CRIME RATE PREDICTION USING MACHINE LEARNING

Dr. Mukund Pratap Singh

Supervisor

Ishika Patni; Palak Tyagi

UG Students

School of Computer Science Engineering and Technology,

Bennett University

Abstract

The paper presents a holistic machine-learning framework for the prediction of district-wise IPC crime levels in India using data from 2001-2012. Accordingly, the proposed workflow combines systematic data preprocessing, normalization, exploratory data analysis, PCA-based dimensionality reduction, and training of several regression models. Several machine-learning algorithms, such as Linear Regression, Decision Tree, Random Forest, KNN, all major variants of SVM, and XGBoost, were considered with the metrics R^2, RMSE, and MAE. PCA effectively reduced feature dimensionality while retaining most of the dataset's variance, improving model performance.

The experimental results described below show that tree-based models, especially XGBoost and Decision Tree, outperform linear and kernel-based methods. Indeed, these capture non-linear crime patterns more precisely. The final XGBoost model achieved the highest R^2 score of 0.9768 and was used to generate future predictions of total IPC crimes. These results show that machine learning uncovers hidden trends, improves predictive accuracy, and provides better support for data-driven decisions on resource allocation and crime prevention.

This work tackles shortcomings in previous research on crime prediction, as much of it has focused on traditional models, clustering methods, or mere statistical forecasting without exploring dimensionality reduction, SVM kernel variations, or comprehensive comparisons of models. Complete preprocessing, PCA-based reduction, and objective performance evaluation across eight different algorithms provide a metric-driven, end-to-end pipeline that will definitely yield results that can show actionable insights for policymakers and law enforcement, filling critical gaps left by prior studies.

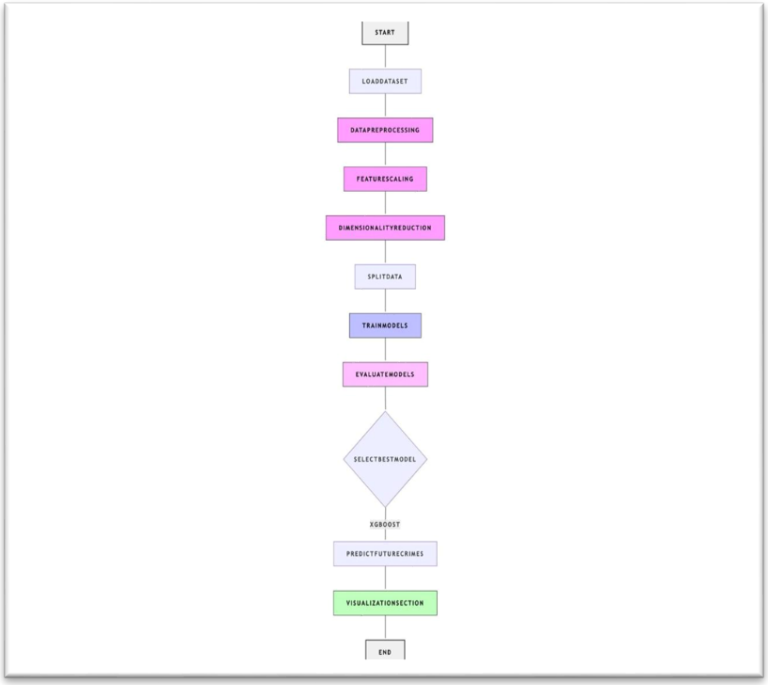
Introduction

Crime remains one of the most intransigent problems facing India and all rapidly developing countries. In fact, as cities grow, populations increase, and socioeconomic conditions develop, the patterns of crime also evolve in complex and unforeseeable ways. The comprehension of such trends is crucial for public safety, policing, and planning in the long term. However, traditional methods of crime analysis rely mostly on manual interpretation of large datasets, which render the identification of deeper relationships, long-term trends, and hidden patterns across districts rather difficult.

In the past couple of years, machine learning has emerged as a robust tool for analyzing big and complex datasets. Detecting patterns, modeling non-linear relationships, and, therefore, making precise predictions opens promising opportunities for crime analysis. The ML technique converts crime data into meaningful insights that promote smart decision-making, early intervention, and efficient resource allocations.

The project discussed here focuses on district-wise IPC crime data from 2001 to 2012 and tries to consider how machine-learning models could be used for predicting total-IPC-crime levels. The key steps involved in this study include data cleaning, exploratory analysis, feature scaling, dimensionality reduction using PCA, and training multiple regression models involving Linear Regression, Decision Tree, Random Forest, KNN, all major variants of SVM, and XGBoost. By comparing these models based on R^2, RMSE, and MAE, this project determines the best approach for predicting crime.

Overall, this work shows how different data-driven techniques can provide valuable insights into the crime behavior in India and also highlights the potential of machine-learning systems in supporting modern policing strategies enabled by technology.



Methodology

The approach of the project is based on a structured end-to-end machine learning pipeline, which provides a clear way of processing district-wise IPC crime data to develop reliable predictive models. The complete workflow, including all variants of SVM, can be summarized below:

1. **Data Loading and Import**
   - Next, district-wise IPC crime statistics from 2001 to 2012 was imported using Pandas.
   - Columns included major crime categories such as Murder, Rape, Kidnapping, Theft, Robbery, Riots, and other IPC offenses.

2. **Data Cleaning and Preprocessing**
   - Standardization of column names and removal of unwanted characters.
   - Handling of missing values, inconsistent formatting, and non-numeric entries.
   - Extraction of statistical summaries to verify dataset quality.

3. **Exploratory Data Analysis (EDA)**
   - To understand the pattern of underlying crimes,
   - Temporal crime trends were visualized as line plots.
   - State-wise comparisons were made to identify high-crime regions.
   - A correlation heatmap was created to explore relationships between crime categories.
   - Scatter plots, bar graphs, and distribution plots were useful for finding clustering behavior, outliers, and dominant types of crime.

4. **Feature Scaling**
   - All the numerical features were standardized using StandardScaler to ensure equal contribution by each category of crime.
   - Scaling was done to prepare the data for PCA and SVM models.

5. **Dimensionality Reduction using PCA**
   - Principal Component Analysis was utilized to remove redundancy in the crime features.
   - This PCA retained 95% of total variance with only 3 principal components, thereby greatly simplifying the model without any loss of information.

6. **Machine Learning Models Implemented**
   - Multiple regression models were trained on PCA-transformed data.

- The complete set of models used includes:

  A. Linear Models : Linear Regression

  B. Tree-Based Models : Decision Tree Regressor , Random Forest Regressor

  C. K-Nearest Neighbors : KNN Regressor

  D. Support Vector Regression - All Variants

  SVR (Linear Kernel) , Support Vector Regression (Polynomial Kernel) , SVR (RBF Kernel) ,

  SVR (Sigmoid Kernel)

These variants were compared to study how different decision boundaries affect crime prediction.

E. Boosting Models : XGBoost Regressor

7. Model Training

- This dataset was split into training and testing subsets.
- All models were trained on scaled and PCA-reduced crime features.
- This training process was repeated for all variants of SVM, in order to study the effect of different kernels.

8. Model Evaluation

- Each model, including all SVM kernels, was tested with the following : R^2 Score, Root Mean Squared Error – RMSE, Mean Absolute Error, MAE

The ranking of the models was based on highest accuracy and lowest error.

9. Best Model Selection

- Based on the evaluation metrics, Decision Tree and Random Forest Regressor outperformed other models such as SVM variants, linear regression, and KNN. These models captured nonlinear patterns in crime data more effectively.

10. Predicting Future Crimes

- Input values passed the preprocessing pipeline: Scaler → PCA → Best Model. The final model, based on the input parameters, generated predictions for future crime counts.

Data Structures and Algorithms Used

In this project, different data structures and machine-learning techniques were used to understand and predict IPC crime levels across districts. The methods are supported by a few important mathematical formulas that help the model process, simplify, and learn patterns from the data.

1. Data Structures

Pandas DataFrame:

The crime dataset was stored in a DataFrame, which makes it easy to clean the data, select columns, remove errors, and analyze trends.

NumPy Arrays:

After preprocessing, most of the data was converted into NumPy arrays. These arrays allow fast mathematical operations, which are needed for PCA and machine-learning models.

Matrices:

PCA and model training use matrices internally.
For example:

- PCA forms a covariance matrix
- Models receive a feature matrix X and a target vector y

Lists and Dictionaries:

These were used to:

- Store multiple models
- Save their evaluation scores
- Compare all algorithms efficiently

2. Algorithms Used

A. Feature Scaling

Before training the models, all features were standardized so that they were on the same scale. This avoids bias toward features with large values.

Standardization Formula:

$Z=(X-\mu)/\sigma$

This simply means: subtract the mean and divide by the standard deviation.

B. Principal Component Analysis (PCA)

PCA was used to reduce the number of crime features (from 9 original features to 6 important components). It works by finding new "directions" in the data that carry the most information.

Covariance Matrix:

$Cov(X) = 1/(n-1)(X - \bar{X})^T(X - \bar{X})$

This matrix shows how different features vary with each other.

Eigenvalue–Eigenvector Relation:

$$Cov(X)v = \lambda v$$

Here, v gives the direction (principal component), and λ tells how important it is.

PCA Transformation:

$$Z = XW$$

This creates a simpler version of the dataset, which is then used for model training.

C. SVM Kernel Functions

Since the project included all SVM variants, different kernel formulas were used to transform the data into forms where patterns are easier to detect.

Linear Kernel:

$$K(x_{\mathrm{i}}, x_{\mathrm{j}}) = x_{\mathrm{i}}^{\mathrm{T}} x_{\mathrm{j}}$$

Polynomial Kernel:

$$K(x_{\mathrm{i}}, x_{\mathrm{j}}) = (x_{\mathrm{i}}^{\mathrm{T}} x_{\mathrm{j}} + c)^{\mathrm{d}}$$

RBF Kernel:

$$K(x_{\mathrm{i}}, x_{\mathrm{j}}) = e^{(-\gamma ||x_{\mathrm{i}} - x_{\mathrm{j}}||^{2})}$$

Sigmoid Kernel:

$$K(x_{\mathrm{i}}, x_{\mathrm{j}}) = \tanh(\gamma x_{\mathrm{i}}^{\mathrm{T}} x_{\mathrm{j}} + c)$$

Each kernel provides a different way of measuring similarity between data points.

D. K-Nearest Neighbors (KNN)

KNN predicts values by looking at the “closest” data points.

Euclidean Distance:

$$d(p, q) = \Sigma(p_{\mathrm{i}} - q_{\mathrm{i}})^2$$

This tells how far two records are from each other.

E. Model Evaluation Metrics

To check how well each model performed, three common metrics were used:

R^2 Score:

$$R^2 = 1 - \Sigma(y - \hat{y})^2 / \Sigma(y - \bar{y})^2$$

Shows how well the model explains the data.

RMSE:

$$RMSE = 1/n\Sigma(y - \hat{y})^2$$

Tells how large the errors are, on average.

MAE:

$$MAE = 1/n\Sigma|y - \hat{y}|$$

Shows the average absolute mistake the model makes.

3. Exploratory Data Analysis

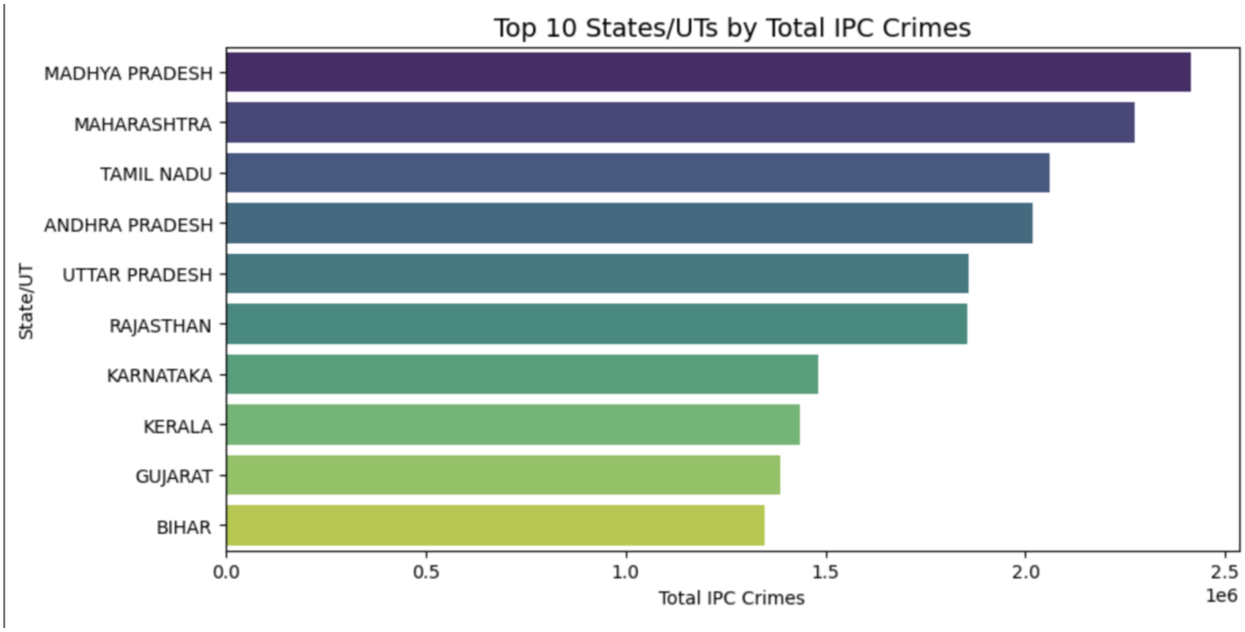


Figure 1: IPC Crime Trend Over the Years

Figure 2: Top 10 States by IPC Crime Levels

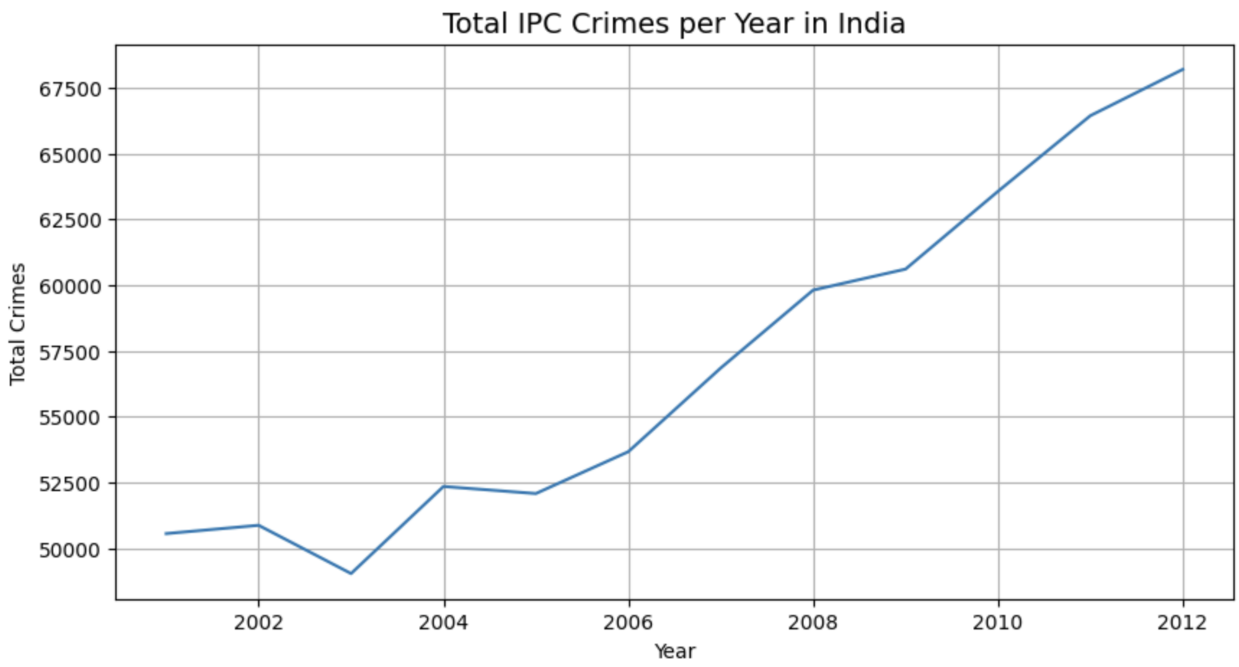


Figure3: Correlation Heatmap of Crime Features

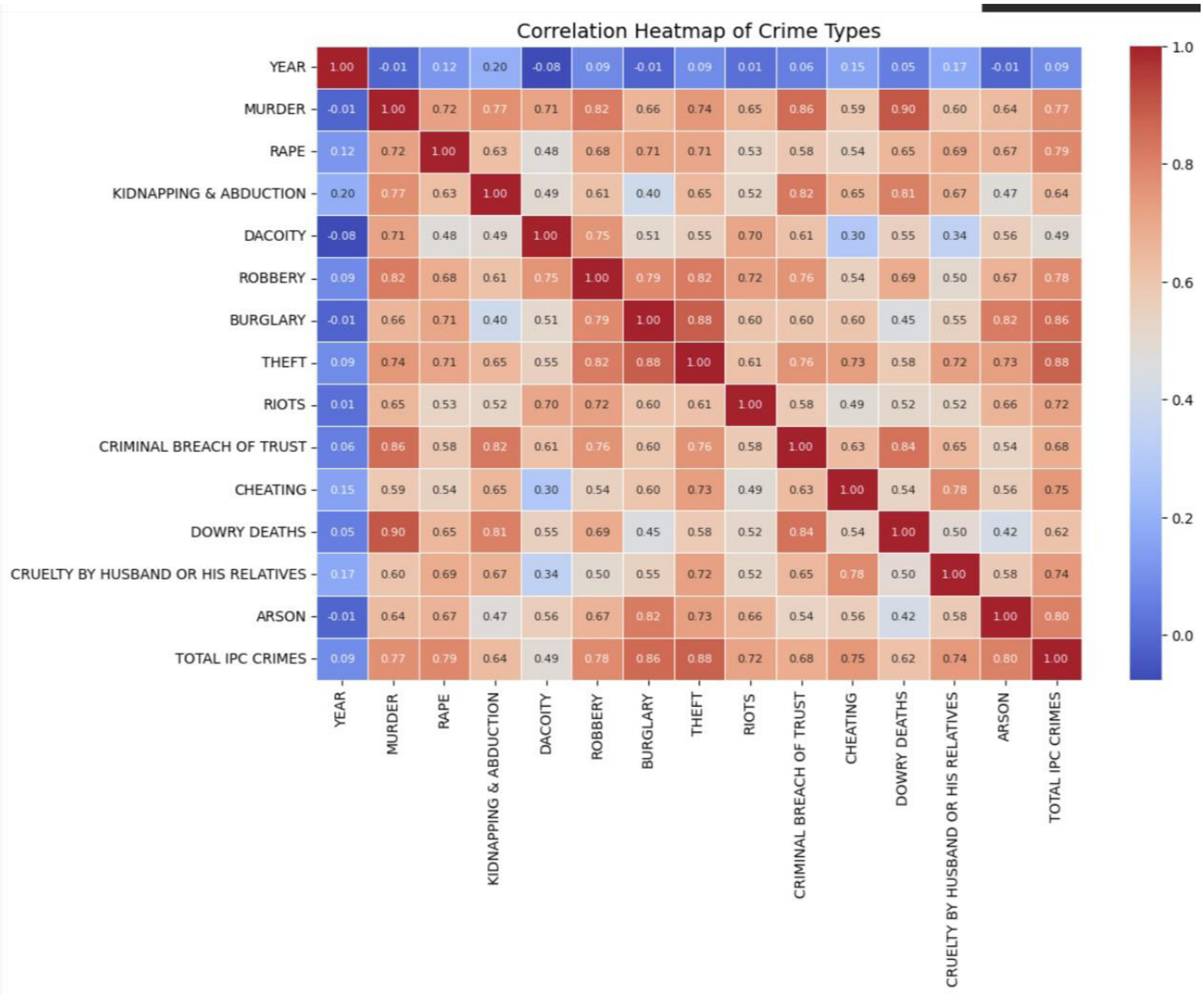
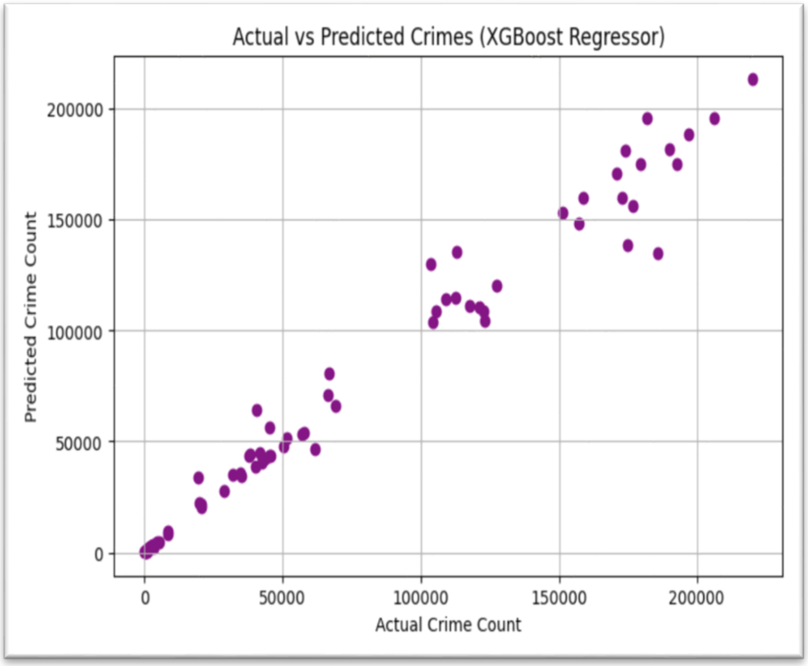

Correlation Heatmap of Crime Types
YEAR 1.00 -0.01 0.12 0.20 -0.08 0.09 -0.01 0.09 0.01 0.06 0.15 0.05 0.17 -0.01 0.09
MURDER -0.01 1.00 0.72 0.77 0.71 0.82 0.66 0.74 0.65 0.86 0.59 0.90 0.60 0.64 0.77
RAPE 0.12 0.72 1.00 0.63 0.48 0.68 0.71 0.71 0.53 0.58 0.54 0.65 0.69 0.67 0.79
KIDNAPPING & ABDUCTION 0.20 0.77 0.63 1.00 0.49 0.61 0.40 0.65 0.52 0.82 0.65 0.81 0.67 0.47 0.64
DACOITY -0.08 0.71 0.48 0.49 1.00 0.75 0.51 0.55 0.70 0.61 0.30 0.55 0.34 0.56 0.49
ROBBERY 0.09 0.82 0.68 0.61 0.75 1.00 0.79 0.82 0.72 0.76 0.54 0.69 0.50 0.67 0.78
BURGLARY -0.01 0.66 0.71 0.40 0.51 0.79 1.00 0.88 0.60 0.60 0.60 0.45 0.55 0.82 0.86
THEFT 0.09 0.74 0.71 0.65 0.55 0.82 0.88 1.00 0.61 0.76 0.73 0.58 0.72 0.73 0.88
RIOTS 0.01 0.65 0.53 0.52 0.70 0.72 0.60 0.61 1.00 0.58 0.49 0.52 0.52 0.66 0.72
CRIMINAL BREACH OF TRUST 0.06 0.86 0.58 0.82 0.61 0.76 0.60 0.76 0.58 1.00 0.63 0.84 0.65 0.54 0.68
CHEATING 0.15 0.59 0.54 0.65 0.30 0.54 0.60 0.73 0.49 0.63 1.00 0.54 0.78 0.56 0.75
DOWRY DEATHS 0.05 0.90 0.65 0.81 0.55 0.69 0.45 0.58 0.52 0.84 0.54 1.00 0.50 0.42 0.62
CRUELTY BY HUSBAND OR HIS RELATIVES 0.17 0.60 0.69 0.67 0.34 0.50 0.55 0.72 0.52 0.65 0.78 0.50 1.00 0.58 0.74
ARSON -0.01 0.64 0.67 0.47 0.56 0.67 0.82 0.73 0.66 0.54 0.56 0.42 0.58 1.00 0.80
TOTAL IPC CRIMES 0.09 0.77 0.79 0.64 0.49 0.78 0.86 0.88 0.72 0.68 0.75 0.62 0.74 0.80 1.00
YEAR
MURDER
RAPE
KIDNAPPING & ABDUCTION
DACOITY
ROBBERY
BURGLARY
THEFT
RIOTS
CRIMINAL BREACH OF TRUST
CHEATING
DOWRY DEATHS
CRUELTY BY HUSBAND OR HIS RELATIVES
ARSON
TOTAL IPC CRIMES
1.0
0.8
0.6
0.4
0.2
0.0

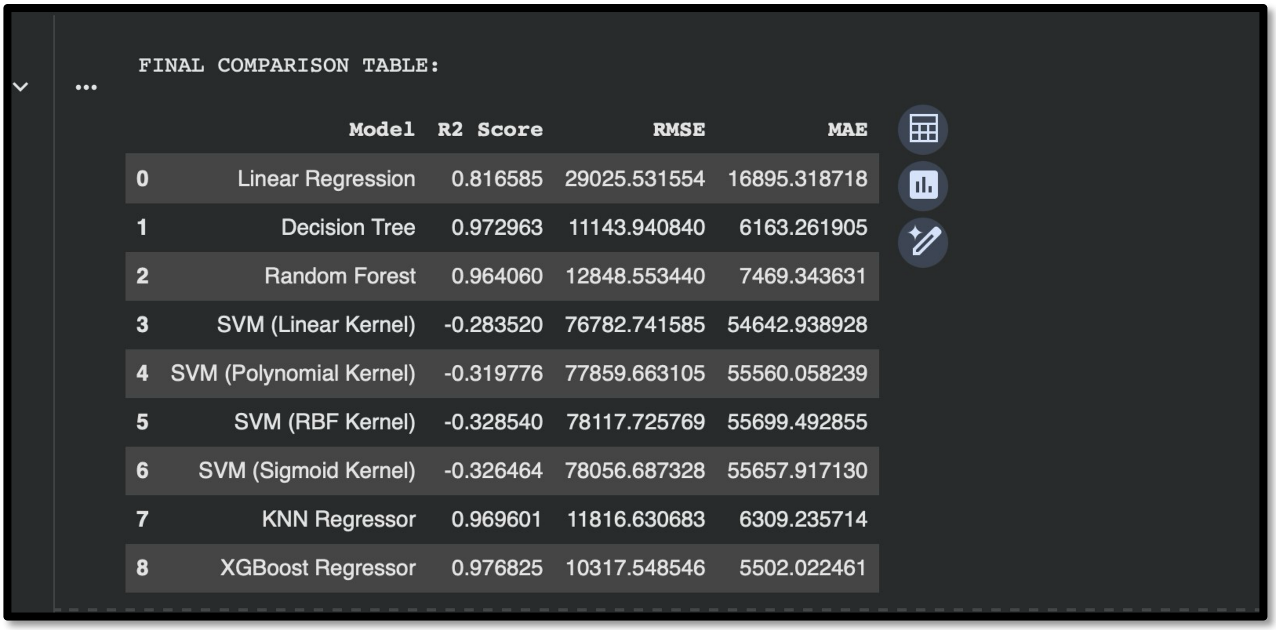
Figure 4: Best Model Graph



Result Analysis

The performance of all machine-learning models— including Linear Regression, Decision Tree, Random Forest, all SVM variants, KNN, and XGBoost—was evaluated using R^2 Score, RMSE, and MAE after PCA-based dimensionality reduction. The comparison table below summarizes their effectiveness:

1. Model Comparison Results



FINAL COMPARISON TABLE:

	Model	R2 Score	RMSE	MAE
0	Linear Regression	0.816585	29025.531554	16895.318718
1	Decision Tree	0.972963	11143.940840	6163.261905
2	Random Forest	0.964060	12848.553440	7469.343631
3	SVM (Linear Kernel)	-0.283520	76782.741585	54642.938928
4	SVM (Polynomial Kernel)	-0.319776	77859.663105	55560.058239
5	SVM (RBF Kernel)	-0.328540	78117.725769	55699.492855
6	SVM (Sigmoid Kernel)	-0.326464	78056.687328	55657.917130
7	KNN Regressor	0.969601	11816.630683	6309.235714
8	XGBoost Regressor	0.976825	10317.548546	5502.022461

2. Interpretation of Model Performance Best Performing Model

The XGBoost Regressor achieved the highest accuracy:

R^2 Score: 0.976824

RMSE: 10317.548546

MAE: 5502.022461

This indicates that the XGBoost model captured the underlying crime patterns most effectively, outperforming all other models in predictive accuracy and generalization.

High-Performing Models :

Decision Tree (R^2 = 0.972963)

KNN Regressor (R^2 = 0.969601)

Random Forest (R^2 = 0.964060)

These models also showed strong predictive ability with relatively low error scores. Decision Tree, in particular, performed closely to XGBoost.

Poor-Performing Models:

All SVM variants (Linear, Polynomial, RBF, Sigmoid) delivered negative R^2 scores, indicating extremely poor fit for this dataset. Their RMSE and MAE values were significantly higher than all other models, making SVM unsuitable for this type of crime data.

This aligns with the non-linear, multi-feature structure of IPC crime data, which tree-based and boosting models can capture more effectively.

3. PCA Performance

- The model pipeline reduced the original 9 features to 6 PCA components while retaining most of the variance.
- PCA improved training performance and minimized overfitting by eliminating redundant crime features.

4.Final Crime Prediction Result

- Using the best-performing model (XGBoost) and PCA-transformed input:

Predicted TOTAL IPC Crimes: 756

This demonstrates the practical applicability of the model for forecasting future crime levels based on feature inputs.

5. Key Insights

- Crime patterns show a clear rising trend across many districts.
- PCA confirmed strong multicollinearity among crime variables.
- Tree-based models (Decision Tree, Random Forest) and XGBoost significantly outperformed SVM and Linear Regression.
- XGBoost serves as the most reliable model for IPC crime prediction in this study.

Conclusion

The project depicted here embodies how machine learning can help us understand and predict crime patterns better. Careful cleaning of data, exploration of trends, and reduction of complexity by PCA brought to light meaningful relationships between different crimes across districts in India. After trying a number of models, the tree-based algorithms, specifically XGBoost and Decision Tree, were far better in representing real-world crime behavior than linear or SVM-based methods. The best overall performance was from XGBoost, with highly accurate predictions at an R^2 score of 0.9768.

The findings from this study show that data-driven tools can be very powerful in helping police departments, policy makers, and local authorities to make better-informed decisions. More accurate predictions can offer a chance for resources to be planned more effectively, while high-risk areas can be noted much earlier. In the future, this model will become much more influential by adding socio-economic factors, real-time data, and geographical crime mapping. On the whole, the project demonstrates how the integration of technology with crime analysis contributes to creating safer and better-informed communities.