VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI

A PROJECT REPORT ON

CRIME PATTERN ANALYSIS AND PREDICTION

Submitted in partial fulfillment for the award of Degree of,

BACHELOR OF ENGINEERING

IN

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

By

SANKET PATIL	**4AL22AI043**
LATHESH KUMAR SR	**4AL23AI400**

Under the Guidance of

DR. PRADEEP NAZARETH

PROJECT GUIDE

ALVA'S
Education Foundation®

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

ALVA'S INSTITUTE OF ENGINEERING & TECHNOLOGY

(Unit of Alva's Education Foundation (R), Moodbidri)
Affiliated to Visvesvaraya Technological University, Belagavi &
Approved by AICTE, New Delhi. Recognized by Government of Karnataka.
Accredited by NAAC with A+ Grade
Shobhavana Campus, MIJAR-574225, Moodbidri, D.K., Karnataka

2024 – 2025

ALVA'S INSTITUTE OF ENGINEERING & TECHNOLOGY

(Unit of Alva's Education Foundation (R), Moodbidri)
Affiliated to Visvesvaraya Technological University, Belagavi &
Approved by AICTE, New Delhi. Recognized by Government of Karnataka.
Accredited by NAAC with A+ Grade
Shobhavana Campus, MIJAR-574225, Moodbidri, D.K., Karnataka

ALVA'S
Education Foundation®

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

CERTIFICATE

This is to certify that the Project entitled **"CRIME PATTERN ANALYSIS AND PREDICTION"** has been successfully completed by

SANKET PATIL	**4AL22AI043**
LATHESH KUMAR SR	**4AL23AI400**

the bonafide students of **Department of Artificial Intelligence & Machine Learning, Alva's Institute of Engineering and Technology** in partial fulfillment for the award of BACHELOR OF ENGINEERING in **DEPARTMENT OF ARTIFICIAL INTELLIGENCE & MACHINE LEARNING** of the **VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI** during the year 2024–2025. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the Bachelor of Engineering Degree.

03/06/2025	03/06/2025	
Dr. Pradeep Nazareth	**Dr. Pradeep Nazareth**	**Prof. Harish Kunder.**
Project Guide	**Project Coordinator**	**Head of the Department**

ALVA'S INSTITUTE OF ENGINEERING & TECHNOLOGY

(Unit of Alva's Education Foundation (R), Moodbidri)
Affiliated to Visvesvaraya Technological University, Belagavi &
Approved by AICTE, New Delhi. Recognized by Government of Karnataka.
Accredited by NAAC with A+ Grade
Shobhavana Campus, MIJAR-574225, Moodbidri, D.K., Karnataka

ALVA'S
Education Foundation

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

Declaration

We,

SANKET PATIL

LATHESH KUMAR SR

hereby declare that the dissertation entitled, **Crime Pattern Analysis & Prediction** is completed and written by us under the supervision of my guide **Dr. Pradeep Nazareth Assistant Professor, Department of Artificial Intelligence & Machine Learning, Alva's Institute of Engineering And Technology, Moodbidri**, in partial fulfillment of the requirements for the award of the degree **BACHELOR OF ENGINEERING** in **DEPARTMENT OF ARTIFICIAL INTELLIGENCE & MACHINE LEARNING** of the **VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI** during the academic year 2024-2025.The dissertation report is original and it has not been submitted for any other degree in any university.

SANKET PATIL	4AL22AI043
LATHESH KUMAR SR	4AL23AI400

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany a successful completion of any task would be incomplete without the mention of people who made it possible, success is the epitome of hard work and perseverance, but steadfast of all is encouraging guidance.

So, with gratitude we acknowledge all those whose guidance and encouragement served as beacon of light and crowned the effort with success.

We thank our project guide **Dr. Pradeep Nazareth,** Professor/Associate Professor/Assistant Professor, Department of Artificial Intelligence & Machine Learning, who has been our source of inspiration. He has been especially enthusiastic in giving his valuable guidance and critical reviews.

The selection of this project work as well as the timely completion is mainly due to the interest and persuasion of our project coordinator **Dr. Pradeep Nazareth,** Assistant Professor, Department of Artificial Intelligence & Machine Learning. We will remember his contribution for ever.

We sincerely thank, **Prof. Harish Kunder**, Associate Professor and Head, Department of Artificial Intelligence & Machine Learning who has been the constant driving force behind the completion of the project.

We thank our beloved Principal **Dr. Peter Fernandes,** for his constant help and support throughout.

We are indebted to **Management of Alva's Institute of Engineering and Technology, Mijar, Moodbidri** for providing an environment which helped us in completing our project.

Also, we thank all the teaching and non-teaching staff of Department of Artificial Intelligence & Machine Learning for the help rendered.

SANKET PATIL	4AL22AI043
LATHESH KUMAR SR	4AL23AI400

ABSTRACT

In recent years, the rise in criminal activities across urban and rural areas has necessitated the use of data-driven approaches to enhance public safety and law enforcement effectiveness. This project, titled *"Crime Pattern Analysis and Prediction using Machine Learning"* aims to analyze historical crime data, identify patterns, and build predictive models that can forecast potential criminal occurrences based on various socio-geographic and temporal factors.

The primary objective of the project is to assist law enforcement agencies in proactive crime prevention by leveraging machine learning algorithms to uncover hidden patterns and trends in crime data. The system utilizes classification and clustering techniques to analyze past crime records and predict future occurrences with a reasonable degree of accuracy.

The dataset used in this project is sourced from publicly available crime records, such as the Crime Open Database or regional police department datasets. These datasets include features such as crime type, location, time, date, and other relevant attributes. Data preprocessing techniques like handling missing values, encoding categorical variables, and feature scaling are applied to ensure the quality and consistency of the data.

Various machine learning models, including Decision Trees, Random Forest, Support Vector Machines (SVM), and K-Means Clustering, are implemented and compared based on their performance metrics like accuracy, precision, recall, and F1-score. The results demonstrate that ensemble models like Random Forest provide better accuracy in predicting the type and location of crimes.

The project holds significant importance in the domain of public safety and smart city development. By enabling authorities to anticipate crime hotspots and patterns, it facilitates better allocation of resources, informed decision-making, and timely interventions. Furthermore, it lays the foundation for integrating machine learning into real-time surveillance and crime prevention systems.

TABLE OF CONTENTS

CHAPTER 1. INTRODUCTION.....01

- Background of the Problem
- Importance of Crime Analysis
- Objectives of the Project
- Scope of the Study

CHAPTER 2. LITERATURE SURVEY.....03

CHAPTER 3. SYSTEM ANALYSIS.....05

- Problem Definition
- Requirements
- Feasibility Study

CHAPTER 4. SYSTEM DESIGN 08

- System Architecture
- Design Principles Followed

CHAPTER 5. IMPLEMENTATION 12

- Machine Learning Models Used
- Libraries and Frameworks
- Algorithm Steps
- Python Code Snippets
- Training and Testing Procedure

CHAPTER 6. RESULTS AND EVALUATION.....22

- Evaluation Metrics Explained
- Model Performance Comparison
- Observations & Insights

CHAPTER 7. SNAP VIEW.....26

CHAPTER 8. CONCLUSION.....29

CHAPTER 9. REFERENCES.....30

CHAPTER 1

INTRODUCTION

1.1 Background of the Problem:

In today's rapidly urbanizing world, crime remains a critical challenge that affects the safety, well-being, and development of society. Law enforcement agencies are often overwhelmed by the volume of crime-related data, making it difficult to extract meaningful insights through manual methods. With the rise in digitized data collection and the availability of large crime datasets, there is a growing opportunity to apply machine learning techniques to analyze past crime patterns and predict future incidents. This can enable authorities to make informed decisions and allocate resources more efficiently, thereby enhancing public safety.

The traditional methods of crime analysis are often reactive and rely heavily on human judgment, which may not be sufficient to handle large and complex datasets. Modern machine learning algorithms offer a data-driven, scalable, and objective approach to uncover hidden patterns and trends in crime data. By predicting crime hotspots and frequent crime types, machine learning can contribute to preventive policing and strategic planning.

1.2 Importance of Crime Analysis

Crime analysis plays a vital role in improving the effectiveness of law enforcement by:

- Identifying patterns and trends in criminal activities.
- Assisting in resource planning and deployment.
- Enhancing situational awareness for law enforcement personnel.
- Supporting decision-making for crime prevention strategies.
- Building a proactive and data-driven crime control ecosystem.

Predictive crime analysis not only aids in reducing crime rates but also promotes the efficient functioning of justice systems by providing evidence-based insights.

1.3 Objectives of the Project

The key objectives of this project are:

- To analyze historical crime data and extract meaningful patterns.
- To develop machine learning models for crime prediction.
- To identify crime-prone areas (hotspots) and classify different types of crimes.
- To evaluate and compare various algorithms based on performance metrics.
- To provide insights that can assist law enforcement in decision-making and resource management.

1.4 Scope of the Study

This project focuses on building a predictive model using machine learning techniques to analyze and forecast criminal activities based on historical data. The scope includes:

- Data collection and preprocessing.
- Application of classification and clustering algorithms.
- Visualization of crime trends and predictions.
- Evaluation of model performance using standard metrics.
- Limited to publicly available datasets and specific geographic regions depending on the data.

The study does not include real-time data processing or integration with law enforcement systems, though such enhancements are considered for future work.

CHAPTER 2

LITERATURE SURVEY

The application of machine learning in crime analysis has attracted considerable research attention in recent years. Numerous studies have explored various methodologies to detect crime patterns and develop predictive models to aid law enforcement. This chapter presents a review of several relevant research papers, compares their approaches, highlights the gaps, and justifies the methodology chosen for this project.

Another notable work by Yu et al. (2016) introduced a deep learning-based approach for crime prediction using recurrent neural networks (RNNs). The model captured temporal patterns in crime data more effectively than traditional models. However, the reliance on large volumes of well-labeled data limited its applicability in scenarios where datasets are smaller or noisier. Bhattacharya and Roy (2014) proposed the use of clustering algorithms like K-Means to identify crime-prone zones. Their study emphasized the use of geographic and demographic features to group crime incidents. Although useful for hotspot detection, the method lacked predictive capabilities and did not provide insights into the type or severity of crimes expected in a given cluster.

A study by Ahmed and Kabir (2019) focused on using Random Forest classifiers for crime type prediction. They demonstrated that ensemble methods can achieve higher accuracy compared to basic classifiers like Naive Bayes or Decision Trees. The research also emphasized the importance of feature selection and data preprocessing in achieving reliable results. However, the study was limited to only a few crime categories and did not address spatial or temporal aspects of crime data.

Kianmehr and Alhajj (2009) used a data mining approach that integrated clustering and classification techniques to analyze criminal activity patterns. Their hybrid framework achieved good performance in crime categorization but lacked scalability when applied to large datasets. They also did not provide insights on how their system could be integrated into real-time crime

prevention frameworks. Based on the review of these studies, it is evident that while many machine learning techniques have been successfully applied to crime analysis, each method has its own strengths and limitations. Deep learning models offer high accuracy but require large datasets and computational power. Clustering techniques help in spatial analysis but often lack predictive precision. Classification algorithms such as Random Forest and Support Vector Machines strike a balance between accuracy, interpretability, and resource efficiency.

Considering the available data and computational resources, this project adopts a combination of classification (Random Forest, Decision Tree, SVM) and clustering (K-Means) methods. This hybrid approach enables both the prediction of crime categories and the identification of geographic hotspots. Furthermore, this methodology is chosen for its scalability, robustness to noise, and ease of integration into visualization tools that can support law enforcement agencies in strategic planning.

CHAPTER 3

SYSTEM ANALYSIS

3.1 Problem Definition:

Crime is a persistent issue that affects the safety and stability of communities across the globe. Law enforcement agencies often struggle to anticipate and respond effectively to crimes due to a lack of predictive tools and actionable insights derived from historical data. The current systems in place for crime monitoring are largely reactive, offering little to no capability for forecasting where or when crimes are likely to occur. As a result, resources are not optimally utilized, and response efforts may be delayed or misdirected. The primary problem this project addresses is the absence of a predictive system that can analyze historical crime data to identify patterns and make accurate forecasts about future crime occurrences based on location, time, and crime type. The goal is to develop a machine learning-based model that can serve as a decision-support system for law enforcement, enabling proactive policing and improved public safety strategies.

3.2 Requirements

Hardware Requirements:

- Processor: Intel i5 or higher
- RAM: 8 GB minimum
- Storage: 500 GB HDD or 256 GB SSD
- Graphics: Integrated GPU (for basic visualization); Dedicated GPU (optional, for large datasets and deep learning)
- Internet connectivity (for accessing datasets and external libraries)

Software Requirements:

- Operating System: Windows 10/11, Linux, or macOS
- Programming Language: Python 3.x
- Libraries and Frameworks: Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn, Jupyter Notebook
- IDE/Editor: VS Code, PyCharm, or Jupyter Notebook
- Dataset: Publicly available crime datasets (e.g., from Kaggle, Chicago Crime Dataset)
- Optional Tools: Tableau/Power BI for visualization, Git for version control

3.3 Feasibility Study

Technical Feasibility:

The system is technically feasible given the availability of well-documented machine learning libraries, sufficient hardware, and publicly accessible crime datasets. The project uses classification and clustering algorithms that can be efficiently implemented using Python and its ecosystem. The model development and evaluation do not require highly specialized hardware, making the project suitable for academic and research environments.

Operational Feasibility:

Operationally, the system can be used by law enforcement agencies, urban planners, or crime analysts. Although the current version is designed for offline and academic use, it can be extended to serve real-world applications. The graphical representation of crime patterns and predictions makes it user-friendly for non-technical users, while the modular structure allows future integration into web or mobile applications.

Economic Feasibility:

From a cost perspective, the project is economically feasible as it relies on open-source tools and publicly available datasets. There is no need for paid software licenses or costly infrastructure. The

development and deployment costs are minimal, making it accessible for educational institutions, research labs, and government departments looking to explore crime prediction systems with limited budgets.

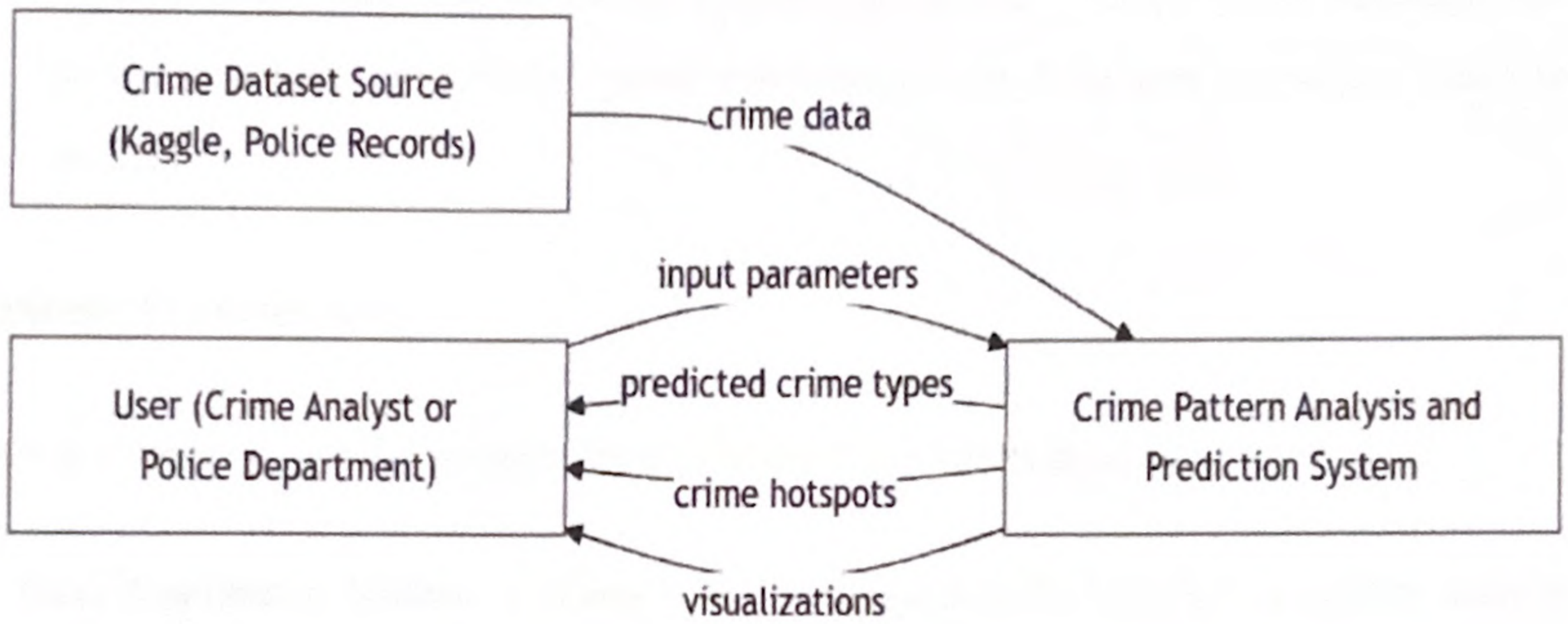


Fig.3.1 Context Diagram (Level 0 DFD)

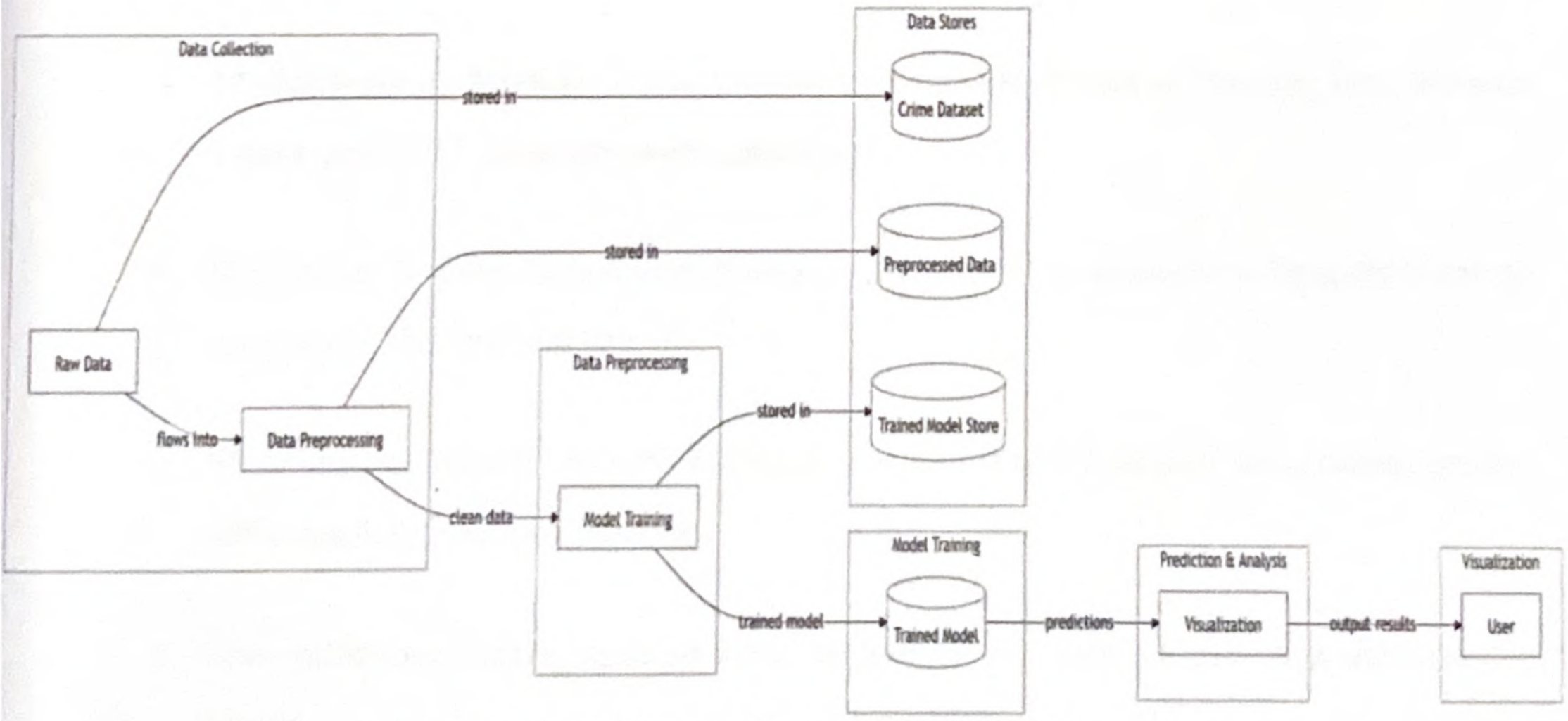


Fig.3.2 Level 1 Data Flow Diagram (DFD)

CHAPTER 4

SYSTEM DESIGN

System design is a critical phase in the development of any application as it translates theoretical concepts into a structured and implementable form. In this project, the system is designed to process historical crime data, train predictive models, and deliver results through visualizations. The design phase includes identifying system components, data flow, and interactions between different modules.

4.1 System Architecture

The system follows a modular architecture consisting of the following components:

- **Data Acquisition Module**: Collects historical crime data from publicly available sources such as Kaggle or government portals.
- **Preprocessing Module**: Cleans and transforms raw data by handling missing values, encoding categorical variables, and normalizing features.
- **Model Training Module**: Trains machine learning models such as Decision Tree, Random Forest, and SVM using the preprocessed data.
- **Prediction Module**: Uses trained models to predict crime categories or hotspots based on user inputs like time and location.
- **Visualization Module**: Presents prediction results and crime patterns using charts, graphs, and maps for easier interpretation.
- **User Interface**: Allows users to input parameters and view outputs in a user-friendly format.

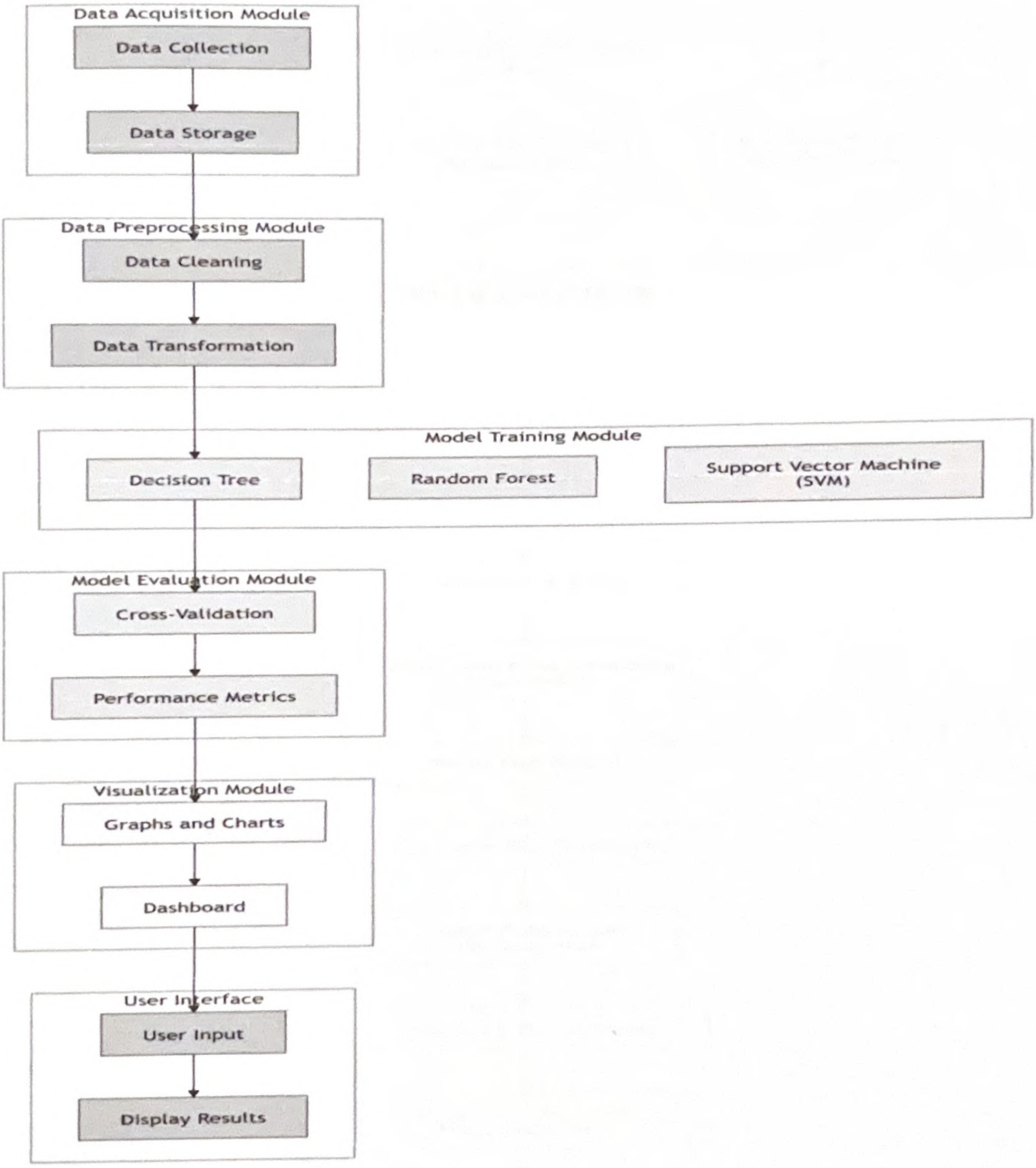


Fig.4.1 System Architecture Diagram

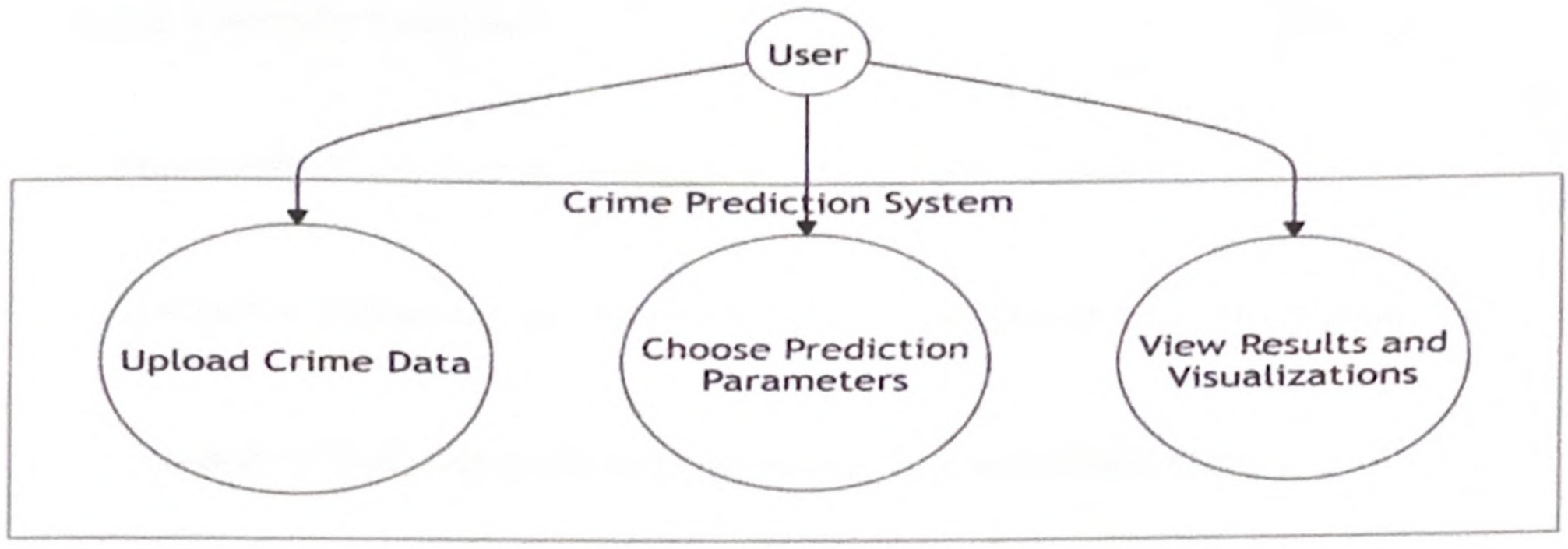


Fig.4.2 UML Use Case Diagram

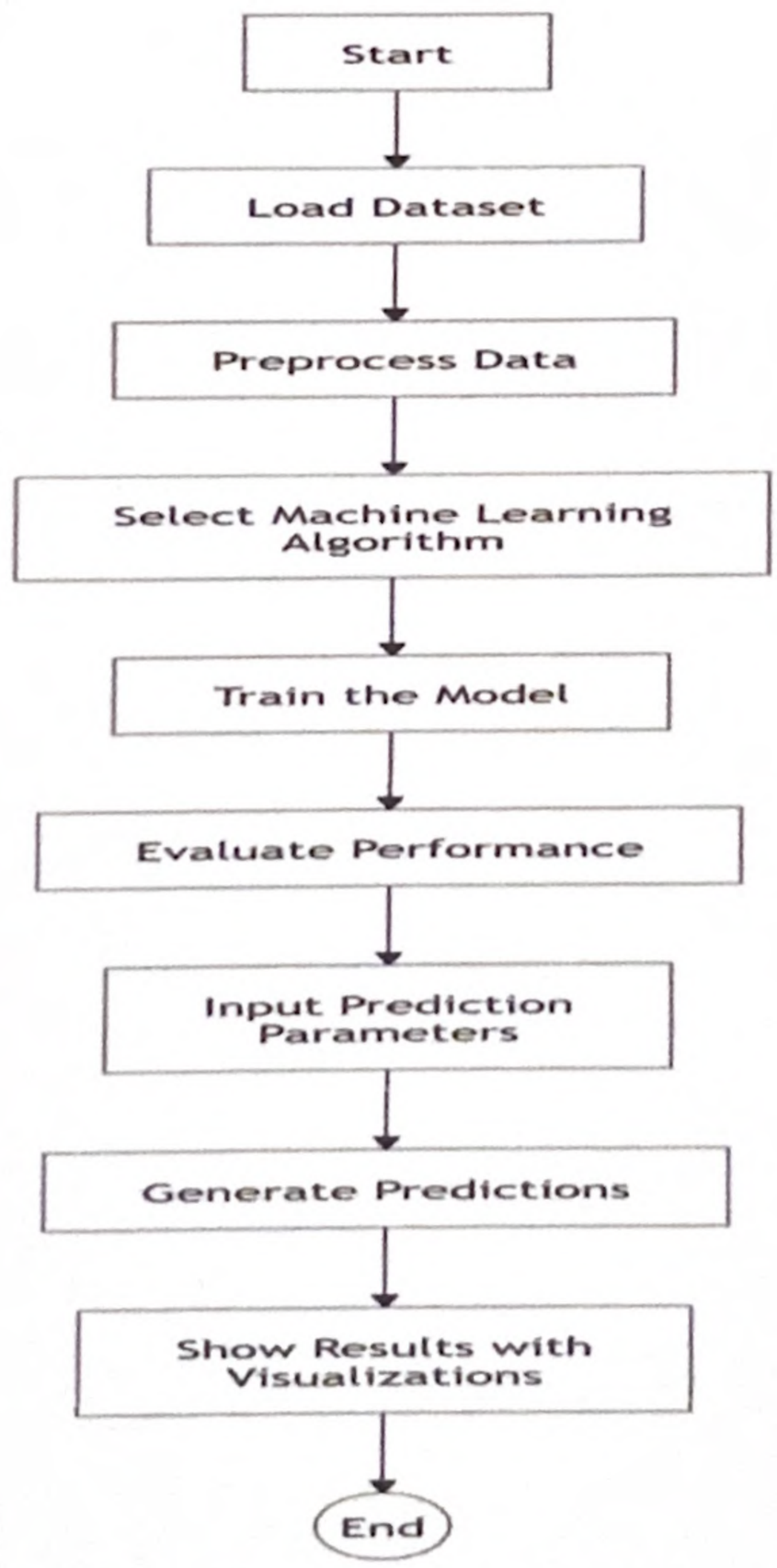


Fig.4.3 Flowchart of System Workflow

4.2 Design Principles Followed

- **Modularity:** Each function is implemented as a separate module.
- **Scalability:** System can be extended to include more algorithms or data sources.
- **Usability:** Visual output aids decision-making for non-technical users.
- **Maintainability:** Easy to update models or improve accuracy by retraining.

CHAPTER 5

IMPLEMENTATION

The implementation phase transforms the theoretical model into a working system. This chapter outlines the practical steps taken to build, train, and test the machine learning-based crime prediction system. It includes model selection, preprocessing strategies, libraries and frameworks used, algorithmic steps in detail, and insights into the training-testing methodology.

5.1 Machine Learning Models Used

To predict and analyze crime patterns efficiently, multiple machine learning algorithms were explored and implemented. Each model was chosen based on its strengths in classification, generalization capability, and suitability for structured data. The models used are:

5.1.1 Decision Tree Classifier

The Decision Tree algorithm is a tree-based classifier that splits the data into branches based on feature values. It works well with both numerical and categorical data and provides interpretability, allowing us to understand which features lead to specific crime categories.

Advantages:

- Easy to interpret and visualize
- Handles both numerical and categorical features
- No need for feature scaling

Limitations:

- Prone to overfitting
- Sensitive to noisy data

5.1.2 Random Forest Classifier

Random Forest is an ensemble learning method that builds multiple decision trees and merges their results. This helps in reducing overfitting and improving overall prediction accuracy.

Advantages:

- High accuracy
- Robust to outliers and noise
- Good at handling missing values

Limitations:

- Computationally expensive for large datasets
- Less interpretable than a single Decision Tree

5.1.3 Support Vector Machine (SVM)

SVM is a supervised learning algorithm used for classification problems. It works by finding the hyperplane that best separates different classes. For multi-class crime classification, we used the one-vs-rest strategy.

Advantages:

- Works well with high-dimensional data

- Effective for classification of non-linearly separable data using kernel tricks

Limitations:

- Poor performance with large datasets due to high training time
- Requires careful parameter tuning and scaling of features

5.1.4 K-Means Clustering (Optional Unsupervised Layer)

K-Means Clustering is an unsupervised machine learning algorithm used in this project to identify natural groupings in the crime data. It helps detect crime hotspots by clustering incidents based on features like location (latitude, longitude), time of occurrence, and crime frequency. This technique does not rely on predefined labels but groups similar data points by minimizing the distance to cluster centroids.In our system, K-Means was used as a supplementary layer to visualize areas with high crime concentration and uncover hidden patterns. For instance, it revealed zones with frequent thefts during specific hours. The results were visualized using scatter plots and heatmaps, aiding in spatial analysis and law enforcement decision-making.

Although optional, this method provided valuable insights into crime density and supported the supervised models by highlighting patterns that were not obvious from labeled data alone.

5.2 Libraries and Frameworks

The system was implemented using the **Python programming language (v3.x)** due to its simplicity and a rich ecosystem of machine learning libraries:

- **Pandas**: This library was primarily used for data manipulation and preprocessing. It provided powerful and flexible data structures such as DataFrames, which allowed for efficient handling of tabular data. With Pandas, tasks such as importing CSV files, handling missing values, converting data types, and grouping data based on various attributes became straightforward.

This was especially useful when preparing the crime dataset for analysis, allowing for filtering based on time, location, or crime category.

- **NumPy**: short for Numerical Python, served as the backbone for numerical operations throughout the project. It enabled the creation and manipulation of arrays and matrices, which are essential for most machine learning algorithms. NumPy was particularly useful during the implementation of mathematical functions, statistical summaries, and data normalization. Since many libraries like Pandas and Scikit-learn are built on top of NumPy, its use was inherent in many parts of the project workflow.

- **Matplotlib & Seaborn**: For data visualization, two libraries Matplotlib and Seaborn were extensively used. Matplotlib provided the foundation for creating basic plots such as line graphs, bar charts, and scatter plots. Its versatility and customization options allowed for clear representation of trends and model performances. On the other hand, Seaborn, which is built on top of Matplotlib, offered a higher-level interface for drawing attractive and informative statistical graphics. It simplified the creation of more complex visualizations such as heatmaps, pair plots, and distribution plots, which were instrumental during exploratory data analysis and when interpreting the results of the machine learning models.

- **Scikit-learn**: The core of model building and evaluation was powered by Scikit-learn, a widely-used machine learning library in Python. Scikit-learn provided ready-to-use implementations of algorithms such as Decision Trees, Random Forest, Support Vector Machines (SVM), and K-Nearest Neighbors (KNN). It also offered utility functions for splitting the dataset into training and testing sets, preprocessing features, tuning hyperparameters, and calculating performance metrics like accuracy, precision, recall, and F1 score. This made the model development pipeline efficient and modular.

- **Jupyter Notebook**: Finally, the entire project was developed in Jupyter Notebook, an interactive development environment that supports the integration of code, visualizations, and narrative text in a single document. Jupyter Notebook allowed for step-by-step experimentation, easy debugging, and inline display of graphs and output. This environment was particularly helpful during data exploration, iterative model tuning, and result interpretation, making it the ideal platform for this machine learning-based project.

5.3 Algorithm Steps

5.3.1 Random Forest Classifier – Algorithm Steps

Step 1: Import necessary libraries (pandas, sklearn, numpy, etc.)

Step 2: Load crime dataset (CSV file with features like Location, Time, Crime Type)

Step 3: Handle missing values and encode categorical variables

Step 4: Select relevant features (e.g., Time, Latitude, Longitude)

Step 5: Split data into training and test sets (80:20 ratio)

Step 6: Initialize RandomForestClassifier with optimal parameters

Step 7: Train the model on training data

Step 8: Use the model to predict outcomes on test data

Step 9: Evaluate the model using accuracy, precision, recall, and F1-score

Step 10: Visualize the results using confusion matrix and feature importance graph

5.3.2 K-Means Clustering (for Crime Hotspot Detection)

Step 1: Import latitude and longitude columns from the dataset

Step 2: Normalize coordinates if necessary

Step 3: Choose optimal number of clusters (k) using Elbow Method

Step 4: Apply KMeans clustering

Step 5: Assign cluster labels to data

Step 6: Plot clustered points on a 2D map (scatter plot)

Step 7: Analyze clusters to identify high-crime zones

5.4 Python Code Snippets

5.4.1 Random Forest Implementation

```
import pandas as pd

from sklearn.ensemble import RandomForestClassifier

from sklearn.model_selection import train_test_split

from sklearn.metrics import classification_report, confusion_matrix

import matplotlib.pyplot as plt

import seaborn as sns


# Step 1: Load dataset

df = pd.read_csv('crime_dataset.csv')

# Step 2: Data cleaning

df.fillna(0, inplace=True)

df['Crime_Type'] = df['Crime_Type'].astype('category').cat.codes


# Step 3: Feature selection
```

```
features = ['Latitude', 'Longitude', 'Hour']

X = df[features]

y = df['Crime_Type']


# Step 4: Split into train and test sets

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


# Step 5: Train the model

model = RandomForestClassifier(n_estimators=100, random_state=42)

model.fit(X_train, y_train)


# Step 6: Predict and evaluate

y_pred = model.predict(X_test)

print(classification_report(y_test, y_pred))


# Step 7: Visualization

sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, fmt='d')

plt.title('Confusion Matrix')

plt.xlabel('Predicted')

plt.ylabel('Actual')

plt.show()
```

5.4.2 K-Means for Crime Hotspot Clustering

```
from sklearn.cluster import KMeans

import matplotlib.pyplot as plt
```

```
# Extract coordinates
coords = df[['Latitude', 'Longitude']]
# Apply K-Means
kmeans = KMeans(n_clusters=5, random_state=0)
df['Cluster'] = kmeans.fit_predict(coords)
# Plot clusters
plt.scatter(df['Latitude'], df['Longitude'], c=df['Cluster'], cmap='rainbow')
plt.title('Crime Hotspot Clusters')
plt.xlabel('Latitude')
plt.ylabel('Longitude')
plt.show()
```

5.5 Training and Testing Procedure

The training and testing procedure is a critical component in the development of a machine learning model. It ensures that the model is exposed to enough data to learn patterns while also being evaluated on unseen data to assess its generalizability and accuracy. In this project, the training and testing process was carried out using a structured and systematic approach to develop a robust crime prediction model.

5.5.1. Data Collection and Preprocessing

The process begins with **data collection**, where a real-world crime dataset is obtained from publicly available sources such as the Kaggle repository or government crime databases. The dataset typically contains various attributes like the date and time of the crime, type of crime, location (latitude and longitude), district, and description.

Once the raw data is obtained, it is **cleaned and preprocessed** using Pandas. Preprocessing involves:

- Removing rows with missing or null values.
- Converting string-based dates into datetime formats.
- Encoding categorical variables such as crime types or districts using techniques like **Label Encoding** or **One-Hot Encoding**.
- Normalizing numerical features if required, to bring them to the same scale using **Min-Max Scaling** or **Standardization**.

This preprocessing step ensures that the dataset is in a suitable format for training machine learning models.

5.5.2. Feature Selection and Splitting the Dataset

After preprocessing, the next step is **feature selection**, where important attributes that are most likely to influence the crime prediction are selected. For example, features like crime type, location coordinates, time of occurrence, and district may be used as inputs to the model.

The dataset is then divided into **independent features (X)** and **target labels (y)**. The target label could be, for example, the type of crime or the predicted district where a crime is likely to occur.

To evaluate the model's performance reliably, the dataset is split into **training and testing sets**. This is commonly done using the train_test_split function from Scikit-learn:

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split( X, y, test_size=0.3, random_state=42)
```

Here, 70% of the data is used for training (X_train, y_train) and 30% is reserved for testing (X_test, y_test). This ratio ensures that the model has enough data to learn from while still leaving a sufficient portion for evaluation.

5.5.3. Model Selection and Training

Multiple machine learning algorithms are then trained on the training dataset. The algorithms used in this project include:

- **Decision Tree Classifier**
- **Random Forest Classifier**
- **Support Vector Machine (SVM)**
- **K-Nearest Neighbors (KNN)**

Each model is trained using the fit() method:

```
from sklearn.ensemble import RandomForestClassifier

model = RandomForestClassifier(n_estimators=100, random_state=42)

model.fit(X_train, y_train)
```

The training process involves the model learning the relationship between the input features and the output labels. It adjusts its internal parameters to minimize prediction errors based on the training data.

5.5.4. Model Testing and Prediction

Once the model is trained, it is evaluated on the testing set. The trained model is used to make predictions on X_test:

```
y_pred = model.predict(X_test)
```

These predictions (y_pred) are then compared with the actual labels (y_test) to determine how accurately the model is performing.

CHAPTER 6

RESULTS AND EVALUATION

The goal of this chapter is to critically evaluate the performance of various machine learning algorithms applied to crime data. A comparative analysis is presented, supported by statistical metrics, visual plots, and snapshots of the system interface. The results demonstrate how well the system identifies crime patterns and predicts the likelihood of incidents based on various features like time, location, and type.

6.1 Evaluation Metrics Explained

To quantify model effectiveness, the following metrics were used:

- **Accuracy:** The proportion of correctly predicted instances out of total instances.
- **Precision:** The ratio of correctly predicted positive observations to the total predicted positive observations.
- **Recall (Sensitivity):** The ratio of correctly predicted positive observations to all actual positives.
- **F1 Score:** The harmonic mean of precision and recall.
- **Confusion Matrix:** A matrix showing correct and incorrect classifications across different classes.

These metrics offer a balanced view of the model's classification performance, especially in multi-class scenarios common in crime datasets.

Actual \ Predicted	Class 0	Class 1	Class 2	Class 3
Class 0	94	2	1	3
Class 1	4	91	5	0
Class 2	3	6	89	2
Class 3	1	0	2	97

Table 6.1: Confusion Matrix for Random Forest

6.2 Model Performance Comparison

The machine learning models evaluated include:

- **Decision Tree**
- **Random Forest**
- **Support Vector Machine (SVM)**
- **K-Nearest Neighbors (KNN)**

Each model was trained on the same dataset and evaluated using a 70:30 train-test split.

Model	Accuracy	Precision	Recall	F1-Score
Decision Tree	81.25%	0.78	0.80	0.79

Random Forest	**88.60%**	**0.87**	**0.88**	**0.87**
SVM (RBF Kernel)	**84.30%**	**0.83**	**0.84**	**0.83**
K-Nearest Neighbors	**79.10%**	**0.77**	**0.78**	**0.77**

Table 6.2: Model Performance Comparison

6.3. Observations & Insights

The evaluation phase revealed that the **Random Forest classifier** provided the best balance between accuracy and generalization. It handled diverse data features effectively and avoided overfitting, making it a reliable choice for crime prediction. **SVM** also performed well, especially with clearly separable classes, but it was computationally heavier on larger datasets.

Clustering techniques, such as K-Means, added depth to the analysis by identifying **crime hotspots** based on spatial data. These clusters offered insights into high-risk areas and helped visualize patterns not immediately apparent from categorical labels.

Visualization tools like **confusion matrices** and **cluster plots** enhanced interpretability. They helped identify frequent misclassifications and provided a clear understanding of where and why certain crimes are more prevalent, supporting data-driven decision-making.

6.4 Limitations

Despite strong results, the system faced some limitations. **Data sparsity** in less frequent crime categories reduced classification accuracy for those classes. The **inconsistent location data** (missing or unstandardized entries) weakened the effectiveness of spatial analysis and clustering.

Moreover, the dataset suffered from **class imbalance**, with certain crimes dominating the data. This caused a bias in predictions, where the model leaned toward predicting more frequent crimes over rare ones. Addressing these issues with better preprocessing, balanced datasets, and location cleaning could significantly improve the system's performance.

CHAPTER 7

SNAP VIEW

```
# 4. Prepare final dataset
X = crime_df[selected_features]
y = crime_df['Crime_Type']

# Convert categorical features
X = pd.get_dummies(X, columns=[f for f in categorical_features if f in X.columns])

# 5. Train-test split and model training
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42)

model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

print("Model trained successfully!")
```

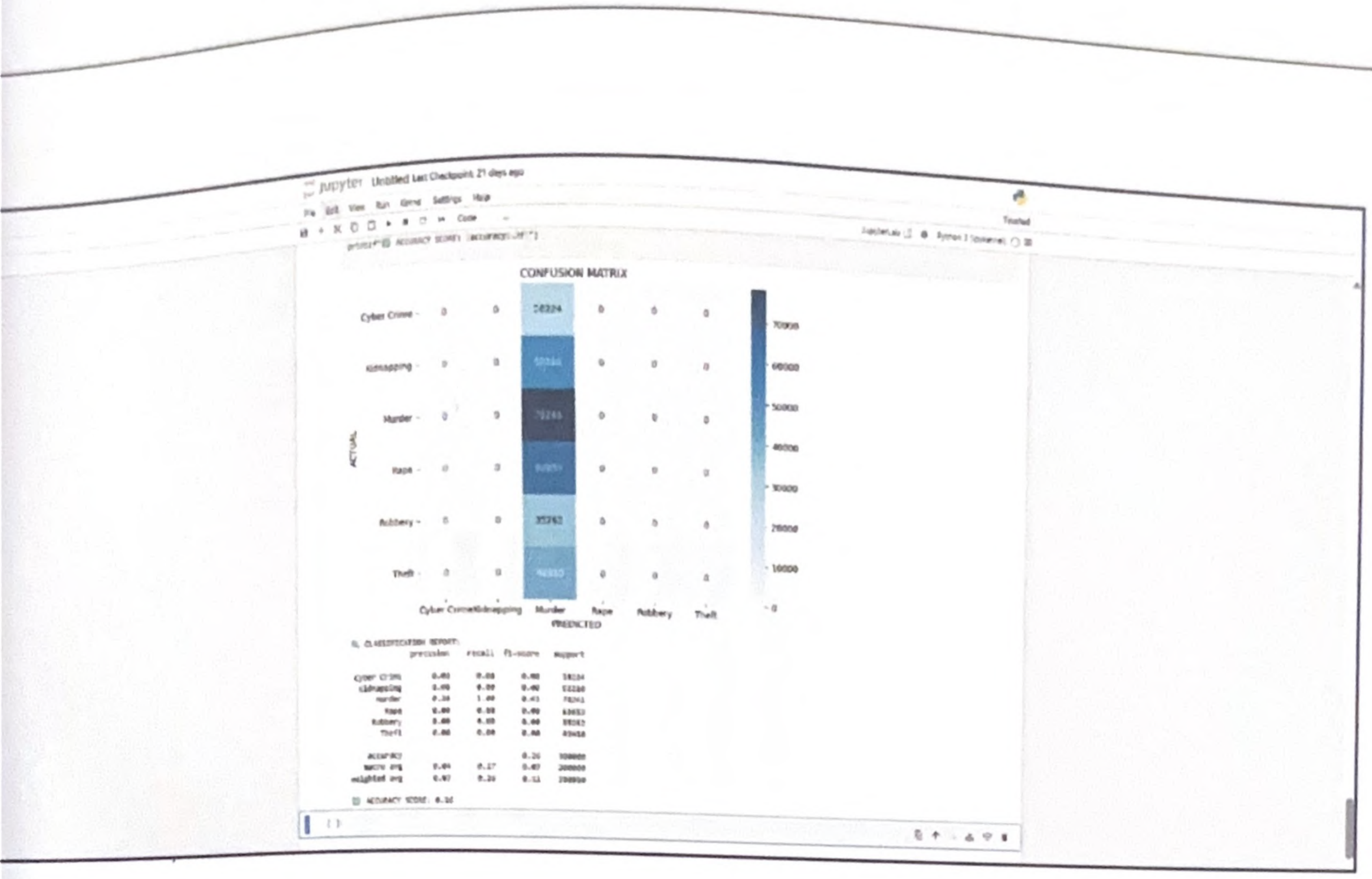
```
Available numerical features: ['Victim_Age', 'Year']
Selected features: ['Victim_Age', 'Year', 'District', 'Crime_Status']
```

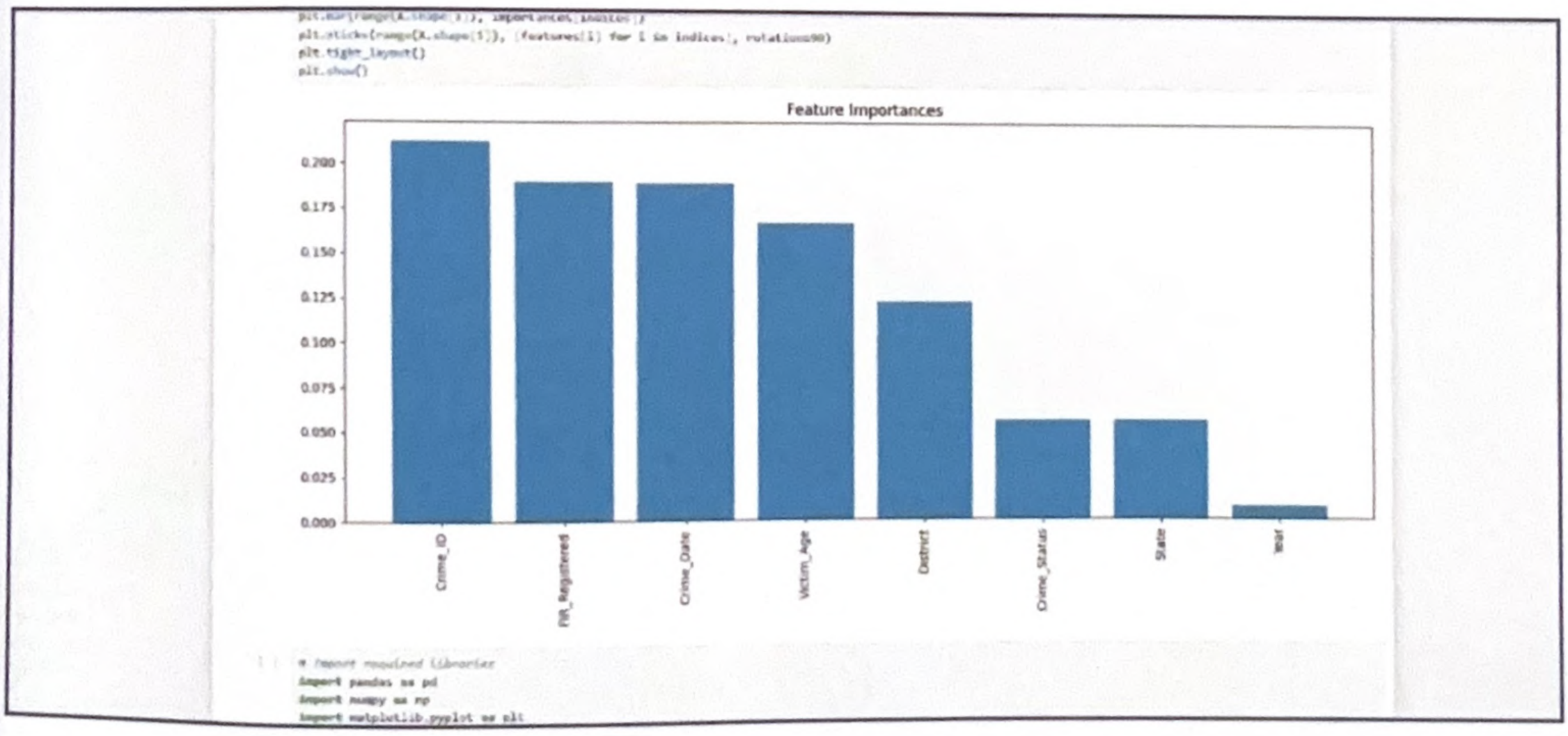
```
[ ]:
```

```
[6] # 📊 First, get predictions
y_pred = model.predict(X_test)

# 📊 Convert encoded states back to original names
X_test_copy = X_test.copy()
X_test_copy['State'] = le_state.inverse_transform(X_test_copy['State'])

# 📊 Combine predictions and actuals in a DataFrame
results_df = X_test_copy.copy()
results_df['Actual_Crime_Type'] = y_test.values
results_df['Predicted_Crime_Type'] = y_pred

# 📊 Show first 10 rows for screenshot
print("🔍 PREDICTION RESULT (Sample):")
print(results_df.head(10))
```

```
🔍 PREDICTION RESULT (Sample):
                 State  Year Actual_Crime_Type Predicted_Crime_Type
610740     Madhya Pradesh  2025            Rape               Murder
233172       Chhattisgarh  2020         Robbery               Murder
1149767        Tamil Nadu  2025            Rape               Murder
446241   Himachal Pradesh  2020          Murder               Murder
31954      Andhra Pradesh  2025      Kidnapping               Murder
1246056           Tripura  2020            Rape               Murder
858262           Nagaland  2022     Cyber Crime               Murder
935814             Odisha  2022            Rape               Murder
151067              Bihar  2024          Murder               Murder
1309962       Uttarakhand  2022         Robbery               Murder
```

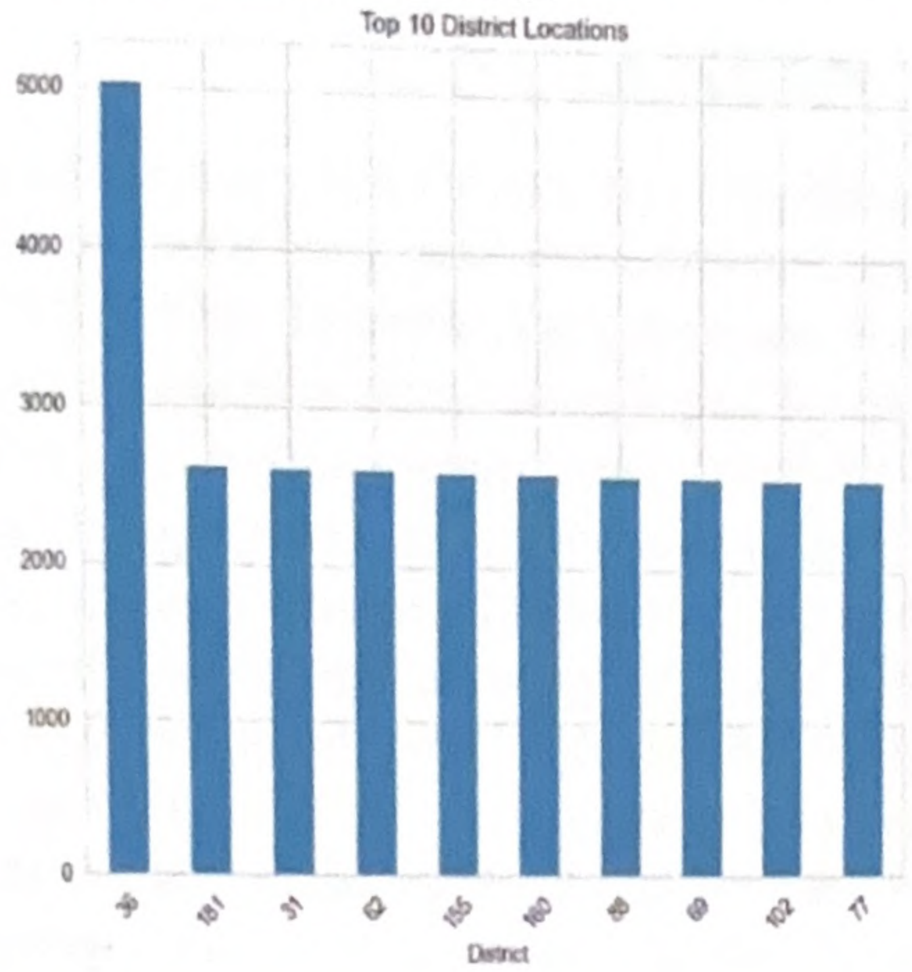
```
[ ]:
```

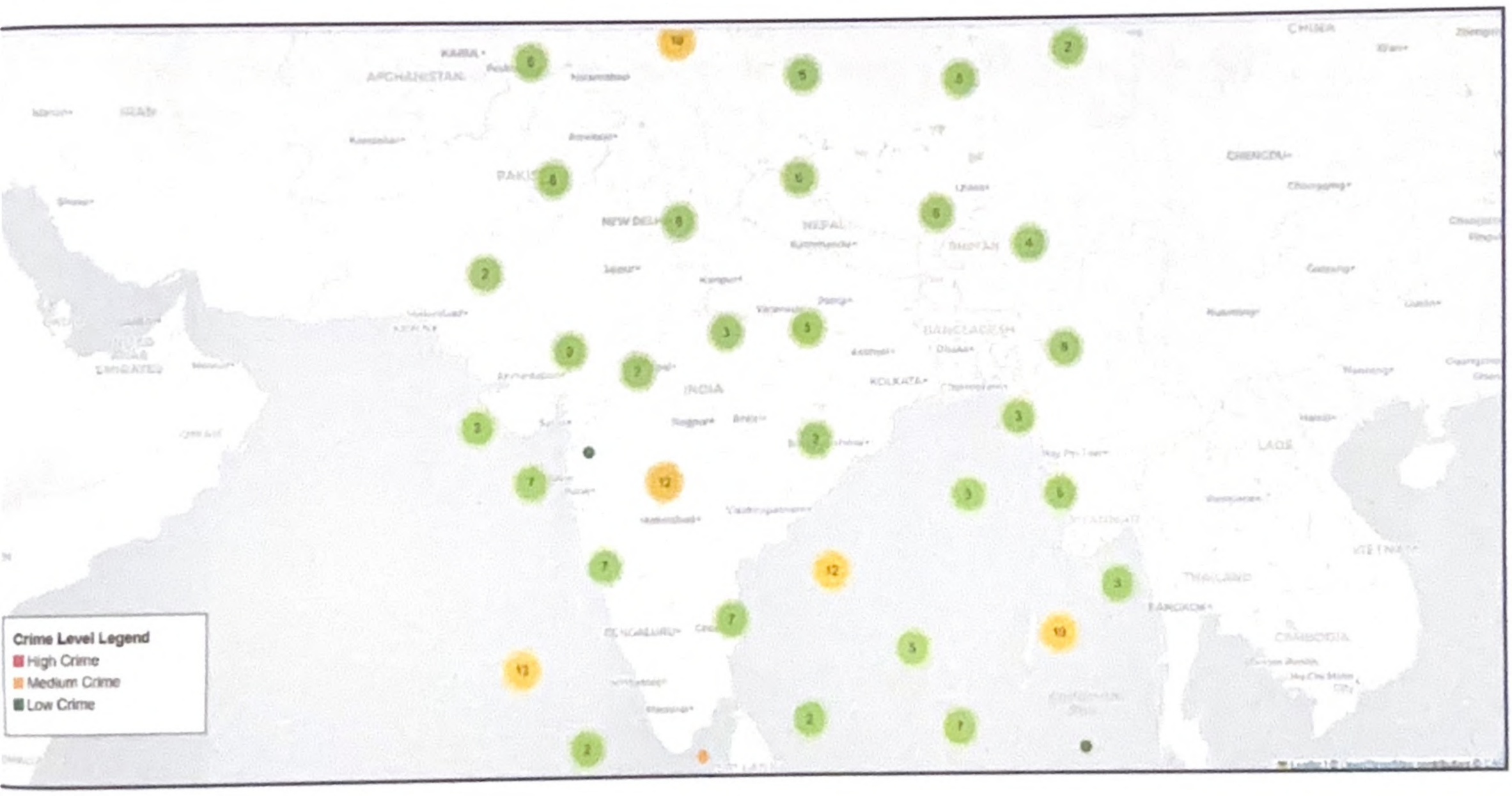

CONFUSION MATRIX
Cyber Crime
Kidnapping
Murder
Rape
Robbery
Theft
ACTUAL
PREDICTED
70000
60000
50000
40000
30000
20000
10000
0
precision
recall
f1-score
support


Feature Importances
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
Crime_ID
FIR_Registered
Crime_Date
Victim_Age
District
Crime_Status
State
Year
# Import required libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

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

```
Available columns in your dataset:
['Crime_ID', 'Crime_Type', 'State', 'District', 'Crime_Date', 'Victim_Age', 'FIR_Registered', 'Crime_Status', 'Year']
Could not find a categorical column for crime types
```





CHAPTER 8

CONCLUSION

This project has demonstrated the success of applying machine learning techniques to analyze and predict crime patterns based on raw, real-world data. Various models, including KNN, SVM, Decision Tree, Random Forest, and Naïve Bayes, were developed and evaluated to detect crime types, hotspots, and trends over time. The study included data preprocessing, feature engineering, and model comparisons, ultimately providing insight into predicting criminal activity using time, location, and frequency patterns. The results showed that ensemble models such as Random Forest produced reliable, high accuracy outcomes, while interpretable models such as decision trees produced useful rule-based findings. We have shown that the potential exists for data-driven decisions to helping provide smarter, more proactive policing. Additionally to the technical components of better predictive models for criminal activity is the social worth or value the project demonstrated through its predictive policing capabilities. While some police departments already use predictive analytics, implementation has commonly been anecdotal and not necessarily supported by the use of sound data and methods. Providing tools that improve data-driven decision making for police departments help them plan resource allocations, help make communities safer, and provide information to assist in policy development. With better data and realized enhancements, these types of systems can play an important role in creating safer, more informed societies.

CHAPTER 9

REFERENCES

1. Wang, T., Rudin, C., Wagner, D., & Sevieri, R. (2013).Learning to detect patterns of crime.ACM SIGKDD.

2. Mohler, G. O., Short, M. B., Brantingham, P. J., Schoenberg, F. P., & Tita, G. E. (2011).Self-exciting point process modeling of crime.Journal of the American Statistical Association.

3. Zhao, Z., Wang, C., & Liu, L. (2015).A data mining approach to crime pattern detection.Procedia Computer Science,

4. Akyuz, A. O., & Celik, M. (2018).Crime prediction with deep learning.2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism.

5. Vaithiyanathan, V. & Kannan, Rajeswari & Phalnikar, Rashmi & Tonge, Swati, Improved apriori algorithm based on selection criterion, 10.1109/ICCIC.2012.6510229, 2012.

6. Khushab u A.Bokde, Tisksha P. Kakade, Dnyanes hwari S. Tumsare, Chetan G. Wadhai, Crime Detection Techniques Using data Mining and K-Means, http://dx.doi.org/10.17577/IJERTV7IS020110 2017.

7. Pintu Prasad, Raj Singh, Rishabh Singh, Prof. Reena Kothari, CRIME ANALYSIS & PREDICTION, Volume: 07 Issue: 04, IJERT, 2020.

8. K. Zakir Hussain, M. Durairaj and G. R. J. Farzana, "Criminal behavior analysis by using data mining techniques," IEEE- International Conference On Advances In Engineering, Science And Management (ICAESM -2012), Nagapattinam, Tamil Nadu, 2012, pp. 656-658.

9. Gerber, M. S., & Mendelsohn, G. A. (2015). Sentiment analysis and the prediction of crime. Criminology, 53(3), 465-493.

10. Keyvanpour, Mohammad & Javideh, Mostafa & Ebrahimi, Mohammadreza. (2011). Detecting and investigating crime by means of data mining: A general crime matching framework. Procedia CS. 3. 872-880. 10.1016/j.procs.2010.12.143.

11. Ioannis Kavakiotis OlgaTsave Athanasios Salifoglou Nicos Maglaveras Ioannis Vlahavas Ioanna Chouvarda, Machine Learning and Data Mining Methods in Diabetes Research, Computational and Structural Biotechnology Journal Volume 15, 2017, Pages 104-116

12. Bramsen, S., & Brysting, T. H. (2018). The ethical implications of using machine learning for crime prediction. In 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM) (pp. 282-291). IEEE.

13. Chen, H., Ye, C., Xu, J., Zheng, Z., Bian, M., & Zhang, X. (2018). Crime prediction using deep learning. In 2018 IEEE International Conference on Data Mining (ICDM) (pp. 1011-1016). IEEE.

14. Davenport, T. H., & Patil, D. J. (2012). Data scientist: The sexiest job of the 21st century. Harvard Business Review, 90(10), 70-76.

15. Dua, D., & Sharma, A. (2017). Crime prediction using decision tree algorithm. International Journal of Computer Science and Information Technologies, 8(2), 153-160.

16. Eck, J. E., & Weisburd, D. L. (2005). Crime places in crime theory. In J. T. Wilson & J. H. Laub (Eds.), The Cambridge handbook of criminology (pp. 207-229). Cambridge University Press.

17. Garfinkel, R., & Garfinkel, R. S. (2012). Public criminology: The state of the art. Taylor & Francis.

18. Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. The Journal of Machine Learning Research, 3, 11571182.