**REPORT**

**Team Name: Fabled Four (EN33)**

**Problem Statement:** Crime Rate Prediction Model

**Libraries Used:**

The implementation for a crime rate prediction model involves the following libraries:

1. **For Data Collection and Processing:**

The following libraries were used to scrape and collect relevant crime data from online sources like Open Govt. Data (OGD) website:

* **BeautifulSoup:** The purpose of BeautifulSoup is for web scraping, parsing HTML and XML documents from the website.
* **Requests:** The purpose of requests library is to make HTTP requests to fetch data from web APIs or webpages.

1. **For Data Preprocessing:**

It is mainly used for handling missing values, normalize numerical data and convert categorical columns to numerical format.

* **Pandas:** The purpose of Pandas library is for data manipulation and analysis of data scraped from the website.
* **Scikitb-learn (Sklearn):** The purpose of this library is to provide tools for machine learning, including preprocessing, model building and evaluation.
* **NumPy:** It provides support for numerical computations and arrays to be made from the data.

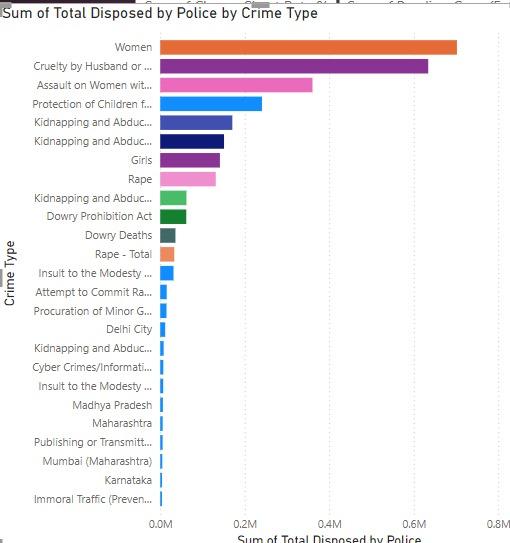
1. **For Data Visualization:**

* **Matplotlib:** It is used for static, animated and interactive visualization. It creates meaningful visualizations to analyze and interpret data.
* **Plotly:** A Python library for creating interactive, high-quality plots and visualizations.
* **Seaborn:** A Python library built on Matplotlib for creating visually appealing and informative statistical graphics.
* **plt.show:** A Matplotlib command to display the generated plots on the screen.
* **pd.DataFrame:** A Pandas data structure representing tabular data in rows and columns, similar to a spreadsheet.
* **plt.tight\_layout:** A Matplotlib function that adjusts plot spacing to prevent overlap between elements.
* **plt.legend:** A Matplotlib function to add a legend to the plot for identifying data elements.

**Types of Visualization used:**

**1. Distribution Visualizations:**

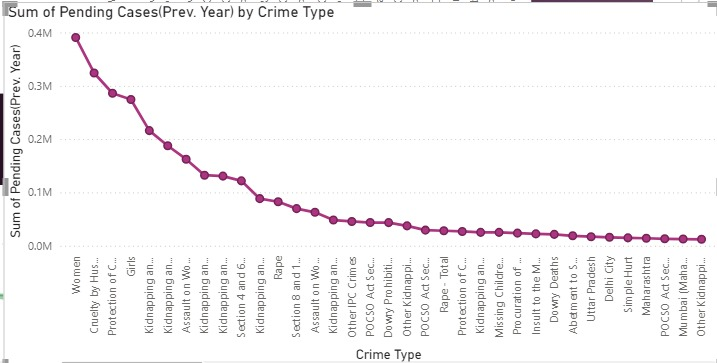
* **Box Plot**: Displays the distribution of data through quartiles, highlighting the median, outliers, and range.
* **Clustered Bar Chart**: Displays multiple bars for each category, grouped together, to compare different subcategories or groups within the same main category.



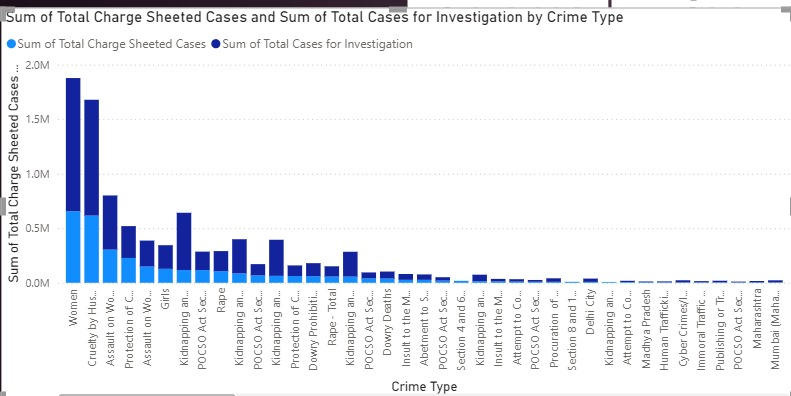
* **Strip Plot**: Displays individual data points along a single axis to show the distribution and variation in the dataset.

**2. Comparison Visualizations:**

* **Line Graph**: Shows the trend or change in data over time or another continuous variable using a series of connected points.

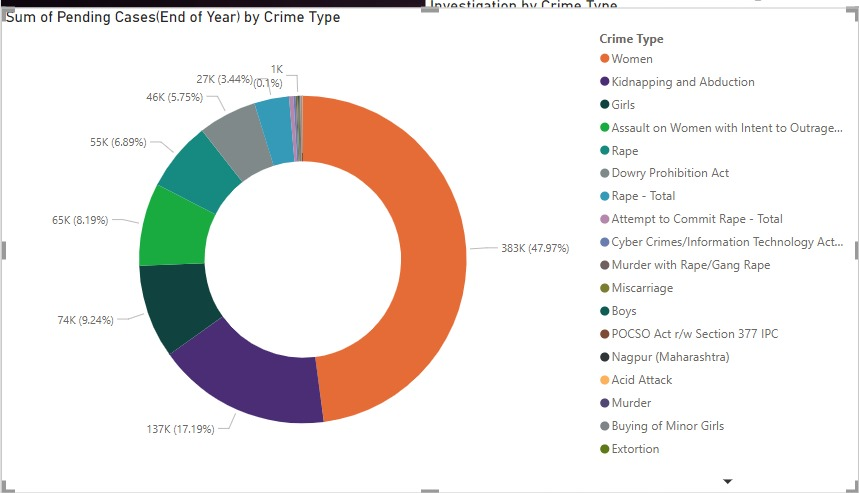


* **Stack Bar Chart**: Visualizes multiple categorical data segments stacked on top of each other to compare parts to a whole.
* **Stack Area Chart**: Similar to the stack bar chart, but uses areas to represent cumulative totals over time or other continuous variables.

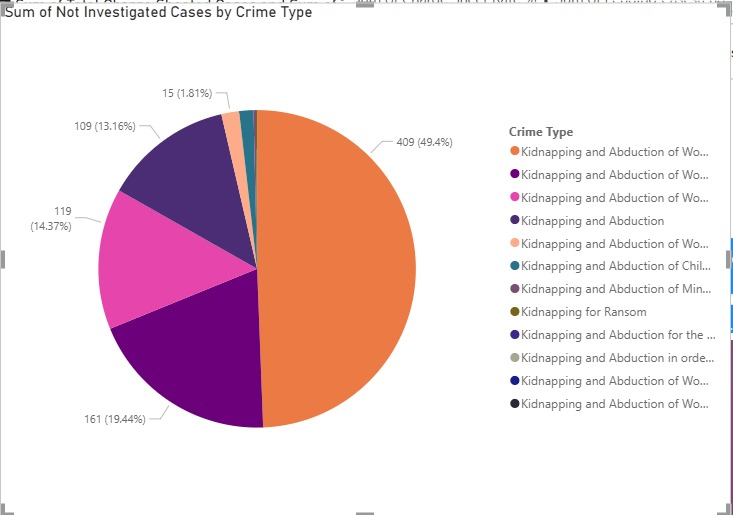


**3. Part-to-Whole Visualizations:**

* **Donut Chart**: A variation of the pie chart, showing proportions of categories with a circular, hollow centre.



* **Pie Chart**: Depicts the proportion of categories as slices of a circle, representing parts of a whole.



**4. Relationship Visualizations:**

* **Heatmap**: Uses color gradients to show relationships between two variables in a matrix, representing intensity or frequency.
* **Scatter Plot**: Displays the relationship between two continuous variables by plotting points on a two-dimensional axis.

**5. Distribution and Comparison (Advanced):**

* **Violin Plot**: Combines aspects of a box plot and kernel density plot, showing the distribution and density of data at different values.

**6. Grids and Layouts:**

* **Grid**: Organizes visual elements in a matrix-like layout to display multiple charts or images for comparison.

**7**. **Demographic:**

* State wise distribution of victims and crimes using visualization methods for crime rate.

**Solution of Crime rate prediction:**

* **Objective:** Predict future crime rates based on historical and socioeconomic data.

1. **Historical Crime Records**: Collect data on past crime incidents, including crime type, location, time, and the number of offenses.
2. **Socioeconomic Indicators**: Gather data on factors such as unemployment rates, education levels, income distribution, and social inequality that may influence crime rates.
3. **Geospatial Data**: Collect data on population density, geographical regions, proximity to key infrastructures (e.g., police stations, schools), and urban development.
4. **Environmental Factors**: Obtain data on weather patterns, such as temperature, rainfall, and time of year, which could impact crime occurrences.

* **Data Preprocessing**

1. **Data Cleaning**: Handle missing values, remove duplicates, and address any outliers or inconsistencies in the dataset.
2. **Feature Engineering**: Create additional features, such as crime trends over time, lag features (e.g., previous month's crime data), and population ratios to enhance predictive power.
3. **Normalization/Scaling**: Standardize or normalize numerical data to ensure that all features are on a similar scale, improving model performance.

* **Model Selection**

1. **Time Series Data**: For predictions based on time, use models like ARIMA, SARIMA, or Long Short-Term Memory (LSTM) networks, which are suited for sequential data.
2. **General Prediction**: For broader crime prediction tasks, apply machine learning models like Random Forest, Gradient Boosting, or Linear Regression to model relationships between variables.

* **Model Training**

1. **Train-Test Split**: Divide the data into training and testing datasets, typically using an 80-20 or 70-30 ratio to ensure proper validation.
2. **Model Training and Hyperparameter Tuning**: Train the model on the training set and optimize hyperparameters using cross-validation techniques to prevent overfitting and ensure robust performance.

* **Evaluation**

1. **Evaluation Metrics for Regression**: Use metrics like Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R-squared (R²) to assess the model’s accuracy and performance.
2. **Test Data Validation**: Validate the predictions by comparing the model’s output against the test data to check for generalization to unseen data.

* **Deployment**

1. **Visualization/Dashboard**: Build an interactive visualization or dashboard to display crime predictions, trends, and patterns for easy interpretation by stakeholders.
2. **Deployment Platforms**: Deploy the model and visualization using web frameworks like Flask or Django. Alternatively, deploy on cloud platforms such as AWS or Azure for scalability and easy access.

* **Monitoring**

1. **Continuous Monitoring**: Regularly monitor the model’s performance to ensure its effectiveness, and track key metrics over time.
2. **Model Update**: Periodically retrain the model with new crime data and update it to reflect changing trends and patterns in crime.

**-**------------------------------------------------------------------------------------------------------