DBSCAN Algorithm with automated parameter selection

Subhadeep Maji
Department of Computer Science
& Engineering
IIT Kharagpur
India

Ravi Shankar Mondal
Tata Consultancy Services
Mumbai
India

Subhadip Banerjee
Infosys Technologies
Mysore
India

ABSTRACT

Clustering is an important problem affecting many fields of engineering and technology. Density based clustering is one of the most widely used clustering techniques. We in this paper propose an improvement to popular density based algorithm known as DBSCAN. The algorithm proposed solves one of the inherent limitations of DBSCAN; its inability to handle multiple densities in the dataset. The algorithm has been tested on number of data sets and results show it can handle multiple densities quite well.

1. INTRODUCTION

Cluster analysis is important in the rapidly growing field known as exploratory data analysis and is being applied to a variety of engineering and scientific disciplines.

Cluster analysis organizes data by abstracting underlying structure either as a grouping of individuals or as a hierarchy of groups. In this paper we propose an improvement to popular clustering algorithm known as “Density based Clustering of Applications with noise” by improving it to work when dataset has different densities and single set of parameters of the original algorithm fails to handle the dataset. Our algorithm is fully automated and extracts the density and radius parameters of the DBSCAN algorithm from the data itself. The complexity of the new approach is within bounds of what is needed by original DBSCAN algorithm. We finally validate our algorithm on a number of datasets.

Rest of this paper is organized as follows. **S**ection 2 outlines the original DBSCAN algorithm and sections 3 describes the algorithm proposed, section 4 shows experimental results obtained on sample datasets and section 5 indicates the references.

2. DBSCAN ALGORITHM

One of the most successful classes of clustering algorithms is the density based clustering algorithms ,which views data as consisting of spatial point processes each of which are distributed with constant but different density. The algorithm detects these densities and assigns them to clusters. Clusters of different densities belong to different processes. Ester et al. (1996) and Sander et al. (1998) introduced the approach of DBSCAN to address the detection of clusters in a spatial database according to a difference in density. The algorithm requires two parameters as user input they are radius parameter and the density parameter.

A. *BASIC IDEA OF DBSCAN*

Given a point p, let p $\in$ D, D is a subset of d-dimensional real space, $d \geq 1$. There are two elements to the definition of density-connected, the Eps-neighborhood

$$\text{N-Eps}(p) = \{q \in D \mid \text{dist}(p, q) <= \text{Eps}\}$$

such that Eps > 0 and MinPts, which is the minimum number of points in the Eps-neighborhood. A point p is said to be density-connected to point q with respect to (w.r.t) Eps and MinPts if there is a collection of points $p_1, p_2, \dots, p_n$ (with p_1 = q and p_n = p) so that

$$p_{i-1} \in \text{N-Eps}(p_i) \text{ (i = 2, 3, . . ., n)}$$

and N-Eps (p_i) (i = 2, 3, . . ., n−1) must contain at least MinPts points. Based on these ideas, a cluster is defined to be a point subset M where any point $p_i \in$ M is density-connected to point

$$p_j \in \text{M} (p_i \neq p_j).$$

In a cluster, point p_i belonging to M is denoted as a core point if the number of points included in N-Eps (p_i) is not less than MinPts. Otherwise, it is denoted as a border point, which is density-connected with at least one core point but with less than MinPts points in its Eps-neighborhood. The cluster can be formed by extending a point into a collection of points which are all density-connected with each other.

A. *DRAWBACKS OF DBSCAN*

The algorithm has some major drawbacks which are as follows:

1. The algorithm has two global parameters "*ε*" and "*Minpts*", estimation of which is difficult for a arbitrary dataset, although the original DBSCAN paper introduces some measures to estimate it but no standalone technique exists.

2. If the dataset has different densities (originating from different point processes in the data) then no single set of "*ε*" and "*Minpts*" can cluster the entire dataset, however DBSCAN does not take this into account.

3. The algorithm fails to scale well to large dimensions majorly because of difficulty in determining a proper distance estimate for large dimensions and complexity of searching the neighbourhood in large dimensions.

3 PROPOSED APPROACH

As stated earlier different densities occur in a dataset from different point processes and automated algorithm should be able to detect the "Eps" and "Minpts" parameter corresponding to each density automatically. We propose the following approach for doing the same.

Our approach is to generate average distance of k-nearest neighbours for each point in the dataset (obtained using a space partitioning tree like KD-tree), considering the average smoothen out the data obtained. Also generating the K-nearest neighbour distances in sorted manner can help identify noise with relative ease as noise points have large k-distance values. Our basic idea is if we can find knees in he k-distance plot then k-distance values corresponding to those knees can be used as estimates "*Eps*" parameter, this is because knee is a region where a sharp change of k-distance value occurs, any k-distance value lesser than this Eps estimate can efficiently cluster data points whose K-NN distance is lesser than that.

So if we can identify all the knees of the graph then a estimate of Eps 's can be obtained, each Eps estimate corresponding to a specific density in the dataset and hence DBSCAN algorithm in a modified manner if run for each estimate of Eps can efficiently determine all the clusters of different densities.

We plan to detect the knee regions using moving average second order differential filter on the sorted K-NN data. As known from property of differential filter it diminishes regions of slowly varying function (here the k-distance) and enhances regions having sharp changes.

Moving average has been used to smooth out small bursts which can otherwise be classified as knee regions by the differential filter. For proper detection of the Eps parameter we detect peak regions of filtered data obtained in the previous step. Note here for N point processes in the data which includes the noisy data we have N-1 peaked regions in the filtered output graph, we detect these regions and use the K -NN value in those region as estimates for Eps.

The k-distance values corresponding to those knees can be used as estimates of "Eps" parameter. *Knee* is a region where a sharp change of k-distance value occurs. Identification of all the *knees* of the graph gives an estimate of all "ε 's". DBSCAN algorithm has been modified in a manner that its sequential execution for each estimate of "ε" can efficiently determine clusters originating from corresponding point processes in a multi-density data.

A. *AVERAGE KNN DETECTION*

Naïve algorithm for finding KNN works poorly. It will take at least O(n^2 log n) because for each data-point we need to compute its distance with all other points and then we have to sort the distances. So, for a large image data it consumes a considerable amount of time to plot the K-NN curve. To overcome this problem, we have used KD tree (one kind of space-partitioning tree) by which we can solve the query of nearest neighbour in O(log n) time. So, finding k nearest neighbours for each n data-points takes O(k*n*log n), but here k is a very small quantity, it is almost constant, so we can assume it doesn't contribute to the asymptotic behaviour of K-NN algorithm and thus O(k*n*log n) becomes O(n*log n).

B. *ESTIMATION OF KNEE REGIONS*

We use a second order filter on the average Knn data.

Let F(x) be a discrete function then the finite difference form of second order filter is

$$F''(x)=F(x+1)+F(x-1)-2*F(x)$$

Moving Average filter applied on discrete function F(x) is

$$\text{m\_avg}(x) = \sum_{l=x-a}^{x+a} F(l) / 2*a$$

where, 2*a is the window size of the filter.
Here we choose the filter window size as 10.

The algorithm uses a 2nd order filter to estimate regions of sharp change in the gradient of the k-NN graph. The filtering is done in steps. Moving average filter is applied at the output of first order filter. The absolute values of the second order filter output are taken and again averaged out. An averaging filter provides a sharp contrast in the k-NN values of the knee-region indices compared to that of other indices. Section 4 shows the application of moving average filter to detect knee regions on datasets.

C. *DETECTION OF Eps VALUES*

The idea is to detect the high peak regions of the filtered data from the previous algorithm output. The results show that given "*N*" number of point processes, which includes noisy data, with sufficiently varying densities and sufficiently high representation in terms of data points, we are bound to get "*N-1*" peaked regions in the filtered graph. These peaks in the graph are meant to contain the indices, which are meant to point to the appropriate **"***Eps***"** values.

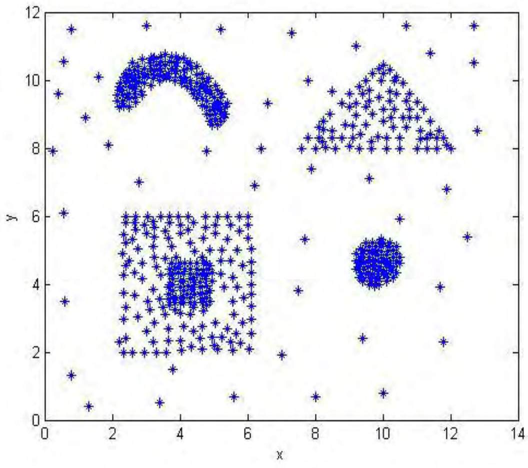
Since there is a direct relationship between the indices or abscissa value of the sorted k-NN graph and that of the filtered K-NN graph, the **"***Eps***"** estimate can be marked on the ordinate of the sorted k-NN graph, properly mapped with the filtered k-NN graph. The algorithm takes as input the output of the moving average filter and rejects the datapoints whose filtered knn values belong to a predominant fluctuation of the filtered data(as that is not significant with respect to knee regions, which is a narrow range of high filtered output values).Then K-means algorithm is run on the remaining datapoints to obtain cluster centroids which corresponds to datapoints whose filtered output are the "Eps" values. The optimal number of clusters is obtained by evaluating the cluster quality with Davies-Bouldin index.

D. *FINAL ALGORITHM*

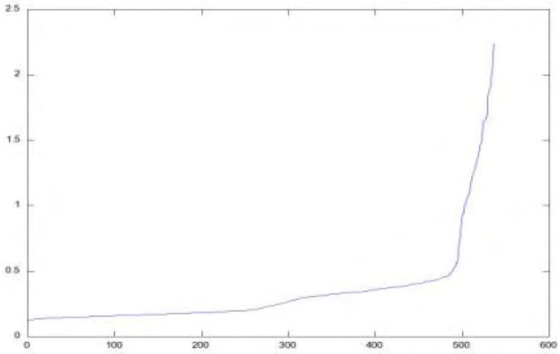
The DBSCAN algorithm is run in multiple steps each time for increasing value of Eps as detected in the previous step, starting with the minimum Eps value. In each step the points which are already clustered are not re-clustered using higher value of Eps. The algorithm reports the clustering as a combined result of each step.

4. EXPERIMENTAL RESULTS

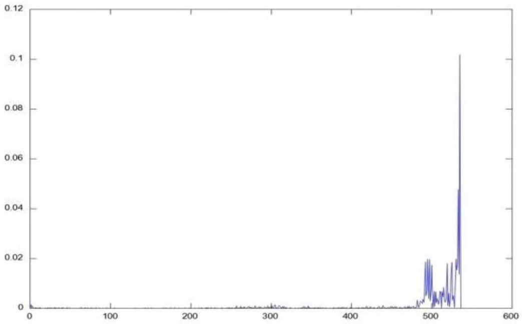
The algorithm is validated on various datasets the following experimental data shows the execution of various stages of the algorithm on one of the datasets. The dataset has multiple point processes giving rise to various densities.



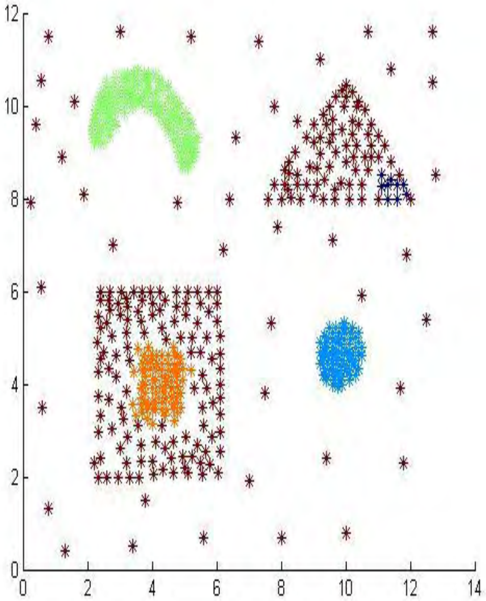
Sample Dataset



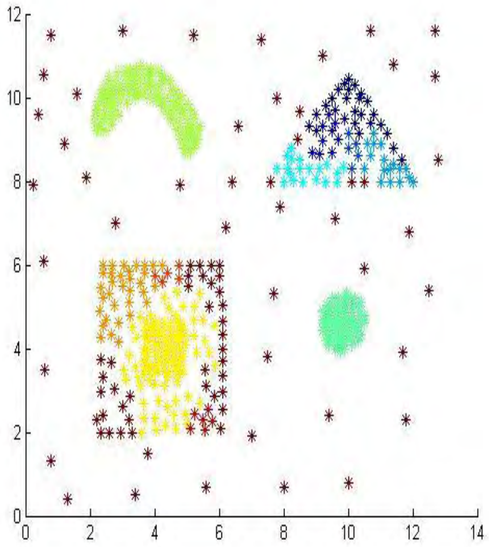
Knn plot of the Dataset



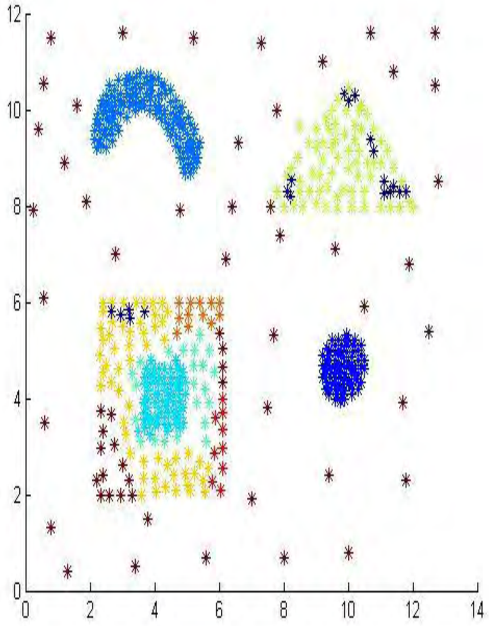
2nd order differential filter output



DBSCAN output with Eps=0.3038



DBSCAN output with Eps=0.4226



Clustering result with our algorithm

The result clearly indicates better clustering than each of the separate Eps value clustering. The lower value of Eps(0.3038) identifies only the high density regions clearly and fails to cluster the low density regions and identifies them as noise. The higher value of Eps(0.4226) breaks one of the higher density clusters(the nested square, indicated in yellow).The result for our algorithm preserves the high density clusters while giving satisfactory results for low density regions and identifying noise correctly.

5. REFERENCES

1. Pattern Classification . Richard O. Duda , Peter E. Heart,David G. Stork..

2. Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu (1996). "A density-based algorithm for discovering clusters in large spatial databases with noise". in Evangelos Simoudis, Jiawei Han, Usama M. Fayyad. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96). AAAI Press. pp. 226–231. ISBN 1-57735-004-9.

3. Sander, Jörg; Ester, Martin; Kriegel, Hans-Peter; Xu, Xiaowei (1998). Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications. Berlin: Springer-Verlag. pp. 169–194. doi:10.1023/A:1009745219419

4. Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel, Jörg Sander (1999). "OPTICS: Ordering Points To Identify the Clustering Structure". ACM SIGMOD international conference on Management of data. ACM Press. pp. 49–60.

5. DECODE: a new method for discovering clusters of different densities in spatial data,Tao Pei, Ajay Jasra, David J. Hand, A. -Xing Zhu, Chenghu Zhou,June 2009 ,Data Mining and Knowledge Discovery , Volume 18 Issue 3

6. An Introductory tutorial on Kd-Tree,Andrew W. Moore,Carnegie Mellon University .

7. MacQueen, J. B. (1967). "Some Methods for classification and Analysis of Multivariate Observations". Proceedings of 5th Berkeley Symposium on Mathematical Statistics and

Probability.

8. Davies, D. L.; Bouldin, D. W. A cluster separation measure.IEEE Trans. Pattern Anal. Mach. Intelligence 1979, 1, 224–227.

9. G. Sheikholeslami, S.chatterjee,A.whang : "Wave cluster : A multi resolution clustering approach for very large spatial databases " .Proc. 24th int. Conf. on very large databases,NewYork,1998. pp 428-439.