Chart Analysis For Crime in Chicago

Table Of Contents

Introduction

The objective of this project is to analyze crime data in Chicago and present the findings through data visualizations. The data is retrieved from the Chicago Open Data Portal crime API and filtered based on date and location. The analysis includes descriptive statistics to understand the crime distribution and trends in the given time frame (January 2019 to July 2019).

Data Retrieval and Preparation

Using the Python `requests` library, we fetch the crime data from the API by applying date and location filters. The retrieved data is converted into a pandas DataFrame for further analysis. The DataFrame consists of columns such as 'date', 'block', 'primary_type', and 'description'.

Descriptive Statistics

To understand the distribution of crime types, we calculate descriptive statistics such as the mean, median, and standard deviation of crime occurrences for each primary type. These statistics provide valuable insights into the overall crime landscape in Chicago during the specified period.

Summary of Descriptive Statistics for Crime Types

The dataset contains information on reported crimes in Chicago between January 1, 2019, and July 16, 2019.This summary highlights the number of occurrences for each crime type, the distribution of crime types, and the most frequently reported crime.

1. Number of Occurrences of Each Crime Type:
The first part of our analysis focused on counting the occurrences of each crime type in the dataset. The "primary_type" column was used to categorize the different types of crimes reported. The number of occurrences for each crime type is presented below:

THEFT	23
BATTERY	19
CRIMINAL DAMAGE	11
DECEPTIVE PRACTICE	7
ASSAULT	6
BURGLARY	6
OTHER OFFENSE	6
CRIMINAL TRESPASS	3
CRIMINAL SEXUAL ASSAULT	1
MOTOR VEHICLE THEFT	1
NARCOTICS	1
PUBLIC PEACE VIOLATION	1
ROBBERY	1
SEX OFFENSE	1

From the above count, we observe that "THEFT" is the most frequently reported crime type, occurring 23 times in the dataset. On the other hand, several crime types, including "CRIMINAL TRESPASS," "CRIMINAL SEXUAL ASSAULT," "MOTOR VEHICLE THEFT," "NARCOTICS," "PUBLIC PEACE VIOLATION," "ROBBERY," and "SEX OFFENSE," occurred only once.

2. Summary of Descriptive Statistics for 'primary_type' column:
Next, we calculated summary statistics for the 'primary_type' column. These statistics provide insights into the distribution of crime types in the dataset. The summary statistics are as follows:

count 87
unique 14
top THEFT
freq 23
Name: primary_type, dtype: object

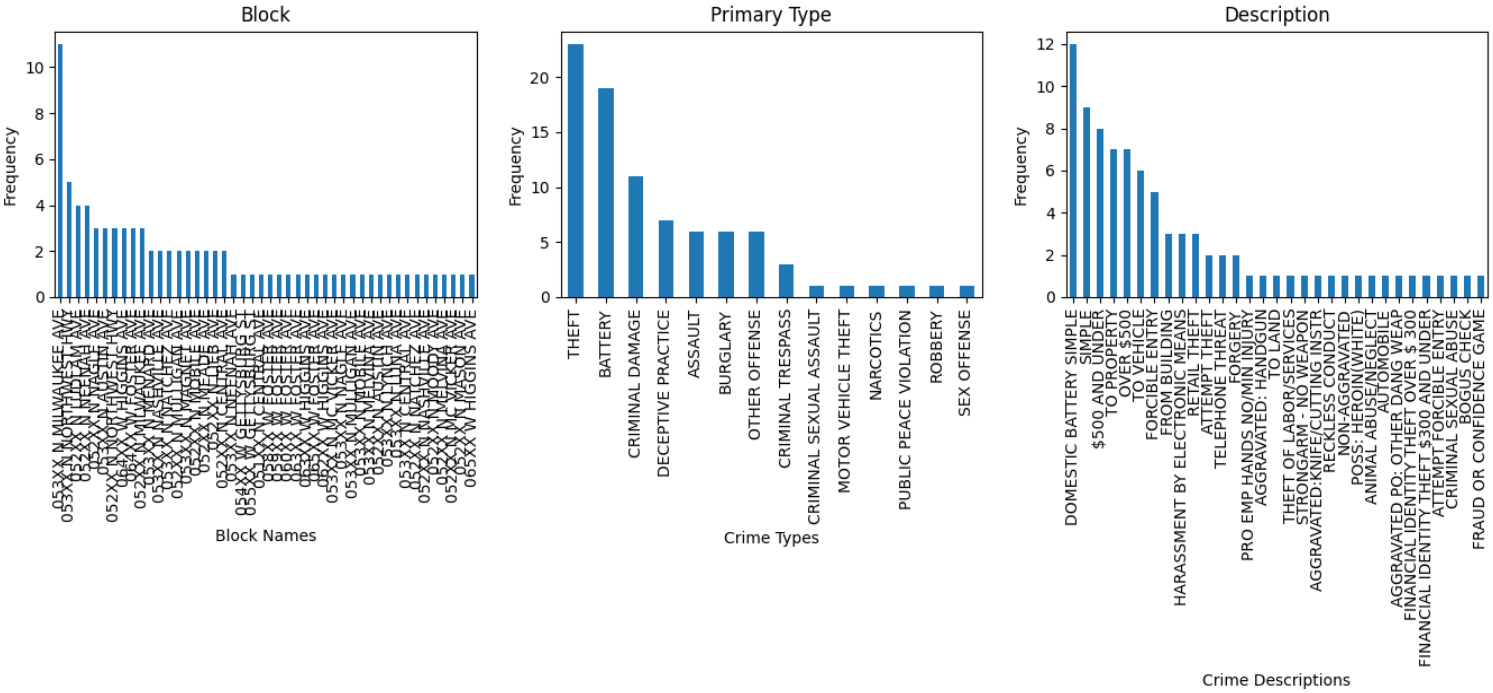
- `count`: The total number of entries in the 'primary_type' column, which is 87. This means there are 87 recorded crimes in the dataset.

- `unique`: The number of unique crime types in the 'primary_type' column, which is 14. This indicates that there are 14 different crime types reported.

- `top`: The most frequently occurring crime type in the 'primary_type' column, which is "THEFT" with a frequency of 23.

- `freq`: The frequency of the most frequently occurring crime type, which is 23. This means that "THEFT" is the most common crime type in the dataset, reported 23 times.

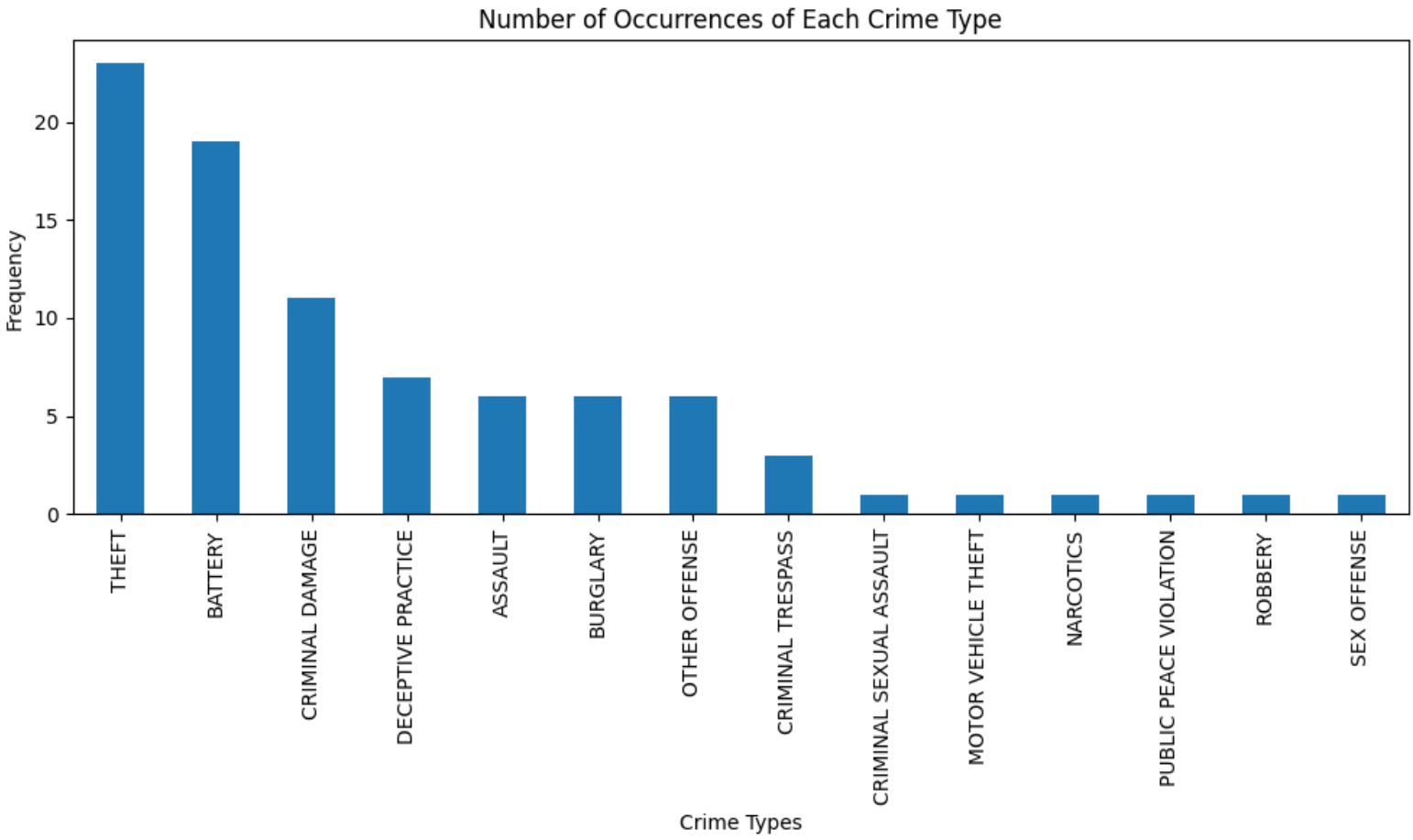
3. Visualization:
To further understand the distribution of crime types, we created a bar plot to visualize the frequency of each crime type. The bar plot provides a clear visual representation of the occurrence of each crime type.(run data.py to view the full version of the graph.)



From the analysis, we can see that "THEFT" is the most prevalent crime type, followed by "BATTERY," "CRIMINAL DAMAGE," and "DECEPTIVE PRACTICE." Conversely, several crime types are relatively rare, with only one occurrence each in the dataset.

Data Visualization

We will visualize the crime data using mat-plot-lib, a Python library used to create plots and charts. Specifically, we generate a bar chart to display the crime counts for each primary type. The chart gives a clear picture of the top crime types and their respective frequencies.



Findings

Data Categories:

The following data categories were selected for analysis:

1. Primary Crime Type
2. Description of Crime
3. Arrest Status
4. Latitude of Crime Location
5. Longitude of Crime Location
6. Top Crimes

Top Crimes: This category represents a special selection of the most frequently occurring crimes in the dataset. It allows us to focus on the primary types of crimes that are the most prevalent in the specified location and time range.

Primary Type: This category denotes the primary type or category of the crime. It includes various types of crimes, such as theft, burglary, assault, robbery, etc. This information helps in identifying the nature of each crime.

Description: The "Description" category provides a detailed description of the specific crime incident. It provides additional context about the nature of the crime.

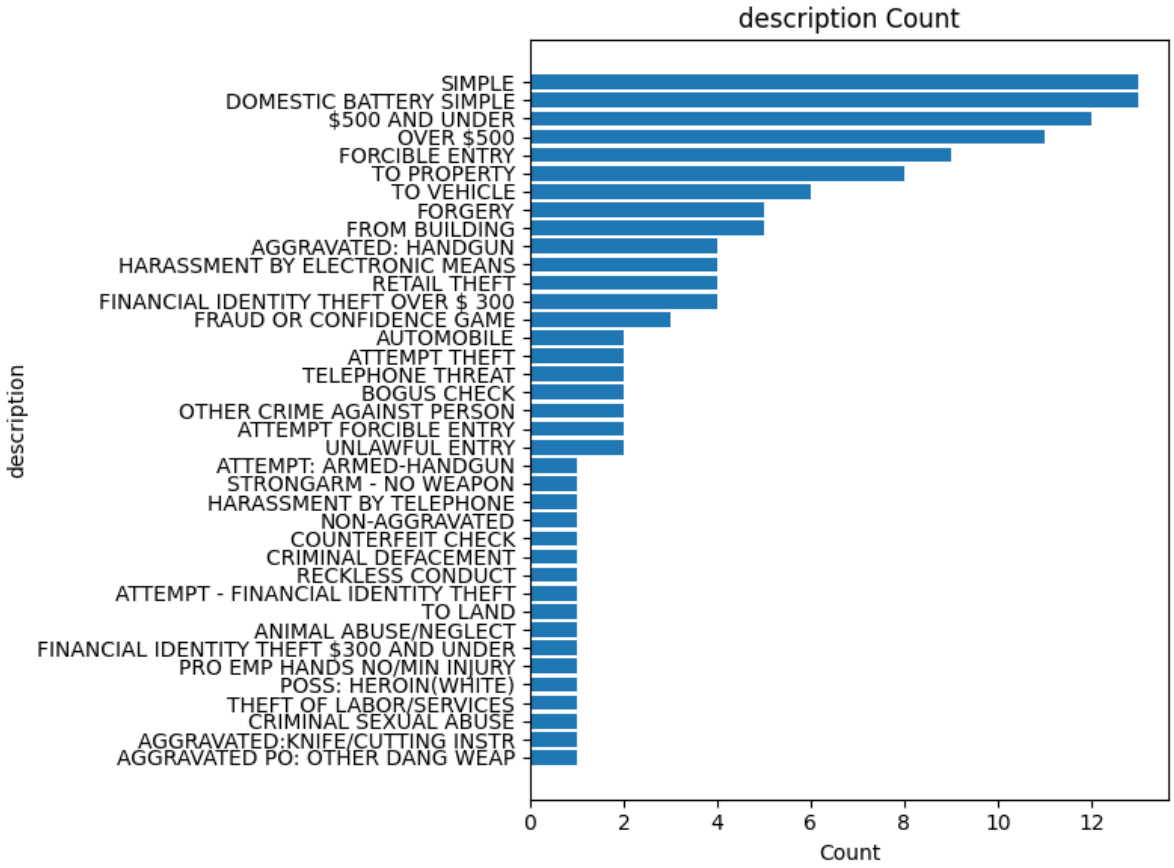
Arrest: The "Arrest" category is a Boolean attribute indicating whether an arrest was made in connection with the crime. It shows whether law enforcement personnel were able to apprehend a suspect.

Latitude and Longitude: These categories represent the geographical coordinates of the crime location. Latitude and longitude values provide the exact geographical location where the crime occurred, allowing for spatial analysis and visualization on maps.

Data Analysis and Visualization:

1. Bar-Charts:

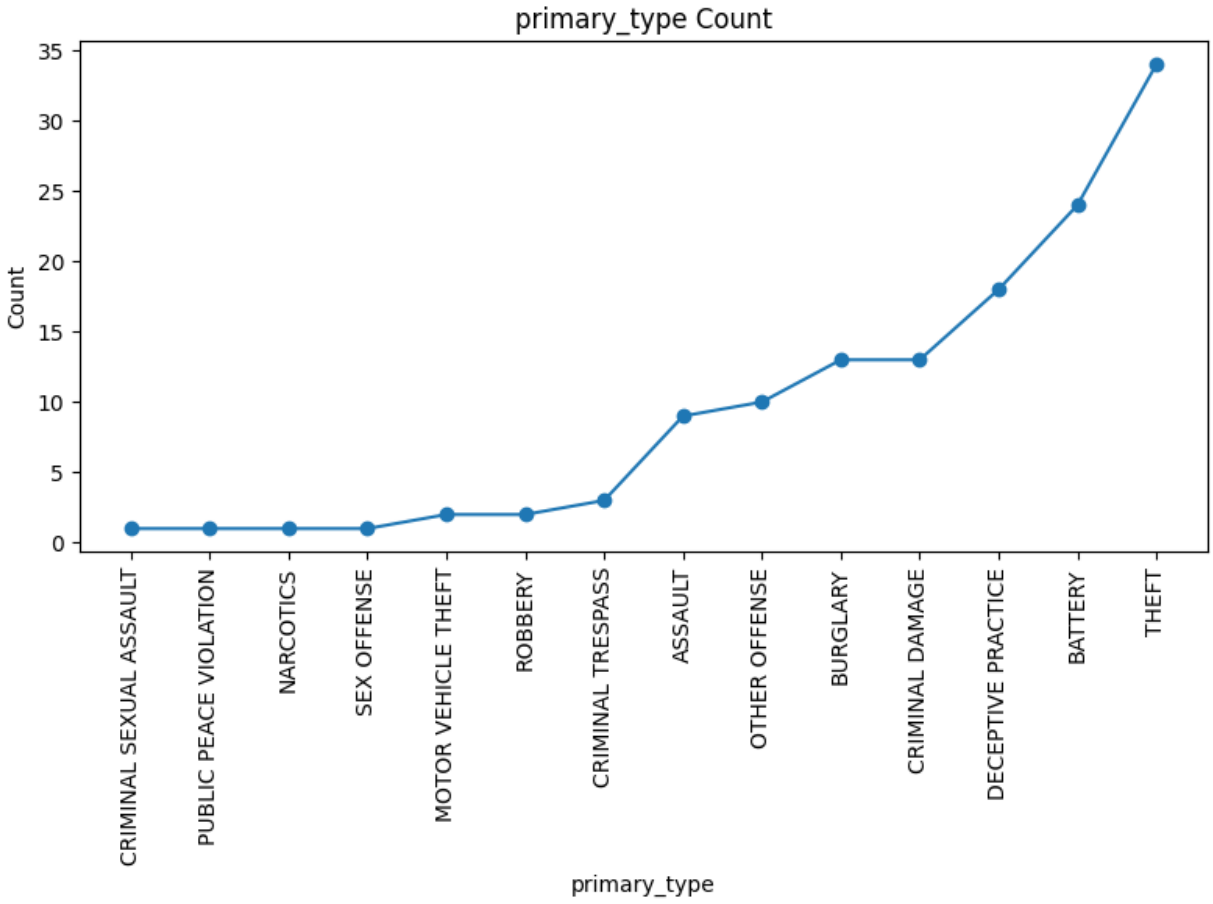
Bar-charts are a visual illustration that employs rectangular bars to display the occurrences or tally of different categories within a given dataset. Each bar represents a category, and the length of the bar corresponds to the frequency or count of that category. Bar charts are created using the `plt.bar()` function in matplotlib or `sns.barplot()` in seaborn..



Bar chart showing the distribution of crime types

2. Line-Charts:

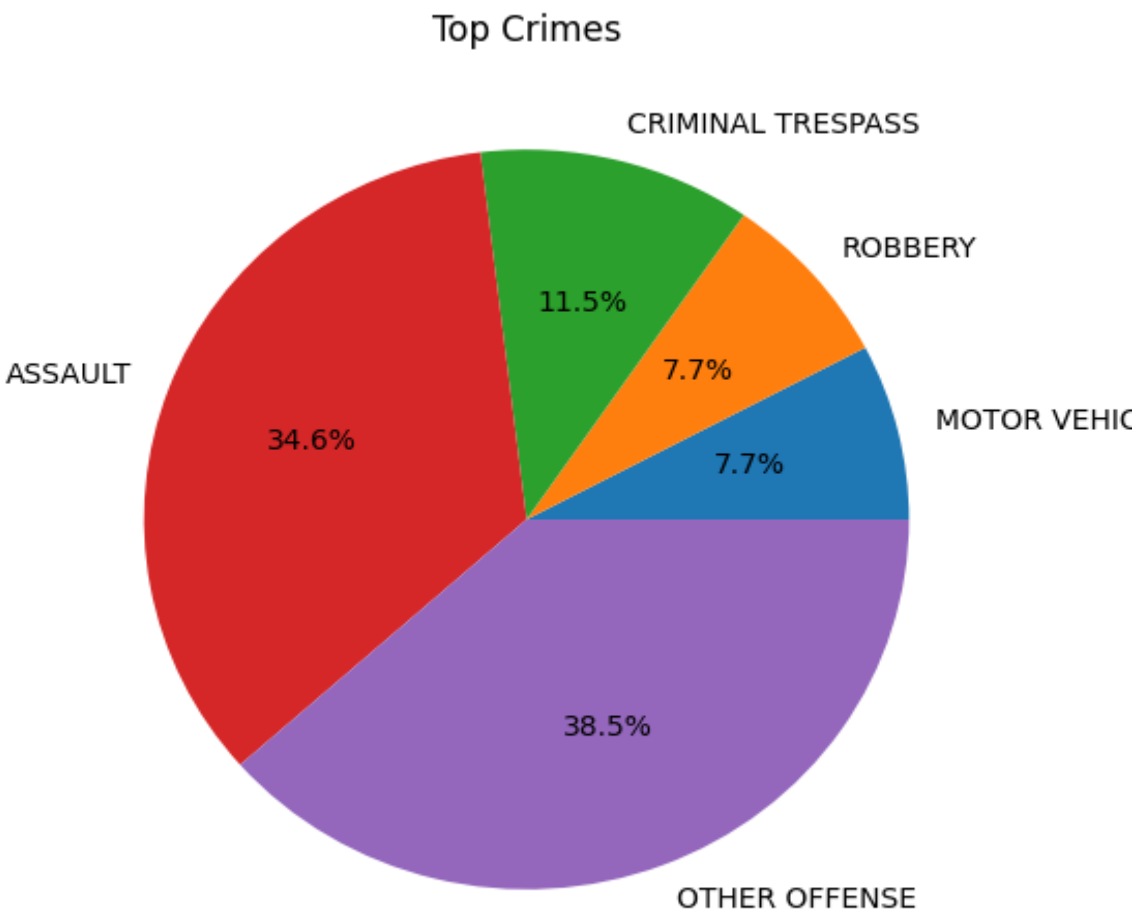
A line chart is a visual depiction that connects data points using straight lines. Its primary purpose is to illustrate trends or variations over a period, relating to time.. In a line chart, the x-axis typically represents the time or an ordered variable, and the y-axis represents the value or count of the data. Line charts can be generated using the `plt.plot()` function in matplotlib or `sns.lineplot()` in seaborn.



Line chart showing primary crime types.

3. Pie-Charts:

A pie chart is a circular graphic that is segmented into slices, each representing the relative proportion of various metrics within a dataset. Pie charts are created using the `plt.pie()` function in matplotlib.Pie charts will display the percentage distribution of top crimes.

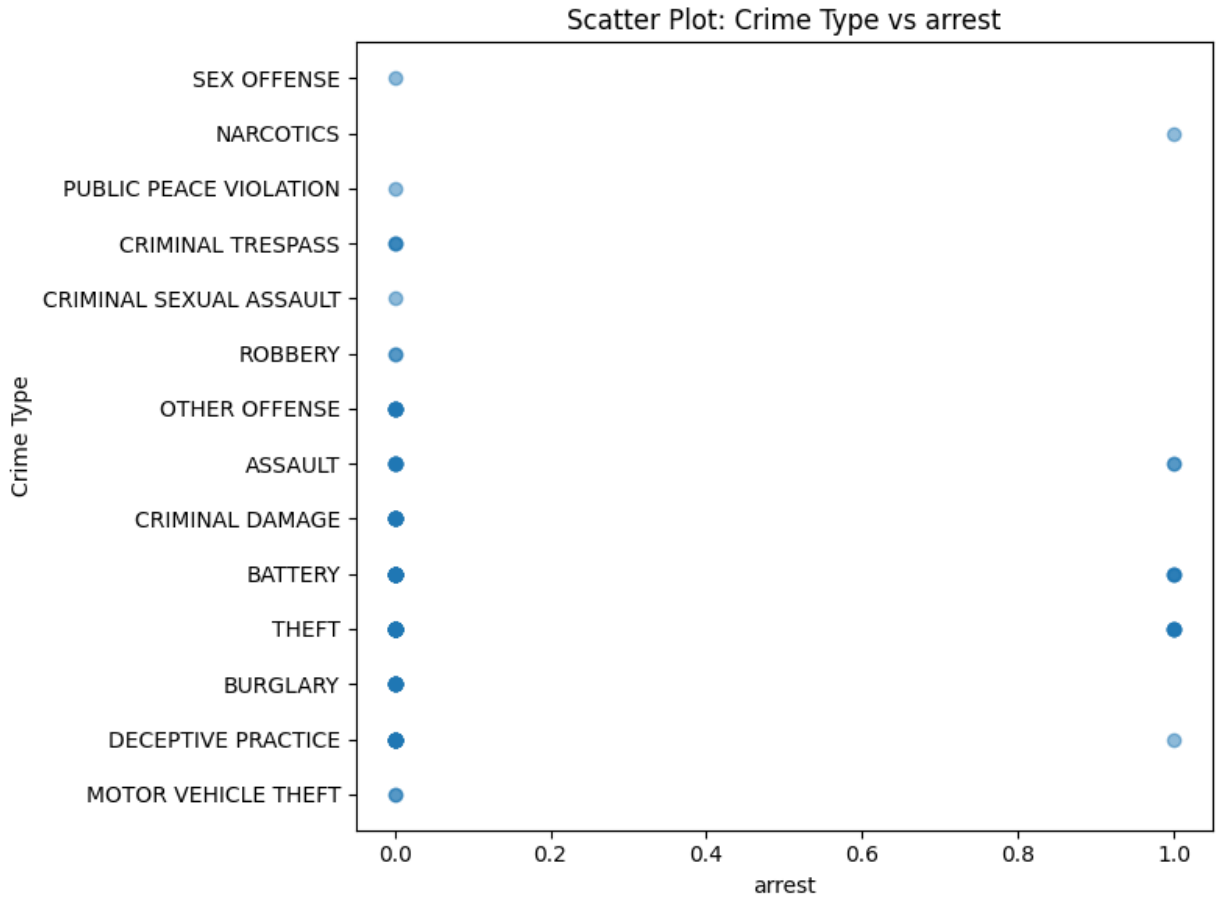


Pie chart showing the percentage distribution of top crimes.

4. Scatter-Plots:(Extra-Credit)

A scatter plot is a graphical representation where individual data points are depicted as dots on a two-dimensional graph. This type of plot is employed to illustrate the relationship or correlation between two continuous variables. Each dot on the graph signifies an observation, and its placement reflects the values of the two variables being compared.

Scatter plots are generated using the `plt.scatter()` function in matplotlib or `sns.scatterplot()` in seaborn.

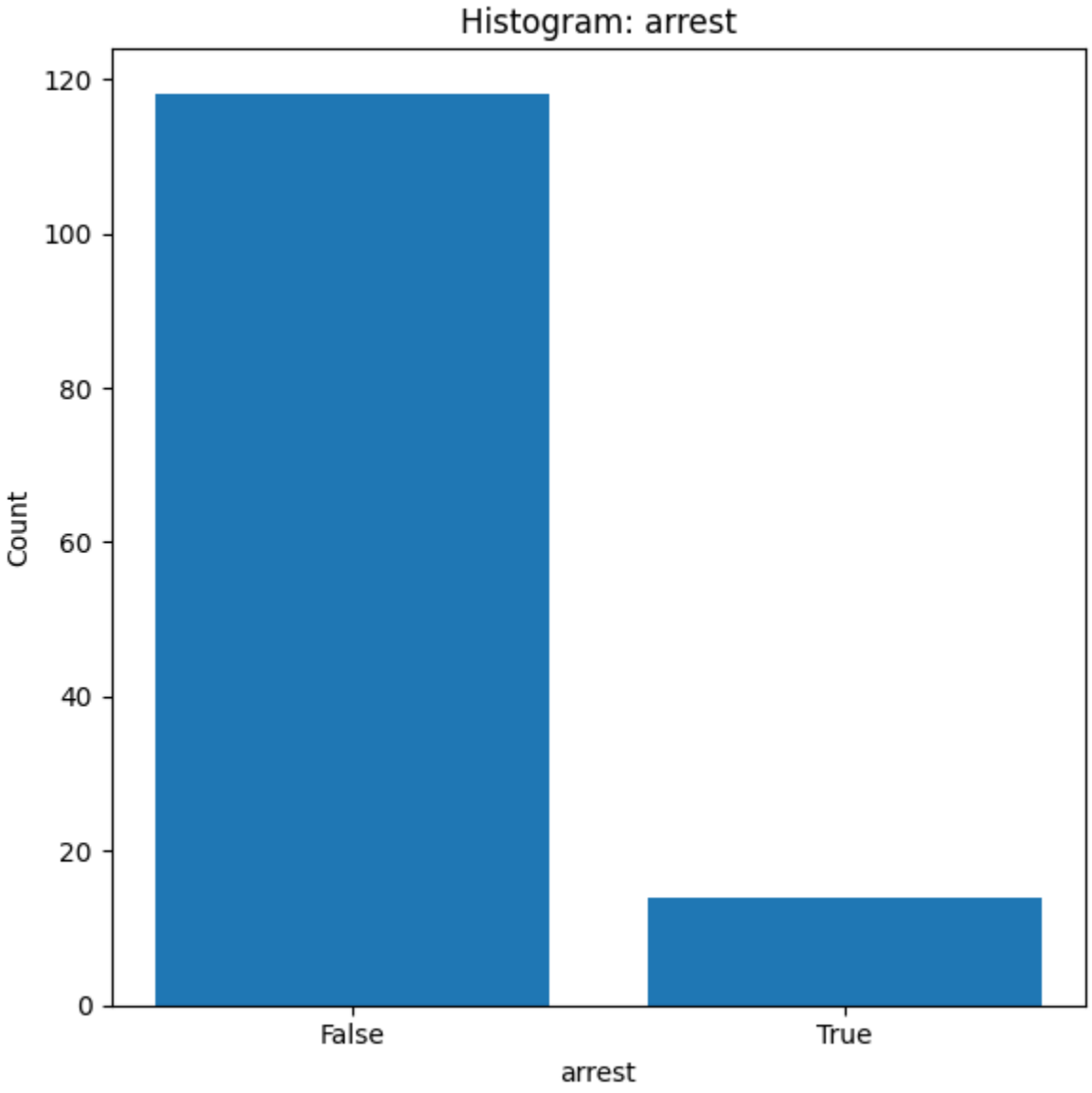


Scatter plot showing the ratio of arrests to non-arrests

5. Histograms:(Extra-Credit)

A histogram is a visual portrayal of the distribution of a continuous variable. It accomplishes this by partitioning the data-set range into specific bins and demonstrating the frequency or count of the data points falling within each bin.The taller the bar, the higher the frequency or count of data points in that particular bin, visually reflecting the distribution of the data across different ranges.

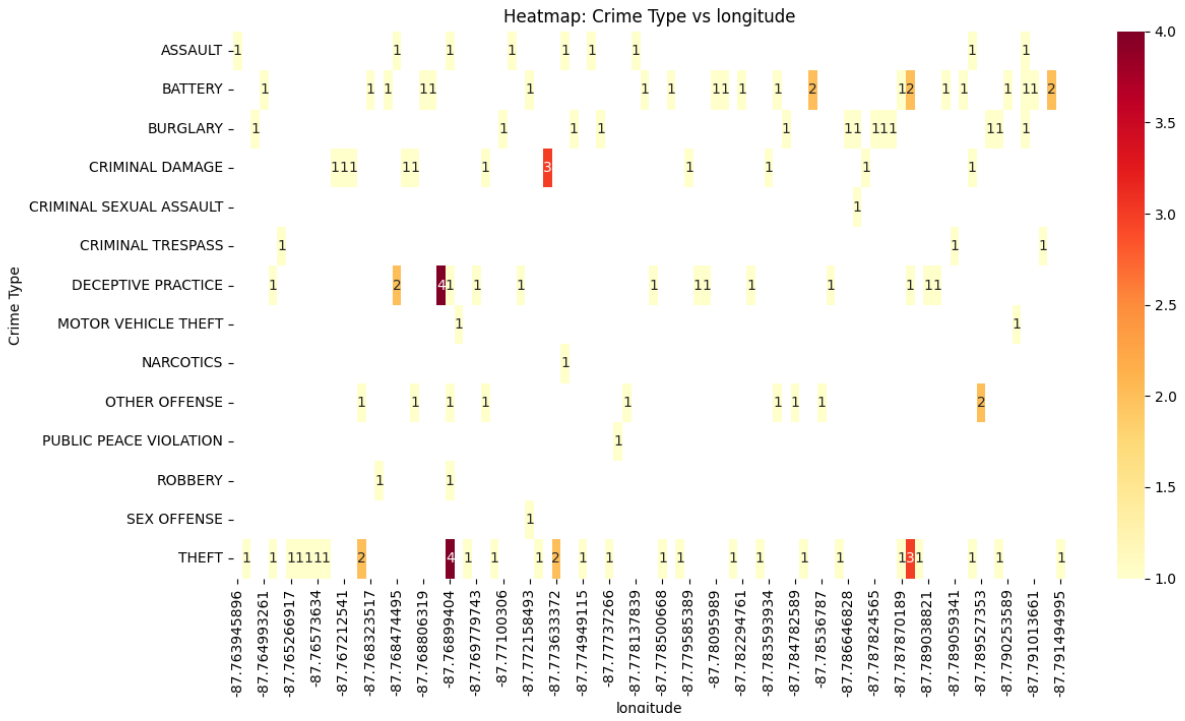
Histograms can be created using the `plt.hist()` function in matplotlib or `sns.histplot()` in seaborn.Histograms will represent the distribution of selected numerical columns, such as the ratio of arrests and non arrests per crime counts.



6. Heatmaps:(Extra-Credit)

A heatmap is a visual representation of data that utilizes colors to portray individual values. In a heatmap, each data point is represented by a specific color, allowing patterns and variations in the data to be easily discerned.It is particularly useful for visualizing the correlation or relationship between two categorical variables. The heatmap displays a grid of squares, and each square is colored based on the value it represents.

The color intensity or shade corresponds to the magnitude of the value. Heatmaps are created using the sns.heatmap() function in the seaborn library. In our analysis, we will generate heatmaps to show the concentration of crime incidents based on their latitude and longitude coordinates. Darker shades of color in the heat map indicate areas with a higher density of crimes, while lighter shades represent areas with lower crime density.
These chart types are valuable tools for data analysis and visualization, as they allow us to explore patterns, trends, and distributions within the crime data. By using these charts, we can gain insights into crime occurrences, identify potential correlations, and make informed decisions based on the visualized data.



Conclusion

Through data analysis and visualization, this project provides valuable insights into the crime trends and patterns in Chicago. The visualizations highlight the distribution of crime types, their occurrence over time, and their geographical concentrations. The findings can assist law enforcement agencies and policymakers in making informed decisions to address crime-related challenges in the city.