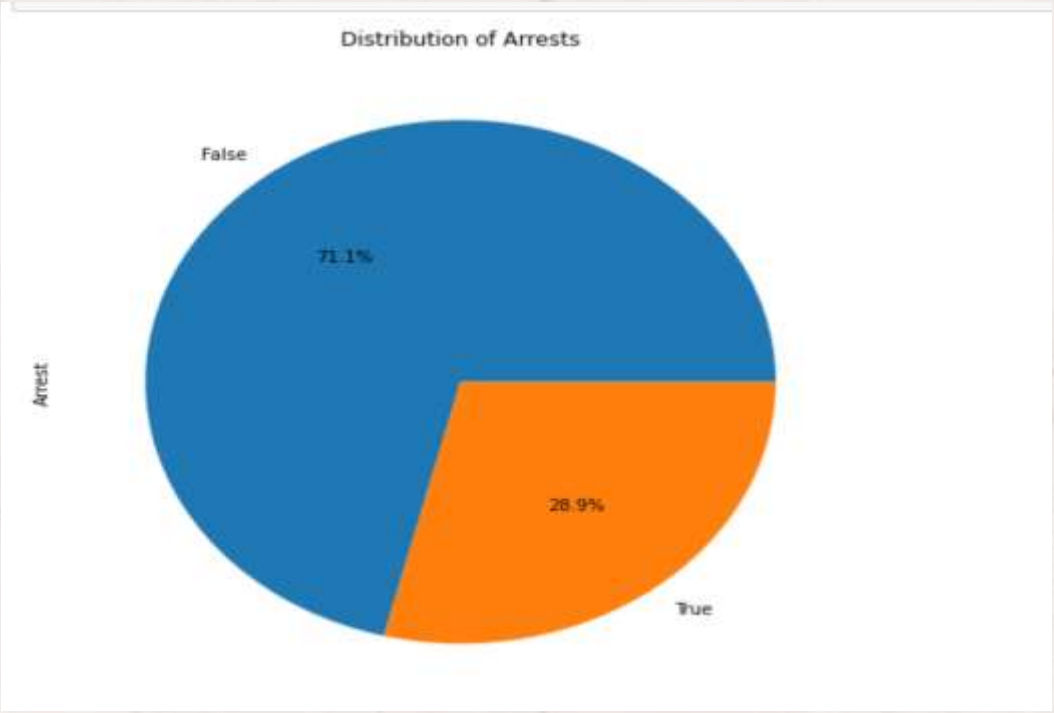
3. Frequency of Crime types



This graph tells us that theft was the most common type of crime that occurred.

Visualization

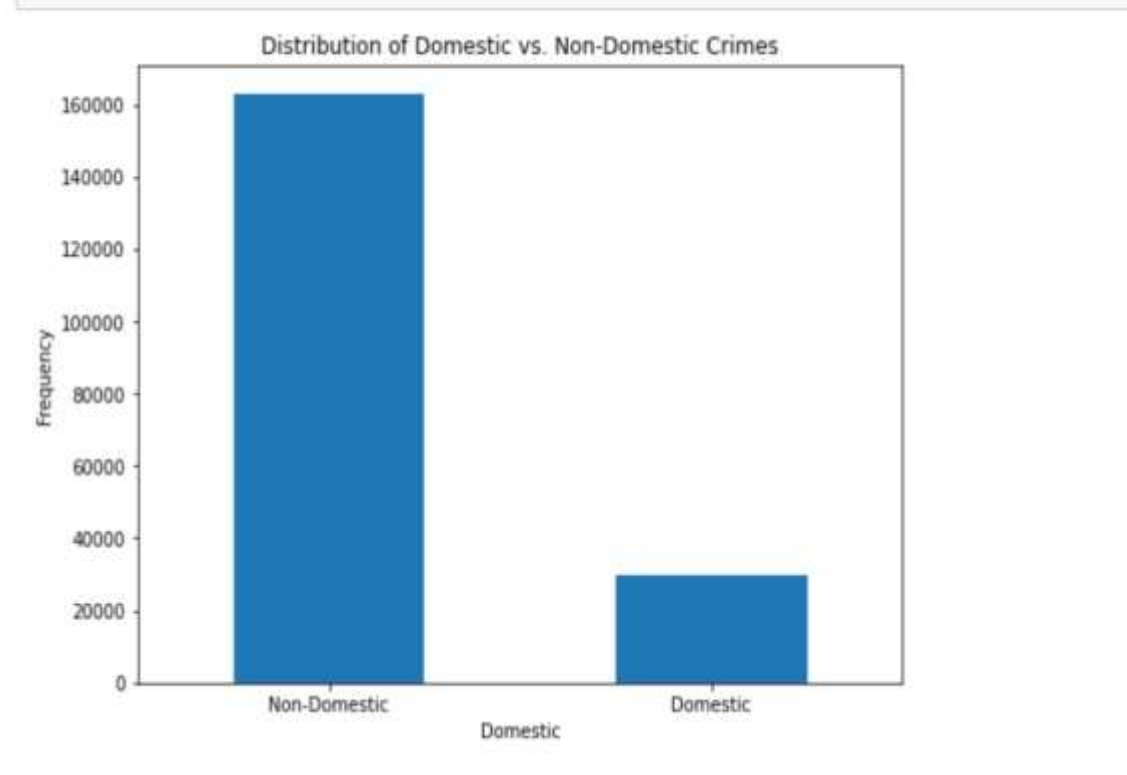
4. Distribution of Arrests



Looking at the pie chart, we can see that around 28.9% of the cases resulted in arrest, while the remaining 71.1% did not lead to arrest.

Visualization

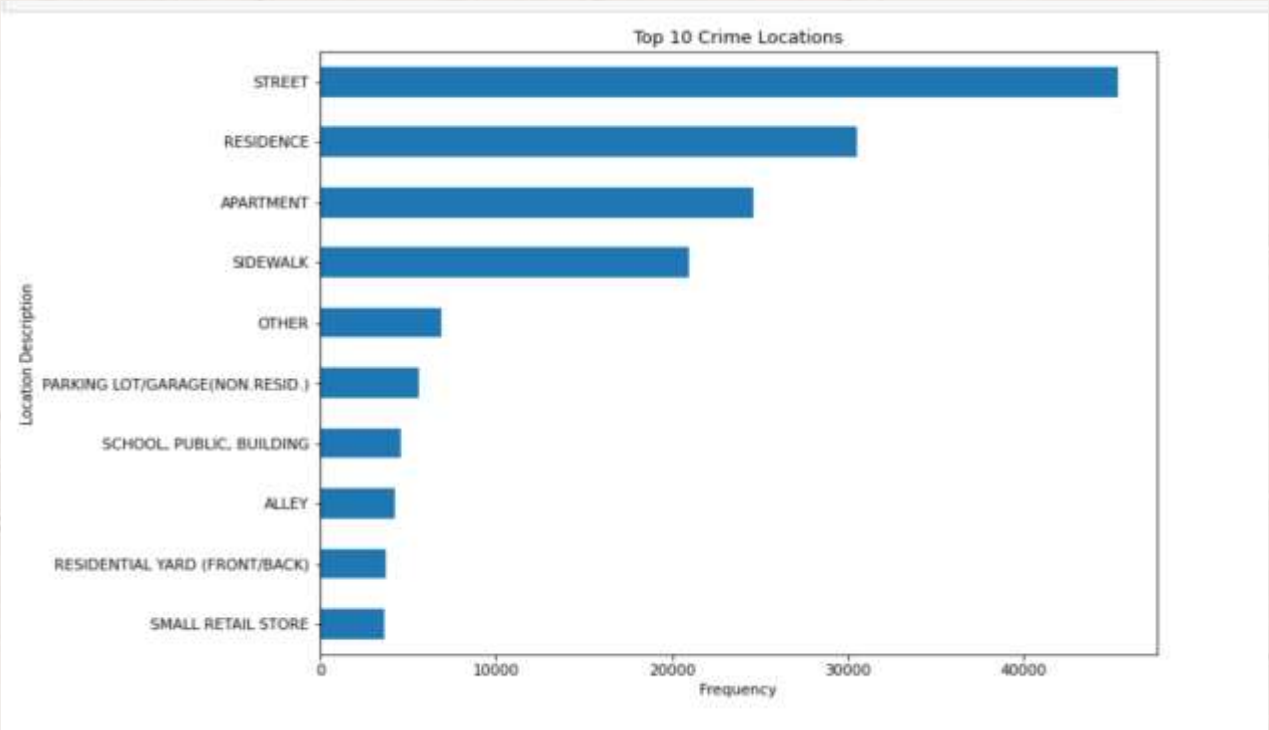
5. Distribution of Domestic vs Non-Domestic Crimes



This graph clearly shows that there are more non-domestic crimes compared to domestic ones.

Visualization

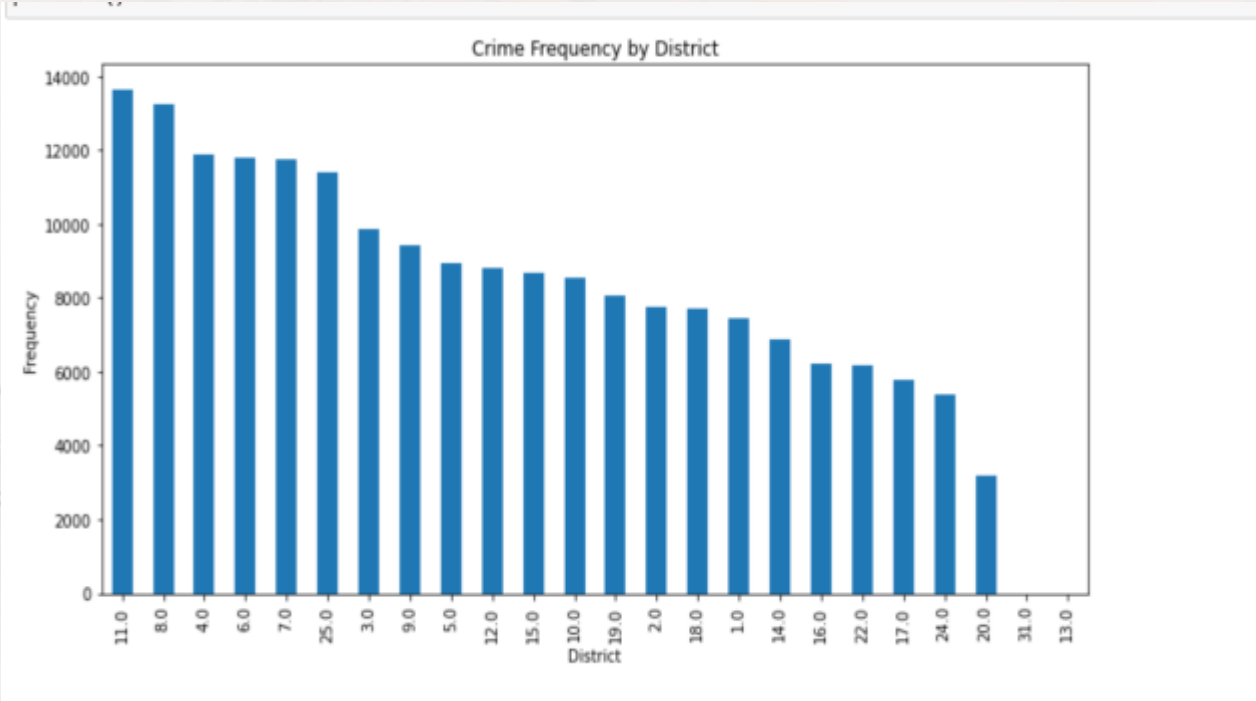
6. Top 10 Crime Locations



From this graph, it's evident that the most common location for incidents is on the street.

Visualization

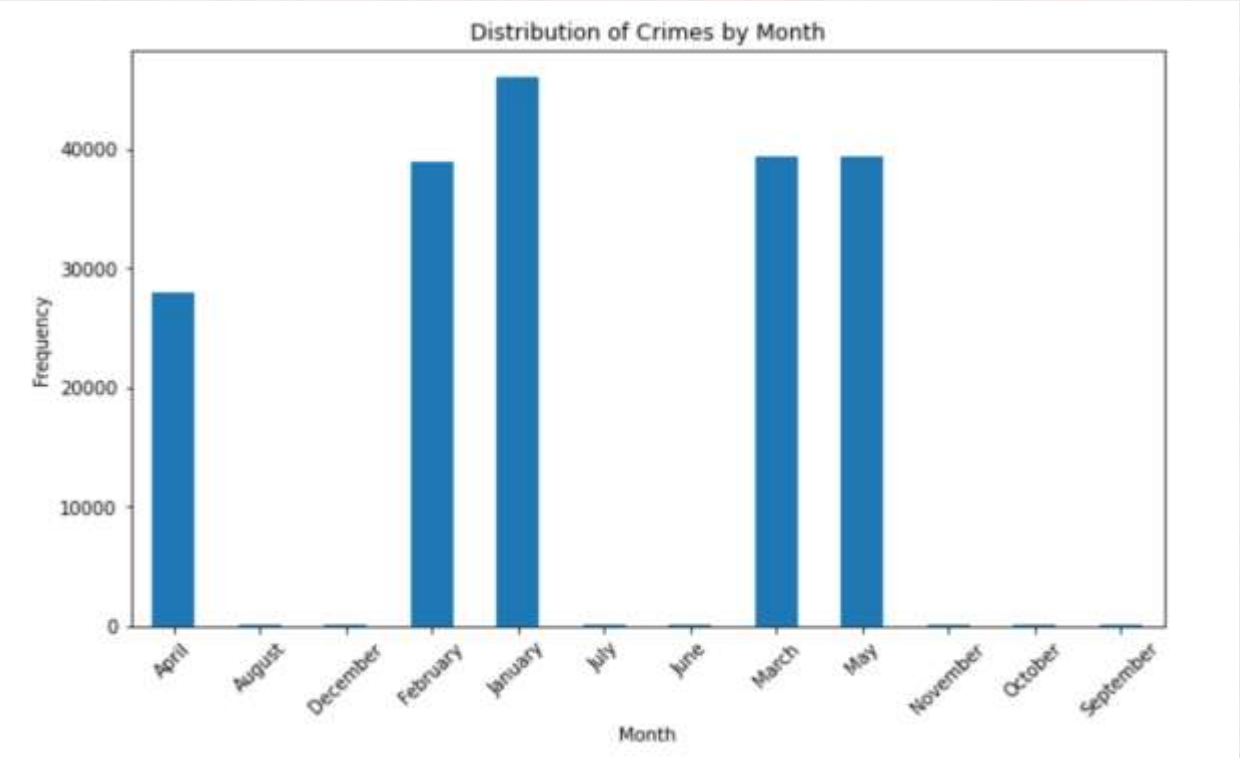
7. Crime Frequency by District



This graph indicates that District 11 experienced the highest occurrence of crimes compared to other districts.

Visualization

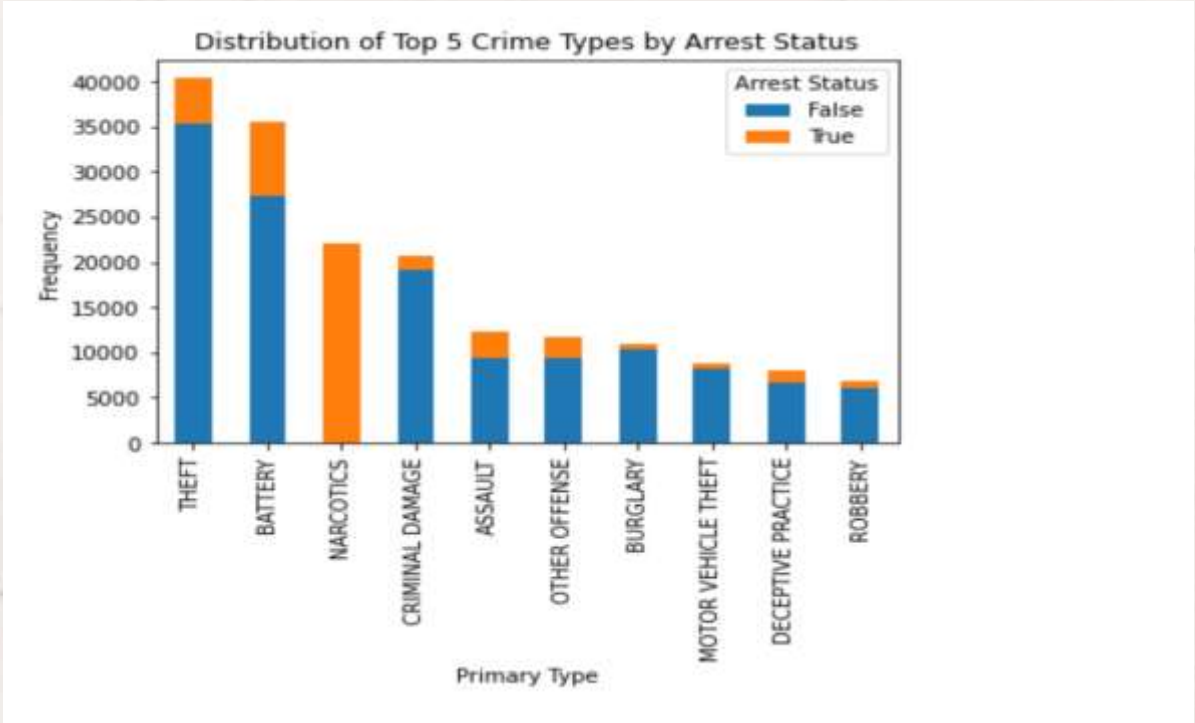
8. Distribution of Crimes by Month



This graph indicates that January had the highest number of reported crimes compared to other months.

Visualization

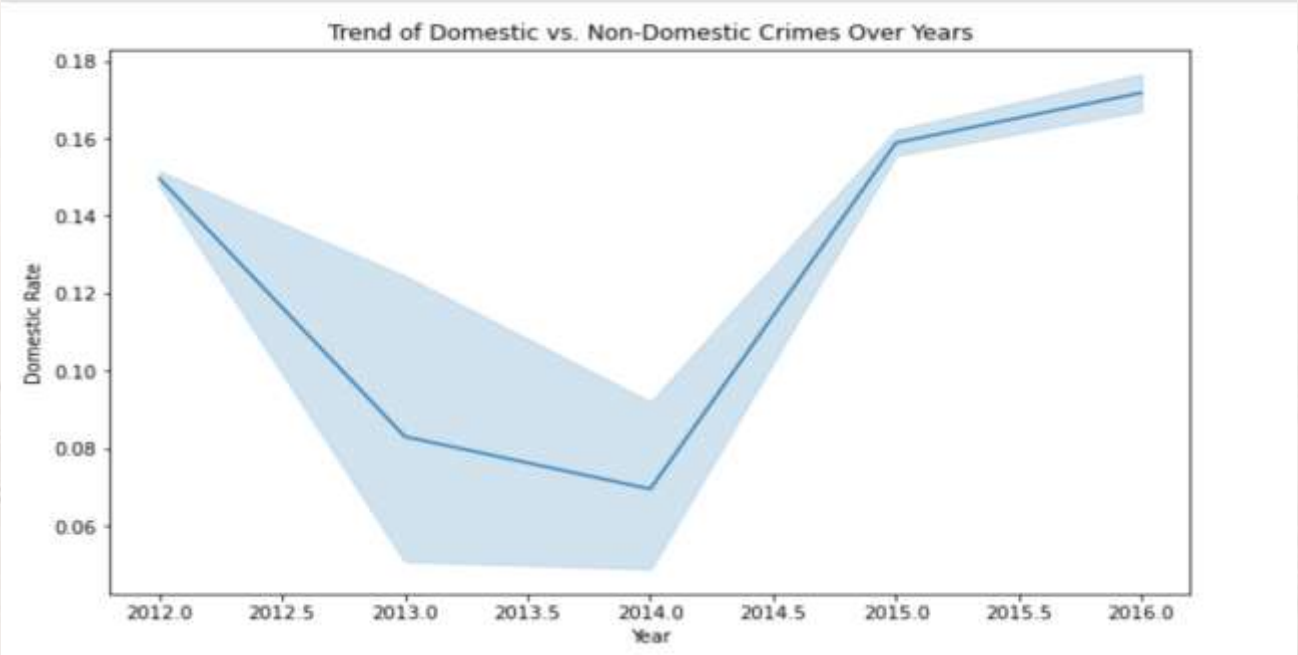
9. Distribution of Top 5 Crime Types by Arrest Status



From this graph, we can analyse that for the crime type "Narcotics," all individuals involved were arrested.

Visualization

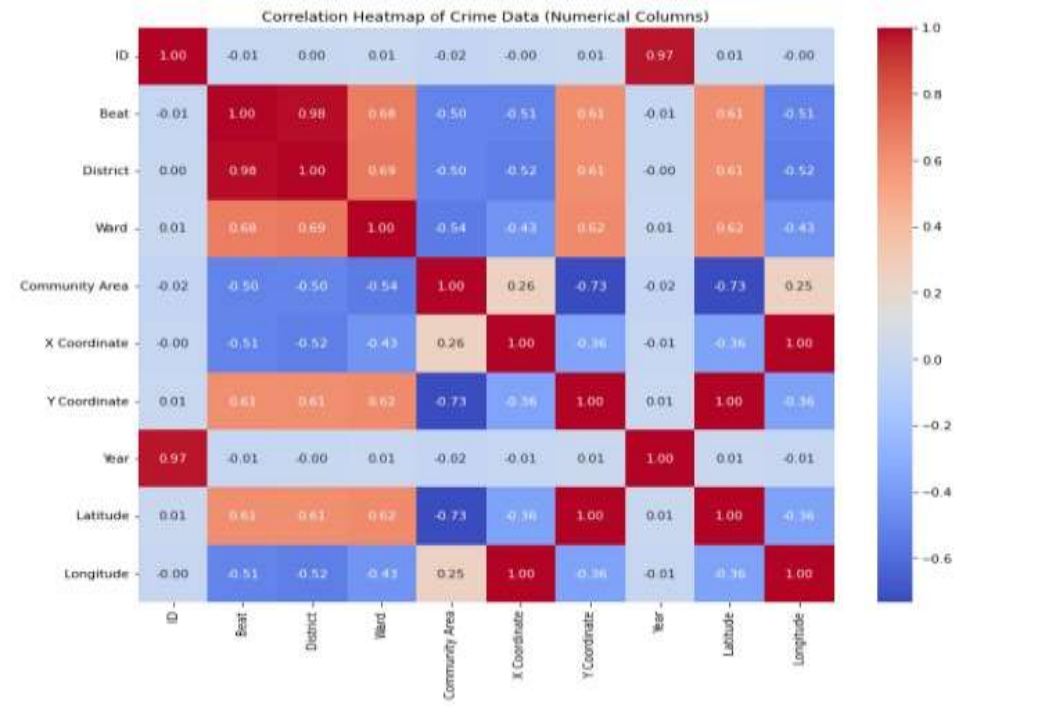
10. Trend of Domestic vs. Non-Domestic Crimes over



From this graph, we can analyze that the incidence of domestic crime peaked in the year 2016.

Visualization

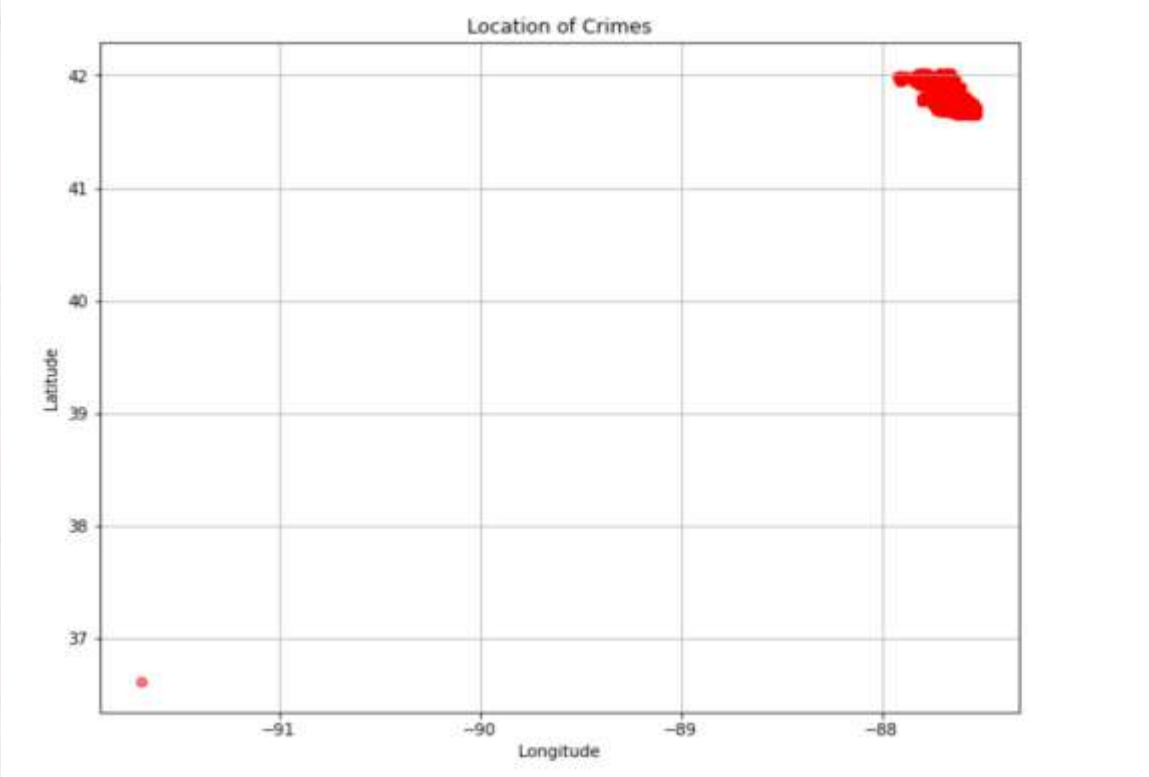
12. Correlation Heatmap of crime Data Numerical columns



When the correlation coefficient is close to 1, it indicates a strong positive correlation between the two attributes, meaning that as one attribute increases, the other tends to increase as well. Conversely, when the correlation coefficient is close to -1, it signifies a strong negative correlation, indicating that as one attribute increases, the other tends to decrease. A correlation coefficient near 0 suggests little to no linear relationship between the attributes.

Visualization

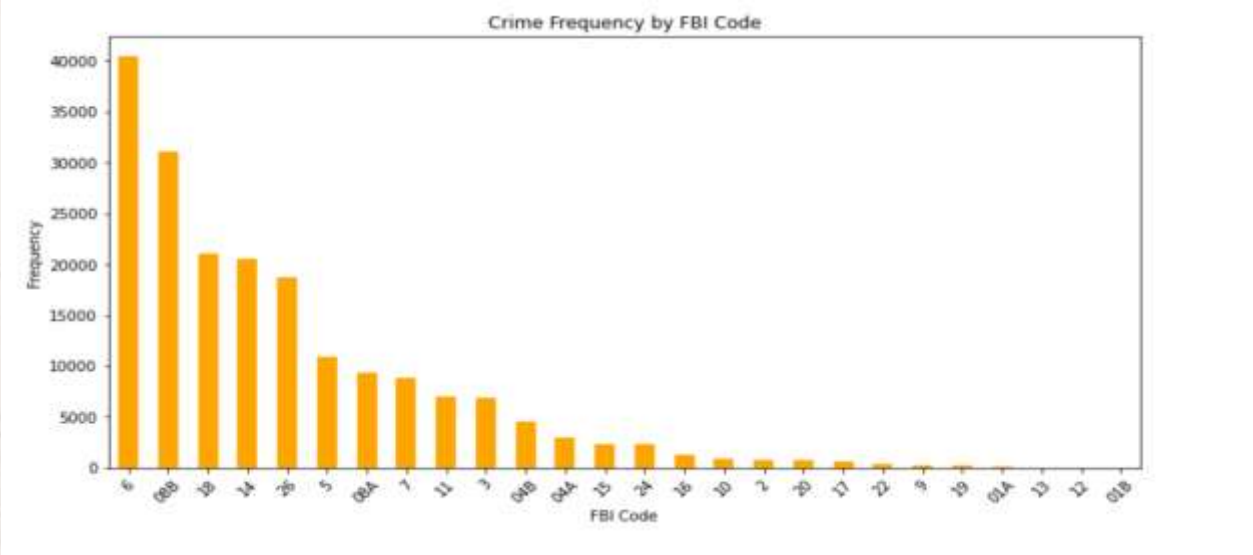
13. Location of crimes



The scatter plot displays the locations of crimes based on latitude and longitude. Each point represents a single crime incident. The dark shade color in the top-right corner indicates a higher concentration of crime incidents in that particular area.

Visualization

14. Crime Frequency by FBI code



From this graph, it indicates that there are more occurrences of crimes classified under the FBI code 6 compared to other FBI codes.

Thank You