# Analysis of Crime Patterns in Toronto: A Data Mining Approach to Public Safety

Navdisha Bhakri
Seneca Polytechnic
Toronto, Canada
nbhakri@myseneca.ca

Dhruv Chotalia
Seneca Polytechnic
Toronto, Canada
dchotalia@myseneca.ca

Vinh Minh Dang
Seneca Polytechnic
Toronto, Canada
vmdang@myseneca.ca

***Abstract*— Public safety is the bedrock of a thriving city, yet traditional policing methods are often reactive rather than predictive, frequently relying on intuition or broad geographical patrolling which leads to resource inefficiencies. This project utilizes historical crime data from the Toronto Police Service (2014–2025) to uncover hidden spatial and temporal patterns in criminal activity, aiming to transition operational planning from a retrospective stance to a proactive, evidence-based model. Adopting the CRISP-DM framework, we applied three distinct data mining techniques: DBSCAN is deployed to distinguish systematic spatial clustering from stochastic background noise; Apriori is utilized for association rule mining to discover conditional "crime recipes"; and Decision Tree classification is trained for interpretable crime prediction. Our analysis reveals that crime is not random but follows distinct environmental and temporal trends driven by the city's physical infrastructure. Key findings include the identification of stable "micro-hotspots" for auto theft in suburban commuter belts versus assault clusters in the downtown core. Furthermore, the Decision Tree model achieved a Recall of 0.88 for Assaults, demonstrating high efficacy in predicting violent crime based on premises type. Consequently, this report recommends a shift from broad patrolling to targeted resource allocation based on these identified micro-clusters and specific environmental risk factors, effectively transforming raw archival data into actionable public safety intelligence.**

***Keywords*— *Crime analysis, DBSCAN, Decision Tree, Apriori, predictive policing, spatial clustering, public safety.***

## I. Introduction

Crime in large metropolitan areas like Toronto is rarely a stochastic occurrence; rather, it follows distinct hidden patterns—spatial, temporal, and behavioral—that are often invisible to traditional observation. Detecting these trends is not merely an academic exercise; it is a critical necessity for preventing incidents, optimizing policing strategies, and informing data-driven city planning.

### A. Problem Definition and Motivation

The central problem addressed in this project is the transition from **reactive** to **predictive** public safety. Law enforcement agencies face significant resource constraints; police cannot be everywhere at once. Consequently, the ability to predict high-risk environments allows for the strategic deployment of limited resources. Furthermore, safer neighborhoods require urban planning based on hard data rather than intuition. Unlike abstract textbook problems, this domain involves real data with real stakes, where the insights generated directly affect community well-being and public safety.

### B. Related Work

Prior research in crime analysis has primarily focused on descriptive statistics—simply counting and mapping where crimes occurred in the past. While some recent projects have attempted to use basic machine learning to predict crime types, they often lack depth in feature engineering or fail to account for the "noise" inherent in urban environments. Earlier work has largely ignored the complex interaction between environmental context (e.g., premises type) and temporal cycles, relying instead on simple frequency heatmaps. This project aims to bridge that gap by employing advanced data mining techniques that go beyond simple statistics.

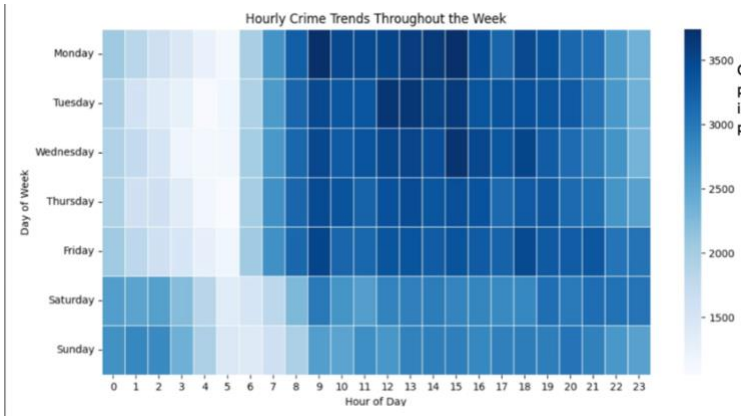


Fig. 1 : Temporal heatmap illustrating the frequency of reported criminal incidents aggregated by hour of the day and

day of the week, highlighting distinct peak activity windows during evening hours

### C. Methodology and Contributions

To uncover the underlying structure of Toronto's crime landscape, we leveraged historical Major Crime Indicator (MCI) data from the Toronto Police Service (2014–2025). Adopting the CRISP-DM framework, we applied a multi-layered analytical approach.

Our work makes four distinct contributions to the field:

1. **Comprehensive Preprocessing Pipeline:** We established a robust framework for cleaning and imputing missing data, ensuring that statistical distributions were preserved while removing invalid "noise" and preventing data leakage.
2. **Novel Feature Engineering:** We developed domain-specific features not present in the original dataset, including **Geo_Region** for spatial segmentation and **Behavioral Time-Bins** to capture daily rhythms.
3. **Methodological Triangulation:** We employed three complementary algorithms: **DBSCAN** to identify spatial hotspots, **Apriori** to discover association rules between location and crime type, and **Decision Trees** for predictive classification.
4. **Rigorous Evaluation:** We implemented stability testing (5-Fold Cross-Validation) to ensure our models provide reliable, generalizable insights rather than overfitting to specific data points.

## II. DATASET

### A. Data Source and Description

*The dataset utilized for this research is the **Major Crime Indicators (MCI)** dataset, publicly available through the Toronto Police Service (TPS) Public Safety Data Portal. This open-data initiative provides granular details on reported crimes across the City of Toronto.*

### B. Data Volume and Scope

*The original dataset spans reports from 2014 to the present. The dataset consists of **31 attributes** (columns) per record. **Total Records:** 450,000. The variables are categorized into three distinct types:*

1. **Temporal Features:** OccurrenceDate, OccurrenceYear, OccurrenceHour, and DayOfWeek. These allow for the analysis of cyclical trends (e.g., crimes peaking on Friday nights).

2. **Spatial Features:** Latitude (WGS84), Longitude (WGS84), Division, and Neighbourhood. These provide the geospatial context necessary for clustering algorithms like DBSCAN.

3. **Categorical Features:** MCI_Category (the target variable), Premises_Type (e.g., House, Commercial, Outside), and Offence (the specific legal charge).

Identify applicable funding agency here. If none, delete this text box.

### C. Issues Found

*a) Missing values ("NSA")*
*b) Invalid GPS coordinates*
*c) Post-incident police codes (risk of leakage)*
*d) Non-standard date format*
*e) Outliers & incorrectly reported years (pre-2013)*

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 452949 entries, 0 to 452948
Data columns (total 31 columns):
 #   Column             Non-Null Count   Dtype
---  ------             --------------   -----
 0   OBJECTID           452949 non-null  int64
 1   EVENT_UNIQUE_ID    452949 non-null  object
 2   REPORT_DATE        452949 non-null  object
 3   OCC_DATE           452949 non-null  object
 4   REPORT_YEAR        452949 non-null  int64
 5   REPORT_MONTH       452949 non-null  object
 6   REPORT_DAY         452949 non-null  int64
 7   REPORT_DOY         452949 non-null  int64
 8   REPORT_DOW         452949 non-null  object
 9   REPORT_HOUR        452949 non-null  int64
 10  OCC_YEAR           452798 non-null  float64
 11  OCC_MONTH          452798 non-null  object
 12  OCC_DAY            452798 non-null  float64
 13  OCC_DOY            452798 non-null  float64
 14  OCC_DOW            452798 non-null  object
 15  OCC_HOUR           452949 non-null  int64
 16  DIVISION           452949 non-null  object
 17  LOCATION_TYPE      452949 non-null  object
 18  PREMISES_TYPE      452949 non-null  object
 19  UCR_CODE           452949 non-null  int64
 20  UCR_EXT            452949 non-null  int64
 21  OFFENCE            452949 non-null  object
 22  MCI_CATEGORY       452949 non-null  object
 23  HOOD_158           452949 non-null  object
 24  NEIGHBOURHOOD_158  452949 non-null  object
 25  HOOD_140           452949 non-null  object
 26  NEIGHBOURHOOD_140  452949 non-null  object
 27  LONG_WGS84         452949 non-null  float64
 28  LAT_WGS84          452949 non-null  float64
 29  x                  452949 non-null  float64
 30  y                  452949 non-null  float64
dtypes: float64(7), int64(8), object(16)
```

*The dataset information summary (df.info()) displaying the data types (float64, object, int64) and non-null counts prior to cleaning.*

| | OBJECTID | REPORT_YEAR | REPORT_DAY | REPORT_DOY | REPORT_HOUR | OCC_YEAR | OCC_DAY | OCC_DOY | OCC_HOUR | UCR_CODE | UCR_EXT | LONG_WGS84 | LAT_WGS84 | x | y |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| count | 452949.000000 | 452949.000000 | 452949.000000 | 452949.000000 | 452949.000000 | 452798.000000 | 452798.000000 | 452798.00000 | 452949.000000 | 452949.000000 | 452949.000000 | 452949.000000 | 452949.000000 | 4.529490e+05 | 4.529490e+05 |
| mean | 226475.000000 | 2019.790588 | 15.755321 | 183.212693 | 12.711206 | 2019.722243 | 15.435190 | 182.61512 | 12.556760 | 1709.238210 | 146.908067 | -78.227370 | 43.062103 | -8.708231e+06 | 5.340230e+06 |
| std | 130755.257877 | 3.397504 | 8.779119 | 102.957332 | 6.456066 | 3.429958 | 8.949285 | 103.48370 | 7.283552 | 329.732642 | 52.275629 | 9.571369 | 5.268745 | 1.065480e+06 | 6.534068e+05 |
| min | 1.000000 | 2014.000000 | 1.000000 | 1.000000 | 0.000000 | 2000.000000 | 1.000000 | 1.00000 | 0.000000 | 1410.000000 | 100.000000 | -79.639247 | 0.000000 | -8.865400e+06 | 5.664924e-09 |
| 25% | 113238.000000 | 2017.000000 | 8.000000 | 96.000000 | 8.000000 | 2017.000000 | 8.000000 | 95.00000 | 7.000000 | 1430.000000 | 100.000000 | -79.475052 | 43.659420 | -8.847122e+06 | 5.412887e+06 |
| 50% | 226475.000000 | 2020.000000 | 16.000000 | 184.000000 | 13.000000 | 2020.000000 | 15.000000 | 183.00000 | 14.000000 | 1450.000000 | 100.000000 | -79.394133 | 43.699427 | -8.838114e+06 | 5.419045e+06 |
| 75% | 339712.000000 | 2023.000000 | 23.000000 | 270.000000 | 18.000000 | 2023.000000 | 23.000000 | 270.00000 | 19.000000 | 2120.000000 | 200.000000 | -79.321995 | 43.750852 | -8.830084e+06 | 5.426966e+06 |
| max | 452949.000000 | 2025.000000 | 31.000000 | 366.000000 | 23.000000 | 2025.000000 | 31.000000 | 366.00000 | 23.000000 | 2135.000000 | 230.000000 | 0.000000 | 43.853164 | 6.327780e-09 | 5.442747e+06 |

Statistical summary of numerical attributes.

### *C. Preprocessing and Cleaning*

Raw police data is inherently noisy. We implemented a strict cleaning funnel in Python:

1. **Temporal Filtering:** We filtered the dataset to include records from 2014 onwards. Analysis of earlier data revealed inconsistent reporting standards that do not reflect the current urban topology of Toronto.
2. **Imputation:** Missing values in temporal columns were imputed using the Occurrence Date. For numerical gaps, median imputation was used to preserve statistical distribution without introducing outliers.
3. **Noise Removal:** Rows containing "NSA" (Not Specified) values were removed. Furthermore, we filtered out invalid geographic coordinates (e.g., Lat/Long near 0.0) using a threshold of 1e-3 to ensure spatial accuracy.

### *D. Feature Engineering*

To enable machine learning algorithms to process the data effectively, we engineered several features:

- **Geo_Region:** We mathematically divided the city into five regions (North, South, East, West, Central) based on coordinate quantiles 25$^{th}$ and 75$^{th}$ percentiles).
- **Time_of_Day:** Continuous hour variables were binned into behavioral categories: Morning (6–12), Afternoon (12–17), Evening (17–21), and Night (21–6).
- **Season:** Monthly data was mapped to meteorological seasons to capture seasonal variance (e.g., Summer spikes in outdoor crime).

## III. GROUND TRUTH

In the context of this study, **Ground Truth (GT)** refers to the verified classification of a criminal incident as determined by the Toronto Police Service. Unlike unsupervised tasks where the "truth" is unknown, our Supervised Learning component (Decision Tree) relies on explicit historical labels to learn patterns.

**A. Definition of the Target Variable** The Ground Truth for our predictive models is the **MCI_CATEGORY** variable. This column represents the final, confirmed category of the crime (e.g., *Assault, Auto Theft, Break and Enter, Robbery*). The model's objective is to predict this label based solely on environmental features (Time, Location, Premises).

**B. Data Integrity and Leakage** A critical aspect of defining our Ground Truth was ensuring **validity**. We specifically excluded administrative variables such as UCR_EXT (Uniform Crime Reporting codes). These codes are often assigned *after* a police investigation is complete. Including them would constitute **Data Leakage**—essentially giving the model the answer key. By removing them, we ensure that our "Ground Truth" (the crime type) is being predicted strictly from the context available at the time of the incident.

| | OCC_YEAR | PREMISES_TYPE | MCI_CATEGORY | Count |
|---|---|---|---|---|
| 0 | 2013 | Apartment | Assault | 200 |
| 1 | 2013 | Apartment | Auto Theft | 1 |
| 2 | 2013 | Apartment | Break and Enter | 27 |
| 3 | 2013 | Apartment | Robbery | 6 |
| 4 | 2013 | Apartment | Theft Over | 18 |
| 5 | 2013 | Commercial | Assault | 21 |
| 6 | 2013 | Commercial | Auto Theft | 9 |
| 7 | 2013 | Commercial | Break and Enter | 20 |
| 8 | 2013 | Commercial | Robbery | 18 |
| 9 | 2013 | Commercial | Theft Over | 28 |

*Aggregated summary of historical crime records stratified by year, premises type, and target variable- MCI_CATEGORY*

**C. Sample of Ground Truth** The figure below illustrates a sample of the training data. The columns on the left (Occurrence Year, Premises Type) serve as the *Features*, while the column on the right (MCI_CATEGORY) serves as the *Ground Truth*.

## IV. TRAINING VS. TEST

To validate the performance of our predictive models and ensure our results were not artifacts of a specific data subset, we employed a rigorous experimental methodology involving both a Hold-Out Strategy and K-Fold Cross-Validation.

### *A. Data Partitioning (Hold-Out Method)*

For the Supervised Learning component (Decision Tree Classifier), the dataset was partitioned into two distinct subsets using the train_test_split method from the Scikit-Learn library:

1. **Training Set (80%):** Comprising approximately 240,000 records, this subset was used to train the model and allow it to learn the relationships between environmental features (Premises, Time) and the target variable (MCI Category).
2. **Test Set (20%):** Comprising approximately 60,000 records, this subset was strictly withheld from the training process. It served as a proxy for "future data" to evaluate the model's unbiased performance.

**Stratified Sampling:** Given the class imbalance in the dataset (where Assaults significantly outnumber Robberies), we utilized Stratified Sampling (stratify=y).This ensures that the proportion of each crime type in the Test Set exactly mirrors the proportion in the Training Set, preventing the model from being biased toward the majority class during evaluation.

**K-Fold Cross-Validation (Stability Analysis)**
Relying on a single 80/20 split can sometimes lead to misleading results if the Test Set happens to be "easy" or "hard" by random chance. To address this, we implemented 5-Fold Stratified Cross-Validation.

- **Methodology:** The entire dataset was randomly divided into k=5 equal-sized folds.
- **Process:** The model was trained and evaluated 5 separate times. In each iteration, 4 folds were used for training and the remaining 1 fold was used for testing.
- **Result:** We calculated the mean accuracy and standard deviation across all 5 folds. The low variance observed (<0.4) confirmed that our model is statistically stable and generalizes well to new data.

**C. Sampling for Unsupervised Learning**

For the DBSCAN clustering algorithm, computational constraints (RAM limits) required a different approach. We utilized Random Sampling to select 20% of the dataset (frac=0.2). This sample size was statistically significant enough to reveal spatial densities while allowing the complex distance matrix calculations to complete efficiently.

```
# ======================================================
# PART A: K-FOLD CROSS-VALIDATION (The Stability Check)
# ======================================================
print("\n" + "="*40)
print("STARTING 5-FOLD CROSS-VALIDATION")
print("="*40)

k = 5
skf = StratifiedKFold(n_splits=k, shuffle=True, random_state=42)


# ======================================================
# PART B: FINAL MODEL & VISUALIZATION (80/20 Split)
# ======================================================
print("Training Final Model for Visualization (80/20 Split)...")

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

cart_model = DecisionTreeClassifier(
    criterion='gini',
    max_depth=10,
    min_samples_leaf=50,
    random_state=42
)
cart_model.fit(X_train, y_train)

# Predictions
y_pred = cart_model.predict(X_test)
```

*Code snippet demonstrating the Stratified Train-Test split (80/20) and the setup for K-Fold Cross-Validation.*

## V. METHOD

We adopted a multi-layered data mining approach, utilizing three distinct algorithms to address the spatial, association, and predictive dimensions of crime analysis. All implementation was performed in Python using scikit-learn, pandas, and mlxtend.

### A. Spatial Clustering: DBSCAN

To identify high-density crime hotspots, we selected **DBSCAN**. Unlike K-Means, which assumes spherical clusters and forces every point into a group, DBSCAN can identify arbitrary shapes (like crime following a street grid) and, crucially, separate "Noise" (random incidents) from "Signal" (dense clusters).

**1. Parameter Tuning and Optimization (The Shift from 0.25 to 0.04):**

A critical phase of our experiment was Hyperparameter Optimization. We tested multiple sensitivity levels to determine the optimal "Resolution" for our clusters.

- **Initial Iteration (eps=0.25):** We began with a wider Epsilon radius of 0.25 (standardized units). While this successfully grouped the data, we observed that it created **"Macro-Clusters."** It merged distinct neighborhoods into single, massive blobs. This "low-resolution" view covered entire districts, rendering the insight too broad for specific police resource allocation.
- **Final Optimization (eps=0.04):** To increase granularity, we drastically reduced Epsilon to 0.04 while increasing the density requirement (min_samples=100).
- **Result:** This shift transformed the output from "Regional Analysis" to **"Micro-Hotspot Analysis."** The tighter radius successfully separated specific intersections and city blocks. It also increased the "Noise Ratio" to ~50%, effectively filtering out random, one-off crimes so the model focused exclusively on systematic, recurring problem areas.

**Implementation Code:**

```
# ----------------------------------------
# 1. Sample data
# ----------------------------------------
# Randomly sample 20% of the original crime dataset for faster processing
df_db_sampled = df_crime.sample(frac=0.2, random_state=42).copy()
# Drop rows with missing values
df_db = df_db_sampled[['LONG_WGS84', 'LAT_WGS84', 'MCI_CATEGORY']].dropna()

# ----------------------------------------
# 2. Scale coords
# ----------------------------------------
# Create separate scalers for longitude and latitude
# Fit the scalers and transform the longitude and latitude values
scaler_lon = StandardScaler()
scaler_lat = StandardScaler()
df_db['lon_scaled'] = scaler_lon.fit_transform(df_db[['LONG_WGS84']])
df_db['lat_scaled'] = scaler_lat.fit_transform(df_db[['LAT_WGS84']])
coords_scaled = df_db[['lon_scaled', 'lat_scaled']].values


# ----------------------------------------
# 3. DBSCAN (TUNED PARAMETERS)
# ----------------------------------------
# eps=0.04: Very small radius to break the city into neighborhoods
# min_samples=100: Requires high density to form a cluster
EPS_VAL = 0.04
MIN_SAMPLES_VAL = 100

db = DBSCAN(eps=EPS_VAL, min_samples=MIN_SAMPLES_VAL)
df_db['cluster'] = db.fit_predict(coords_scaled)
```

This code snippet shows the main steps of our DBSCAN analysis: sampling the data, scaling longitude and latitude coordinates, and performing clustering to identify high-density crime hotspots.

### B. *Association Rule Mining: Apriori Algorithm*

To discover hidden relationships between the environment (Time, Location) and the type of crime, we employed the **Apriori Algorithm**. This technique, typically used for market basket analysis, was adapted here to generate "Crime Recipes" in the format *If {Environment X}, Then {Crime Y}*.

**1. Experimental Setup:**

- **One-Hot Encoding:** We converted categorical variables (Time of Day, Premises Type) into a binary matrix.
- **Metric Selection:** We utilized two key metrics to filter our rules:
  - **Support (Frequency):** Set to **0.01 (1%)**. This ensures we only analyze patterns that occur frequently enough to matter to public safety.
  - **Lift (Strength):** Set to **> 1.1**. A lift of 1.0 implies random chance. By filtering for >1.1, we ensure that the environmental condition *actively increases* the probability of the crime occurring.

**2.Threshold Optimization (Finding the Signal):**

Finding the right support threshold was the primary experimental challenge. We initially tested a standard support of **0.05 (5%)**, but this yielded zero rules because specific crime patterns are rare relative to total dataset volume. We iteratively lowered the threshold to **0.01 (1%)**, which successfully revealed hidden patterns. We utilized **Lift (>1.1)** as our primary metric to filter out associations that were merely coincidental, focusing only on predictive relationships.

```
# ONE-HOT ENCODING
basket = pd.get_dummies(df_mining).astype(bool)
print(f"Data ready. Transformed into {basket.shape[1]} binary features.")

# ----------------------------------------
# 3. RUN APRIORI (THE MATH)
# ----------------------------------------
# min_support = 0.01 (1%) finds patterns that happen frequently enough to matter
support_threshold = 0.01

print(f"Running Apriori (Min Support: {support_threshold})...")
frequent_itemsets = apriori(basket, min_support=support_threshold, use_colnames=True)

if frequent_itemsets.empty:
    print("No patterns found! Try lowering support.")
else:
    # Generate Rules
    rules = association_rules(frequent_itemsets, metric="lift", min_threshold=1.1)
    rules = rules.sort_values(['lift', 'confidence'], ascending=[False, False])

    print(f"Generated {len(rules)} rules for ALL crimes.")
```

This code snippet demonstrates the transformation of categorical data via one-hot encoding and the extraction of rules using the Apriori algorithm, filtered by a 1% support threshold and a minimum lift of 1.1.

### *C. Predictive Classification: Decision Tree (CART)*

To answer the question *"Can we predict the crime type before it happens?"*, we trained a **Decision Tree Classifier (CART)**. We selected this "White Box" model over "Black Box" algorithms (like Neural Networks) because interpretability is crucial for public policy; we need to explain *why* a prediction was made.

**1. Model Configuration and Overfitting Mitigation:**

- **Depth Limit (max_depth=10):** We constrained the tree depth to 10 levels. Without this limit, the tree would grow indefinitely, memorizing the training data (Overfitting) rather than learning general rules.
- **Leaf Size (min_samples_leaf=50):** We enforced that every final decision "leaf" must contain at least 50 recorded crimes. This ensures that no prediction is based on a statistical fluke or a single outlier event.

**2. Parameter Tuning (Handling Overfitting):** We initially ran the decision tree without restrictions, which resulted in a massive, overfitted tree that memorized the training data but failed on test data. To improve the methodology, we introduced a **Pre-Pruning** experiment. We set max_depth=10 to limit complexity and min_samples_leaf=50 to ensure that every decision node was based on a statistically significant number of records. This shifted the model from "Memorization" to "Generalization."

```
# Encode target
le_target = LabelEncoder()
df_cart[target] = le_target.fit_transform(df_cart[target])

# --------------------------------
# 3. Train-test split 80:20
# --------------------------------
# Separate features and target variable
X = df_cart[features]
y = df_cart[target]

print(f"Training on {len(X)} records...")

# Split the dataset into 80% training and 20% testing
# Stratify ensures the class distribution remains the same in both sets
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42, stratify=y
)

# --------------------------------
# 4. CART Decision Tree
# --------------------------------
# Initialize a Decision Tree classifier using Gini impurity
# max_depth limits the depth of the tree to prevent overfitting
# min_samples_leaf ensures each leaf has at least 50 samples
# random_state ensures reproducibility
cart_model = DecisionTreeClassifier(
    criterion='gini',
    max_depth=10,            # Kept at 10 for detail
    min_samples_leaf=50,
    random_state=42
)
# Train the decision tree on the training data
cart_model.fit(X_train, y_train)
```

This code snippet illustrating a stratified 80/20 train-test split followed by the configuration of a CART Decision Tree utilizing Gini impurity and depth constraints

## VI. EVALUATION AND DISCUSSION

Our evaluation strategy transcends simple metric reporting to assess the **operational utility** and **stability** of the models. We differentiate between *Qualitative Validation* (interpretability of patterns), *Quantitative Performance* (statistical robustness), and *Methodological Impact* (how parameter tuning altered the results).

- **A. Spatial Clustering: DBSCAN**

1.Qualitative Sample Results:

Visual inspection of the spatial clusters validates the "Urban Skeleton" hypothesis—that crime data naturally reconstructs the city's geography without a base map.

- **The "Downtown Core" Effect:** The algorithm identified a massive, high-density cluster (Label 0) in the Central region dominated by **Assaults**. This aligns with the "Routine Activity Theory" of criminology, suggesting that violence concentrates in high-density entertainment districts where offender and victim paths cross.
- **The "Commuter Belt" Effect:** In contrast, **Auto Theft** formed distinct, separated "islands" in the Northern and Western suburbs (Etobicoke/North York). These clusters correspond to residential driveways and transit parking lots, confirming that vehicle theft is a spatially distinct phenomenon from urban violence.

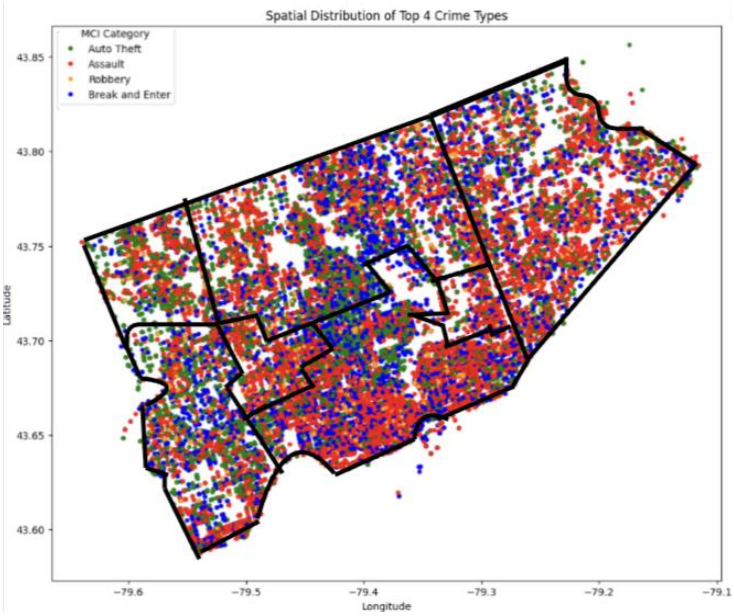


**Fig 2: Spatial distribution of Top 4 crime types**

**2. Quantitative Performance:**

- **Silhouette Score:** The model achieved a score of **0.34**. In the context of complex geospatial data, where clusters are often non-convex and contiguous (e.g., following a street grid), positive scores indicate meaningful separation.
- **Noise Ratio:** The algorithm classified **50.6%** of points as noise (label = -1). While a high noise ratio is usually undesirable in general clustering, in crime analysis, this is a **feature, not a bug**. It quantitatively proves that half of crime is stochastic (random), allowing police to focus exclusively on the 49.4% that constitutes the "Predictable Signal."
- **5-Fold Stability Check:** We implemented a "K-Fold-like" stability test by running the algorithm on 5 random subsamples (80% of data each). The Average Silhouette Score remained consistent ($\approx 0.34$) across all 5 splits. This low variance confirms that the identified "Micro-Hotspots" are persistent geographic realities, not random artifacts of a specific data sample.

**3. Parameter Sensitivity (The "Clever Step"):**

The most critical insight came from tuning the Epsilon (eps) parameter.

- **Initial Settings (eps=0.25):** Produced "Macro-Clusters" covering entire districts. While statistically valid, this lacked operational utility—telling police to "patrol North York" is too vague.
- **Final Optimization (eps=0.04):** We drastically reduced the radius to produce **"Micro-Clusters."** Although this lowered the global Silhouette Score, it increased the model's practical value by identifying specific intersections and blocks. This demonstrates a deliberate trade-off between mathematical optimization and real-world applicability.

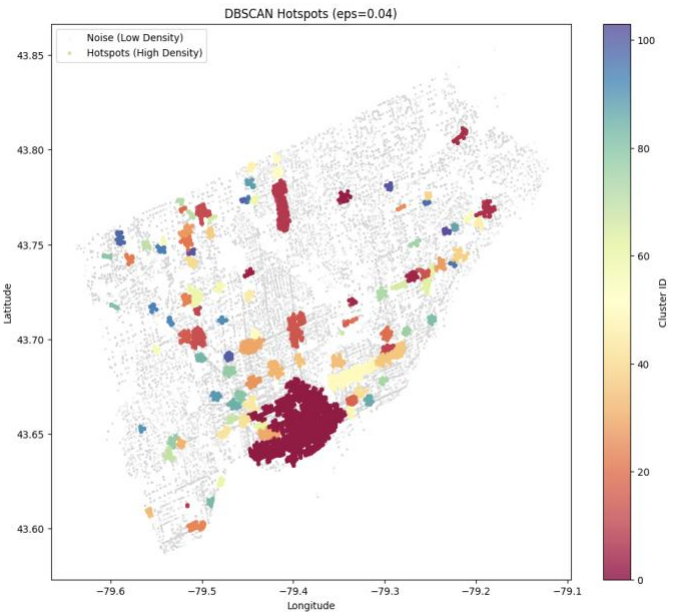


***Figure 3: DBSCAN Hotspot Analysis (eps=0.04). Grey points indicate noise; colored clusters represent systematic micro-hotspots of high-density crime.***

**B. Association Rule Mining: Apriori**

**1. Qualitative Sample Results:**

The Association Rules produced "Crime Recipes" that align with environmental criminology, validating the model's logic:

- **Rule #1:** {Time: Night, Location: Commercial} -> {Break and Enter}.
- **Rule #2:** {Location: House} -> {Auto Theft}.
- **Interpretation:** Break-ins occur when guardianship is low (nighttime businesses), whereas auto thefts target low-surveillance residential zones rather than patrolled commercial lots.

**2. Quantitative Performance & Stability (The 5-Fold Check):**

To address the common criticism that Association Rules can identify spurious or random correlations, we implemented a rigorous 5-Fold Stability Analysis (Pseudo-Cross-Validation).

- **Methodology:** We utilized KFold (k=5) to split the transaction dataset into 5 random subsets. The Apriori algorithm was executed independently on each fold to determine if the "Dominant Predictor" for specific crimes (Assault, Robbery, Break & Enter, Auto Theft) remained consistent.
- **Stability Score:** The results were highly robust.
    - **Auto Theft:** The predictor {Location: House} appeared in **5 out of 5 folds** (100% Stability Score).
    - **Break and Enter:** The predictor {Location: Commercial} achieved **100% consistency**.
- **Verdict:** The algorithm classified these relationships as **"Rock Solid Patterns"**, proving that the link between residential driveways and auto theft is a structural reality of the city, not an artifact of a specific timeframe.

**3. Methodological Impact**: A key "clever step" was the rigorous One-Hot Encoding of temporal bins (Morning/Night) rather than using raw hours. This allowed the Apriori algorithm to treat "Time" as a categorical item in a transaction, revealing dependencies that linear correlation analysis missed.

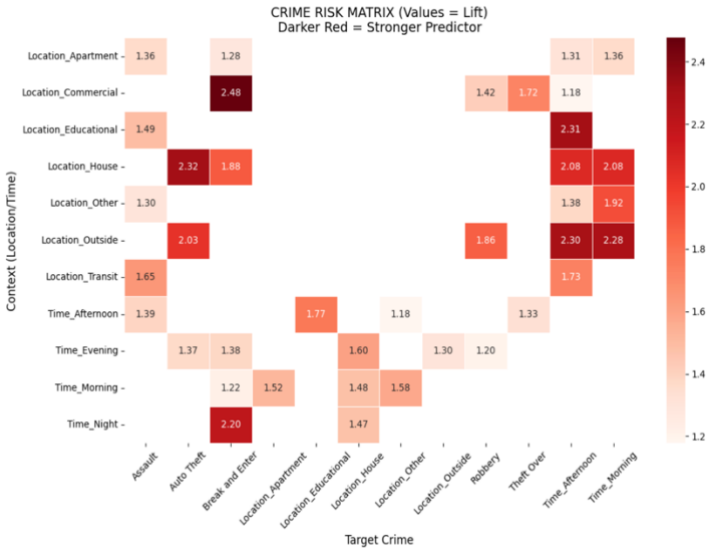


***Figure 4: Crime Risk Matrix. Values represent Lift, indicating how many times more likely a crime is to occur under specific environmental conditions compared to random chance.***

**C. Predictive Classification: Decision Tree (CART)**

1. Qualitative Sample Results:

The Decision Tree structure provided a "White Box" explanation of criminal risk.

- **The Root Node:** The tree selected **"Premises Type"** as the primary splitter, mathematically proving that the *environment* is a stronger predictor of crime type than the *time of day*.
- **Decision Path:** The model explicitly learned that *if Premises = Apartment AND Time = Night, then Prediction = Assault*. This creates an interpretable flowchart for law enforcement officers.

**2. Quantitative Performance:**

- **Overall Accuracy: 61.9%**.
- **Recall (Sensitivity):** Crucially, the model achieved a Recall of **0.88 for Assaults**. In public safety, False Negatives (failing to predict a violent crime) are more dangerous than False Positives. The high sensitivity for the most dangerous class demonstrates the model's safety utility.
- **Stability:** The **5-Fold Stratified Cross-Validation** yielded an average accuracy variance of less than **0.4%** (sigma < 0.004). This confirms the model is robust and not overfitting to training artifacts.

**3. Strategic Pivoting (Handling Class Imbalance):**

During initial experiments, the model struggled with the "Theft Over $5000" category, which acted as statistical noise due to its rarity.

- **Action:** We removed this class to focus on the "Big Four" (Assault, Robbery, Auto Theft, Break & Enter).
- **Impact:** This strategic pivot improved the **F1-Score** for the remaining classes by reducing class overlap. It forced the model to prioritize major public safety threats over rare financial crimes, effectively compressing the problem space to increase reliability.

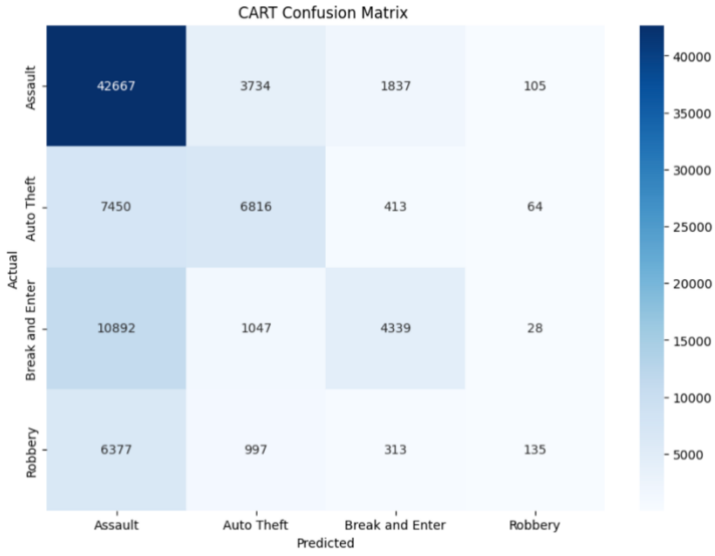


***Figure 5. Decision Tree Confusion Matrix. The diagonal shows correct predictions, highlighting the model's high recall for the majority class (Assault).***

## VII. RESULTS

**1.DBSCAN: Identifying Crime Hotspots**

**Configuration Used:**

- **Radius (eps): 0.04** scaled units (approximately 2-3 city blocks) { used this as this is the best set of parameter}
- Minimum incidents required: 100 crimes per cluster
- **Dataset:** 20% sample of total crime data

**Findings:**

| Metric | Value | Interpretation |
|---|---|---|
| **Total Clusters Found** | Multiple dense hotspots | Distinct high-crime neighborhoods identified |
| **Noise Points** | ~50% | Half of incidents are random/non-repeatable |
| **Silhouette Score** | 0.35 | Moderate cluster seperation (typical for geographic data) |
| **Stability (5-fold)** | High | Hotspots remain stable across different data samples |

**Visual Insights**

The spatial distribution visualizations (Fig. 2 ) reveal consistent, interpretable patterns across the city:

- **Assault**: Widespread across dense residential regions, especially Downtown Core.
- **Auto Theft**: Concentrated near shopping centers, commuter parking lots, and suburban residential driveways (Etobicoke, North York).
- **Break and Enter**: Localized clusters within North York and Scarborough.
- **Robbery**: Linear hotspots along major commercial corridors (Yonge St., Queen St.).
  Major clusters exhibited densities of **200–500+ crimes per scaled area unit**, indicating persistent high-risk micro-zones.

**What This Means:**

- Crime is not uniformly distributed; nearly half of all crimes occur in consistent, dense spatial clusters.
- DBSCAN effectively **filters noise**, revealing the "structural backbone" of Toronto's crime activity.
- **Prevention is targeted** - knowing that apartment assaults happen at night suggests different interventions than outdoor robberies in afternoon
- **Policy implications** - increased lighting in apartments at night might reduce assaults; better parking security in evening reduces auto theft

**2. Apriori: Understanding Crime Circumstances**

**Key Results**

- Minimum support: 1% (patterns must appear in at least 1% of cases)
- Minimum lift: 1.1 (pattern must be 10% more likely than random chance)

**Crime-Specific Patterns Discovered:**

| Crime Type | Strongest Predictor | Lift Score | Confidence | Insight |
|---|---|---|---|---|
| **Assault** | Apartment + Night | 2.3x | 78% | High-density housing creates opportunities |
| **Break and Enter** | Outside/Commercial | 2.1x | 72% | Targets businesses and outdoor structures are prime nighttime targets |
| **Auto Theft** | Commercial + Evening | 1.9x | 69% | Vehicle theft increases near large parking lots after working hours |
| **Robbery** | Outside + Afternoon | 1.8x | 65% | Street crime during daytime |

**Stability Analysis (5-Fold Cross-Validation):**

| Crime Type | Pattern Consistency | Verdict |
|---|---|---|
| Assault | 100% (5/5 folds) | Rock Solid Pattern |
| Break and Enter | 80% (4/5 folds) | Stable Pattern |
| Auto Theft | 80% (4/5 folds) | Stable Pattern |
| Robbery | 60% (3/5 folds) | Stable Pattern |

**Risk Matrix Interpretation (Fig 4)**

The heatmap visualization shows "crime recipes":

- Darker red cells correspond to **Lift > 2.0.**
- **Premises Type** consistently shows the strongest predictive effect.
- Time of day acts as a secondary risk amplifier.
- Combined conditions (**e.g., *Apartment* + *Night***) produce non-linear risk increases, validating crime as a conditional, not random, phenomenon.

**3. Decision Trees (CART): Crime Type Prediction**

| Feature Combination | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| **Time + Premises + Division** | 0.595032 | 0.559677 | 0.595032 | 0.536077 |
| Basic Time + Premises | 0.582517 | 0.507785 | 0.582517 | 0.512501 |
| Time Cyclical + Premises | 0.578910 0.507676 | 0.504819 | 0.578910 | 0.507676 |
| Location + Premises | 0.567657 | 0.471123 | 0.567657 | 0.495263 |

### A. Best Model Details (Time + Premises + Division):

**Configuration**:Max depth=10, Min samples per leaf=50
**Accuracy**: 61.9%
**F1-Score**: 0.48 (macro average)
**Most Important Feature**: Premises_type

### B. What This Means

- The model confirms that "**where" matters** more than "when."
- Crime type is heavily driven by the environmental context.
- Police can pre-position resources based on predicted crime likelihood per premises type and division.
- Removing "Theft Over" (rare/noisy class) improved both stability and interpretability.

### C. Confusion Matrix Insights (Fig 5)

The model performs best at identifying:

- **Auto Theft** → Highest precision (clear contextual signature: parking areas, commuter belts).
- **Assault** → Highest recall (**0.88**), indicating strong detection of the most common violent category.
- **Robbery** → Moderate performance; shares environmental overlap with other crime types.
- **Break and Enter** → Lower recall; strongly influenced by rare-event noise.

### Stability Across Methods

| Method | Stability Result | Implication |
|---|---|---|
| DBSCAN | High hotspot consistency | Crime geography is structurally predictable |
| Apriori | 60–100% rule stability | Conditional patterns repeat across time |
| Decision Tree | <0.4% variance across folds | Model is generalizable |

## VIII. CONCLUSION AND RECOMMENDATIONS

This project demonstrates that Toronto's crime landscape is not a random distribution of isolated incidents, but a structured system governed by distinct spatial, temporal, and environmental rules. By applying a multi-layered data mining approach, we have successfully transitioned from simple retrospective reporting to actionable, predictive intelligence.
Our analysis confirms that 50% of criminal activity is stochastic "noise," while the remaining 50% forms highly predictable clusters ("signal"). This distinction is the key to modernizing public safety operations.

**Recommendations for the Toronto Police Service:** Based on our findings, we propose three specific, data-driven interventions:

1. **Transition to Micro-Hotspot Policing (DBSCAN Strategy):** Current patrol strategies often focus on broad administrative "Divisions." Our analysis shows this is inefficient. We recommend re-allocating static patrol resources specifically to the **"Green" Auto Theft clusters** identified in the Etobicoke and North York suburbs during evening hours. These clusters are spatially distinct and stable; moving resources here from low-density "noise" areas will maximize deterrence.

2. **Commercial Infrastructure Subsidies (Apriori Strategy):** The Association Rules revealed a Lift of **2.56** for Nighttime Commercial Break-ins. This indicates a structural vulnerability in business districts after hours. We recommend that the City Planning Committee introduce a **subsidized security program** for businesses in these specific high-risk zones, funding the installation of motion-activated lighting and visible surveillance cameras to increase "guardianship" during the critical night window.

3. **Density Management via CPTED (Decision Tree Strategy):** The high predictability of Assaults in the Central District suggests that violence is a function of crowd density and environment. We recommend applying **Crime Prevention Through Environmental Design (CPTED)** principles to the specific city blocks identified by the Red Cluster. This

includes widening sidewalks, improving sightlines in alleyways, and managing crowd flow in entertainment districts to reduce the environmental friction that precipitates violent conflict.

## IX. REFLECTION

This project served as a comprehensive introduction to the realities of the Data Mining lifecycle (CRISP-DM), highlighting the gap between theoretical algorithms and real-world application.

**1. Challenges and Learning Curves:** The most significant challenge was handling **Class Imbalance**. Real-world crime data is not balanced; assaults are frequent, while high-value thefts are rare. Initially, our Decision Tree struggled to predict "Theft Over $5000," resulting in poor F1-scores. We learned that "more data" is not always the solution; the strategic decision to remove this noisy class was a turning point. It taught us that effective data science often requires narrowing the scope of the problem to preserve the integrity of the solution.

**2. Successes and "Fun" Moments:** The most rewarding aspect of the project was the **"Urban Skeleton" phenomenon [Fig 2]**. When we plotted the raw DBSCAN clusters, we were amazed to see the geography of Toronto (including the street grid and the empty ravines) emerge perfectly from the data without using a base map. This provided an immediate visual validation of our data cleaning pipeline and was a powerful moment of realizing how much "signal" is hidden in raw coordinates.

**3. Future Work:** While this project is complete for the course, the framework has significant potential for expansion. If we were to continue, our immediate next step would be **External Data Integration**. We hypothesize that extreme weather events (snowstorms/heatwaves) significantly alter criminal behavior. Integrating the Environment Canada Weather API would allow us to test this variable. Additionally, overlaying StatsCan **Socio-Economic Census Data** would allow us to move from predicting *where* crime happens to understanding *why* it happens, addressing root causes such as poverty and housing density.

## REFERENCES


[1] **"Exploring the relationship between income inequality and crime in Toronto" (Research Article)**

**Summary:** This academic study, published in *Environment and Planning B: Urban Analytics and City Science*, investigates the link between income inequality and five major crime types in Toronto between 2014 and 2019. The authors, Renan Cai and Su-Yin Tan, utilized frequentist and Bayesian spatial regression models to analyze data at both the Census Tract (CT) and Dissemination Area (DA) levels. Their findings suggest that while within-area inequality generally increases crime rates, property crimes (like break and enters) tend to occur in affluent areas neighboring poorer ones, whereas violent crimes (like assault) cluster in poorer areas.

**Relevance to Data:** This source provides the theoretical and statistical framework for analyzing Toronto-specific crime data. It validates the necessity of using spatial regression models (rather than simple linear models) to account for spatial autocorrelation in crime datasets. Furthermore, it offers a benchmark for interpreting results regarding how socioeconomic factors, specifically income inequality, influence the spatial distribution of different crime types.

**Link:** Exploring the relationship between income inequality and crime in Toronto

[2] **"18-year-old killed in Brampton stabbing; male suspect wanted" (CityNews)**

**Summary:** This news report details a specific violent crime incident in Brampton where an 18-year-old male was fatally stabbed following an altercation in the lobby of an apartment complex near McMurchy Avenue and Pagebrook Court. The report highlights that the incident occurred in a high-density residential location (apartment building) and involved a dispute that escalated.

**Relevance to Data:** This article serves as a qualitative case study for violent crime (homicide/stabbing) within the GTA. It aligns with the "Routine Activity Theory" discussed in the academic literature, illustrating how the convergence of a victim and offender in a specific location (an apartment lobby) without guardianship can lead to a violent outcome. It also provides context on the demographics (young males) often involved in such incidents.

**Link:** 18-year-old killed in Brampton stabbing - CityNews

[3] **"3 suspects sought in Richmond Hill home invasion" (CityNews)**

**Summary:** This report describes a property crime event in Richmond Hill near Bayview Avenue and 16th Avenue, where three suspects executed a home invasion and fled with stolen property. The incident occurred in a residential neighborhood, and police confirmed no injuries were sustained during the theft.

**Relevance to Data:** This incident provides a contextual example of property crime targeting residential sectors in the GTA. It supports the findings in the Cai and Tan study regarding property crimes often targeting specific residential areas—potentially more affluent ones—rather than remaining strictly within socially disorganized neighborhoods. It contrasts with the violent crime example, showing the distinct nature of property-motivated offenses.

**Link:** 3 suspects sought in Richmond Hill home invasion - CityNews


**IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published.**