1.0 Introduction

1.1 Project Background

Crime is a significant concern in many urban areas, including Chicago. According to U.S. News & World Report (n.d.), Chicago's overall crime rate, especially the violent crime rate, is higher than the US average. Gangs in Chicago have a role in the city's crime rate. The city has struggled with various types of crimes, including violent crimes, property crimes, and drug-related offenses. This project aims to analyze crime data in Chicago to understand the patterns and trends, identify the root causes, and propose potential solutions to reduce crime rates.

1.2 Problem Statement

The dataset consists of various attributes such as incident ID, date, type of crime, location, and arrest to determine the outcomes of crimes. Despite its availability, this data is often unstructured and not effectively utilized to derive meaningful insights that can help in crime prevention and policy formulation.

1.3 Objectives

The objectives of this project are:

- To visualize crime data based on various attributes.
- To analyze trends and patterns in crime over the years.
- To identify high-crime areas and the types of crimes prevalent in those areas.
- To propose data-driven strategies for crime reduction.
- To present the analysis and support it with data visualization.

2.0 Problem Identification

To better understand and analyze our dataset, our team has implemented the 5 Whys method for root cause identification. This method involves iteratively asking "why" to explore the cause-and-effect relationships underlying a particular problem. We opted for this approach because it allows us to delve deeply into the origins of any issue we encounter. Moreover, it fosters a collaborative environment where team members can contribute ideas for continuous improvement. By utilizing the 5 Whys technique, we build confidence in our ability to address and resolve problems effectively, ensuring the robustness of our analysis process.

Who	The dataset is provided by the Chicago Police Department (CPD). It includes reported incidents of crime in the City of Chicago sourced from the CLEAR (Citizen Law Enforcement Analysis and Reporting)
What	The dataset consists of reported incidents of crime, excluding murders, where data exists for each victim.
When	The data spans from 2012-2017.
Where	The dataset can be found at Kaggle.
Why	It provides the public with insights into reported crime trends and patterns within Chicago. The objective of this project is to identify patterns and root causes of crimes to develop effective prevention strategies.

3.0 Potential Solutions

The high crime rates in Chicago have far-reaching implications for the safety and well-being of its residents. The root causes, as identified through the 5 Whys technique, include systemic inequality, lack of education, and economic opportunities. Various solutions can be implemented to address these underlying issues.

There are 4 possible solutions :

1. **Increase Funding for Education and Job Training Programs**
    a. Reason: Improving access to quality education and job training can provide residents with the skills necessary to secure better employment, thereby reducing poverty and crime rates. Education is a key factor in promoting social mobility and1 economic stability.
    b. Implementation: Allocate more resources to schools in underserved areas, develop vocational training centers, and partner with local businesses to provide apprenticeships and internships.
2. **Community Policing and Violence Prevention Programs**
    a. Reason: Community policing fosters trust between law enforcement and the community, which can lead to better cooperation in crime prevention. Violence prevention programs can address the immediate causes of violent behavior.
    b. Implementation: Train police officers in community engagement techniques, establish neighborhood watch programs, and create intervention programs for at-risk youth.
3. **Improve Economic Opportunities through Business Incentives and Support**
    a. Reason: Economic development can revitalize communities by creating jobs and improving living conditions. Incentives for businesses can encourage investment in high-crime areas.
    b. Implementation: Offer tax breaks and grants to businesses that open in underserved areas, provide support for small business startups, and create job fairs and employment resource centers.
4. **Enhance Mental Health and Social Services**
    a. Reason: Addressing mental health issues and providing social services can reduce the factors that contribute to criminal behavior. Mental health support can help individuals manage stress and avoid negative coping mechanisms.

b. Implementation: Increase funding for mental health clinics, provide training for social workers, and integrate mental health services into schools and community centers.

Based on the potential solution, we **recommend** implementing **targeted economic development programs**. The dataset provides detailed information on crime incidents, including location, time, and type of crime. By analyzing this data, we can identify high-crime areas and correlate them with socioeconomic factors such as unemployment and poverty rates. Implementing targeted economic development programs addresses the root causes of crime by creating job opportunities and improving living conditions. This approach reduces crime rates and promotes long-term economic stability and community development.

There a few **strategies to implement our solution** which is :

1. Data analysis and identification:

Objective: Identify high-crime areas and correlate these with socioeconomic indicators.

Steps:

I. Utilize the crime dataset to map crime incidents by location.
II. Integrate additional datasets that provide information on unemployment rates, poverty levels, and other relevant socioeconomic factors.
III. Use statistical analysis to determine the correlation between crime rates and socioeconomic indicators.

2. Developing economic development programs:

Objective: Create programs that target economic growth and job creation in high-crime areas.

Steps:

I. Collaborate with local government and community organizations to design economic development initiatives.
II. Focus on areas identified as high-crime zones with significant socioeconomic challenges.
III. Prioritize initiatives such as small business grants, vocational training, and infrastructure development.

3.

4. **Business Incentives and Support:**

   Objective: Encourage businesses to invest in high-crime areas.

   Steps:

   I. Offer tax breaks and financial incentives to businesses that establish operations in targeted areas.
   II. Provide support for small business startups through grants, loans, and mentoring programs.
   III. Organize job fairs and employment resource centers to connect residents with job opportunities.

5. **Monitoring and Evaluation:**

   Objective: Assess the effectiveness of the economic development programs.

   Steps:

   I. Establish key performance indicators (KPIs) to measure the impact on crime rates and economic conditions.
   II. Conduct regular evaluations to track progress and make necessary adjustments to the programs.
   III. Engage with community members to gather feedback and ensure the programs are meeting their needs.

The expected outcomes of these programs include a reduction in crime rates by providing improved economic opportunities that decrease the motivation for criminal behavior. Additionally, economic growth will result from increased business activity and job creation, leading to the economic revitalization of targeted areas. Furthermore, these enhanced economic conditions will contribute to a better quality of life, elevating living standards and overall community well-being.

4.0 Finding Data

Data Source Selection

To develop a comprehensive understanding of crime in Chicago and the factors contributing to it, we need to identify and utilize relevant datasets. The chosen dataset for this analysis is from Kaggle, titled Chicago Crime Visualization (https://www.kaggle.com/datasets/currie32/crimes-in-chicago?select=Chicago_Crimes_2012_to_2017.csv). This dataset provides detailed information on crime incidents in Chicago, including the type of crime, location, date, and outcomes.

Dataset Overview

The Chicago Crime dataset includes the following key variables:

- **ID:** Unique identifier for each crime incident.
- **Case Number:** Case number assigned to the crime incident.
- **Date:** The date and time when the crime was reported.
- **Block:** The block where the crime occurred.
- **IUCR:** Illinois Uniform Crime Reporting code for the crime.
- **Primary Type:** The specific category of crime (e.g., theft, assault).
- **Description:** Detailed description of the crime.
- **Location Description:** Description of the location where the crime occurred.
- **Arrest:** Indicates the outcomes whether an arrest was made.
- **Domestic:** Indicates whether the crime was domestic-related.
- **Beat:** The police beat where the crime occurred.
- **District:** The police district where the crime occurred.
- **Ward:** The ward where the crime occurred.
- **Community Area:** The community area where the crime occurred.
- **FBI Code:** The FBI code for the crime.
- **X Coordinate:** X coordinate of the location.
- **Y Coordinate:** Y coordinate of the location.

- **Year:** The year when the crime was reported.
- **Update on:** The date when the record was last updated.
- **Latitude:** Latitude of the location.
- **Longitude:** Longitude of the location.
- **Location:** The full location coordinates.

Data Relevance

The dataset is highly relevant for our analysis as it allows us to:

- Analyze trends in crime over time.
- Identify high-crime areas and types of crimes prevalent in different neighbourhoods.
- Examine the outcomes of crime incidents to understand the effectiveness of law enforcement.

Additional Data Sources

To enrich our analysis, we will integrate additional datasets that provide context on socioeconomic factors. Potential sources include:

- **Demographic Data:** Information on population demographics, income levels, and education attainment from sources like the U.S. Census Bureau.
- **Economic Data:** Employment rates, business establishments, and economic activities from sources like the Bureau of Labor Statistics.
- **Educational Data:** School performance and funding levels from sources like the National Center for Education Statistics.

Data Integration

Integrating these datasets with the crime data will enable a more comprehensive analysis. We can explore correlations between crime rates and factors such as poverty, education, and economic opportunities. This integrated approach will provide a holistic view of the underlying causes of crime and support the development of targeted interventions.

5.0 Data Management towards Quality Data.

5.1 Data Definition

VARIABLE	MEANING	DATA TYPE	SCALE
ID	Unique ID for each crime incident.	int	1-271,354
Case Number	Every incident has a distinct case number.	object	-
Date	When the crime was reported.	object	2012-2017
Block	Where the incident took place.	object	“001XX S HALSTED ST”
IUCR	Illinois Uniform Crime Reporting code.	object	3 or 4 digits code (e.g., “0110”)
Primary Type	The crime’s primary description.	object	-
Description	Description of the crime.	object	-
Location Description	Description of the crime’s location.	object	-
Arrest	Indicates whether or not an arrest was made.	object	true/false
Domestic	Indicates whether the incident was domestic related.	object	true/false
Beat	Beat is where the scene of the incident occurred.	int	111-1933
District	District is where the	object	1-25

	scene of the incident occurred.		
Ward	Which event took place.	object	1-50
Community Area	The location of the occurrence in the community.	object	1-77
FBI Code	The specific crime's FBI Code.	object	-
X Coordinate	The location's X coordinates.	object	-
Y Coordinate	The location's Y coordinates.	object	-
Year	The year the incident happened.	object	2012-2017
Update on	The date when the record was last updated.	object	-
Latitude	Latitude of the location.	object	-
Longitude	Longitude of the location.	object	-
Location	The full location coordinates.	object	-

5.2 Preparing Data

1. Importing all required libraries.

```
[1]: #preparing the data
     #importing libraries
     import pandas as pd
     import numpy as np
     import seaborn as sns
     import matplotlib.pyplot as plt
     from sklearn.model_selection import train_test_split
     from sklearn.metrics import accuracy_score
     from sklearn.metrics import classification_report
     import re
     import string
     import random as rnd
     import seaborn as sb
     from nltk.corpus import stopwords
     from nltk.stem import PorterStemmer
     import re
     from sklearn.metrics import classification_report
     from sklearn.model_selection import train_test_split
     from sklearn.naive_bayes import MultinomialNB
     from sklearn.ensemble import ExtraTreesClassifier
     from datetime import datetime
     import seaborn as sns
     import plotly.graph_objs as go
     import plotly.express as px
     from plotly.subplots import make_subplots
     from collections import Counter
     import matplotlib.pyplot as plt
     from matplotlib.pyplot import pie, axis, show
     from matplotlib import rcParams
     import math
     import matplotlib.pyplot as plt
     from matplotlib.gridspec import GridSpec
     %matplotlib inline
     from sklearn.model_selection import train_test_split
     from sklearn.linear_model import LogisticRegression
     from sklearn.svm import SVC, LinearSVC
     from sklearn.ensemble import RandomForestClassifier
```

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn. linear_model import Perceptron
from sklearn.linear_model import SGDClassifier
from sklearn.tree import DecisionTreeClassifier
from nltk.stem import WordNetLemmatizer
from sklearn.model_selection import train_test_split
from sklearn import metrics
import nltk
from nltk.corpus import wordnet
```

2. Preparing the data frame file.

```
# Locating the dataframe file and displaying some basic information of the data
crimes_dataframe = pd.read_csv('Chicago_Crimes_2012_to_2017.csv')
crimes_dataframe.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1418365 entries, 0 to 1418364
Data columns (total 23 columns):
 #   Column                Non-Null Count    Dtype
---  ------                --------------    -----
 0   Unnamed: 0            1418365 non-null  int64
 1   ID                    1418365 non-null  int64
 2   Case Number           1418365 non-null  object
 3   Date                  1418365 non-null  object
 4   Block                 1418365 non-null  object
 5   IUCR                  1418365 non-null  object
 6   Primary Type          1418365 non-null  object
 7   Description           1418365 non-null  object
 8   Location Description  1418365 non-null  object
 9   Arrest                1418365 non-null  bool
 10  Domestic              1418365 non-null  bool
 11  Beat                  1418365 non-null  int64
 12  District              1418365 non-null  float64
 13  Ward                  1418365 non-null  float64
 14  Community Area        1418365 non-null  float64
 15  FBI Code              1418365 non-null  object
 16  X Coordinate          1418365 non-null  float64
 17  Y Coordinate          1418365 non-null  float64
 18  Year                  1418365 non-null  int64
 19  Updated On            1418365 non-null  object
 20  Latitude              1418365 non-null  float64
 21  Longitude             1418365 non-null  float64
 22  Location              1418365 non-null  object
dtypes: bool(2), float64(7), int64(4), object(10)
memory usage: 230.0+ MB
```

3. Displaying all the column names (feature names) available in crime_df.

```
# printing out all the feature names
print(crimes_dataframe.columns.values)
```

```
['Unnamed: 0' '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']
```

4. Displaying the first five records of crime_df.

```
# displaying the first five records of the dataframe
crimes_dataframe.head()
```

	Unnamed: 0	ID	Case Number	Date	Block	IUCR	Primary Type	Description	Location Description	Arrest	...	Ward	Community Area	FBI Code	X Coordinate	Y Coordinate	
0	3	10508693	HZ250496	05/03/2016 11:40:00 PM	013XX S SAWYER AVE	0486	BATTERY	DOMESTIC BATTERY SIMPLE	APARTMENT	True	...	24.0	29.0	08B	1154907.0	1893681.0	2
1	89	10508695	HZ250409	05/03/2016 09:40:00 PM	061XX S DREXEL AVE	0486	BATTERY	DOMESTIC BATTERY SIMPLE	RESIDENCE	False	...	20.0	42.0	08B	1183066.0	1864330.0	2
2	197	10508697	HZ250503	05/03/2016 11:31:00 PM	053XX W CHICAGO AVE	0470	PUBLIC PEACE VIOLATION	RECKLESS CONDUCT	STREET	False	...	37.0	25.0	24	1140789.0	1904819.0	2
3	673	10508698	HZ250424	05/03/2016 10:10:00 PM	049XX W FULTON ST	0460	BATTERY	SIMPLE	SIDEWALK	False	...	28.0	25.0	08B	1143223.0	1901475.0	2
4	911	10508699	HZ250455	05/03/2016 10:00:00 PM	003XX N LOTUS AVE	0820	THEFT	$500 AND UNDER	RESIDENCE	False	...	28.0	25.0	06	1139890.0	1901675.0	2

5 rows × 23 columns

5. Displaying the last five records of crime_df.

```
# displaying the last five records of the dataframe
crimes_dataframe.tail()
```

	Unnamed: 0	ID	Case Number	Date	Block	IUCR	Primary Type	Description	Location Description	Arrest	...	Ward	Community Area	FBI Code	Coo
1418360	6250330	10508679	HZ250507	05/03/2016 11:33:00 PM	026XX W 23RD PL	0486	BATTERY	DOMESTIC BATTERY SIMPLE	APARTMENT	True	...	28.0	30.0	08B	11
1418361	6251089	10508680	HZ250491	05/03/2016 11:30:00 PM	073XX S HARVARD AVE	1310	CRIMINAL DAMAGE	TO PROPERTY	APARTMENT	True	...	17.0	69.0	14	11
1418362	6251349	10508681	HZ250479	05/03/2016 12:15:00 AM	024XX W 63RD ST	041A	BATTERY	AGGRAVATED: HANDGUN	SIDEWALK	False	...	15.0	66.0	04B	11
1418363	6253257	10508690	HZ250370	05/03/2016 09:07:00 PM	082XX S EXCHANGE AVE	0486	BATTERY	DOMESTIC BATTERY SIMPLE	SIDEWALK	False	...	7.0	46.0	08B	11
1418364	6253474	10508692	HZ250517	05/03/2016 11:38:00 PM	001XX E 75TH ST	5007	OTHER OFFENSE	OTHER WEAPONS VIOLATION	PARKING LOT/GARAGE(NON.RESID.)	True	...	6.0	69.0	26	11

5 rows × 23 columns

6. Displaying some basic statistical information of crime_df.

```
# displaying some basic statistical information of the dataframe
crimes_dataframe.describe(include=['O'])
```

	Case Number	Date	Block	IUCR	Primary Type	Description	Location Description	FBI Code	Updated On	Location
count	1418365	1418365	1418365	1418365	1418365	1418365	1418365	1418365	1418365	1418365
unique	1418258	571105	32546	363	33	340	140	26	950	368079
top	HZ140230	01/01/2012 12:01:00 AM	001XX N STATE ST	0820	THEFT	SIMPLE	STREET	06	02/04/2016 06:33:39 AM	(41.883500187, -87.627876698)
freq	6	126	3567	132848	321950	147845	325084	321950	906327	2093

5.3 Data Cleaning and Sandardization

7. Display basic statistical information for every column in each dataset.

```
#Display basic statistical information for every column in each dataset
crimes_df.describe(include='all')
```

	Unnamed: 0	ID	Case Number	Date	Block	IUCR	Primary Type	Description	Location Description	Arrest	...	Ward	Community Area	
count	1.418365e+06	1.418365e+06	1418365	1418365	1418365	1418365	1418365	1418365	1418365	1418365	...	1.418365e+06	1.418365e+06	141
unique	NaN	NaN	1418258	571105	32546	363	33	340	140	2	...	NaN	NaN	
top	NaN	NaN	HZ140230	01/01/2012 12:01:00 AM	001XX N STATE ST	0820	THEFT	SIMPLE	STREET	False	...	NaN	NaN	
freq	NaN	NaN	6	126	3567	132848	321950	147845	325084	1047308	...	NaN	NaN	3:
mean	3.273045e+06	9.574675e+06	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	2.285474e+01	3.747455e+01	
std	1.183059e+06	8.011218e+05	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	1.379378e+01	2.142995e+01	
min	3.000000e+00	2.022400e+04	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	1.000000e+00	0.000000e+00	
25%	2.696813e+06	8.987180e+06	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	1.000000e+01	2.300000e+01	
50%	3.052281e+06	9.575732e+06	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	2.300000e+01	3.200000e+01	
75%	3.409553e+06	1.019113e+07	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	3.400000e+01	5.700000e+01	
max	6.253474e+06	1.082334e+07	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	5.000000e+01	7.700000e+01	

11 rows × 23 columns

8. Displaying some information of the crimes based on the crimes types.

```
# displaying the information of crimes based on the crime types
# we also require this information as we are about to clean the data by removing any duplicate countries and NaN values
crimes_dataframe.groupby(by=["Description"], dropna=False).count()
```

Description	Unnamed: 0	ID	Case Number	Date	Block	IUCR	Primary Type	Location Description	Arrest	Domestic	...	Ward	Community Area	FBI Code	X Coordinate
$500 AND UNDER	132848	132848	132848	132848	132848	132848	132848	132848	132848	132848	...	132848	132848	132848	132848
ABUSE/NEGLECT: CARE FACILITY	32	32	32	32	32	32	32	32	32	32	...	32	32	32	32
AGG CRIM SEX ABUSE FAM MEMBER	537	537	537	537	537	537	537	537	537	537	...	537	537	537	537
AGG CRIMINAL SEXUAL ABUSE	858	858	858	858	858	858	858	858	858	858	...	858	858	858	858
AGG PO HANDS ETC SERIOUS INJ	80	80	80	80	80	80	80	80	80	80	...	80	80	80	80
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
VIOLATION OF STALKING NO CONTACT ORDER	105	105	105	105	105	105	105	105	105	105	...	105	105	105	105
VIOLATION OF SUMMARY CLOSURE	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1
VIOLENT OFFENDER: ANNUAL REGISTRATION	86	86	86	86	86	86	86	86	86	86	...	86	86	86	86
VIOLENT OFFENDER: DUTY TO REGISTER	22	22	22	22	22	22	22	22	22	22	...	22	22	22	22
VIOLENT OFFENDER: FAIL TO REGISTER NEW ADDRESS	14	14	14	14	14	14	14	14	14	14	...	14	14	14	14

We require this information as we are about to clean the data by removing any duplicate and NaN values.

9. Dropping all the NaN values in the data frame.

```
# dropping all Nan values in the dataframe
crimes_dataframe.dropna(inplace = True)
```

There are a lot of NaN values and we need to remove them from the data frame before we can start analysing and visualising them in graphs.

10. Displaying some information of the crimes based on the crimes types NaN values has been cleaned up.

```
# displaying the information of crimes based on the crime types that has drop the Nan values
# we also require this information to check whether we have performed data cleaning or have not
crimes_dataframe.groupby(by=["Description"], dropna=False).count()
```

Description	Unnamed: 0	ID	Case Number	Date	Block	IUCR	Primary Type	Location Description	Arrest	Domestic	...	Ward	Community Area	FBI Code	X Coordinate
$500 AND UNDER	132848	132848	132848	132848	132848	132848	132848	132848	132848	132848	...	132848	132848	132848	132848
ABUSE/NEGLECT: CARE FACILITY	32	32	32	32	32	32	32	32	32	32	...	32	32	32	32
AGG CRIM SEX ABUSE FAM MEMBER	537	537	537	537	537	537	537	537	537	537	...	537	537	537	537
AGG CRIMINAL SEXUAL ABUSE	858	858	858	858	858	858	858	858	858	858	...	858	858	858	858
AGG PO HANDS ETC SERIOUS INJ	80	80	80	80	80	80	80	80	80	80	...	80	80	80	80
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
VIOLATION OF STALKING NO CONTACT ORDER	105	105	105	105	105	105	105	105	105	105	...	105	105	105	105
VIOLATION OF SUMMARY CLOSURE	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1
VIOLENT OFFENDER: ANNUAL REGISTRATION	86	86	86	86	86	86	86	86	86	86	...	86	86	86	86
VIOLENT OFFENDER: DUTY TO REGISTER	22	22	22	22	22	22	22	22	22	22	...	22	22	22	22
VIOLENT OFFENDER: FAIL TO REGISTER NEW ADDRESS	14	14	14	14	14	14	14	14	14	14	...	14	14	14	14

340 rows × 22 columns

11. Checking for missing values in crime_df.

```
# Check for missing values
print("\nMissing Values in the Dataset:")
print(crimes_df.isnull().sum())
```

```
Missing Values in the Dataset:
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
```

12. Display the counting of all the records of the data frame to see any duplicate records.

```
# counting the records of the dataframe
# we also need this information to check whether there is a duplicate record or not
crimes_dataframe.value_counts()
```

```
Unnamed: 0  ID       Case Number  Date                  Block                IUCR  Primary Type        Description                    Location D
escription           Arrest  Domestic  Beat  District  Ward  Community Area  FBI Code  X Coordinate  Y Coordinate  Year  Updated On             Lati
tude   Longitude   Location
3           10508693  HZ250496     05/03/2016 11:40:00 PM  013XX S SAWYER AVE     0486  BATTERY             DOMESTIC BATTERY SIMPLE        APARTMENT
True    True      1022  10.0      24.0  29.0            08B       1154907.0     1893681.0     2016  05/10/2016 03:56:50 PM  41.864073  -87.706819  (4
1.864073157, -87.706818608)    1
3289148     9862700   HX512386     11/18/2014 06:30:00 PM  080XX S GREEN ST       0610  BURGLARY            FORCIBLE ENTRY                 RESIDENCE-
GARAGE               False   False     621   6.0       21.0  71.0            05        1172033.0     1851612.0     2014  02/04/2016 06:33:39 AM  41.7
48271  -87.645186  (41.748271333, -87.645186436)    1
3289156     9862709   HX512439     11/19/2014 11:00:00 AM  000XX W RANDOLPH ST    1330  CRIMINAL TRESPASS   TO LAND                        RESTAURANT
True    False     111   1.0       42.0  32.0            26        1176157.0     1901285.0     2014  02/04/2016 06:33:39 AM  41.884487  -87.628582  (4
1.884486956, -87.628581948)    1
3289155     9862708   HX512248     11/19/2014 08:40:00 AM  078XX S MUSKEGON AVE   0820  THEFT               $500 AND UNDER                 APARTMENT
False   True      421   4.0       7.0   43.0            06        1196484.0     1853528.0     2014  02/04/2016 06:33:39 AM  41.752957  -87.555528  (4
1.752957436, -87.55552768)     1
3289154     9862707   HX512471     11/19/2014 12:10:00 PM  012XX N LOCKWOOD AVE   2092  NARCOTICS           SOLICIT NARCOTICS ON PUBLICWAY  SIDEWALK
True    False     2532  25.0      37.0  25.0            26        1140801.0     1907643.0     2014  02/04/2016 06:33:39 AM  41.902657  -87.758258  (4
1.902657453  -87.758258361)    1
```

13. Drop any duplicate records in the data frame.

```
[11]: # removing duplicate records
      crimes_dataframe.drop_duplicates(inplace = True)
```

14. Display the counting of all records of the data frame again to see if there any duplicate records left.

```
# counting the records of the dataframe
# also, we need this information to check whether there is a duplicate record or not
# As you can see, there is no duplicate record found anymore in this dataframe
crimes_dataframe.value_counts()
crimes_dataframe.to_csv('Chicago_Crimes_2012_to_2017.csv', index=False)
```

There is no results after running the cell so there is no duplicate record found anymore in this dataframe.

5.4 Exploring Data

15. Count the number of general arrest in ascending order.

```
[14]: #Count the number of general arrest rows in ascending order
      crimes_df.groupby(["Arrest"]).size().reset_index(name='counts').sort_values(by=["counts"], ascending=False).head(10)
```

[14]:

	Arrest	counts
0	False	1047308
1	True	371057

16. Displaying some basic statistic about the crimes in general .

```
# counting the records of the dataframe
# also, we need this information to check whether there is a duplicate record or not
# As you can see, there is no duplicate record found anymore in this dataframe
crimes_dataframe.value_counts()
crimes_dataframe.to_csv('Chicago_Crimes_2012_to_2017.csv', index=False)
```

```
# diplaying some basic stats of crimes in Chicago from 2012 to 2017
print("\n\033[1mBasic Stats (Crimes in Chicago from 2012-2017)\033[0m")
print("_"*50)
print("\nCrimes Types:", crimes_dataframe["Description"].nunique(), "unique types of crimes")
print("\nLocation:", crimes_dataframe["Location Description"].nunique(), "unique location")
```

```
Basic Stats (Crimes in Chicago from 2012-2017)
__________________________________________________

Crimes Types: 340 unique types of crimes

Location: 140 unique location
```

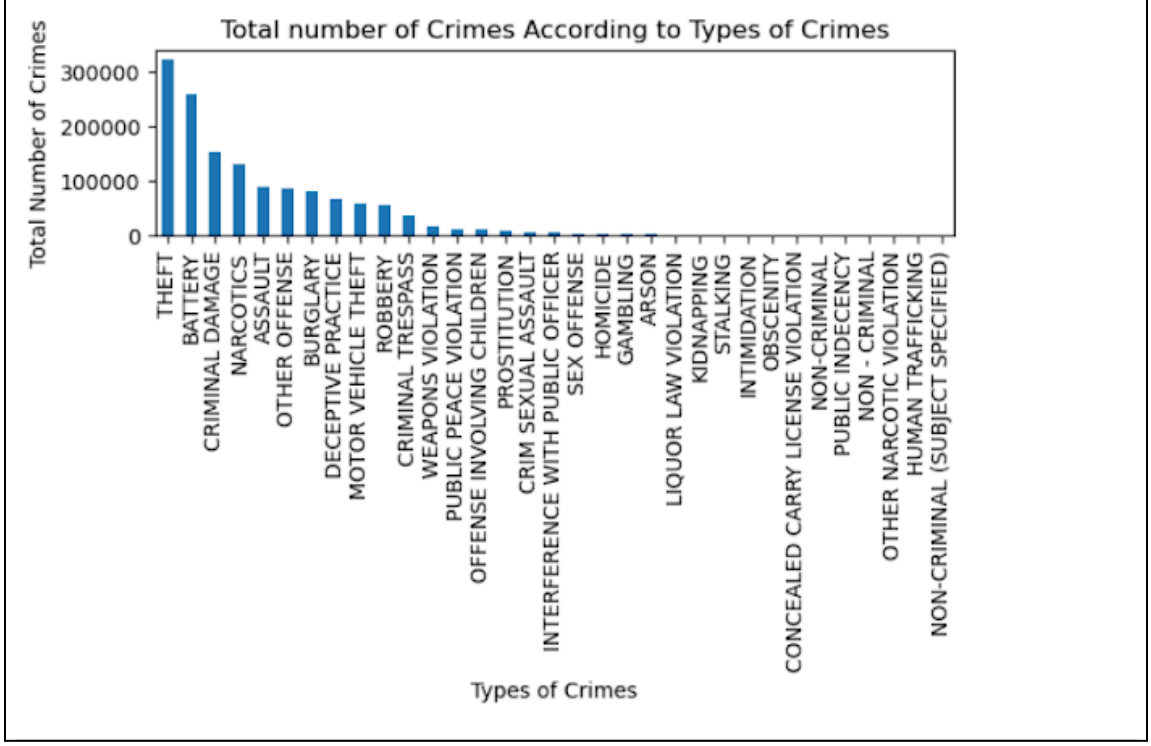
This is to display the basic statistics about crimes in general, and it is just for informational purposes.

6.0 Visualizing Data to Solve The Problems

6.1 Categorical Components

17. Visualising and representing the total number of crimes according to crimes types.

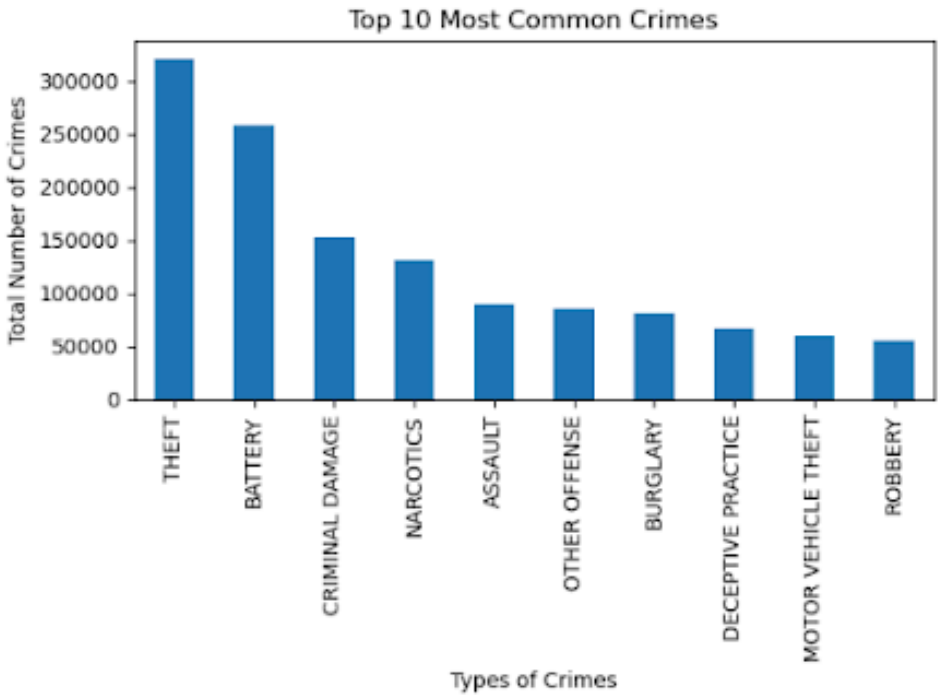
```
# visualizing and representing the 'Total number of Crimes According to Types of Crimes' in a vertical
# Plot the value counts of 'Primary Type' to find the most common crime
crimes_dataframe['Primary Type'].value_counts().plot.bar()
plt.title("Total number of Crimes According to Types of Crimes")
plt.ylabel("Total Number of Crimes")
plt.xlabel("Types of Crimes")

# Adjust layout to prevent labels from getting cut off
plt.tight_layout()

# Show plot
plt.show()
```



The visualization of the total number of crimes by crime types is represented using a vertical bar chart. This chart provides a clear view of how different types of crimes vary in frequency. The analysis reveals that theft is the most prevalent crime type, followed by battery and criminal damage. This information is crucial for law enforcement and policymakers as it highlights the need for targeted interventions and resource allocation. By understanding which crimes are most common, strategies can be developed to address these specific issues, potentially reducing their occurrence.

18. Visualising and representing top 10 most common crimes.

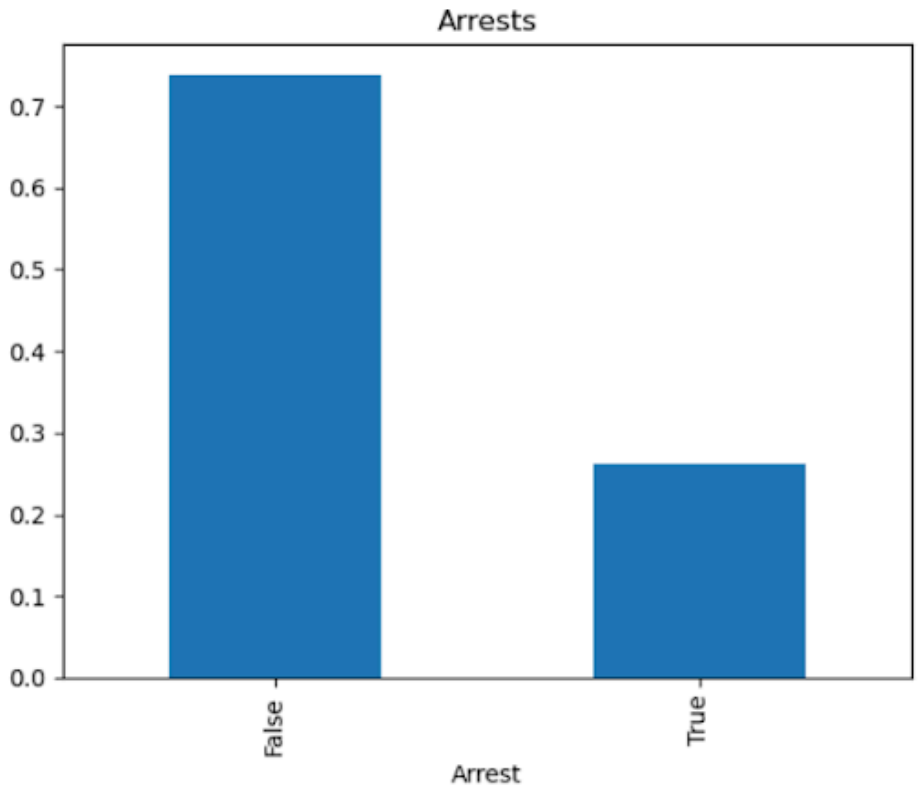
```
# Calculate the top 10 most common crimes
top_10_crimes = crimes_dataframe['Primary Type'].value_counts().head(10)

# Plot the top 10 most common crimes
top_10_crimes.plot.bar()
plt.title("Top 10 Most Common Crimes")
plt.xlabel("Types of Crimes")
plt.ylabel("Total Number of Crimes")

# Adjust layout to prevent labels from getting cut off
plt.tight_layout()

# Show plot
plt.show()
```



The top 10 most common crimes are also visualized using a vertical bar chart, which allows for an easy comparison between different crime types. Theft emerges as the most frequent crime, followed by battery, criminal damage, narcotics, assault, other offenses, burglary, deceptive practice, motor vehicle theft, and robbery. This ranking provides valuable insights into the criminal landscape, indicating that property crimes and violent crimes are significant concerns. The prevalence of narcotics-related crimes suggests ongoing issues with drug use and distribution. Law enforcement agencies can use this data to prioritize their efforts and develop specialized units to tackle the most common crimes effectively.

19. Visualising and representing the arrest success rate.

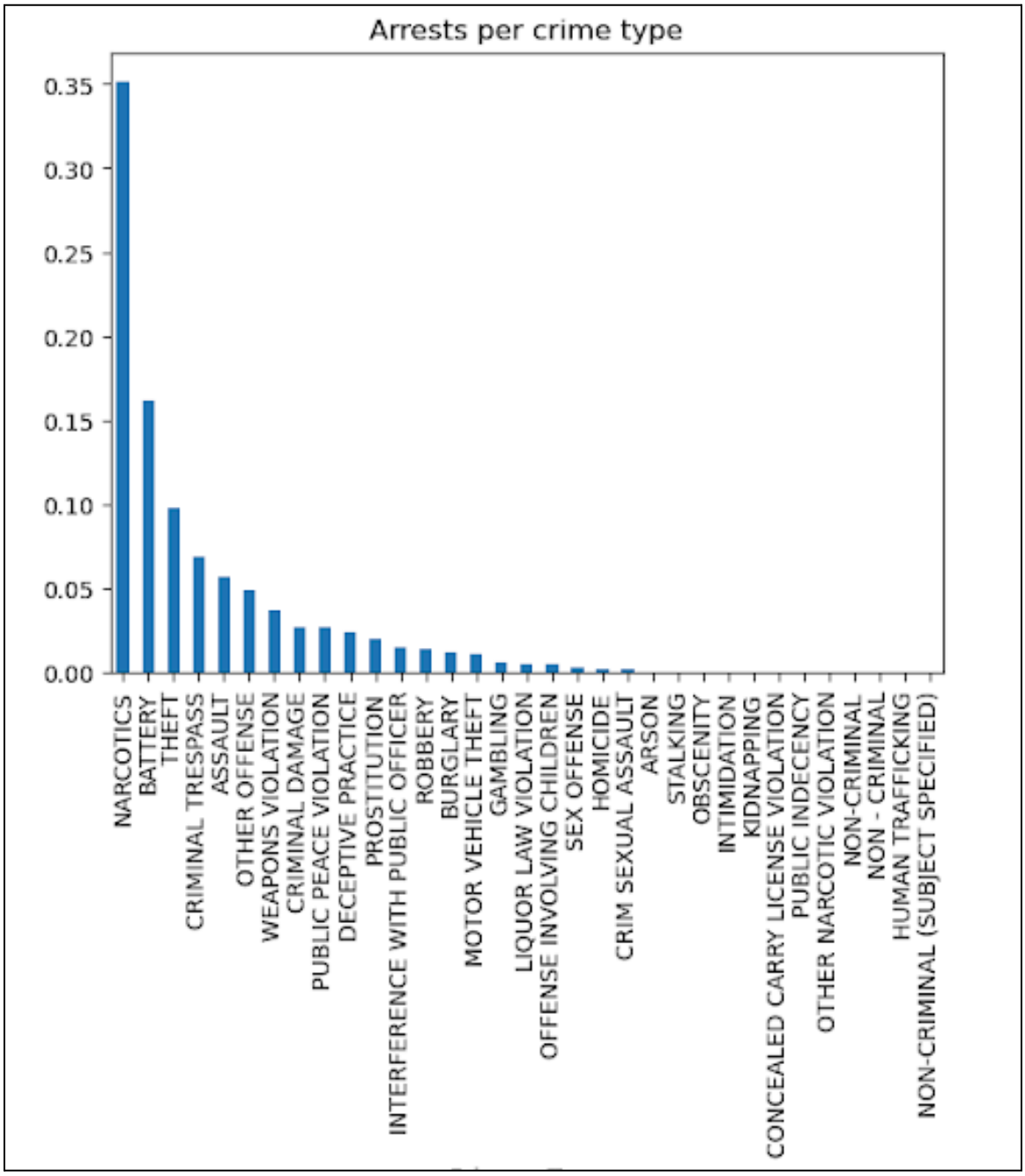
```
# Calculate the arrest success rate
arrest_success_rate = crimes_dataframe['Arrest'].value_counts(normalize=True)

# Plot the arrest success rate
arrest_success_rate.plot.bar()
plt.title("Arrests")
plt.show()
```



The arrest success rate is represented through a vertical bar chart, distinguishing between successful arrests (true) and unsuccessful ones (false). The chart reveals that many crimes do not result in arrests, which is a critical insight for criminal justice agencies. This finding may point to challenges in the investigative process, insufficient evidence, or other barriers to securing arrests. Addressing these issues could involve improving forensic capabilities, increasing the number of detectives, or enhancing cooperation with the community to gather more reliable information. Ensuring that more crimes lead to arrests can improve public confidence in the justice system and deter future criminal activity.

20. Visualising and representing the number of arrests per crime type.

```
# Calculate the arrest rate for all types of crimes
arrest_rate_per_crime = crimes_dataframe[crimes_dataframe['Arrest'] == True]['Primary Type'].value_counts(normalize=True)

# Plot the arrest rate per crime type
arrest_rate_per_crime.plot.bar()
plt.title("Arrests per crime type")
plt.show()
```



21. We use vertical bar chart is used to represent the number of crimes that occur on each day of the week. The analysis shows that Friday has the highest number of crimes, followed

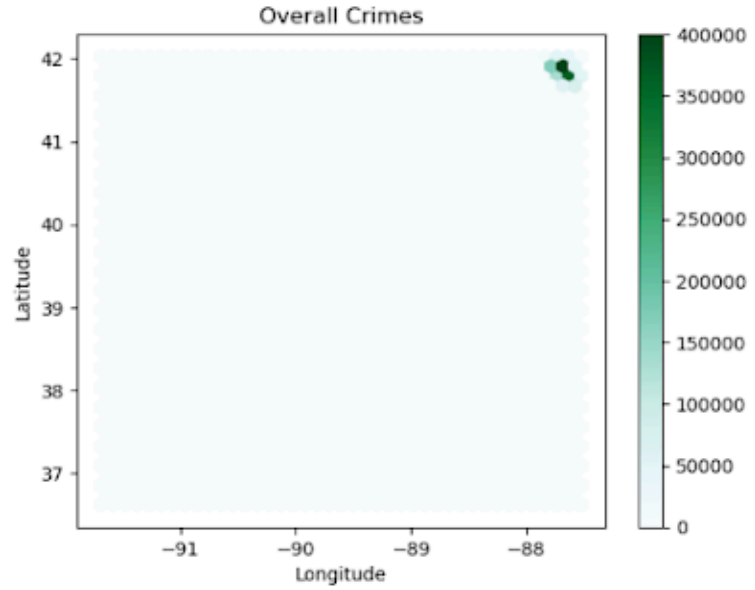
by Saturday and Thursday. This pattern suggests that criminal activity increases towards the end of the week, which could be associated with increased social activities, nightlife, and possibly higher alcohol consumption. Law enforcement agencies can use this information to allocate more resources and personnel during these peak times to prevent and respond to crimes more effectively. Community programs aimed at promoting safety during weekends can also be beneficial.

6.2 Spatial Components

22. Visualising and representing the common location of the crimes.

```
# To see the location of all crimes use a heatmap

# Plot hexbin for overall crimes
plt.figure(figsize=(10, 8))
crimes_dataframe.plot.hexbin(x='Longitude', y='Latitude', gridsize=40)
plt.title("Overall Crimes")
plt.show()
```

```
<Figure size 1000x800 with 0 Axes>
```

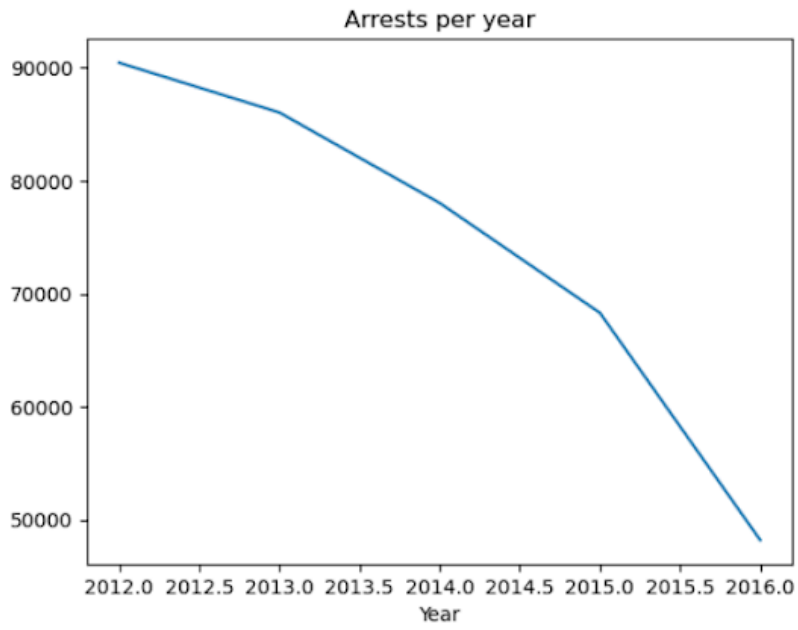


A heatmap is used to visualize the common locations where crimes occur based on the recorded latitude and longitude data. This type of visualization is particularly effective in identifying crime hotspots and areas where criminal activity is concentrated. The analysis shows that certain neighborhoods and intersections experience higher crime rates, which can inform the deployment of police patrols and community safety programs. Understanding the geographical distribution of crimes helps develop localized strategies, such as increasing surveillance in high-crime areas or improving street lighting to deter criminal activities. Community engagement in these hotspots can also play a role in crime prevention and building trust between residents and law enforcement.

6.3. Temporal Components

23. Visualising and representing the number of arrests per year.

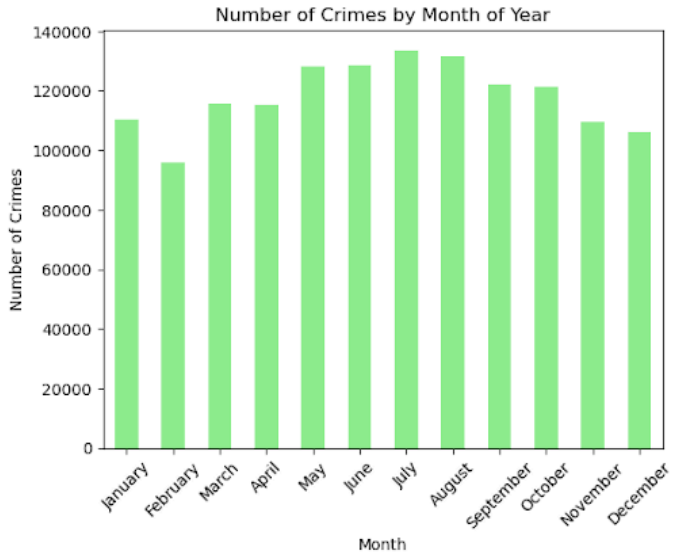
```
# Filter the DataFrame for arrests and plot the number of arrests per year
arrests_per_year = crimes_dataframe[crimes_dataframe['Arrest'] == True]['Year'].value_counts().sort_index()

# Plot the arrests per year
arrests_per_year.plot.line()
plt.title("Arrests per year")
plt.show()
```



The line chart illustrates the number of arrests made yearly from 2012 to 2017. The trend shows a decrease in arrests over this period, suggesting several possible scenarios: changes in law enforcement practices, shifts in crime reporting, or variations in crime rates. This downward trend warrants further investigation to understand its causes. It could reflect positive developments such as improved crime prevention strategies or indicate challenges like reduced police funding or manpower. Policymakers and law enforcement leaders must dig deeper into this trend to ensure that the decrease in arrests does not correspond to a rise in unaddressed criminal activity.

24. Visualising and representing the number of crimes by month of year.

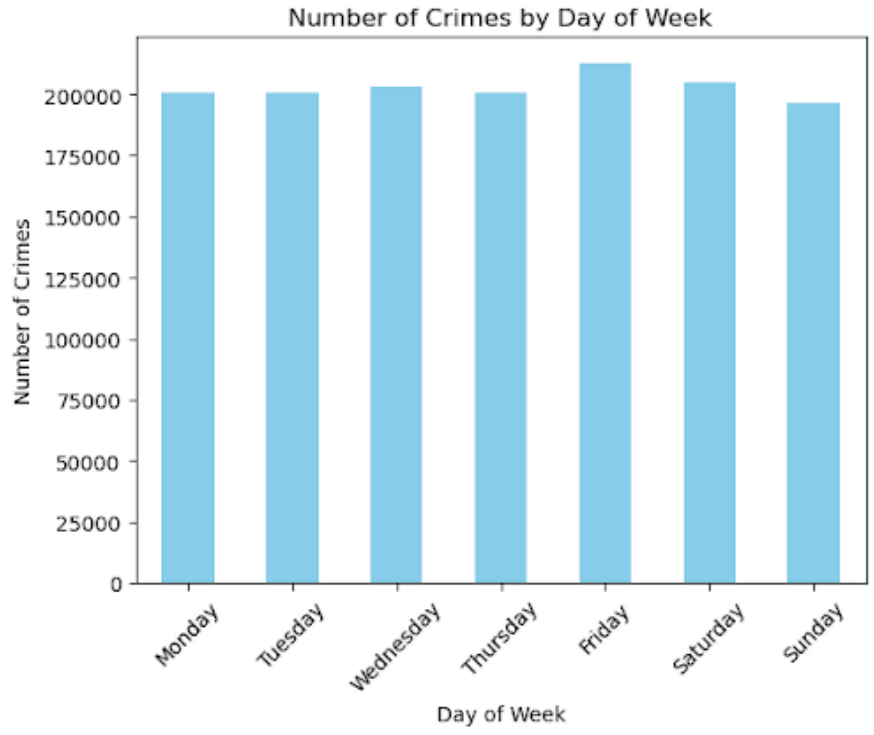
```
[21]: # Convert 'Date' column to datetime
crimes_dataframe['Date'] = pd.to_datetime(crimes_dataframe['Date'], format='%m/%d/%Y %I:%M:%S %p')

# Extract month of the year
crimes_dataframe['Month_of_Year'] = crimes_dataframe['Date'].dt.month

# Define month names
months = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October', 'November', 'December']

# Count number of crimes by month of the year
crimes_by_month = crimes_dataframe['Month_of_Year'].value_counts().sort_index()

# Plot the number of crimes by month of the year
crimes_by_month.plot(kind='bar', color='lightgreen')
plt.title('Number of Crimes by Month of Year')
plt.xlabel('Month')
plt.ylabel('Number of Crimes')
plt.xticks(range(0, 12), [months[i-1] for i in range(1, 13)], rotation=45)
plt.show()
```



The number of crimes by month of the year is visualized using a vertical bar chart. The analysis indicates that July sees the highest number of crimes, with a noticeable increase in criminal activity during summer. This seasonal trend may be influenced by school holidays, warmer weather, and more outdoor events, which can create more opportunities for crimes. Understanding these patterns allows for the implementation of seasonal crime prevention strategies, such as increased police presence during summer events, community outreach programs to educate residents about staying safe, and initiatives to engage youth in positive activities during their break.

25. Visualising and representing to see the number of crimes by day of week.

```
# Convert 'Date' column to datetime
crimes_dataframe['Date'] = pd.to_datetime(crimes_dataframe['Date'], format='%m/%d/%Y %I:%M:%S %p')

# Extract day of the week
crimes_dataframe['Day_of_Week'] = crimes_dataframe['Date'].dt.dayofweek

# Define day names
days = ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday']

# Count number of crimes by day of the week
crimes_by_day = crimes_dataframe['Day_of_Week'].value_counts().sort_index()

# Plot the number of crimes by day of the week
crimes_by_day.plot(kind='bar', color='skyblue')
plt.title('Number of Crimes by Day of Week')
plt.xlabel('Day of Week')
plt.ylabel('Number of Crimes')
plt.xticks(crimes_by_day.index, [days[i] for i in crimes_by_day.index], rotation=45)
plt.show()
```

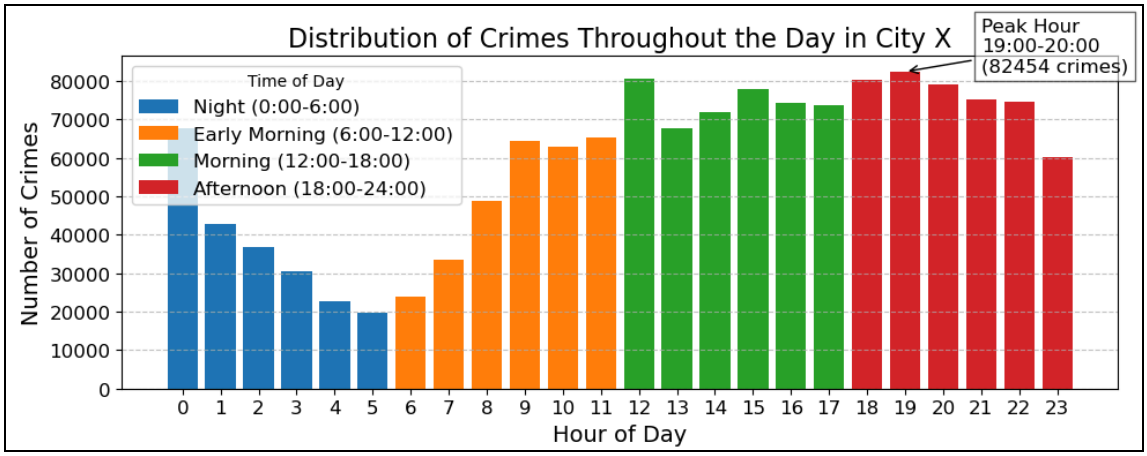


A vertical bar chart is used to represent the number of crimes that occur on each day of the week. The analysis shows that Friday has the highest number of crimes, followed by Saturday and Thursday. This pattern suggests that criminal activity increases towards the end of the week, which could be associated with increased social activities, nightlife, and possibly higher alcohol consumption. Law enforcement agencies can use this information to allocate more resources and personnel during these peak times to prevent and respond to crimes more effectively. Community programs aimed at promoting safety during weekends can also be beneficial.

26. Visualising and representing the number of crimes by hour of day.

```
# Load the data
# Assuming crimes_df is already loaded with the necessary data

# Convert 'Date' column to datetime
crimes_df['Date'] = pd.to_datetime(crimes_df['Date'], format='%m/%d/%Y %I:%M:%S %p')

# Extract hour of the day
crimes_df['Hour_of_Day'] = crimes_df['Date'].dt.hour

# Count number of crimes by hour of the day
crimes_by_hour = crimes_df['Hour_of_Day'].value_counts().sort_index()

# Define time of day categories
time_of_day = ['Night', 'Early Morning', 'Morning', 'Afternoon', 'Evening']
colors = ['#1f77b4', '#ff7f0e', '#2ca02c', '#d62728', '#9467bd']
bins = [0, 6, 12, 18, 24]

# Plot the number of crimes by hour of the day
fig, ax = plt.subplots(figsize=(10, 4))

for i in range(len(bins) - 1):
    start, end = bins[i], bins[i + 1]
    mask = (crimes_by_hour.index >= start) & (crimes_by_hour.index < end)
    ax.bar(crimes_by_hour.index[mask], crimes_by_hour[mask], color=colors[i], label=f'{time_of_day[i]} ({start}:00-{end}:00)')

# Add a more descriptive title
plt.title('Distribution of Crimes Throughout the Day in City X', fontsize=16)
```

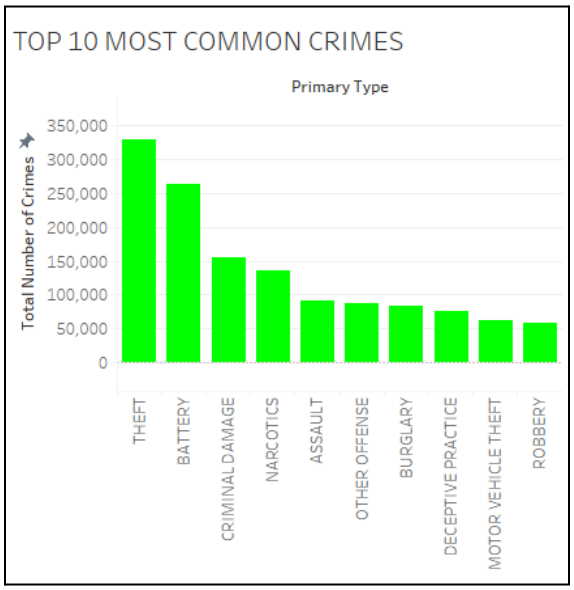
```
# Improve axis labels for clarity
plt.xlabel('Hour of Day', fontsize=14)
plt.ylabel('Number of Crimes', fontsize=14)
plt.xticks(range(0, 24), fontsize=12)
plt.yticks(fontsize=12)
plt.grid(axis='y', linestyle='--', alpha=0.7)

# Add annotations to highlight peak hours
peak_hour = crimes_by_hour.idxmax()
peak_count = crimes_by_hour.max()
ax.annotate(f'Peak Hour\n{peak_hour}:00-{peak_hour + 1}:00\n({peak_count} crimes)',
            xy=(peak_hour, peak_count),
            xytext=(peak_hour + 2, peak_count + 20),
            arrowprops=dict(facecolor='black', arrowstyle='->'),
            fontsize=12, bbox=dict(facecolor='white', alpha=0.6))

# Add legend
plt.legend(title='Time of Day', fontsize=12)

plt.tight_layout()
plt.show()
```

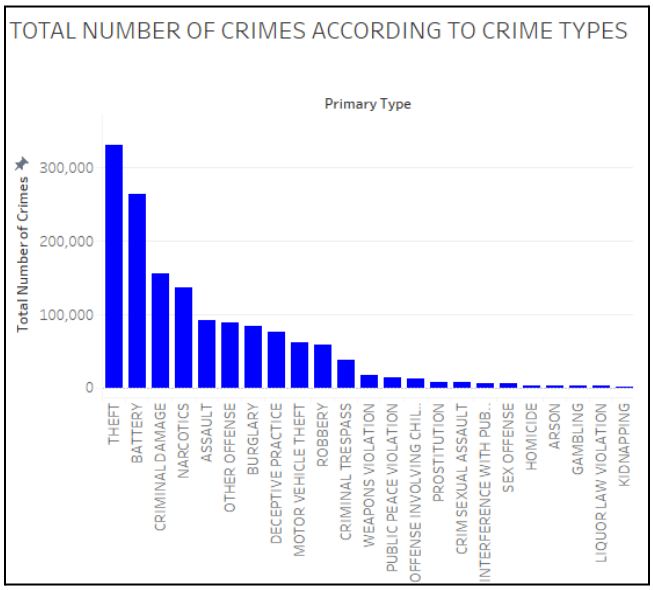


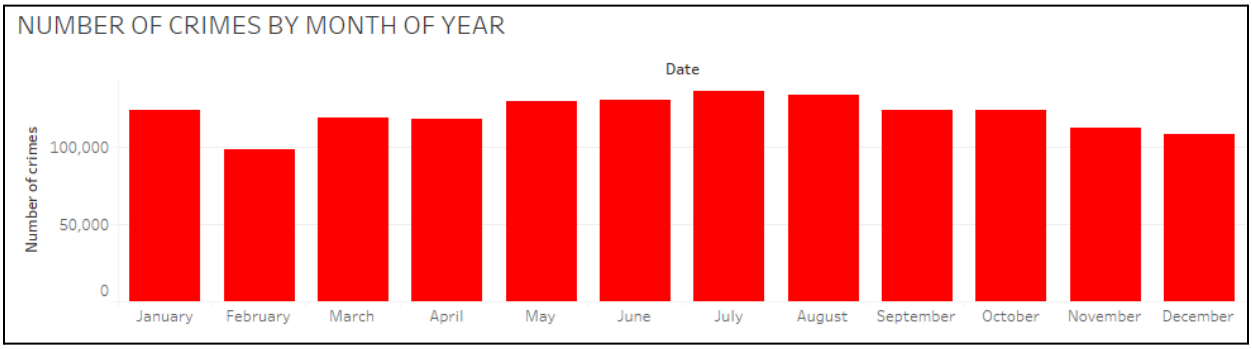
A vertical bar chart is used to represent the number of crimes by the hour of the day. The analysis shows that 7:00 p.m. is the peak time with 82454 crimes for criminal activity, with a significant number of crimes occurring in the evening hours. This finding suggests that evening routines, such as commuting home from work or attending social events, may be associated with higher crime risks. Law enforcement can use this information to schedule more patrols and increase vigilance during these critical hours. Additionally, public safety campaigns can be designed to raise awareness about staying alert and taking precautions during evening hours, potentially reducing the incidence of crimes.

6.4 Data From Tableau

This is some example of our data we created using tableau :







7.0 References

Crimes in Chicago. (n.d.). Kaggle. https://www.kaggle.com/datasets/currie32/crimes-in-chicago?select=Chicago_Crimes_2012_to_2017.csv

U.S. News & World Report. (n.d.). Crime in Chicago, IL. U.S. News & World Report. Retrieved July 4, 2024, from https://realestate.usnews.com/places/illinois/chicago/crime#:~:text=How%20safe%20is%20Chicago%2C%20IL,higher%20than%20the%20national%20rate.