ISSN : 2231-0711

Divya G et al | International Journal of Computer Science Engineering and Technology(IJCSET) | July 2014 | Vol 4, Issue 7,231-234

Suitability of Clustering Algorithms for Crime Hotspot Analysis

Divya G
SCT College of Engineering
Pappanamcode
Trivandrum

Rejimol Robinson R R
Asst. Professor
SCT College of Engineering
Pappanamcode, Trivandrum

Kalai Selvan
Joint Director
BCG, CDAC
Trivandrum

Abstract-Crime analysis is a field that needs immediate attention due to the drastic increase of number of crimes. Most of the crimes, from the past experiences of the police, are said to be concentrated in some areas, called hotspots and keep on recurring. Clustering algorithms are best applied to crime analysis, but suitability of broad spectrum of clustering algorithm for an application is an issue to be addressed. In this paper we evaluate three clustering algorithms i.e. hierarchical clustering, k-means clustering and DBSCAN clustering with the intent of finding the best one suitable for crime hotspot analysis. Each one of the clustering algorithm evaluated here need inputs such as number of clusters, neighbour distance, minimum number of points etc. are needed by a cluster. The cluster similarity measure is the Euclidean distance. The results suggest that DBSCAN is much more suitable to crime hotspot analysis due to its inherent nature of being density driven.

Keywords: Hotspot Analysis, Clustering, Data Mining, Average Link Algorithm, K-means Algorithm, DBSCAN, Davies- Bouldin index.

I. INTRODUCTION

Crime analysis is gaining importance day by day as the number of crimes is increasing drastically. This is true especially in case of crimes against women, which is the prime focus for police nowadays. Crime analytics is a process of gathering and analysing recurring patterns in crime. Crime analytics aids police in intelligently making decisions and proactively planning the defence mechanism.

Biggest challenge involved in crime analytics is processing large amount of historical data. Also the police personnel may have a very busy schedule to process the data. If crime analytics can be performed so that the data is processed and patterns are generated out of given data and patterns thus generated if displayed on the GIS based map, can help police personnel in making decisions faster. Police personnel can take proactive actions such as assigning more force at places where there is high probability of crime occurrence.

Hotspots are high concentration areas of some activity. Crime hotspots are concentration of crimes in specific geographical area, recurring over time. Experience of police shows that certain geographical areas called crime generators have high probability of occurrence of crime.Though there is no theoretical basis, but police can use the knowledge of hotspots in reducing the crime rates. Hotspots are areas of imaginary boundary where there is recurrence of crime incidents.

Clustering is a process of grouping of elements in a dataset in such a way that the elements in a cluster are similar to one another than the elements in other cluster. Grouping of elements happens by evaluating the elements of dataset over some similarity metrics. Clustering is a generic representation for a class of algorithms, used for the extracting similarity in dataset. This data mining methodology may require a set of parameters to be specified such as number of clusters, minimum radius of a cluster, minimum set of points around a data element to qualify as a cluster etc.

Clustering is an unsupervised learning, where the training set of data is not provided as the clusters learn the patters by themselves. The typical steps of clustering are representation of patterns, definition of data proximity measure appropriate to the data domain, grouping, abstraction and assessment of output [1].

The objective of this paper is to evaluate the different clustering algorithm that can be used effectively for crime hotspot analysis.

This paper is structured as follows. Sect. II provides an overview of type of clustering algorithms. Sect. III briefly describes hierarchical clustering and provides the algorithm for average link clustering. Sect. IV presents the partitioning clustering algorithm, K-means with the capability of K-means++. Sect. V describes DBSCAN, density based clustering, and accesses its suitability to cluster geographical data. Experiment based on geographical data to detect crime hotspots are presented in Sect. VI. The evaluation of the three algorithms is provided in Sect. VII. In Sect. VIII, we summarize the evaluation and discuss the future work.

II. CLUSTERING ALGORITHMS

A particular algorithm cannot be correct or wrong. Suitability of cluster algorithm depends on the kind of application and dataset. As said “clustering is in the eye of the beholder"[2]. The specific requirements of an application may require usage of an algorithm. Suitability of algorithms for an application should be experimentally found.

The ranges of algorithms for clustering are numerous. References [3] and [4] survey the broad range of clustering algorithms. Clustering can be hard, where points belong to one set or other but not both, or soft, where the points can belong to different clusters.Clustering algorithms can be broadly classified into hierarchical and partitioning.Hierarchical clustering builds a tree structure from the dataset, starting from grouping individual point to form a parent node combining all the points in the dataset. In effect levels of elements of dataset are formed, each

level grouping a subset of elements. Partitioning clusters provide a distinct grouping of the elements of dataset, where the elements are grouped in disjoint set of clusters.

The simplest of the partitioning clustering is nearest neighbour clustering. This method aims to find the nearest neighbours of the given elements of datasets [5]. K-nearest neighbour clustering aims to form K clusters by taking each element and assigning it to clusters having shortest distance. The first cluster is formed by assuming the K to be first point. Though the simplicity of KNN gives it an edge over others but, decision about the value of K is critical. The final clusters may not be as close as the actual ones.

K means algorithm assumes K clusters and tries to move element to the cluster having closer mean. This recursive algorithm will produce K clusters containing the elements of the dataset partitioned so that points in a cluster are closer to the mean than others.

Density based clustering techniques look for areas of high density. Areas that are sparse are ignored. DBSCAN [7] is a popular density based clustering algorithm. This algorithm takes into consideration two parameters (i.e. minimum distance and minimum number of points) and generates the clusters automatically. OPTICS [8] is another density based clustering that does not require specifying the parameter, minimum distance, and produces a structure like hierarchical clustering.

III. HIERARCHICAL CLUSTERING

Hierarchical clustering technique considers each point as a single cluster and merges them together until all the points are merged into one cluster. The merging of clusters is often represented as dendrogram. A dendrogram is a tree-like structure used to represent the hierarchy of cluster merging in hierarchical clustering.

Let the dataset consists of $< x_1, x_2, x_3.....,x_n>$, where n is the number of elements of the dataset. Each element of dataset x_i can contain many attributes, represented by $< a_1,a_2,a_3, ...,a_m >$, where m is the number of attributes of an element of dataset.

A. Algorithm

Input: $< x_1, x_2, x_3.....,x_n>$

Output: $<L_i>$,$<C_{ij}>$ *where i is the number of levels and j is the number of clusters in each level.*

1. *Assign* $C_i = x_i$ *for all the elements of the dataset.*
2. *Calculate the adjacency matrix for all the clusters*

$$A_{ij}= d(x_i,x_j)$$

whereA_{ij} *is an n x n matrix and* $d(x_i,x_j)$ *can be defined as Euclidean distance.*

3. *Let level l=1;*
4. *Let p =n be the number of clusters at each level.*
5. *Find the most similar clusters from the adjacency matrix say* C_u *and* C_t *using a similarity criterion.*
6. *Merge the clusters and p=p-1.*
7. *Perform steps 3 and 4 for all clusters in a level l.*
8. *If total number of clusters =1 then stop.*
9. *If no more clusters are there to be added then l=l+1, update* A_{ij} *and go to step 4.*

The algorithm above provides the details of the hierarchical clustering. The result of the algorithm is the level and clusters in each level. Adjacency matrix for all the elements is calculated by finding out the distance between every two elements. Euclidean distance is used for calculation of adjacency matrix. Euclidean distance between two elements is defined as the square root of the sum of squares of the difference of each attribute of the two elements. Initially individual elements of the dataset are all treated as clusters. So at level 1 there will be as many as number of clusters as there are number of elements in the dataset. Subsequently, the clusters are merged using the similarity criteria and adjacency matrix is updated. The similarity criteria for the distance between the clusters can be based on three metrics: single link, complete link and average link. Single link will consider the distance between the clusters as the distance between closest elements of the cluster. Complete link takes into account the distance between the farthest elements in the two clusters. Average link calculates the similarity criteria for clusters by calculating the average of the distances of all the elements in both the clusters.

The equation for average link hierarchical clustering is as given below:

$$D(M,N) = \frac{1}{n*m}\sum_{i=1}^{n}\sum_{j=1}^{m} d(x(i),y(j)) \qquad (1)$$

where M and N are two clusters and d(x(i),y(j)) is the distance between the elements of the two clusters. m and n are the number of points in the two clusters.

IV. PARTITIONAL CLUSTERING

Partitioning clustering will group the elements of dataset into different clusters in such a way that one element belongs to only one cluster. Most popular and simplest partitioning algorithm is k-means algorithm. K-means assumes two parameters: k, number of clusters and initial value of centroids.

A. Algorithm

Input: $< x_1, x_2, x_3.....,x_n>$

Output:$<C_j>$ *where j is jth clusters.*

1. *Assume the value of k, the number of clusters.*
2. *For all the k clusters assume initial value of centroids.*
3. *For each element* x_i *calculate the Euclidean distance from the element to the centroid of all clusters*
4. *Move the element to closest cluster.*
5. *Calculate the new centroid.*
6. *If there are no more changes then stop else repeat steps 3,4 and 5.*

The output is k clusters, each of the clusters contains a subset of elements of the dataset. Here, starting from the first element find the closest cluster by calculating the Euclidean distance. Each element is moved to the closest cluster. Centroid of the cluster is recomputed. This process continues and halts when there are no changes to clusters. K-means is very efficient algorithm. But, the efficiency is limited by two factors: number of clusters and

the initial value of centroids. The initial value of centroid can be calculated using k-means++ algorithm [9]. k-means++ uses weighted probability distribution to choose centroids. Another method for choosing the initial value of centroids is randomly selecting k values from the dataset. Yet another limiting factor is that all the elements will be clustered. Noise elements will not be considered. Even if a point is standalone, it will be grouped in some cluster.

V. DENSITY BASED CLUSTERING

Density based clustering methods aims to build clusters with high density areas than the remainder of the dataset. The sparse areas are considered as noise and are ignored. DBSCAN (Density-based spatial clustering of application s with noise) is the most popular density based algorithm. This algorithm needs two parameters: *eps*, minimum distance of a point from a cluster to consider it as a neighbour and minimum number of neighbour points to group them into a cluster. The concept of minimum distance from the cluster is called density reachability.

A. Algorithm

Input: $< x_1, x_2, x_3, \ldots, x_n >$, *eps, minPoints*
Output:$<C_j>$ *where j is jth clusters.*

1. *Mark each point* x_i *UNCLASSIFIED.*
2. *Calculate the adjacency matrix of the input.*
3. *Let i=1*
4. *Consider* x_i *check if* x_i *is UNCLASSIFIED.*
5. *If yes then get neighbours of* x_i *such that the distance of each point to* x_i *is not more than eps.*
   - 5.1 *If number of such neighbours is greater than min points create new cluster and add* x_i *and all its neighbours to this new cluster. Also mark all of them CLASSIFIED.*
   - 5.2 *Otherwise mark this as ISNOISE.*
6. *i=i+1*
7. *If i>n stop.*

In this algorithm the points having nearby neighbours are formed as clusters and others which do not have enough density are ignored as noise. So we get only the clusters having high density. All the points are initially marked as UNCLASSIFIED. Each point is then visited and checked to see if it has enough neighbours in the distance *eps*. If so it is considered as clusters and all the neighbours are added to this cluster. If it does not have enough points then it is declared as noise. So the algorithm successively recognizes all the clusters with high density.

VI. EXPERIMENT

We have evaluated the suitability of clustering algorithm over a dataset of around 300 elements. Each of the element represents some crime happened along with its latitude and longitude. All the three algorithms were evaluated over this dataset. The various criterion were:

- Number of clusters(except K means)
- Average number of elements in clusters.
- Running time of algorithm in miiliseconds.
- Davies – Bouldin index.

Number of clusters provides the count of clusters identified by each algorithm. But in case of K means we need to provide the number of clusters. When the clusters and its elements are identified, average number of elements in each cluster will let us conclude that clusters are large enough. Also by using average we can eliminate clusters having relatively lower elements. Running time of the algorithm gives us the time elapsed between providing input to producing outputs. It was programmatically computed from the starting of the code to ending of the algorithm.

Davies – Bouldin index is an evaluation index for clustering. It can be calculated from the below given formula.

$$DBI = \frac{1}{k}\sum_{i=0}^{k} \max \left(\sum_{j=0.i\neq j}^{k} \frac{avg(i)+avg(j)}{distance(c(i),c(j))}\right) \quad (2)$$

where
k is the number of clusters.
avg(i) is the average distance of elements of the cluster to its centroid.
c(i) is the centroid of cluster.
distance(c(i),c(j)) is the distance between centroids of cluster *i* and *j*. It is measured as Euclidean distance.
For Davies- Bouldin index the smaller value is preferred.

VII. RESULTS AND DISCUSSION

The dataset of around 300 crime locations was provided as input to the algorithms. Each of the location contains the latitude and longitude of each point. The results of the evaluation over the different evaluation criterion are presented in tables below.

The results for the evaluation of hierarchical clustering are shown in Table1. The total running time for the hierarchical clustering is very high compared to other algorithms. From the input dataset 11 levels of clusters were formed, starting from each element being a cluster and finally all points merged into one cluster. The Davies-Bouldin index is very high. The lowest value of Davies – Bouldin index is generated at 10^{th} level where the number of clusters is two.

For K-means clustering, the input parameter is the number of clusters. K-means algorithm was evaluated by providing different values for k i.e. the number of clusters. Each criterion was recorded for each input value of k. Results are provided in Table 2.The lowest value of Davies – Bouldin is achieved when the value of k is five.

DBSCAN Algorithm achieves the best performance. The input values that are to be provided are the neighbourhood distance and the minimum number of points for an element in the dataset to be declared as a cluster. The most attractive feature of DBSCAN algorithm is that the number of clusters is automatically found and stand-alone points are labelled noise. The elements designated as noise are not included in the clusters. Smallest Davies – Bouldin index is attained when neighbourhood distance is 0.5 and minimum number of points ranges from 10 to 20.

Table 1: Evaluation of hierarchical clustering

LEVELS	EVALUATION CRITERIA			
	No of Clusters	Running Time (in milliseconds)	Average number of elements in each cluster	Davies – Bouldin index
1	321	28757	1	0
2	161		1	Infinity
3	81		3	Infinity
4	41		7	374.7319
5	21		15	25.1641
6	11		29	41.1954
7	6		53	56.5601
8	4		80	30.0025
9	3		107	10.5658
10	2		160	0.0007498
11	1		321	0

Table 2: Evaluation of K means Clustering algorithm

PARAMETERS (that can be changed)	VALUE OF INPUT PARAMETERS	Evaluation Criterion		
		Running Time (in milliseconds)	Average number of elements in each cluster	Davies – Bouldin index
K,number of clusters	30	1301	10	0.8146526
	25	386	12	0.8060029
	20	391	16	0.7425921
	15	358	21	0.7179025
	10	293	32	0.6043710
	5	303	64	0.340549

Table 3: Evaluation of DBSCAN Algorithm

PARAMETERS (that can be changed)	VALUE OF INPUT PARAMETERS	EVALUATION CRITERIA			
		No of Clusters	Running Time (in milliseconds)	Average number of elements in each cluster	Davies – Bouldin index
Neighbour Threshold , Minimum number of points	NeighborThreshold=0.5, minPoints=10,15,20	2	583	160	0.0007498
	NeighborThreshold=0.05, minPoints=10	2	432	158	4.7728222
	NeighborThreshold=0.05, minPoints=15,20	3	469	103	1.6291838
	NeighborThreshold=0.005, minPoints=10,15,20	4	419	74	0.9802176
	NeighborThreshold=0.0005, minPoints=10,15,20	5	437	52	1.2913722

VIII. CONCLUSION

Crime analysis is an important area as the number of crime is increasing these days. Data mining provides immense tools for crime analysis. Research on application of data mining algorithms in crime analysis is an area that needs much ore contribution. Crime hotspot analysis can provide the police personnel, knowledge on the crime generators. Police can take proactive actions to control crime in hotspot areas. DBSCAN is the effective algorithm for crime hotspot analysis.

REFERENCES

[1] A.K. JAIN, M.N. MURTY and P.J. FLYNN, "*Data Clustering : A Review*", ACM Computing Surveys, Vol 31, No.3, September 1999.

[2] Estivill- Castro, Vladimir, "Why so many clustering algorithms – A Positive Paper", ACM SIGKDD Explorations NewletterVol 4 Issue 1, pp 65-75, June 2002.

[3] Xindong Wu et al. , "*Top 10 algorithms in data mining*" Springer Knowledge and Information Systems, Vol 14, Issue 1, pp 1-37, January 2008

[4] RuiXu, Donald Wunsch II,"*Survey of Clustering Algorithms*", IEEE Transactions on Neural Networks, Vol. 16, No. 3, May 2005

[5] SebastienBubeck, Ulrike von Luxburg," *Nearest Neighbor Clustering: A Baseline Method for Consistent Clustering with Arbitrary Objective Functions*", Journal of Machine Learning Research Vol 10,pp 657-698,2009

[6] Nitin Bhatia, Vanadana," *Survey of Nearest Neighbor Techniques*", International Journal of Computer Science and Information Security (IJCSIS), Vol.8, No.2, pp 302-305, 2010.

[7] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and XiaoweiXu, "*A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise* ",Proceedings of the Second International Conference on Knowledge Discovery and Data Mining KDD-96, page 226-231. AAAI Press, (1996).

[8] MihaelAnkerst, Markus M. Breunig, Hans-Peter Kriegel, JörgSander , "*OPTICS: Ordering Points To Identify the Clustering Structure*", ACM SIGMOD international conference on Management of data, ACM Press, pp. 49–60, 1999.

[9] Arthur, D. and Vassilvitskii, S. (2007). "k-means++: the advantages of careful seeding". Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms. Society for Industrial and Applied Mathematics Philadelphia, PA, USA. pp. 1027–1035.

[10] Kriegel, Hans-Peter; Kröger, Peer; Sander, Jörg; Zimek, Arthur (2011). "Density-based Clustering". WIREs Data Mining and Knowledge Discovery 1 (3): 231–240