Crime in Chicago (Clustering)

Addis Ababa University Institute of Technology
School of Information Technology and Engineering
Department of Artificial Intelligence

Group Name: Thomas Kitaba (GSE/9938/17)
Rediet Girmay GSE/0945/17(GSE)
Instructor: Dr Fanahun Bogale

Step 1: Introduction

This project is prepared as a subsumption to this question made by Dr Fantahun Bogale for the course named Machine learning, and this was the question “you are expected to solve a Machine Learning problem of a reasonable depth. It should include several learning elements; enhancement techniques.
You can solve the problem using regression, classification or clustering methods.
You are expected to submit your project titles through your representative.
The project should be deployed. “

Step 2: Identify the problem:

These days lots of crimes are being committed in different places by different criminals. Police and other entities who are responsible to fight crime need to gather their resources in an efficient manner to identify, minimize and eliminate crimes before they even happen.
To be able to accomplish this, responsible entities need to get accurate, timely, non redundant and relevant information related to the crimes. After wards the data can be analyzed and interpreted to serve as input for decision making process.

Here are list of the problems: (this is actually what we want to solve)

- lack of timely information to deploy the right number and type of security personnel on locations where specific types of crimes are known to happen.

- large amount of money and personnel is required to analyse and prepare report by analyzing large amount of data gathered in a span of large time period.

- hard to get to know the magnitude and type of crime happening in specific locations.

- In many occasions the Management needs to have specific information to make the best decisions, which were impossible to get with existing technology
Example: accurate prediction of crime rate, arrests.

- Here are list of the problems in a more concise manner

- Location Based information: areas related to different crimes.
- Temporal Based information: knowing time of high crime activity has significant for decision making like how many personnel to deploy on the field at what time.
- Police Activity pattern: Study law enforcement response and behavior
- Crime Type Clustering: Group similar crimes together to study their behavior

Step 3: Chose a task: (defines What we want to do/accomplish)

Note: Task is chosen based on our problem
Selected task: So taking this into account Clustering task is selected.
Why: many of the problems listed require clustering task

NB: for the purpose of learning we will create a model that clusters crimes based on their location, so that the we can study their behaviour.

Step 4: The Dataset

The dateset used for this project is fetched from Keggle (the world's largest online data science and machine learning community)

Name of Dateset: Crime in Chicago Number of Features: 22
Task Type: Classification and Clustering

	Feature Name	Condensed Description
1	**ID**	Unique identifier for the crime record.
2	**Case Number**	Unique police case identifier (RD number).
3	**Date**	Date and time when the crime occurred (approximate in some cases).
4	**Block**	Partially masked address (e.g., "010XX S MICHIGAN AVE") of the incident.
5	**IUCR**	Illinois Uniform Crime Reporting code for categorizing the crime.
6	**Primary Type**	Main category of the crime (e.g., THEFT, BATTERY, NARCOTICS).
7	**Description**	Subcategory of the crime (e.g., OVER $500 under THEFT).
8	**Location Description**	Type of place where the crime happened (e.g., STREET, RESIDENCE, PARKING LOT).
9	**Arrest**	Indicates if an arrest was made (True or False).
10	**Domestic**	Whether the incident was related to domestic violence (True or False).
11	**Beat**	The smallest police patrol unit area where the crime happened.
12	**District**	Larger police area containing several beats.
13	**Ward**	City council district where the incident occurred.
14	**Community Area**	One of 77 defined community areas in Chicago where the crime took place.
15	**FBI Code**	Crime classification code from the FBI's system.
16	**X Coordinate**	X (East-West) coordinate in the state plane projection (used for mapping).
17	**Y Coordinate**	Y (North-South) coordinate in the state plane projection.
18	**Year**	The year when the incident occurred.
19	**Updated On**	When the record was last updated in the database.
20	**Latitude**	Geographic latitude of the crime location (masked slightly for

	Feature Name	Condensed Description
		privacy).
21	**Longitude**	Geographic longitude of the crime location (masked slightly for privacy).
22	**Location**	Combined lat/long in point format, used for mapping and GIS operations.

Step 5: How to chose a clustering Algorithm

To select the appropriate clustering algorithm we can use the following methods

4.1 Study the Dataset:
The first one is study the structure of our datasets to know the clustering type existing with in the datasets. Once we have information about the clustering type then it will be easier to chose the appropriate clustering algorithm.

Question: How do we study the dataset?

Answer: Here are some of the ways we can use to study the structure of our dataset
4.2 use Pandas built in functions like to see and understand the structure

- header():
- describe():
- columns():

4.3 Using dimension reductions like PDA and t-SNE:

4.3.1 PCA (Principal Component Analysis):
Linearly projects high-dimensional data into fewer dimensions while preserving variance. Useful to get a rough idea of data structure and clusters.

4.3.2 t-SNE (t-distributed Stochastic Neighbor Embedding):
Non-linear technique that captures local relationships and visualizes complex cluster structure better than PCA, especially for visualization.

4.4 Train multiple models and chose the best one:

Here we can train multiple clustering models on the dateset then compare them using different metrics, then select the algorithm with the desired result.
This one is simpler and will certainly help up pinpoint the best clustering algorithm for our problem.

Note: it is always better to view the structure of the data, as well as explore different models, so we will use both methods

Step 6: The Clustering Algorithms used

To solve our machine learning problem we have managed to test all the three clustering algorithms. Here are the three clustering algorithms we used in our project.

5.1 Kmeans:

K-Means is like picking meeting spots for a group of scattered friends. It tries to group everyone around a few central locations.

Algorithm steps or How it works

1. **Choose the Number of Groups (K):**
   Decide how many clusters you want (e.g., K = 3).
   The algorithm randomly picks K starting points (initial "centers").
2. **Assign Each Friend to the Nearest Center:**
   Every data point (friend) joins the closest center.
3. **Update the Centers:**
   For each group, calculate the average position of the people in it.
   Move the center to this average point.
4. **Repeat Until Stable:**
   Keep reassigning and updating centers until things stop changing much.

5.2 DBSCAN:

DBSCAN is a density-based clustering algorithm that groups together closely packed data points, marking as outliers points that lie alone in low-density regions.

How DBSCAN Works

1. Everyone Starts Separate

Each data point starts unvisited and unassigned.

2. Explore the Neighborhood

Pick any unvisited point and find how many points fall within its **ε (epsilon)** radius.

3. Decide Its Role

If it has at least **MinPts** neighbors → it's a **core point** and starts a new cluster.

If it has fewer than **MinPts** neighbors:

But lies within the ε-neighborhood of a core point → it's a **border point** and joins that cluster.

Otherwise → it's labeled as **noise** (an outlier).

4. Expand the Cluster

If the point is a core point, all directly reachable **core** points and their **border** points are added to the same cluster.

5. Keep Going

Repeat the process until all points are either:

Assigned to a cluster, or Marked as noise.

5.3 Agglomerate:

Clustering builds groups by merging the closest pairs step-by-step. Think of it as building a friendship family tree.

How Agglomerative Clustering Works

1. **tart with Everyone Separate**

Each person (data point) begins as their own individual group.

2. **Find Closest Pairs**

Look at all the current groups and find the two that are most similar (i.e., closest).

3. **Merge Them**

Combine these two closest groups into one larger group.

4. **Repeat**

Continue finding and merging the closest groups until:All points are in a single group, **or** You decide to stop at a desired number of clusters.

5. **Visualize with a Dendrogram** A **dendrogram** is a tree diagram that shows the merging process. The **height** of each branch represents how far apart the groups were when they merged.

6. **Choose Final Clusters**

Decide how many clusters you want by drawing a horizontal **cut across the dendrogram**. Each vertical line it intersects becomes a final cluster.

Step 7: Evaluation Matrics

These are the evaluation matrices we used to measure how good the clusters created are.

6.1 Type of Evaluation Matrics

6.1.1. Silhouette Score

What it measures:

How similar a data point is to its own cluster (cohesion) compared to other clusters (separation).

Formula: $s(i) = \frac{b(i)-a(i)}{max(a(i),b(i))}$

· a(i): average distance to all other points in the **same cluster**

· b(i): average distance to all points in the **nearest other cluster**

· Silhouette score for point i:

6.1.2 Davies–Bouldin Index (DBI)

What it measures:

The average “similarity” between each cluster and its most similar one. **Lower is better.**

Formula:

$$\text{DBI} = \frac{1}{k}\sum_{i=1}^{n} \underbrace{max}_{j!=i} R_{ij}$$

- Si : average distance of points in cluster iii to its centroid (intra-cluster distance)
- Mij: distance between centroids of clusters i and j.

6.1.3 Calinski–Harabasz Index (Variance Ratio Criterion)

What it measures:

Ratio of between-cluster dispersion to within-cluster dispersion. **Higher is better.**

Formula: $\text{CH Index} = \frac{\text{Tr}(B_k)}{\text{Tr}(W_k)} \cdot \frac{n-k}{k-1}$

- Tr(Bk): trace of between-cluster dispersion matrix
- Tr(Wk): trace of within-cluster dispersion matrix
- nnn: total number of data points
- kkk: number of clusters

6.2 Clustering Evaluation Interpretation Table

Metric	Score Range	Interpretation
Silhouette Score	> 0.75	Excellent
	> 0.50 to 0.75	Good
	> 0.25 to 0.50	Moderate
	> 0.00 to 0.25	Poor
	≤ 0.00	Not Applicable
Davies-Bouldin Index	< 0.50	Excellent
	0.50 to < 1.00	Good
	1.00 to < 2.00	Moderate
	≥ 2.00	Poor
Calinski-Harabasz	> 1000	Excellent
	> 500 to 1000	Good
	> 100 to 500	Moderate
	≤ 100	Poor

Step 7: Final Note

So based on the problems identified at the beginning of this project, we trained an unsupervised machine learning model that solves a clustering task, and based on the model a web app is deployed on streamlit server this is the link to the website is https://thomaskitaba-3-crime-in-c-best-cluster-for-chicago-crime-fvqvoe.streamlit.app/

Git address: https://github.com/thomaskitaba/3-crime_in_chicago.git