Crime Prediction and Analysis Using Machine Learning

YASHIKA CHAURSIA (23MCA1004)

NAINA MOGHA (23MCA1082)

## **ABSTRACT:** Crime prediction and analysis are critical components in safeguarding communities. Leveraging data science and machine learning, our study focuses on predicting criminal occurrences, identifying high-risk areas, and visualizing patterns. By employing classification models (including logistic regression, gradient boosting, and random forest), clustering techniques (such as K-means and DBSCAN), and ensemble learning (via gradient boosting and random forest), we empower law enforcement agencies with actionable insights. Our goal is to develop models to predict crime hotspots and extract valuable insights from unstructured crime data.

## **INTRODUCTION:**

Crime detection involves the use of various techniques and technologies to identify, analyse, and predict criminal activities. This includes the analysis of crime incidents over time, precision, recall, and F1 score for different classes, as well as the use of ensemble classifiers and deep learning techniques such as the Random Forest Classifier. It covers the implementation of various machine learning models such as Random Forest Classifier, Gradient Boosting Regressor, DBSCAN, K-Means, and Logistic Regression. The document also includes the preprocessing of data, feature extraction, and label encoding. Additionally, it delves into performance evaluation metrics such as accuracy, precision, recall, and F1 score for different classes. The use of visualization techniques, including scatter plots and bar charts. Overall, the document offers a detailed exploration of the application of machine learning algorithms in crime prediction and analysis. The insights into the distribution of crime types, density plot of crime incidents, and the accuracy, macro avg, and weighted avg of crime prediction models. These methods and tools are essential for law enforcement agencies and policymakers to understand crime patterns, allocate resources effectively, and develop strategies for crime prevention and intervention, the importance of crime detection in maintaining public safety, reducing criminal activities, and ensuring the efficient allocation of law enforcement resources. You can also emphasize the role of advanced technologies and data analysis in enhancing crime detection and prevention efforts.

## **TOOLS USED:**

**Python Programming Language:**

Python serves as the primary programming language for developing the project due to its simplicity, readability, and extensive library support.

**Matplotlib:**

Matplotlib is a plotting library for Python used to visualize data, including images, training/validation accuracy, loss curves, and other metrics. It provides a wide range of functionalities for creating static, interactive, and publication-quality plots.

**NumPy:**

NumPy is a fundamental package for scientific computing with Python, providing support for multidimensional arrays, mathematical functions, linear algebra operations, and random number generation. It is commonly used for data manipulation and preprocessing tasks.

**Sklearn:**

scikit-learn is a powerful machine learning library that provides a wide range of tools for various tasks such as classification, regression, clustering, dimensionality reduction, and more. It offers a plethora of machine learning algorithms, including decision trees, support vector machines, k-means clustering, and more.

**Seaborn:**

Seaborn is a Python data visualization library built on top of Matplotlib. It simplifies the creation of aesthetically pleasing statistical plots. Seaborn provides a high-level interface for creating various types of plots, including scatter plots, bar plots, box plots, and heatmaps.

**Colab :**

Google Colab, a cloud-based Jupyter notebook environment, may be used for collaborative development and execution of the project. Colab provides free access to GPU and TPU accelerators, enabling faster model training and experimentation without the need for powerful hardware resources.

## **METHODOLOGY**

**Data Collection:**

Extract crime data from Kaggle.

Features: crime type, year, month, day, hour, minute, time, neighbourhood, latitude, longitude.

**Data Preprocessing:**

Drop missing values columns

* Identify columns with a significant number of missing values.
* Evaluate whether these columns provide valuable information.
* If not, drop those columns from your dataset.

Label encoding

* converts categorical variables (textual labels) into numerical representations.
* Assign a unique integer to each category in a categorical column.

Normalization

* Normalization scales numerical features to a common range
* Standardize features to have zero mean and unit variance.
* Common techniques include Min-Max scaling and Z-score normalization.

Split data into train and test

* Reserve a portion of your data 20% for testing.
* Use the remaining data for training.

**Classification technique:**

**Logistic Regression**

* is a statistical method used for predicting a binary outcome (yes/no, true/false, 0/1) based on one or more independent variables.
* Achieved an accuracy of **69 %**.
* **Sigmoid Function (Logistic Function)**:
  + Logistic regression uses the sigmoid function to model the probability of an event:
  + P(Y=1∣X)=1/1+e−z​
  + Here, z represents a linear combination of input features.
* **Linear Combination**:
* The linear combination is calculated as:
* z=β0​+β1​X1​+β2​X2​+…+βn​Xn​
* Each X\_i represents an input feature, and each β\_i is a coefficient.

**Support Vector Machine**

* SVM maps data to a high-dimensional feature space, even when the data are not linearly separable.
* It identifies a separator (hyperplane) between categories.
* The key concept is to maximize the margin (distance) between the hyperplane and the nearest data points (support vectors).
* SVM can handle both linear and non-linear data by using different kernel functions.
* Achieving **70%** accuracy.

**Clustering techniques:**

## **Density-Based Spatial Clustering**

* DBSCAN identifies clusters based on the density of data points.
* It doesn’t assume a fixed number of clusters.
* Points within a dense region are considered part of the same cluster.
* silhouette score :-0.74

## **K-means**

* K-means partitions data into K distinct clusters.
* It assumes spherical, equally sized clusters.
* Iteratively assigns data points to the nearest cluster centre (centroid).
* Minimizes the within-cluster variance.
* It aims to minimize the sum of squared distances within each cluster.
* silhouette score :-0.80

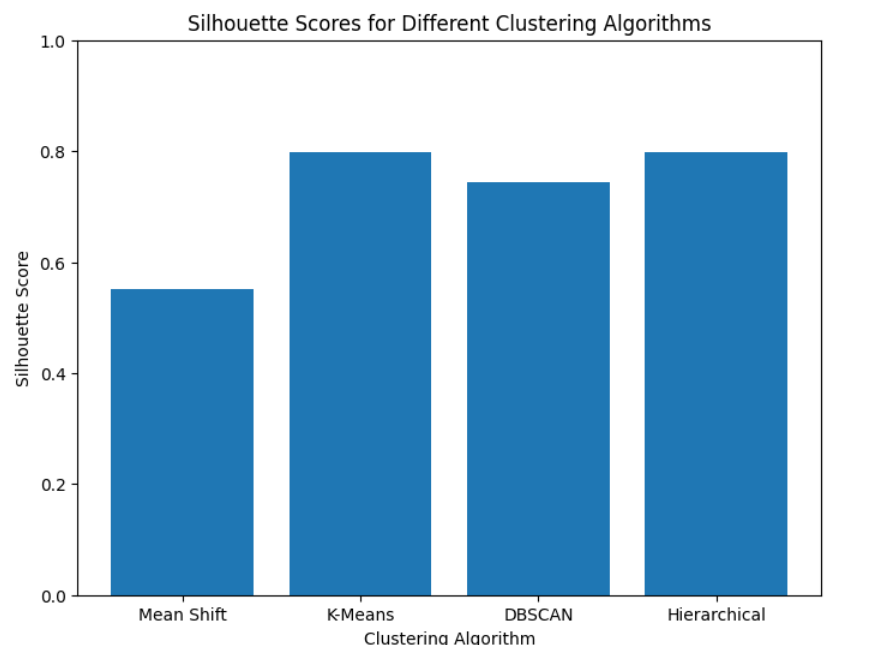
**Mean Shift**

* It identifies clusters by iteratively shifting data points towards the densest regions of the data distribution.
* Unlike K-Means, it does not require specifying the number of clusters beforehand.
* Mean Shift is particularly useful for datasets with arbitrary shapes and clusters that are not well-separated by linear boundaries.
* Silhouette score:-0.55

**Agglomerative Clustering**

* It starts by treating each individual data point as a single cluster.
* Then, it iteratively merges clusters based on their similarity until forming one large cluster containing all objects.
* Agglomerative Clustering is particularly good at identifying small clusters.
* Silhouette score:-0.80

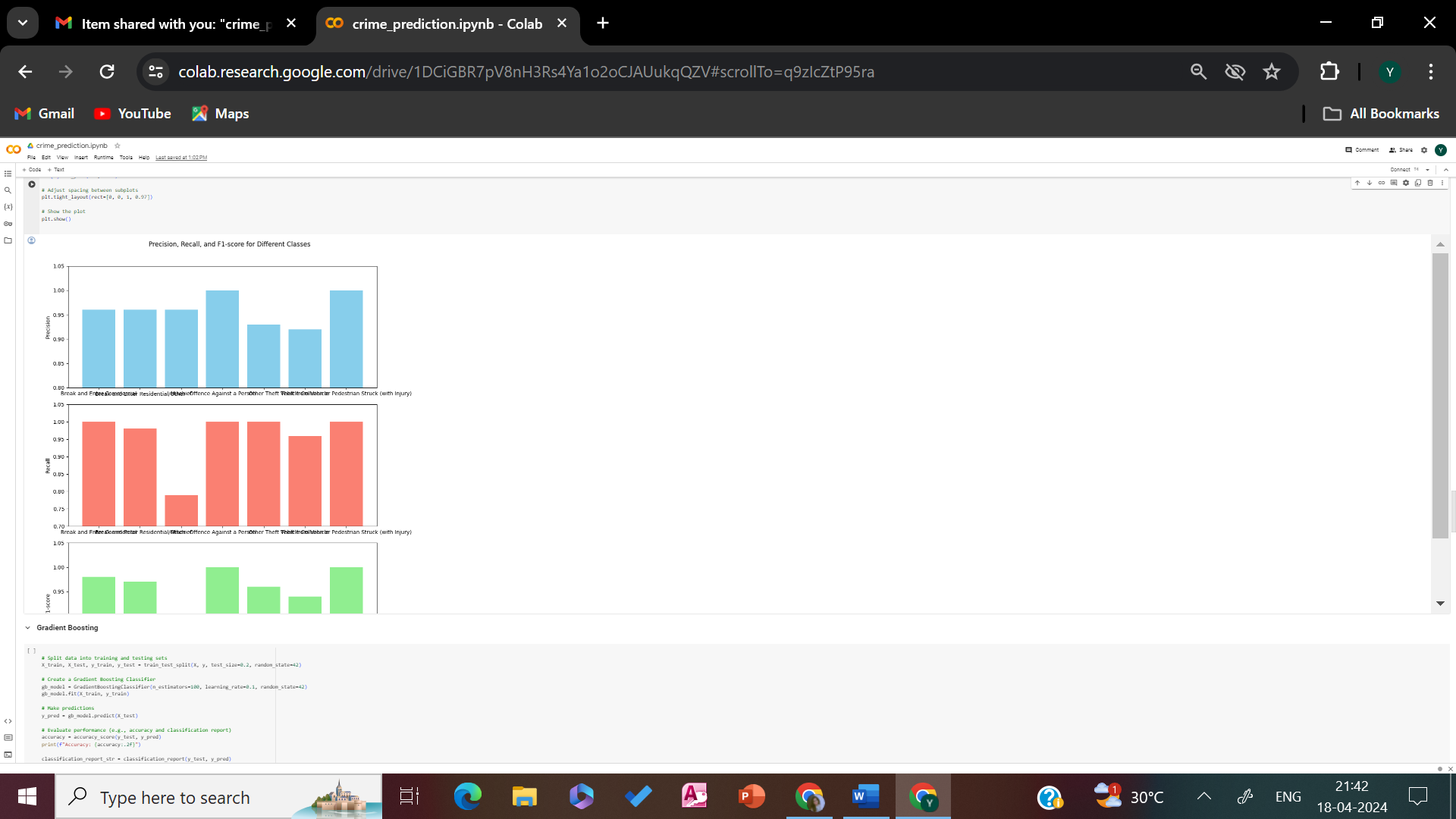
**Comparison Graph**

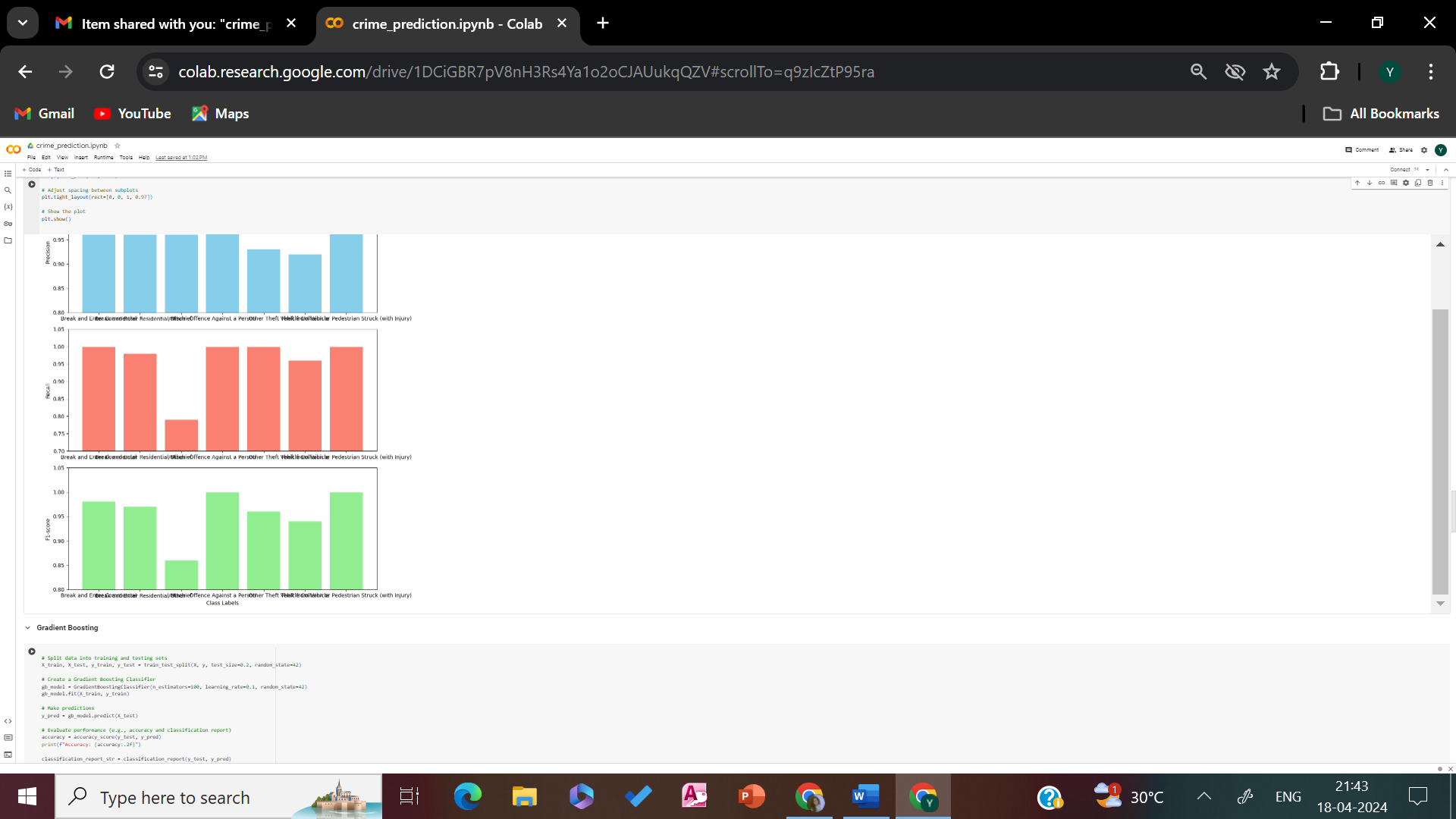
****

**Ensemble learning technique**

**Random Forest Classifier**

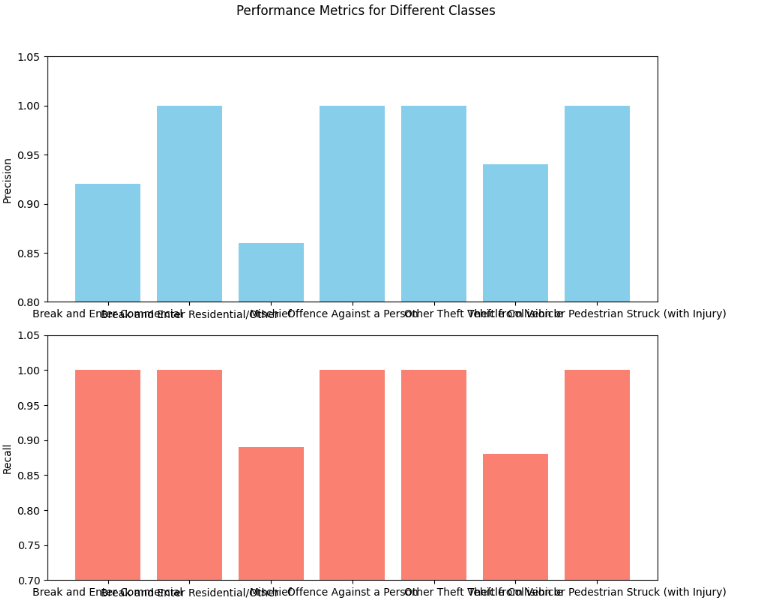
The model has high precision, recall, and F1-score across multiple classes, with an overall accuracy of 0.97. This suggests that the Random Forest Classifier model performs well on the given dataset.

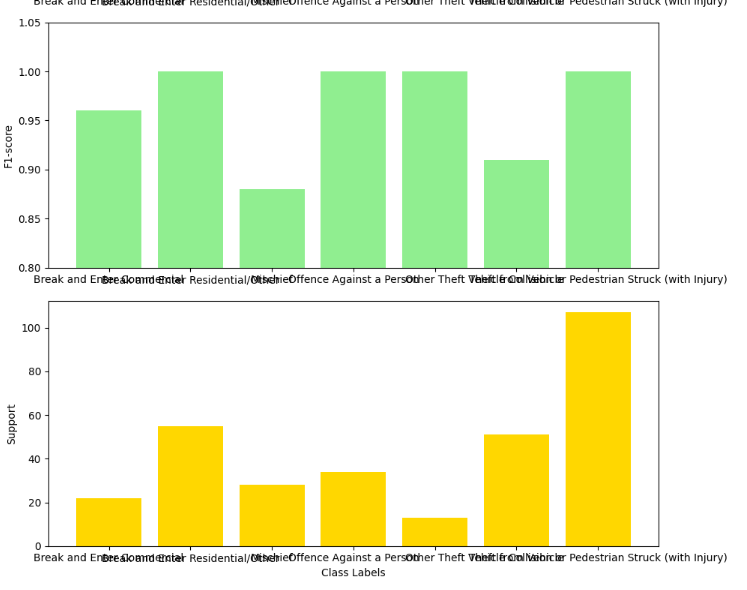




**Gradient Boosting**

The model has high precision, recall, and F1-score across multiple classes, with an overall accuracy of 0.97. This suggests that the Gradient Boosting Classifier model performs well on the given dataset, similar to the Random Forest Classifier model mentioned earlier. Both models have similar accuracy and performance metrics across different classes.

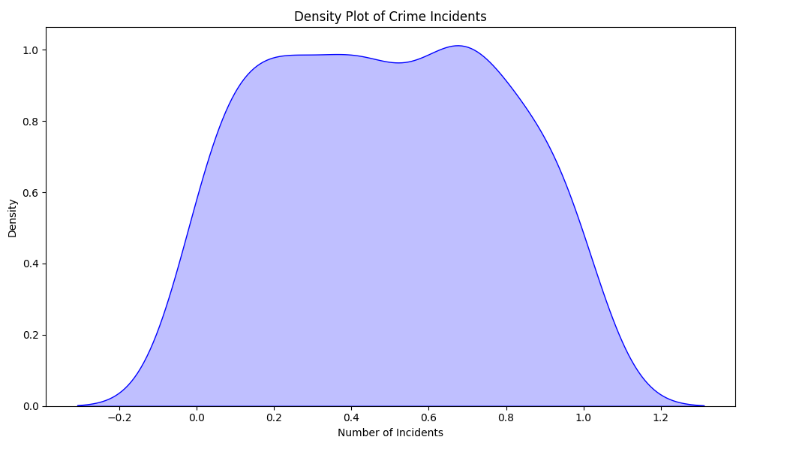




**RESULT:**

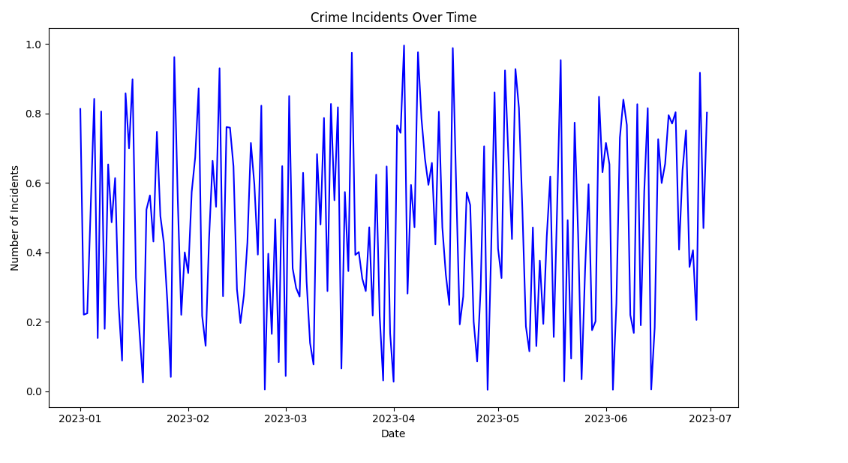
**1.Density Plot**

A density plot to visualize the distribution of crime incidents over time, providing insights into the density or concentration of incidents across the specified time period.



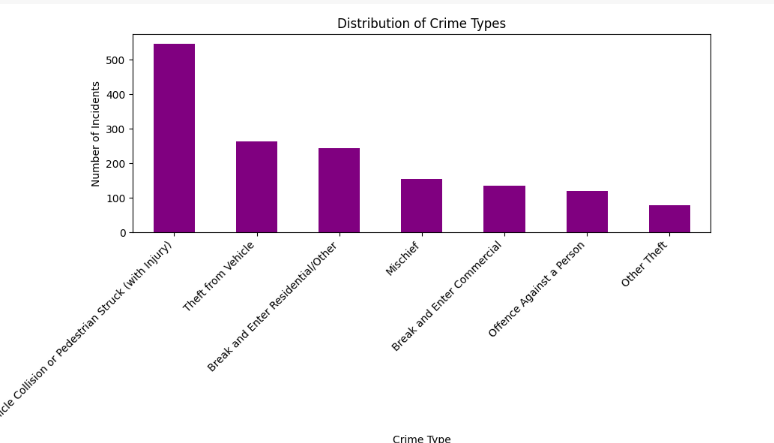
**2.Time Series**

A time series plot to visualize the trend of crime incidents over the specified time period, providing insights into how the number of incidents varies over time.



**3. Distribution of Crime Types**

A bar chart to visualize the distribution of crime types based on the count of incidents, providing insights into which crime types are most prevalent in the dataset.



**4. Hourly Crime Distribution**

A bar plot to visualize the distribution of crime incidents by hour of the day, providing insights into the variations in crime activity throughout the day.

