A Novel Method to Find Appropriate ϵ for DBSCAN

Jamshid Esmaelnejad[1,2], Jafar Habibi[1,3], and Soheil Hassas Yeganeh[1,4]

[1] Computer Engineering Department, Sharif University of Technology, Tehran, Iran
[2] esmaelnezhad@ce.sharif.edu
[3] jhabibi@sharif.edu
[4] hassas@ce.sharif.edu

Abstract. Clustering is one of the most useful methods of data mining, in which a set of real or abstract objects are categorized into clusters. The DBSCAN clustering method, one of the most famous density based clustering methods, categorizes points in dense areas into same clusters. In DBSCAN a point is said to be dense if the ϵ-radius circular area around it contains at least $MinPts$ points. To find such dense areas, region queries are fired. Two points are defined as density connected if the distance between them is less than ϵ and at least one of them is dense. Finally, density connected parts of the data set extracted as clusters. The significant issue of such a method is that its parameters (ϵ and $MinPts$) are very hard for a user to guess. So, it is better to remove them or to replace them with some other parameters that are simpler to estimate. In this paper, we have focused on the DBSCAN algorithm, tried to remove the ϵ and replace it with another parameter named ρ (Noise ratio of the data set). Using this method will not reduce the number of parameters but the ρ parameter is usually much more simpler to set than the ϵ. Even in some applications the user knows the noise ratio of the data set in advance. Being a relative (not absolute) measure is another advantage of ρ over ϵ. We have also proposed a novel visualization technique that may help users to set the ϵ value interactively. Also experimental results have been represented to show that our algorithm gets almost similar results to the original DBSCAN with ϵ set to an appropriate value.

Keywords: Data Mining, Clustering, Density Based Clustering, Parameter Estimation.

1 Introduction

Clustering is the process of categorizing a set of data into groups of similar objects. A wide range of algorithms have been proposed for clustering. One of well known clustering algorithms is *DBSCAN* [1] which categorizes objects according to a connectivity mechanism, based on density around a data point. The main idea of DBSCAN is that any two points are neighbors if and only if they are similar enough and at least the area around one of them is dense. Any connected component of data points forms a cluster, and any point that

N.T. Nguyen, M.T. Le, and J. Świątek (Eds.): ACIIDS 2010, Part I, LNAI 5990, pp. 93–102, 2010.

does not have any dense neighbor is marked as noise. So according to the given definition, DBSCAN needs two parameters to extract neighborhood relations: First, The threshold of distance or ϵ: Any two points with distance less than ϵ are considered neighbors and second, the threshold of density or $MinPts$: Any point with more than $MinPts$ neighbors are marked as dense.In other words, if the distance between two points is less than ϵ and at least one of them has $MinPts$ points around, they are neighbors.

The problem. The main advantage of DBSCAN is its rare capability of finding clusters with either concave or convex shapes. But, the problem with DBSCAN is that the quality of the clustering results strongly depends on the values of its parameters. Setting appropriate values for these parameters is too hard and time consuming, and may not be feasible without *prior in-depth knowledge* about the data set. Removing each of those parameters or replacing them with another parameter which is simpler to estimate will increase the usability of the algorithm.

Our contribution. We have proposed a heuristic method to find an appropriate value of ϵ to get better results from the DBSCAN. In our method, ϵ is estimated based on ρ, the noise ratio of data set and $MinPts$, the other parameter of DBSCAN. Using noise ratio has three advantages over using ϵ:

1. Noise ratio is much more easier than ϵ to be estimated by the user.
2. Noise ratio is not dependent on the distance function used in the clustering.
3. In some applications the user knows or can easily compute the noise ratio of his data set.

Drawbacks. There are two main drawbacks in our method. The first one is the fact that, although, noise ratio is an easily identifiable and understandable parameter but yet, we could not absolutely omit epsilon because we replaced it with the noise ratio and the second one is that this method is highly dependent of the value of $MinPts$ which is itself a parameter.

2 Related Works

This paper mainly focuses on parameter estimation for DBSCAN. As mentioned before DBSCAN is one of the well known density based clustering algorithms that forms a basis for many other clustering algorithms.

2.1 Density Based Clustering

There are a wide variety of density based clustering algorithms proposed, but the main idea of those algorithms is to find the clusters based on the density of regions in a data set. The density of data is the ratio of the number of data in each region to the volume of data in that region. The final clusters of such clustering techniques will be continuously dense regions of the data set. Using

this heuristic makes density based clustering methods capable of finding clusters with non-convex shapes. On the other hand, any other clustering method that does not use this mechanism is only capable of finding convex shaped clusters (e.g. K-Means[2], C-Means[3], or AGNES[4]). In the following, we will give a brief description for DBSCAN [1] which is a well known example of density based clustering methods. The detailed description of DBSCAN and other density based clustering methods like OPTICS [5] and DENCLUE [6] and a comparison between them can be found in the extended version of this paper.

The DBSCAN [1], the main focus of this paper, defines density of the region around a data point as the number of points in a neighborhood, N_ϵ (a circle with constant radius named ϵ). To be tolerant to noise, the region is mentioned as dense if this number is greater than a threshold, $MinPts$. Then connected dense regions are called clusters. The time complexity of this method is $O(n^2)$, which can be reduced to $O(n \lg n)$ if a spatial indexing is utilized in the algorithm. Using the DBSCAN, two parameters, ϵ and $MinPts$, must be set at the beginning.

There are also many other density based algorithms like DBRS[7,8], ADBSCAN[9], and Circlusters[10,11] which are all extensions to the DBSCAN.

The main focus of this paper is to remove the ϵ parameter from the DBSACN. So, in the following section we will describe the researches for parameter estimation of this algorithm in detail.

2.2 Parameter Estimation

There are many researches around density based clustering, but there are only one research accomplished about parameter estimation in density based clustering which is the AEC algorithm (Automatic Eps Calculation) [12,13]. The AEC algorithm is an iterative algorithm that in each iteration a set of points is selected randomly. Then it calculates three coefficients: distance between the points, number of points located in a stripe between the points and density of the stripe. Then the algorithm chooses the best possible result, which is the minimal distance between clusters. The calculated result has an inïňĆuence on the sets of points created in the next iteration [12]. The AEC algorithm is only capable of estimating ϵ in simple data sets with small noise ratio, while noisy data sets are common in practice. Also, it needs some other parameters to estimate the ϵ (e.g. the width of the stripe) that are not easier to estimate than the ϵ .And finally, the time complexity of the AEC algorithm is much more than the time complexity of the DBSCAN in practice.

3 Estimating the ϵ

To estimate the value of ϵ, we have to answer an important question: What is the optimum value of the ϵ? A simple naive answer may be that the more DBSCAN succeeds, the more satisfying the ϵ is. But when does DBSCAN carry out its task well? In figure 1, the dense central points(the red ones) are in a cluster, say C, and the points around (the blue ones) are noises around C. Now If one can

set ϵ for this data set to a value so that it holds the following properties, then DBSCAN will put all the red points exclusively in a single cluster just as one may expect from a good clustering algorithm:

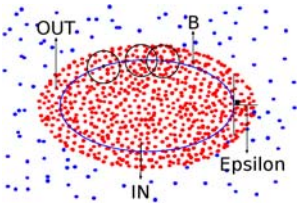
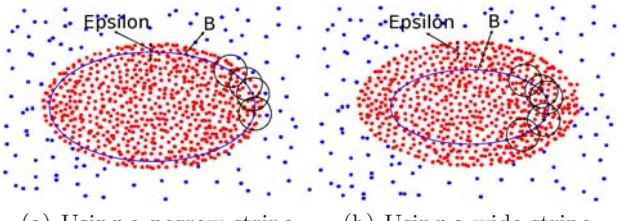


Fig. 1. A propriate ϵ to cover all and only red points

- As shown in figure 1, there exists a closed line like B, such that it separates C's points into two subsets IN and OUT. The maximum perpendicular distance of points in OUT from B is equal to ϵ. In fact, there is a stripe around C with ϵ as its width.
- Running DBSCAN with this $MinPts$ and ϵ, all the points in IN are known as core points.
- For each two point $p, q \in IN$, p is density-reachable from q and vice versa.

As an explanation, second and third properties are needed because DBSCAN should put all red points in figure 1 in a same cluster. As you see in figure 2(a) and figure 2(b), if the stipe's width is less than ϵ, then DBSCAN may include some noises in C and if it is greater than ϵ, DBSCAN may lose some boundary points of C or even split C into two or more clusters.



(a) Using a narrow stripe (b) Using a wide stripe

Fig. 2. The stripe made by B, should not be too much narrow or wide

Now we have introduced the properties of the special ϵ which we are trying to estimate. If we run DBSCAN with an appropriate ϵ value, all the inner-cluster points will be marked as core points. Suppose that E is such a value and we will propose a method to find an ϵ which has the properties given in the previous part.

3.1 Simplifying Assumptions

We have made an assumption about the data set for simplification. Suppose that S is an area from the problem space in which there are no borders of any cluster. We have assumed that the data set has uniform distribution inside the clusters. In other words, the assumption is that the following value does not change as S moves around in the data set space,

$$\xi = \frac{vol(S)}{num(S)}$$

in which, $vol(S)$ means the area of S in two dimensional space and volume in 3 or higher dimensions, and $num(s)$ is the number of points inside S. This assumption may seem far away from reality but the generated results using it are very remarkable.

Now let's get back to our main subject. The goal is to find a value like F which is smaller than E and if we assign it as ϵ, DBSCAN will mark all the points of the clusters as core points except a stripe around each cluster with width of F. It's not hard to conclude the following equality for which you can find a complete proof in the extended version of this paper.

$$F = \sqrt[k]{\frac{1}{2}}E \tag{1}$$

In which, k is the number of dimensions of the data set and $C(k)$ is the coefficient which is used in the computation formula of $vol(S)$.

3.2 A Useful Visualization for Finding ϵ Value

It is really hard to decide what is the optimum value of ϵ in a data set. Some visualization techniques can be employed to make the decision easier. In this section, we will introduce a chart which helps the user to decide on the appropriate ϵ value. The larger the ϵ is, the more core points we will have. So, for each point in the data set and a specified $MinPts$ value, there exists a minimum ϵ value that makes the point, a core point. We call such a value $MinDist$ and it is formally defined in Definition 1.

Definition 1 (MinDist). *For any point p in the data set $MinDist$ is defined as follows*

$$MinDist(p, MinPts) = \min\{\epsilon \in \mathbb{R} | \epsilon > 0 \text{ and } |N_\epsilon(p)| \geq MinPts\}$$

To sketch a histogram based on $MinDist$ of the points, we used rounding. Suppose that $N(m, k)$ is the number of nodes that has $[MinDist]_k = m$, in which $[x]_m$ means rounding number x to m digits of precision. Figure 3 shows the N function for data set 8.8 borrowed from a data set named chameleon [14] with $MinPts = 6$. The horizontal axis corresponds to different values of m for which

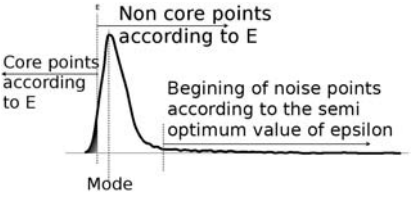


Fig. 3. The proposed visualization for data set 8.8 with $MinPts = 6$

there exists at least one point p in data set with $MinDist(p, 3) = m$. The vertical axis corresponds to the N value related to a special m. It is important to note that this continuous chart is created by connecting the consecutive points of the N function. There are some important notes about this figure:

- The area under the chart equals to the size of the data set. Note that this area is, in fact, the sum of all points with $[MinDist(p)]_3 = X$ (for every X). It can also be concluded that the hatched area in figure 3 is equal to the number of points with $[MinDist(p)]_3 \leq E$.
- It is obvious that the value of ϵ can be chosen from $[0, +\infty]$, but too small or too large ϵ's are not useful. So, the interval in which we decide on ϵ is restricted to $[M_{min}, M_{max}]$ where $M_{min} = min\{MinDist(p)|p \in DB\}$ $M_{max} = Max\{MinDist(p)|p \in DB\}$.Also, an explanation on the fact that we don't lose any useful ϵ is brought in the extended version.

And there are also some other important notes about this chart in the extended version of the paper.

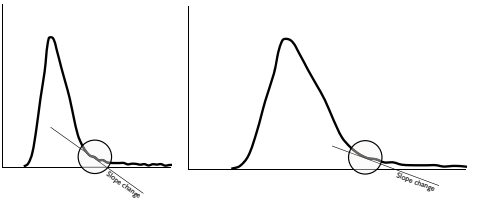
3.3 Using the Chart to Set ϵ

Now, the problem is to find the proper ϵ value, or in other words to find the place of ϵ on the $x-$axis. In this section we will present the main heuristic to set the value of ϵ. Too small ϵ values will mark some non-noise points as noise and too large value will mark some noise points as cores. Choosing an ϵ value in an area shown by disk in figure 4(a) seems a good heuristic. As you see, here is the place in which the absolute value of the slope of chart and so the density of area around the points is decreasing very quickly.

As shown in figure 4(b), stretching the chart will cause changes in its slope. So this simple geometric heuristic, or any other simple heuristic dependent on the slope of the chart will not help in all cases. Generally, we could not get any satisfying result from geometrical approaches.

Now we present a novel heuristic idea which has produced very satisfying results. Suppose that we have the noise ratio of our data set: ρ $(0 \leq \rho \leq 1)$. So we just have to set the value to somewhere in the x-axis of the chart so that:

$$\rho \simeq \frac{The\ area\ under\ the\ chart\ to\ right\ of\ X}{The\ whole\ area\ under\ the\ chart}$$



(a) The appropriate ϵ should be somewhere near or in the disk

(b) The angles change in the stretched version of the chart

Fig. 4. Heuristic based on the curve of the chart will not work

Then :

$$\rho \times n \simeq number\ of\ noise\ points$$

where n is the size of data set.

Now if we assume that $MinDist(p) >= MinDist(q)$ for any pair of points (p, q) that p is noise and q is not, the points to the right of the chosen x value are exactly the noise points. And finally, from equation 1 we have $\epsilon = \frac{1}{\sqrt{2}}x$. So, we can estimate the ϵ value from the noise ratio, ρ.

4 Experimental Results

Using the heuristic described in previous section, we can use ρ instead of ϵ. In the followings, we have run DBSCAN three times for three different data sets. And for each data set, two times with different $MinPts$s. All this three data sets are brought from a data set set named Chameleon [14].

We have represented three figures for each execution of the DBSCAN. In each set, the left figure, shows the introduced chart and the vertical line determines the generated ϵ. The centered figure is the clustering result of DBSCAN execution, each color representing a cluster. The right figure which is the same data set representation while we have eliminated the noise points is used to show the noise removal capability of our method.

Figure 5 shows the results related to a data set named 7.10 setting $MinPts$ to 12 and 30. As you see, while we have a convincing result in figure 5(b), choosing a high $MinPts$ value, (30), results into figure 5(f) that has some eliminated points which are not noise. Note that $MinPts$ is also a very effective parameter for DBSCAN.

Figure 6 is related to a data set named 4.8. With $MinPts$ values of 10 and 15. As you see DBSCAN have determined all the clusters perfectly. But, in figure 6(b), in addition to the main clusters, you can find three new small clusters

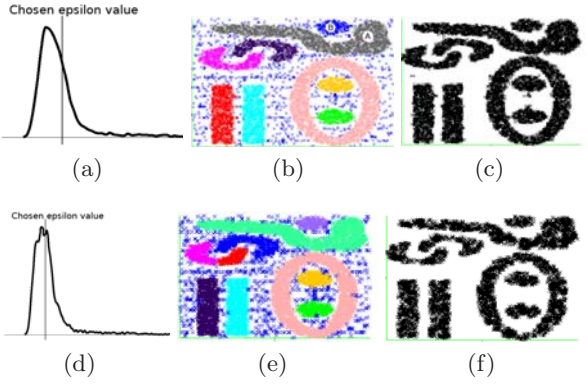


Fig. 5. Results for data set 7.10 with $MinPts = 12$ and 30, $\rho = 0.10$ and the resulting $\epsilon = 0.0162$ and 0.0233

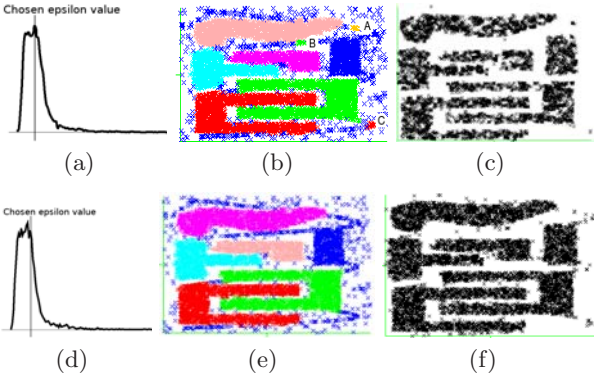


Fig. 6. Results for data set 4.8 with $MinPts = 10$ and 15, $\rho = 0.10$ and the resulting $\epsilon = 0.0176$ and 0.0219

(a yellow, a green and a red one which are labeled with letters A, B and C respectively). In figure 6(c) there are some non-noise eliminated points. This time, the problem is due to the smallness of $MinPts$ parameter because as you see in figure 6(e), the problem is solved.

Finally, figure 7 shows the results for data set 8.8 with $MinPts$ values of 6 and 10. There are two huge clusters lying in left down side of the data set in figure 7(b) labeled with letters A and B. The points at the top of this two clusters are not dense enough to call them a cluster and not so sparse that we call them noise. This problem (having clusters with different densities in the data set) is in fact a basic problem of DBSCAN. Actually, we could not find any better results for this data set than figure 7(e).

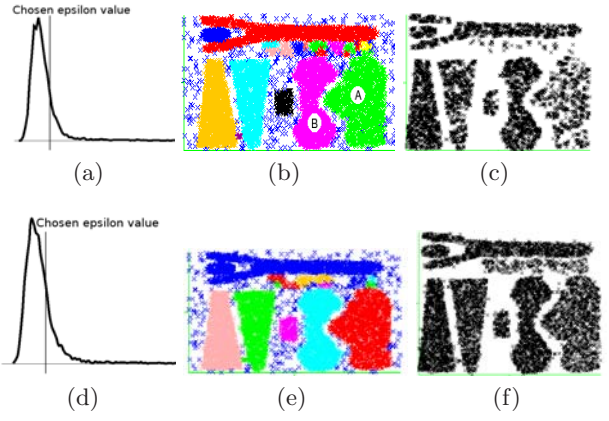


Fig. 7. Results for data set 8.8 with $MinPts = 6$ and $10, \rho = 0.05$ and the resulting $\epsilon = 0.0160$ and 0.205 respectively

5 Conclusions and Future Works

DBSCAN, the most famous density based clustering method, requires two parameters ϵ and $MinPts$ to be set for clustering.Finding an appropriate ϵ value for a data set is a tedious and time consuming task. In this paper, we have proposed an estimation method based on noise ratio of the data set, ρ. Using noise ratio is much more simpler than the ϵ because it is relative, more probable to be known in advance, and also easier to estimate. The main idea of the estimation heuristic method is based on visualization technique that can also be used to estimate the ϵ value interactively. As shown in the experimental results, our method is capable of estimating a proper ϵ value for real and complex data sets. Also it has been shown that the algorithm has wonderful density based noise removal capability, that can be used to cleanse the data for future usage.

There are some issue that can be addressed as future works. First of all, it is important to propose a method to estimate the optimum value of $MinPts$ in a data set. To estimate $MinPts$ the first step is to accurately define a satisfying clustering;The next step then, may be testing all $MinPts$ values and choosing the $MinPts$ value which results into the best clustering result.

References

1. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Simoudis, E., Han, J., Fayyad, U. (eds.) Second International Conference on Knowledge Discovery and Data Mining, Portland, Oregon, pp. 226–231. AAAI Press, Menlo Park (1996)
2. MacQueen, J.B.: Some methods for classification and analysis ofmultivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematics, Statistics and Probabilities, vol. 1, pp. 281–297 (1967)

3. Dunn, J.C.: A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. Journal of Cybernetics and Systems 3(3), 32–57 (1973)
4. Han, J., Kamber, M.: Data Mining: Concepts and Techniques, 2nd edn. The Morgan Kaufmann Series in DataManagement Systems. Morgan Kaufmann, San Francisco (2006)
5. Ankerst, M., Breunig, M.M., Kriegel, H.P., Sander, J.: Optics: ordering points to identify the clustering structure. In: Proceedings of 1999 ACM International Conference on Management of Data (SIGMOD 1999), vol. 28, pp. 49–60. ACM, New York (1999)
6. Hinneburg, A., Keim, D.A.: An efficient approach to clustering in large multimedia databases with noise. In: Knowledge Discovery and Data Mining, pp. 58–65 (1998)
7. Wang, X., Hamilton, H.J.: Dbrs: A density-based spatial clustering method with random sampling. In: Proceedings of the 7th PAKDD, Seoul, Korea, pp. 563–575 (2003)
8. Wang, X., Rostoker, C., Hamilton, H.J.: Density-Based Spatial Clustering in the Presence of Obstacles and Facilitators. In: Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D. (eds.) PKDD 2004. LNCS (LNAI), vol. 3202, pp. 446–458. Springer, Heidelberg (2004)
9. Yeganeh, S.H., Habibi, J., Abolhassani, H., Tehrani, M.A., Esmaelnezhad, J.: An approximation algorithm for finding skeletal points for density based clustering approaches. In: Proceedings of the IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2009, part of the IEEE Symposium Series on Computational Intelligence 2009, March 2009, pp. 403–410. IEEE, Los Alamitos (2009)
10. Yeganeh, S.H., Habibi, J., Abolhassani, H., Shirali-Shahreza, S.: A novel clustering algorithm based on circlusters to find arbitrary shaped clusters. In: International Conference on Computer and Electrical Engineering, pp. 619–624. IEEE Computer Society, Los Alamitos (2008)
11. Shirali-Shahreza, S., Hassas-Yeganeh, S., Abolhassani, H., Habibi, J.: Circluster: Storing cluster shapes for clustering. To appear in the Proceedings of the 4th IEEE International Conference on Intelligent Systems, Varna, Bulgaria (September 2008)
12. Gorawski, M., Malczok, R.: AEC Algorithm: A Heuristic Approach to Calculating Density-Based Clustering Eps Parameter. In: Yakhno, T., Neuhold, E.J. (eds.) ADVIS 2006. LNCS, vol. 4243, pp. 90–99. Springer, Heidelberg (2006)
13. Gorawski, M., Malczok, R.: Towards Automatic Eps Calculation in Density-Based Clustering. In: Manolopoulos, Y., Pokorný, J., Sellis, T.K. (eds.) ADBIS 2006. LNCS, vol. 4152, pp. 313–328. Springer, Heidelberg (2006)
14. Karypis, G.: Chameleon data set (2008), http://glaros.dtc.umn.edu/gkhome/cluto/cluto/download