Adaptive DBSCAN Algorithm Based on Sample Density Gradient

Ningwei Xie*, Z Miao, J Wang and Q Zhang
Beijing Jiaotong University, No.3 Shangyuancun, Haidian District, Beijing, 100044, P. R. China
*Email: 18120323@bjtu.edu.cn

Abstract. DBSCAN is a classic and commonly applied density-based clustering algorithm, but its clustering accuracy depends on the choice of two input parameters. This paper presents a new algorithm for adaptive parameter determination in DBSCAN. The assumption of this new algorithm is that regions with larger sample density gradient usually correspond to the edge areas of clusters. The main idea is to generate some pre-clusters and determine the values of parameters by their statistics information. We first randomly pick pairs of points in the sample space to form circles, which are called “disks”. Then we estimate the density information of a disk by random sampling, and define the criteria of disk quality to select disks with larger sample density gradient. Finally, we obtain the suitable parameters of DBSCAN in terms of the distributions of radius and points number of these extracted disks by Gaussian kernel density estimation. Experimental results show that the new algorithm improves the accuracy of DBSCAN and performs better than classic algorithms like k-means and Birch in some cases.

1. Introduction
Typical representatives of clustering methods are distance-based clustering algorithms, hierarchical clustering algorithm, and density-based clustering algorithms, etc [1]. K-means, independently discovered in different scientific fields by multiple researchers, is by far the most popular clustering tool used in various applications [2]. In k-means, clusters are represented by their center points (centroid) that minimize the mean squared distance from each data point to its nearest centre [3]. The Birch (Balanced Iterative Reducing and Clustering using Hierarchies) designed by T. Zhang [4], performs efficient clustering for huge dataset with multiple classes [5].

The algorithm DBSCAN (Density-Based Spatial Clustering of Applications with Noise) was proposed by Ester et al. [6]. It classifies data points by expanding and aggregating adjacent areas with similar density, efficiently handles irregular and abnormal data. But its performance highly depends on the manual selection of two input parameters $Eps$ and $minPts$. Several papers have already put forward parameter selection methods. In [6], the value of $Eps$ is determined by user’s observation of the $k$-dist graph, but $minPts$ need to be specified. [7] assumes $minPts = 4$ and introduces the consideration of distance distribution to estimate $Eps$, but number of clusters is required. [8] presents a new method of data partition based on $k$-dist graph to determine $Eps$, but $minPts$ still need to be set by users. [9] presents the recursive density-based clustering algorithm which modifies dynamically the parameters of DBSCAN, however requiring manual initialization.
In view of this, this paper proposes a new adaptive algorithm based on sample density gradient for better parameter selection in DBSCAN. We automatically select parameters by statistics information of pre-clusters, which are extracted by the criteria of disk quality, defined according to the concept of sample density gradient. Given that the pre-clusters are located in clusters area and their radius and sample number can reflect the data structure, we can finally choose suitable values for Eps and minPts by analysing the two features using Gaussian kernel density estimation. Compared with other existed parameter selection methods, the new algorithm can adaptively give the values of two parameters at the same time without any prior knowledge of the dataset and improve the clustering accuracy of DBSCAN.

2. Related technologies

2.1. DBSCAN Algorithm
In density-based clustering algorithms, a cluster is considered as a dense region of objects and an object of one cluster must be surrounded by objects of the same cluster. The main idea of DBSCAN is to partition density-connected objects into the same cluster [10].

Some basic concepts of the algorithm DBSCAN are listed below.

- Eps-neighbourhood: a d dimensional hypersphere region with a given object p as its core and Eps as its radius.
- Core points: if the Eps-neighbourhood of the given object p contains more than minPts data points, p is called a core point.
- Directly density-reachable: given the values of Eps and minPts, if there is an object p belongs to the Eps-neighbourhood of a core point q, then p is directly density-reachable from q.

DBSCAN checks the Eps-neighbourhood of each data point in D to search the core point of clusters, iteratively aggregate objects that are directly reachable from these core objects and finally output the collection of clusters, when no new points are added to any cluster.

2.2. Gaussian kernel density estimation
Kernel density estimation is a nonparametric method to estimate unknown density functions in probability theory, proposed by Rosenblatt [11] and Emanuel Parzen [12]. It studies the characteristics of data distribution directly from samples, independent with any prior knowledge.

If \( x_1, x_2, ..., x_n \) are independently and identically distributed random variables on the d dimensional space \( \mathbb{R}^d \), the probability distribution density function \( f(x) \) is estimated as below:

\[
\hat{f}_h(x) = \frac{1}{n} \sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right)
\]

(1)

\( h \) is the bandwidth (or smooth parameter), \( n \) represents the number of samples, \( K(\bullet) \) is the kernel function. In subsequent approach, Gaussian kernel function is applied to fit the probability density functions of radius and sample number of pre-clusters:

\[
K(x - x_i) = \exp(-\|x - x_i\|^2/2\sigma^2)
\]

(2)

3. Adaptive DBSCAN algorithm based on sample density gradient
The new algorithm consists of four parts: the generation of disks, the estimation of density information in disks, the good disk selection and the parameter determination by Gaussian kernel density estimation.

3.1. Generation of disks
We randomly pick two data points \( x \) and \( y \) in data space, form a circle with the segment \( xy \) as diameter, and note the center and the radius of circle as \( g \) and \( r \). We form a concentric circle with radius \( R = \sqrt{2}r \) (\( \sqrt{2} \) is a fixed constant). These two circles are called a "disk", which divide the data
space into three parts: the area within the inner circle, denoted as \( IN \); the area between the inner and outer circle, denoted as \( BETWEEN \); the area out of the outer circle, denoted \( OUT \). Figure 1 shows the diagram of a disk.

![Figure 1. Definition of a “disk”](image)

A hyper-parameter \( r_{gen} \) is introduced to control the ratio of the number of generated disks to the samples number in the dataset and is usually set to be 4.

### 3.2. Estimation of the density information in disks

After randomly generating enough disks, we need to obtain their density information by following steps:

- **Generate a disk.**
- **Sample data objects** \( S_{total} \) times in the entire dataset, randomly taking \( C \) points at a time. Where \( S_{total} \) and \( C \) are hyper-parameters.
- **During each time of sampling**, the sample numbers in the two areas \( IN \) and \( BETWEEN \), are denoted as \( n_{in} \) and \( n_{between} \).
- **After completing all samplings**, sum up cumulative sample numbers in \( IN \) and \( BETWEEN \), denoted as \( N_{in} \) and \( N_{between} \). The number of samplings satisfying \( n_{between} = 0 \) is denoted as \( S_{between} \).

We define two disk features as follow.

\[
q = \frac{S_{between}}{S_{total}}
\]

\[
R_{between,in} = \begin{cases} 
\frac{N_{between}}{N_{in}} & \text{if } N_{in} \neq 0 \\
0 & \text{if } N_{in} = 0 
\end{cases}
\]

### 3.3. Good disk selection

In density-based clustering algorithms, the density is usually considered as the number of objects contained in unit area. In light of this definition, we define the sample density gradient in a disk \( \nabla D_{disk} \) as the difference of sample numbers between \( IN \) and \( BETWEEN \) (\( IN \) and \( BETWEEN \) are equal in area, their sample numbers correspond to the density of the two areas).

\[
\nabla D_{disk} = N_{in} - N_{between}
\]

When a disk is located in centre area of a cluster, its \( IN \) and \( BETWEEN \) areas are similar in density, \( \nabla D_{disk} \) is relatively small. When the disk gradually moves to the edge area of the cluster, the difference of density in \( IN \) and \( BETWEEN \) increases, \( \nabla D_{disk} \) gets larger. When the disk slips out of the cluster, both of the two areas fall into the sparse region of points, \( \nabla D_{disk} \) decreases. Therefore, disks with larger sample density gradient theoretically correspond to those in edge area of the clusters. Since it is difficult to use the definition of sample density gradient to select the disk (the thresholds are difficult to determine), we design the criteria of disk quality, consisting of two conditions, to find good disks.

- **Condition 1**: \( q \geq q_{min} \).
- **Condition 2**: \( R_{between,in} \geq R_{min} \).
Where hyper-parameters $q_{min}$ and $R_{min}$ are the criteria thresholds. In practice, their optimal values are $q_{min} = 0.5$, and $R_{min} = 0.4$.

![Figure 2. Possible forms of disks satisfied Condition 1.](image)

If a disk satisfies Condition 1, the probability of extracting samples in BETWEEN area is relatively small. This comforts not only to the case of disks located at edge of clusters, showed in figure 2 (a) and (b), but also to the case of small disks outside clusters, shown in figure 2 (c) and large disks wrapping entire clusters, showed in figure 2 (d). Thus, we add Condition 2, in which selecting disks with lager $R_{BETWEEN,IN}$ can eliminate the disks with sudden drop of density from IN to BETWEEN. Besides, small disks outside clusters formed by adjacent outliers are eliminated for their $R_{BETWEEN,IN} = 0$.

After this step, the retained disks are considered as the pre-clusters, whose statistics information is analyzed in next step to determine the values of $Eps$ and minPts.

3.4. Parameter determination by Gaussian kernel density estimation

After disk selection, we focus on the outer radius and the sample number of the pre-clusters. These two features of disk can reflect the density characteristics of the clusters and adapt to different data structures. Firstly, the radius of selected disks varies with the size of cluster. Figure 3 (a) illustrates that the average of disk radius is proportional to the size of a circle cluster. Secondly, the radius of selected disks can adapt to the shape and the density of cluster. Figure 3 (b) shows an irregular cluster consisting three parts of different shapes. In figure 3 (c), the blue, green and red lines indicate the variation of the average disk radius along with point density in rectangle, triangle and circle part of the cluster respectively.

![Figure 3. The radius of pre-clusters varies along with size, density and shape of clusters.](image)

Thus, we estimate the probability density functions of outer radius $R$ and sample numbers $P = N_{in} + N_{between}$ of pre-clusters, denoted as $f_{R}(x)$ and $f_{P}(x)$. The values of $Eps$ and minPts are set as bellow.

$$Eps = \arg \max_x f_{R}(x)$$  \hspace{1cm} (6)

$$\text{minPts} = \arg \max_x f_{P}(x)$$  \hspace{1cm} (7)
4. Experiments and results
We carry out experiments on four synthetic datasets and compared with three related clustering algorithms (k-means [2], DBSCAN [6] and Birch [4]). SD1, SD2, SD3 are two-dimensional datasets containing 1860, 2078 and 1530 objects respectively. Iris is a four-dimensional dataset containing 150 objects of 3 classes. The F-Measure method [13] is applied to measure the clustering accuracy.

4.1. Clustering Visualization of our algorithm
Figure 4 (a), (c) and (e) show the extracted pre-clusters for the two-dimensional datasets SD1, SD2 and SD3 respectively. Where the objects are marked by blue points, the disks are represented by black circles. Figure 4 (b), (d) and (f) show the results of clustering using the adaptive algorithm. Where different clusters are marked in different colors, and outliers are marked in gray.

Figure 4 indicate that the adaptive algorithm is able to find high-density regions in the dataset by pre-clusters, distinguish the outliers and make appropriate cluster partitioning. In particular, the adaptative algorithm performs well for the concentric clusters and the clusters with different density.

4.2. Comparison test results
The parameters of traditional DBSCAN are taken by using the method proposed in paper [6]. We carry out 20 tests for each algorithm, and choose the average accuracy (table 1).

| Dataset | Dimension | Sample Size | Clustering Algorithm   | Parameters | Accuracy |
|---------|-----------|-------------|------------------------|------------|----------|
|         |           |             |                        | Eps | minPts |
| SD1     | 2         | 1860        | Adaptive-DBSCAN        | 0.3366 | 6       | 98.50%   |
|         |           |             | DBSCAN                 | 0.2533 | 4       | 98.50%   |
|         |           |             | K-Means                | 0.2533 | 6       | 98.50%   |
|         |           |             | Birch                  | 0.2533 | 6       | 98.50%   |
| SD2     | 2         | 2078        | Adaptive-DBSCAN        | 0.5154 | 7       | 98.78%   |
|         |           |             | DBSCAN                 | 0.2413 | 4       | 98.65%   |
|         |           |             | K-Means                | 0.2413 | 4       | 98.65%   |
|         |           |             | Birch                  | 0.2413 | 4       | 98.65%   |
| SD3     | 2         | 1530        | Adaptive-DBSCAN        | 0.4465 | 6       | 97.04%   |
|         |           |             | DBSCAN                 | 0.4465 | 6       | 97.04%   |
|         |           |             | K-Means                | 0.4465 | 6       | 97.04%   |
|         |           |             | Birch                  | 0.4465 | 6       | 97.04%   |
In terms of accuracy, both the adaptative algorithm and the traditional DBSCAN algorithm perform well when processing datasets (SD2 and Iris) containing hyper spherical clusters or datasets of arbitrary shape clusters (SD3). In high-dimensional datasets (Iris), the distribution of objects is more random, therefore the differences in Euclidean distance between objects become insignificant. As a result, the accuracy of all algorithms decreases. Compared with the traditional DBSCAN, the adaptative algorithm can choose better parameters in dealing with more complex datasets.

We propose a new algorithm for adaptively determining the two parameters of DBSCAN. No prior knowledge of datasets is required, the new algorithm extracts density information in regions with large density gradient, choose appropriately the parameters by combining Gaussian kernel density estimation, and finally acquires higher average accuracy than traditional DBSCAN. The future direction of research is to reduce the time complexity as well as to improve the performance in high-dimensional datasets.

**Acknowledgments**
This work is supported by the NSFC 61672089, 61273274, 61572064, and National Key Technology R&D Program of China 2012BAH01F03.

**References**
[1] Jain A K, Murty M N and Flynn P J 1999 Data clustering: a review ACM Computing Surveys 31 3 p 264
[2] Jain A K 2010 Data Clustering: 50 Years Beyond K-means Pattern Recognition Lett. 31 8 p651
[3] Kanungo T, Mount D M, Netanyahu N S, Piatko C D, Silverman R and Wu A Y 2002 An efficient k-means clustering algorithm: analysis and implementation IEEE Transactions on Pattern Analysis and Machine Intelligence 24 7 p 0
[4] Zhang T, Ramakrishnan, R and Livny M 1996 BIRCH: an efficient data clustering method for very large databases ACM SIGMOD Int. Conf. on Management of Data p 103
[5] Du H Z and Li Y B 2011 An Improved BIRCH Clustering Algorithm and Application in Thermal Power Int. Conf. on Web Information Systems & Mining IEEE
[6] Ester M, Kriegel H P and Xu X 1996 A density-based algorithm for discovering clusters in large spatial databases with noise Int. Conf. on Knowledge Discovery and Data Mining (AAAI Press) p 226
[7] S Yue, P Li, J Guo and S Z 2005 A statistical information-based clustering approach in distance space J. of Zhejiang University: Science 6 1 p 71
[8] Y Yu, A Zhou. 2011 An improved algorithm of DBSCAN Computer Technology and Development 21 2 p 30
[9] Z S, Q Y 2002 Web files clustering based on Web-Log Mining J. of Software 13 1 p 99
[10] Rai, Pradeep, Singh and Shubha 2011 A survey of clustering techniques *Int. J. of Computer App.* **7** 12 p 1
[11] M Rosenblatt 1956 Remarks on some nonparametric estimates of a density function *Annals of Mathematical Statistics* **27** 6 p 832
[12] E Parzen 1962 On estimation of a probability density function and mode *Annals of Mathematical Statistics* **33** 8 p 1065
[13] Hoff T P V and Lindgren A G 2002 Adaptive step-size control in blind source separation *Neurocomputing* p 119