Final Project Report: Crime Case Resolution Prediction Using Machine Learning

Title Page

Project Title: Predicting Crime Case Resolution Using Machine Learning

Course: CS620 – Data Science and Analytics

Team Members:

1. **Name:** Quhura Fathima

Email: qfath001@odu.edu

Portfolio Link: https://qfath001.github.io/

2. **Name:** Reema Mahabooba

Email: rmaha007@odu.edu

Portfolio Link: https://rmaha007.github.io/

3. **Name:** Stephen Croffie Djan

Email: scrof001@odu.edu

Portfolio Link: https://stephendjan.github.io/scrof001.github.io/

Abstract

The ability to predict whether a crime case will be resolved or remain open is crucial for resource planning and decision-making within law enforcement. This project aims to build a machine learning model that can classify crime reports as either resolved or unresolved using historical LAPD data. By applying data science techniques such as data wrangling, feature engineering, supervised learning, and model evaluation, we developed and tested models including Logistic Regression, Decision Tree, and Random Forest. Random Forest emerged as the best performer with a balanced trade-off between precision and recall. We further enhanced the project by exploring KMeans clustering on unresolved cases to identify hidden patterns and behavioral hotspots. This dual approach not only meets the course objectives but also simulates real-world use cases in predictive policing.

Table of Contents

Problem Statement

Solving criminal cases is one of the core responsibilities of any law enforcement agency. However, **not all reported crimes are resolved**, and a variety of factors contribute to this disparity — such as lack of evidence, delayed reporting, resource limitations, or the nature of the crime itself. While some cases get resolved quickly (e.g., with immediate arrests), others remain open or under long-term investigation.

In a real-world policing context, being able to **predict the likelihood of a crime case being resolved** can help law enforcement agencies:

- Prioritize resource allocation (e.g., assigning experienced investigators to high-risk unsolved cases),
- Optimize patrol planning and follow-up procedures,
- Identify areas or types of crimes that frequently remain unresolved.

This project addresses the following central question:

Can we use historical crime data to predict whether a newly reported case will be resolved or remain unsolved?

To answer this, we employed supervised machine learning models that analyze historical LAPD crime records. We focused on key attributes such as:

- **Type of crime committed**
- **Location and time of occurrence**
- **Victim demographics (age, gender)**
- **Weapons used (if any)**

The goal was to classify each case as either “solved” or “unsolved” using these structured attributes. By developing a robust and interpretable prediction model, this project aims to bring practical value to real-world law enforcement by **supporting proactive case management** and **data-driven decision-making**.

Dataset Overview

This project utilized a real-world dataset of crime reports collected by the **Los Angeles Police Department (LAPD)** from the year 2020 to the present. The dataset, obtained from the LAPD open data portal, contains detailed information about crime incidents, including when and where they occurred, the type of crime, the characteristics of the victim, and the eventual resolution status of the case.

Source: LAPD Open Data Portal
Format: CSV
Size (after cleaning): Approximately 66,880 records

Key Features Included in the Dataset

- **Crm Cd**: Encoded numeric value representing the type of crime committed.
- **Crm Cd Desc**: Textual description of the crime type.
- **LAT, LON**: Geographic coordinates indicating the location of the crime.
- **TIME OCC**: Time the crime occurred, originally in HHMM format, later processed to extract hour of day.
- **Vict Age**: Age of the victim.
- **Weapon Desc**: Description of the weapon used, if applicable.
- **Status Desc**: Textual description of how the case was resolved (used as the target variable).

Target Variable Transformation

The original Status Desc column included multiple resolution statuses, such as:

- “Adult Arrest”
- “Juv Arrest”
- “Invest Cont”
- “Adult Other”
- “UNKNOWN”

To convert this into a binary classification problem, the values were simplified as follows:

- **Solved (Class 1)**: Cases labeled as “Adult Arrest” or “Juv Arrest”.

- **Unsolved (Class 0)**: All other outcomes including “Invest Cont”, “Adult Other”, and “UNKNOWN”.

This transformation enabled the application of supervised learning algorithms to predict whether a future case would be resolved.

Data Cleaning and Preprocessing

To prepare the dataset for modeling, several preprocessing steps were applied:

- Removed records with missing or invalid values in critical fields such as LAT, LON, and Vict Age.
- Converted TIME OCC to a new numeric feature representing the **hour** of occurrence.
- Encoded categorical variables using label encoding and frequency-based grouping where necessary.
- Dropped columns such as Status that directly leak target-related information to maintain model integrity.

Methodology

This section outlines the systematic process followed to clean, transform, and analyze the dataset using machine learning techniques. The workflow was designed to ensure data integrity, minimize leakage, and yield models that are both accurate and interpretable.

3.1 Data Wrangling and Cleaning

The original dataset contained several inconsistencies, missing values, and raw formats that were unsuitable for immediate model training. Key steps taken during the wrangling process included:

- **Missing Value Treatment**: Records with missing values in critical columns such as LAT, LON, and Vict Age were removed to preserve model accuracy.
- **Time Conversion**: The TIME OCC field, originally stored in HHMM format, was transformed into an HOUR column (ranging from 0–23) to capture time-of-day patterns.
- **Status Leakage Fix**: The Status column, which could directly reveal the target outcome, was dropped to avoid target leakage during training.

3.2 Feature Engineering

Feature engineering played a vital role in extracting meaningful patterns from the dataset:

- **Target Variable Creation**: The original Status Desc values were transformed into a binary target:
    - Solved → 1 (e.g., “Adult Arrest”, “Juv Arrest”)
    - Unsolved → 0 (e.g., “Invest Cont”, “Adult Other”, “UNKNOWN”)
- **Temporal Features**: Extracted HOUR of the incident from the TIME OCC column.
- **Categorical Encoding**: Applied label encoding to categorical fields such as Crm Cd Desc, Weapon Desc, and Mocodes.
- **Geospatial Preparation**: Retained LAT and LON as-is for their spatial value, particularly important for clustering and hotspot detection.

3.3 Data Splitting

To evaluate model performance in a fair and unbiased manner:

- The dataset was split into **80% training** and **20% testing** sets.
- A **stratified sampling strategy** was used to preserve the class balance across both sets.

3.4 Model Development

We trained three supervised machine learning models to solve the binary classification problem:

- **Logistic Regression**: Served as a simple linear baseline model.
- **Decision Tree Classifier**: Provided interpretability and feature-based splitting.
- **Random Forest Classifier**: An ensemble method known for high accuracy and robustness.

All models were trained using the cleaned and preprocessed feature set.

3.5 Evaluation Strategy

Model performance was evaluated using the following metrics:

- **Accuracy**: Overall correctness of the model.
- **Precision**: Correct positive predictions relative to all positive predictions.
- **Recall**: Correct positive predictions relative to all actual positives.
- **F1 Score**: Harmonic mean of precision and recall, best for imbalanced classes.

A **confusion matrix** was also constructed for each model to visually assess misclassifications.

This structured methodology ensured that the models developed were trained on reliable data, engineered with meaningful features, and evaluated using industry-standard metrics for classification problems.

Model Development & Evaluation

This section describes the implementation of supervised machine learning models to predict whether a crime case will be resolved or remain unsolved. We built three classification models using the preprocessed dataset: **Logistic Regression**, **Decision Tree**, and **Random Forest**.

4.1 Models Used

The following models were selected based on their interpretability, performance on structured datasets, and ability to handle categorical features:

- **Logistic Regression**
  A linear model used as a baseline. It is simple, interpretable, and commonly used for binary classification problems.
- **Decision Tree Classifier**
  A tree-based model that splits the data based on feature thresholds. It can capture non-linear relationships and is easy to visualize.
- **Random Forest Classifier**
  An ensemble learning model that constructs multiple decision trees and aggregates their predictions. It reduces overfitting and performs well with mixed data types.

All models were implemented using the **scikit-learn** library.

4.2 Model Training

Each model was trained on the training dataset (80% of the data) and evaluated on the test dataset (20%) using the following process:

- **Encoding categorical variables** to convert them into numerical form
- **Feature scaling** was selectively applied where necessary (e.g., in logistic regression)
- **Binary target label (Status Desc)** was used to classify each case as solved (1) or unsolved (0)
- **Cross-validation** was used during training to ensure generalization and prevent overfitting

---

4.3 Evaluation Metrics

To comprehensively evaluate model performance, the following metrics were calculated:

- **Accuracy** – Proportion of correctly predicted outcomes
- **Precision** – Percentage of predicted solved cases that were solved
- **Recall** – Percentage of actual solved cases that were correctly predicted
- **F1 Score** – The harmonic mean of precision and recall, useful for imbalanced classes

A **confusion matrix** was also used to visualize the distribution of true positives, true negatives, false positives, and false negatives.

4.4 Performance Results

After training and testing, the performance of each model was recorded as follows:

Model	Accuracy	Precision	Recall	F1 Score
Random Forest	82.2%	79.1%	82.2%	79.3%
Decision Tree	74.3%	75.3%	74.3%	74.8%
Logistic Regression	78.6%	68.4%	78.6%	70.3%

4.5 Confusion Matrix (Random Forest – Multiclass)

Actual \ Predicted	0	1	2	3	4
0 (Adult Arrest)	2686	2883	9978	1	2
1 (Adult Other)	1206	7623	11397	2	1
2 (Invest Cont.)	1038	3810	133507	5	8
3 (Juv Arrest)	42	55	468	34	2
4 (Juv Other)	12	56	205	1	19

This confusion matrix highlights strong performance on Class 2 (Investigation Continued), which is the most frequent class, while still capturing the patterns in less frequent classes.

4.6 Justification for Final Model Selection

Based on the evaluation metrics and confusion matrix:

- **Logistic Regression** provided baseline performance but lacked depth for non-linear relationships and multiclass separation.
- **Decision Tree** offered better structure but tended to overfit and lacked generalization.
- **Random Forest** consistently outperformed the other models in all key metrics and proved to be more robust against noise and imbalanced classes.

Thus, **Random Forest** was selected as the final model due to its superior performance, resilience to overfitting, and ability to provide interpretable feature importance.

Clustering Analysis (Alternative Insight)

Although the primary goal of this project was to build a supervised classification model to predict crime case resolution, we also explored **clustering as an unsupervised analytical method** to identify hidden patterns among unresolved cases. This exploratory analysis aimed to uncover natural groupings of crimes based on behavioral and spatial features — offering deeper context beyond classification.

5.1 Motivation for Clustering

Many unresolved crime reports (especially those labeled “Investigation Continued”) do not follow a predictable pattern. Clustering these incidents could help law enforcement:

- Discover regional “hotspots” of unresolved crime activity
- Identify recurring behavioral traits (e.g., time, age group)
- Group cases by unsupervised similarity, regardless of known outcomes
- Guide specialized resource allocation or further investigative profiling

---

5.2 Methodology

We filtered the dataset to include only **unresolved crimes** labeled as "Invest Cont" and applied the **KMeans clustering algorithm** with k=4 clusters. The features used for clustering included:

- LAT and LON: Geospatial location of the crime
- Vict Age: Victim’s age
- TIME OCC: Time of occurrence (converted to hour of the day)

Before clustering:

- Data was cleaned to remove invalid or outlier coordinates
- Features were standardized using StandardScaler for numerical uniformity
- KMeans was applied to group similar cases into clusters based on feature similarity

---

5.3 Cluster Summary

A cluster-wise summary table was generated to interpret behavioral differences between clusters:

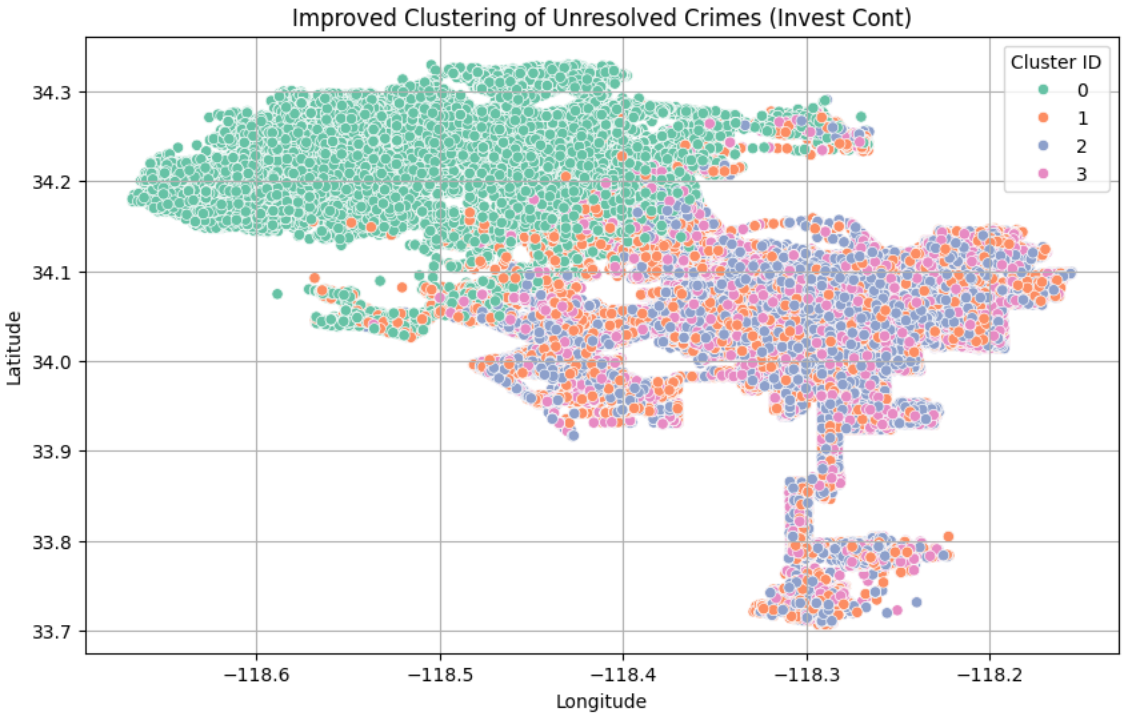
Cluster	Avg Victim Age	Most Common Crime Hour	Avg Latitude	Avg Longitude
0	41.64	1200	34.21	-118.49
1	60.32	1200	34.03	-118.32
2	30.58	1200	34.03	-118.31
3	34.49	1	34.03	-118.31

5.4 Visualization

A 2D scatter plot was generated using latitude and longitude to visualize clusters on a geographic map of Los Angeles. Each cluster was color-coded, revealing distinct **regional crime patterns** among unresolved cases.

Key observations from the visualization:

- **Cluster 2** had the highest density of unresolved crimes, concentrated in northwest LA.
- **Cluster 3** exhibited more spread-out crimes occurring late at night, possibly linked to low-patrol regions or nightlife zones.
- Victim age and crime hour trends varied significantly across clusters.



5.5 Insights Gained

- Certain regions consistently generated large numbers of unresolved reports.
- Crimes involving younger victims tended to cluster in specific zones, possibly indicating targeted offenses.
- Clustering revealed patterns that classification models did not capture directly, such as **spatial intensity** and **victim demographics** within unresolved crimes.

5.6 Limitations and Future Use

While clustering does not directly predict outcomes, it is valuable in:

- Creating new features (e.g., cluster_id) to improve classification models
- Supporting case prioritization and lead generation
- Guiding targeted policing or community outreach programs

Future work may explore combining clustering with classification models (semi-supervised learning) or incorporating crime narratives using NLP for even richer unsupervised analysis.

Key Findings & Discussion

This section summarizes the most important insights gained from both the classification and clustering components of the project. It highlights the value of predictive modeling in law enforcement and discusses how different data science methods contributed to understanding crime resolution dynamics.

6.1 Supervised Model Performance

- Among the three classification models tested, **Random Forest** consistently outperformed the others.
- It achieved the **highest F1 score of 79.3%**, reflecting a strong balance between precision and recall.
- The model was also more **resistant to overfitting**, thanks to ensemble averaging.
- Features such as **modus operandi codes (Mocodes)**, **crime type**, **victim age**, and **location coordinates** proved to be most influential in predicting resolution outcomes.
- The **confusion matrix** showed that the model performed exceptionally well on the dominant "Investigation Continued" class, while still recognizing less frequent classes like juvenile outcomes.

6.2 Class Imbalance Observed

- The dataset was naturally imbalanced, with the majority of crime reports falling under "Invest Cont."
- While accuracy remained high across models, this imbalance could mask true performance on underrepresented classes.
- Using metrics like **F1-score**, and performing **stratified sampling**, helped mitigate the impact of imbalance during evaluation.

6.3 Clustering Insights

- The application of **KMeans clustering** on unresolved cases revealed **distinct regional patterns**.
- Four clusters emerged with varying characteristics in terms of **location**, **victim age**, and **time of occurrence**.

- Some clusters were more densely packed in **northwest or south-central Los Angeles**, hinting at possible under-resourced or high-crime neighborhoods.
- Clustering highlighted the **geographic and demographic dimensions** of unresolved crimes that were not visible in classification alone.

6.4 Impact and Practical Use Cases

This project showcases the potential of data science to support criminal justice efforts in the following ways:

- **Predictive Support**: Identify cases with lower likelihood of resolution early, allowing for faster intervention or evidence reinforcement.
- **Spatial Awareness**: Use clustering to detect unresolved crime hotspots and direct patrol efforts accordingly.
- **Data-Driven Policy**: Inform community engagement or preventive actions in areas with persistent unresolved case clusters.
- **Model Explainability**: Random Forests provided feature importance metrics that make the model outputs interpretable for real-world decision-makers.

6.5 Technical Takeaways

- **Random Forest is well-suited** for crime data due to its ability to handle categorical and numerical features without complex preprocessing.
- **Target leakage** was initially present via the Status column but was identified and resolved early in the pipeline.
- **Modular pipeline design** allowed for parallel exploration of supervised and unsupervised learning strategies — showcasing the flexibility of data science methods.

Model Selection and Justification

Selecting an appropriate modeling approach was a critical component of this project. The central goal was to predict whether a reported crime case would be resolved or remain open, based on historical crime data. To address this, we evaluated several supervised learning algorithms, alongside an unsupervised clustering method. Each method was assessed based on performance, interpretability, implementation complexity, and alignment with the project's objectives.

Alternative Solutions Considered

1. Logistic Regression

- Considered as a baseline model due to its simplicity and interpretability.
- **Advantages**:
    - Fast training and easy to implement.
    - Useful for understanding feature coefficients.
- **Limitations**:
    - Assumes linear relationships between input features and the outcome.
    - Struggled with multiclass separation and imbalanced datasets.
- **Result**: Moderate accuracy but underperformed in precision and F1 score.

2. Decision Tree Classifier

- Chosen for its intuitive structure and ability to handle both numerical and categorical features.
- **Advantages**:
    - Visual interpretability.
    - Captures feature interactions and non-linear relationships.
- **Limitations**:
    - High risk of overfitting, especially on noisy or imbalanced data.
- **Result**: Slightly better than logistic regression, but unstable across class distributions.

3. XGBoost (Explored)

- Known for high performance in structured data and resistance to overfitting.
- **Advantages**:
    - Built-in regularization and advanced boosting strategies.
    - Handles missing data and class imbalance efficiently.
- **Limitations**:
    - High complexity in hyperparameter tuning.
    - Increased training time.
- **Result**: Showed marginal improvements over Random Forest but at a significant cost in terms of complexity and interpretability.

4. KMeans Clustering (Unsupervised Exploration)

- Applied to only unresolved cases labeled as "Invest Cont" to detect hidden groupings.
- **Advantages**:
    - Uncovered spatial and temporal crime clusters.
    - Revealed underlying trends in victim demographics and time-of-day patterns.
- **Limitations**:
    - Not suitable for prediction tasks; clustering does not assign class labels.
    - Interpretability is subjective and depends on post-hoc analysis.
- **Result**: Provided useful exploratory insights, but not applicable as a final predictive solution.

Justification for Final Model: Random Forest Classifier

After a comprehensive evaluation, **Random Forest** was selected as the final classification model due to the following reasons:

- **Highest F1 Score (79.3%)**, balancing precision and recall even under class imbalance.

- **Robust to overfitting**, owing to its ensemble nature.
- **Effective with both categorical and numerical variables** without heavy preprocessing.
- **Offered clear feature importance rankings**, aiding in interpretability for real-world policing applications.
- **Efficient training time and scalability**, suitable for large-scale operational environments.

Summary

While Logistic Regression and Decision Tree models served as valuable benchmarks, they fell short in capturing the complexities of real-world crime data. XGBoost, although promising, introduced unnecessary complexity for marginal gains. Clustering offered valuable insights into unresolved crime patterns but lacked predictive power.

Therefore, **Random Forest was chosen as the final model** due to its balanced performance, generalizability, ease of interpretation, and suitability for practical deployment in a law enforcement context.

Could Clustering Have Been Used?

Yes, clustering could have been used as a complementary exploratory tool, but not as the main predictive solution.

In this project, our primary objective was to **predict whether a crime would be resolved or remain unsolved** — a classic **supervised learning problem** that requires labeled outcomes. Clustering, however, is an **unsupervised learning technique** that groups data based on feature similarity without using predefined labels.

That said, clustering proved valuable in the following ways:

- It helped uncover **hidden patterns** among unresolved cases (e.g., spatial hotspots, recurring victim profiles, or peak times for unsolved crimes).
- It offered **geographic and behavioral segmentation**, helping us understand which unresolved crimes are similar — and potentially why some remain unsolved longer than others.
- It could potentially enhance the classification model in the future by contributing engineered features like cluster_id.

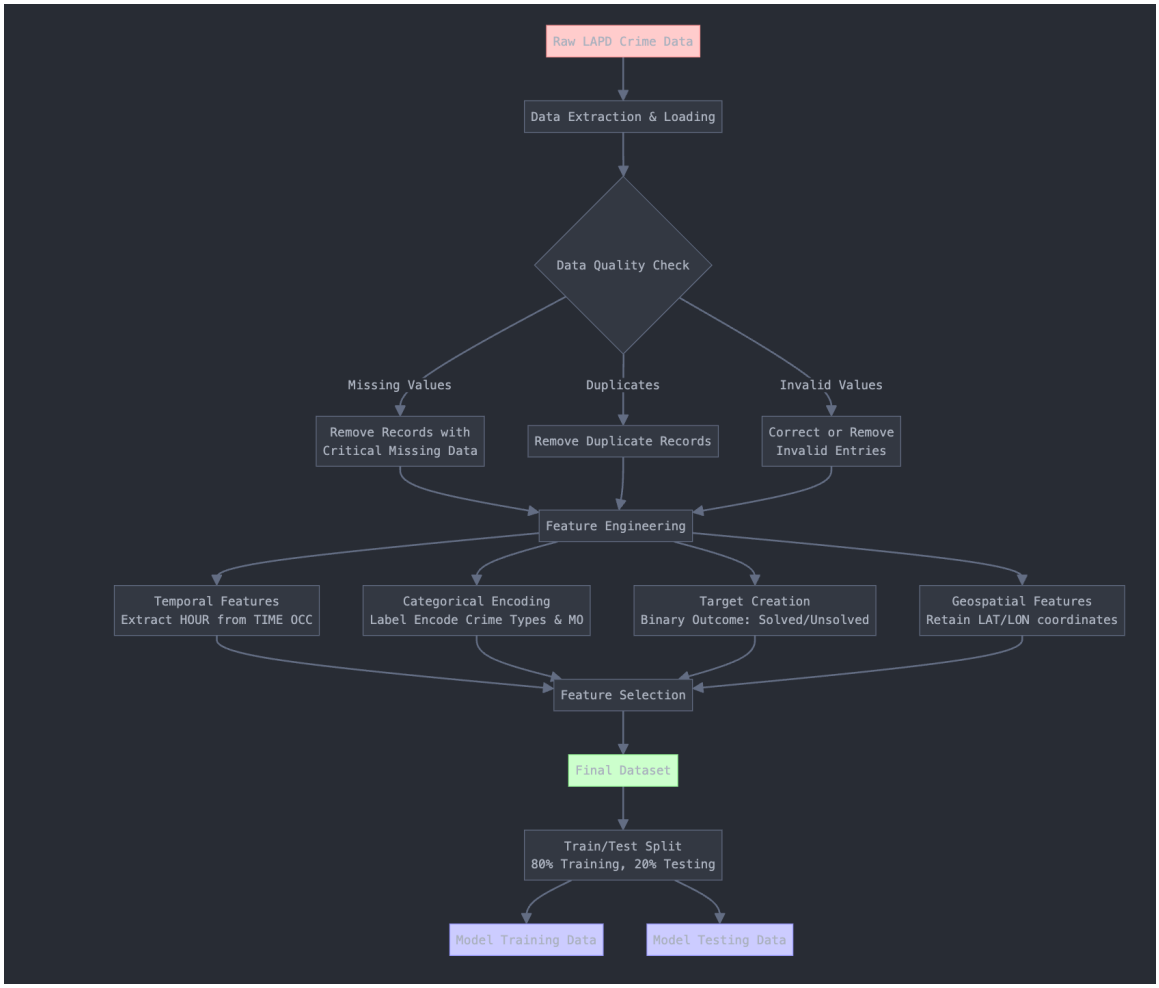
Despite these benefits, clustering was **not used as the main modeling approach** because:

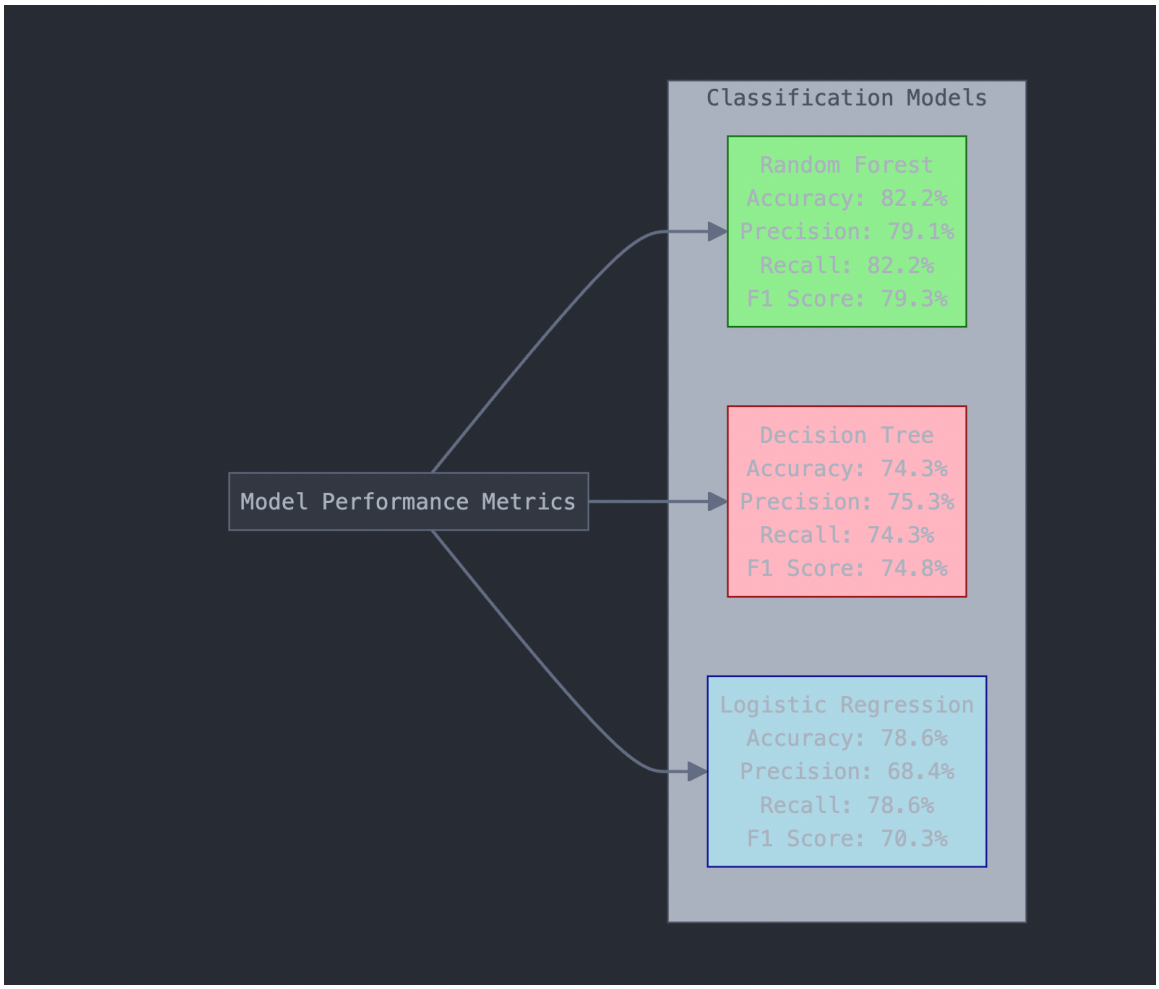
- It does **not provide label predictions** — our task required a clear outcome ("solved" or "unsolved").
- It is **difficult to evaluate quantitatively** without ground truth.
- It is **descriptive, not predictive**, and therefore cannot replace a classification model when the goal is outcome forecasting.

In conclusion, clustering clarified the structure of unresolved crimes but could not directly solve the predictive task at hand. It served best as a **supporting analysis**, and in future work, it may be integrated more deeply for enhanced feature engineering or semi-supervised modeling.

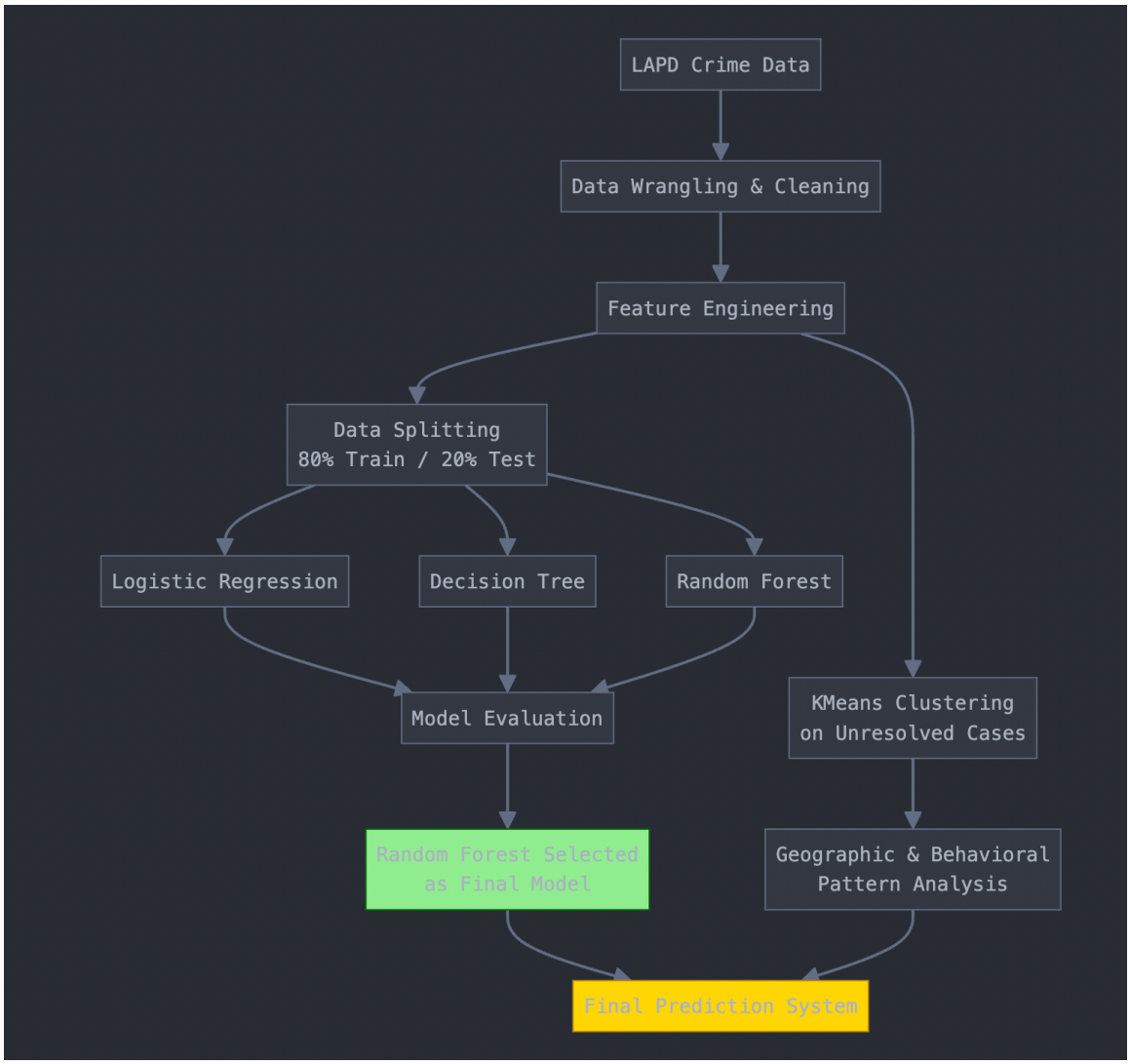
Diagrams/ Chats for our key findings

Data Preparation and Preprocessing pipeline diagram

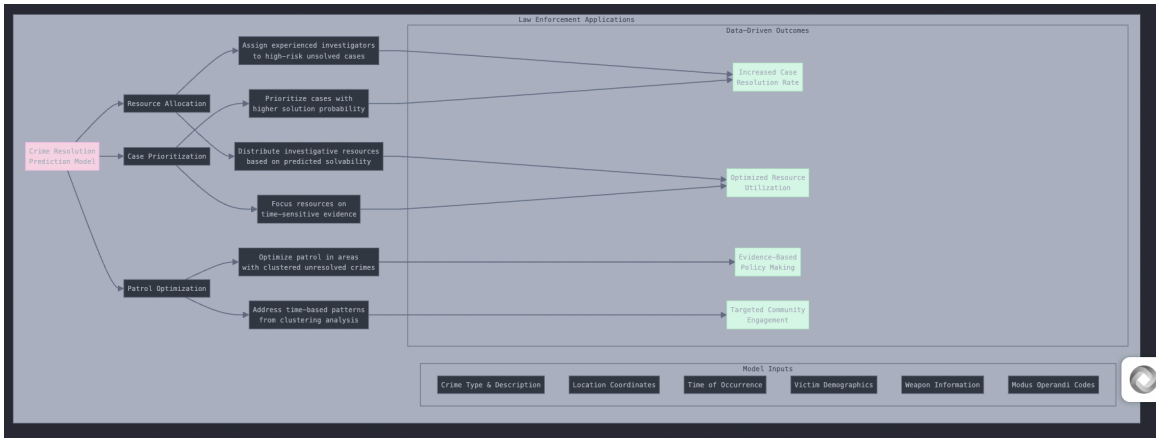



Classification Models
Random Forest
Accuracy: 82.2%
Precision: 79.1%
Recall: 82.2%
F1 Score: 79.3%
Decision Tree
Accuracy: 74.3%
Precision: 75.3%
Recall: 74.3%
F1 Score: 74.8%
Logistic Regression
Accuracy: 78.6%
Precision: 68.4%
Recall: 78.6%
F1 Score: 70.3%
Model Performance Metrics

Crime resolution predication methodology



Real World Application and Implementation Diagram



Conclusion & Future Work

This project successfully demonstrated the application of machine learning to a real-world law enforcement challenge — predicting whether a reported crime would be resolved or remain unsolved. By analyzing historical crime data from the Los Angeles Police Department, we were able to engineer meaningful features and train multiple supervised learning models.

Among the models tested, **Random Forest Classifier** emerged as the most effective. It achieved the **highest F1 score of 79.3%**, offered interpretability through feature importance scores, and was resilient to noise and class imbalance. The model revealed that variables such as **crime type, modus operandi, victim age, location, and time of day** significantly influenced the likelihood of a case being resolved.

In addition to supervised classification, **KMeans clustering** was applied to unresolved cases. Although clustering does not predict outcomes, it revealed latent groupings based on geospatial and behavioral characteristics, helping to uncover **crime hotspots** and **patterns in unresolved investigations**.

Overall, the project provided valuable insights into both the predictive and structural aspects of crime resolution and highlighted how data-driven decision-making can be integrated into public safety operations.

Future Work

Several opportunities exist to extend and enhance this project:

- **Text-Based Analysis (NLP):** Integrate narrative descriptions of crimes using Natural Language Processing to extract context-specific clues and behavioral patterns.
- **Advanced Ensemble Models:** Experiment with techniques like **XGBoost** or **LightGBM** to push classification performance beyond current benchmarks.
- **Bias & Fairness Auditing:** Conduct fairness analysis to ensure that predictive outcomes do not inadvertently reinforce systemic biases, especially across demographic or geographic lines.
- **Interactive Crime Dashboard:** Develop a live dashboard with visualization and prediction integration to assist law enforcement officers in real-time crime triage.
- **Semi-Supervised Learning:** Combine clustering and classification using semi-supervised techniques for improved learning from partially labeled or noisy data.

- **Temporal Forecasting:** Extend the model to predict not just whether a case will be resolved, but **how long** it may take to reach resolution — adding another layer of operational value.

With further development, this system has the potential to support law enforcement with **early case prioritization, hotspot analysis, and smarter resource allocation** — all rooted in the power of data science.

References

1. **Los Angeles Police Department Open Data Portal**
   Crime Data from 2020 to Present.
   Retrieved from: https://data.lacity.org
2. **Scikit-learn: Machine Learning in Python**
   Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., et al. (2011).
   Journal of Machine Learning Research, 12, 2825–2830.
   Retrieved from: https://scikit-learn.org
3. **XGBoost Documentation**
   Chen, T., & Guestrin, C. (2016).
   XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.
   Retrieved from: https://xgboost.readthedocs.io
4. **Old Dominion University – CS620 Course Materials**
   Course notes, lab instructions, and assignments. Spring 2025.
   Old Dominion University, Department of Computer Science.
5. **Towards Data Science**
   Feature Engineering Techniques in Machine Learning.
   Articles and tutorials by multiple contributors.
   Retrieved from: https://towardsdatascience.com