Boundary Detecting Algorithm for Each Cluster based on DBSCAN

Yarui Guo[1,a],Jingzhe Wang[1,b], Kun Wang[1,c]

[1]School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China.

[a]1026901536@qq.com, [b]498155639@qq.com, [c]kunw2012@163.com

Keywords: Clustering; Cluster numbers; Boundary; Point density; Border degree

Abstract. Detecting Detecting the boundary of each cluster in a data set is a tough problem for many existed boundary detecting algorithms. In order to solve that problem, a clustering boundary detecting algorithm based on DBSCAN named BDAEC(Boundary Detecting Algorithm for Each Cluster based on DBSCAN: BDAEC) is proposed . Firstly, according to the core point percent and the density value of each data object, all the core points are extracted by this algorithm from the data set. Then, many connected undirected graphs will be constituted by these core points. And the cluster numbers of the data set can be known by those connected undirected graphs for each one of them represents a cluster. Finally, Eps field will be diveded into two fields: the positive field and the negative field. And the boundary of each cluster or the whole data set can be detected by the distribution characteristics of the data objects which are located in the positive field and negative field of the given data object. The experimental results on many data sets with noise show that BDAEC algorithm can obtain the numbers and the boundaries of the clusters with different size or shapes effectively.

1 Introduction

Advances in information technologies have led to the continual collection and rapid accumulation of data in repositories. Patterns, like cluster[1], classification[2] and outlier analysis[3], are used to find the interesting models from the repositories to help us extract the useful information. Besides, boundary detection[4] is also an emerging pattern which has quickly developed during these years. Boundary detection aims at finding the boundary objects which are located in the edge of the clusters[5]. Compared with the other objects in a cluster, boundary objects have their unique features. For example, the boundary of the patients with benign tumor may means the patients who are easily developed into malignant tumor in the field of medicine. Searching and finding those patients will contribute to the early prevention and diagnosis of malignant tumors.

Now days, methods, like BORDER[6], BERGE[7], have been proposed in the field of boundary detection. BORDER algorithm can receive a preferable result on the data set without noise compared with the data set with noises. On the noisy data set, BORDER will see all the noisy objects as the boundary objects.That is to say: BORDER can not avoid the interfere of the noisy objects. BERGE algorithm can apply to the data sets with noises or without noise, and it will get good results.

Although BORDER and BOUND algorithms can obtain the whole boundary of the data set effectively, they cannot get the boundary of each cluster in the data set. In order to extract the boundary of each cluster in a data set, we propose a clustering boundary detecting algorithm based on DBSCAN[8] to solve this problem.

DBSCAN algorithm is proposed as a clustering method. It brought us the concept of boundary points. It mentions that boundary points which belong to non-core points are located in the Eps-field of core points. Although it is proposed as a clustering method, it can get the boundary of each cluster in a data set through cluster division and boundary points identification. DBSCAN algorithm still has some deficiencies in the field of boundary detection. First, parameter Eps from DBSCAN is used to control the point density, cluster numbers and the thickness of a cluster's boundary. So Eps is hard to take a appropriate value to satisfy them all. Second, it can not control the thickness of the cluster's boundary flexibly. Third, boundary points are recognized by the point's density, this will lead to the points which are located in the inner cluster to be recognized as the boundary points. Been noticed of those, we have improved DBSCAN. And on the basis of DBSCAN, we proposed BDAEC to extract the boundary of each cluster.

The paper is organized as follows: Section 2 introduces BDAEC algorithm in detail. Section 3 compares BDAEC algorithm with other boundary detection algorithms to validate the validity of BDAEC algorithm. Section 4 time complexity analysis of BDAEC algorithm. Section 5 parameter discussion of BDAEC algorithm. Section 6 concludes the paper.

2 BDAEC Algorithm

First of all, we introduce the definition of BDAEC algorithm.

Definition 1 ***Point Density:*** Given a data set *D*, distance metric *M*, $\forall p \in D$, *p*'s point density, denotes as *den(p)*, is the number of objects which located in the Eps-field of p.

$$den(p)=|N_{eps}(p)| \qquad (1)$$

p's Eps-field denotes as $N_{eps}(p)$. $|N_{eps}(p)|$ means the number of objects in p's Eps-field.Here the distance metric *M* is Euclidean Distance[9].

The average point density of the data set is calculated in this way:

$$ave\_den=\frac{1}{n}\sum_{i=1}^{n} den(p_i) \qquad (2)$$

n is the objects number in the data set, p_i is the i-th object.

Definition 2 ***Core Points Set:*** The objects in the data set who's point density is bigger than *cp*ave_den*. Core Point Set is denoted as *core_pts*.

$$core\_pts=\{p \in D \mid den(p) \quad \} \geqq cp*ave\_den \qquad (3)$$

cp represents core point percent. It is used to control the number of core points, and $cp \in [0,1]$.

Definition 3 Mutual *Connected:* $\forall p,q \in D$, if $dist(p,q) \le con\_dist$, then point *p* and point *q* are Mutual Connect. Here *con_dist* is connected radius. It is the ω time of *Eps*. And *con_dist* is calculated as follows:

$$con\_dist = \omega * Eps \qquad (4)$$

Here, *ω* is denoted as connected degree, and $\omega \in (0,1]$

Definition 4 ***Candidate Points Set:*** Candidate Point Sets is consisted of core points set and non-core points set. Candidate Point Sets is denoted as *cdt_pts*.

Non-core Points Set: objects in the data set which is Mutual Connected with any core point. Non-core Points Set is denoted as ncp_pts.

$$ncp\_pts=\{q \in D \mid dist(p,q) \quad \} \leqq con\_dist \qquad (5)$$

And $p \in core\_pts \, \mathrm{I} \;\; q \notin core\_pts$。

$$cdt\_pts = core\_pts \cup ncp\_pts \qquad (6)$$

Definition 5 ***Maximum Density Point:*** $\forall p \in cdt\_pts$, *p*'s Maximum Density Point is the object which belongs to $N_{eps}(p)$ and its Point Density is bigger than any objects else in *p*'s $N_{eps}(p)$.

Definition 6 ***Positive Field*[10] *and Negative Field*[10]**: *q* is *p*'s Maximum Density Point, for $\forall r \in N_{eps}(p) \, \mathrm{I} \;\; r \neq p$, *p*'s Positive Field, denotes as $P_{eps}(p)$, is defined as follows:

$$P_{eps}(p)=\{r \mid 0 \quad \leqq cos(pq,pr) \leqq 1, r \in cdt\_pts\} \qquad (7)$$

p's Negative Field, denotes as $N_{eps}(p)$, is defined as follows:

$$F_{eps}(p)=\{r \mid -1 \quad \leqq cos(pq,pr) < 0, r \in cdt\_pts\} \cup \{p\} \qquad (8)$$

Definition 7 ***Border Degree***: Border Degree represents an object's degree of been a boundary point. $\forall p \in cdt\_pts$, *p*'s border degree, denoted as *BD(p)*, is defined as follows:.

$$BD_p = |P_{eps}(p)| \,/\, |F_{eps}(p)| \qquad (9)$$

And $|P_{eps}(p)|$ or $|F_{eps}(p)|$ represents the number of objects in *p*'s Positive Field or Negative Field.

Definition 8 ***Border***: the boundary of the whole data set.

$$Border=bor(C_1) \cup bor(C_2) \cup ...... \cup bor(C_K) \qquad (10)$$

k represents the number of the clusters in a data set. C_i is the ith cluster. $bor(C_i)$ represents the boundary set of the ith cluster. Each object from the cluster C_i has its border degree value. $bor(C_i)$ has *RANK* objects, and it is comprised by the objects which have the largest *RANK* border degree values.

$$RANK = bp*|C_i| \qquad (11)$$

bp represents boundary percent, it is used to control the thickness of the boundary, and $bp \in (0,1)$. The bigger the *bp* is, the thicker the boundary of the C_i. $|C_i|$ represents the number of objects in the cluster C_i.

Then the specific steps of BDAEC algorithm are as follows:

BDAEC Algorithm

Input:data set *D*, Eps radius *eps*, core points percent *cp*, connected degree *ω*,
boundary percent *bp*.

Output:boundary of each cluster in the data set, boundary of the whole data set.

Step 1: According to definition 1, calculating the Point Density of every object in the data set.

Step 2: According to definition 2, calculating the Core Points Set of the data set. An edge will be put between the core points which are satisfied the concept of Mutual Connected. Many connected undirected graphs will be constituted in this way. And the cluster numbers and the cluster division of the data set can be known by those connected undirected graphs for each one of them represents a cluster and has its unique class label.

Step 3: According to definition 4, calculating the Non-core Points Set and the Candidate Points Set, and mark the objects in the Non-core Points Set with a core point’s class label. And that core point must be the nearest core point to this object.

Step 4: According to definition 5, calculating the Maximum Density Point of each object in the Candidate Points Set. And calculating the Positive Field and the Negative Field of every object in the Candidate Points Set according to definition 6. Then calculating the Border Degree value of each object in the Candidate Points Set according to definition 7.

Step 4: According to Border Degree, cluster division, boundary percent and definition 8, calculating the boundary set of each cluster. The boundary of the whole data set is the union of every cluster’s boundary set.

3 Experimental Results and Analysis

In order to validate the effectiveness of BDAEC algorithm, we performed experiments on many data sets with different algorithms. First, we compare BDAEC with boundary detection algorithms to validate the effectiveness of the whole boundary extraction of BDAEC; Then we compare BDAEC with DBSCAN to validate the effectiveness of boundary extraction for each cluster of BDAEC.

Experimental Environment: CPU: Intel(R) Core(TM) i3-2130 3.40GHz; Memory:4GB; Operating System: Microsoft Windows 7; Algorithm Writing Environment: MATLAB2012.

First, we compare BDAEC with boundary detection algorithms BORDER and BERGE.

There are 5034 objects (including noisy objects) in the comprehensive data set which is showed in Fig1(a). As is shown in Fig1(a), there are 5 different clusters with different sizes in the data set. The result of BORDER(k=120, n=1200) is shown in Fig1(b); The result of BERGE (k=5, w=0.2, *beta*=0.9) is shown in Fig1(c); The result of BDAEC(*eps*=6.2, *cp*=0.85, ω=0.48, *bp*=0.13) is shown in Fig1(d)-Fig1(h), Fig1(d) is the whole boundary of the comprehensive data set, Fig1(e)-Fig1(i) is the boundary of each cluster in the comprehensive data set.

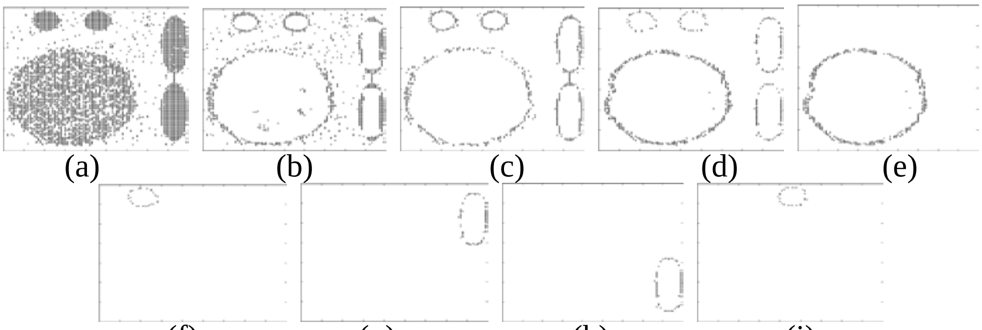

Figure 1: BDAEC compares with BORDER and BERGE

The experiment shows: BORDER, BERGE and BDAEC algorithms can find the boundary of the whole data set. BORDER can not avoid the interfere of the noises, and all the noises in the data set will be the boundary under the algorithm of BORDER. BERGE and BDAEC can avoid the interfere of noises, and they can get the boundary of the whole data set clearly. What's more, unlike the BERGE, BDAEC also can get the boundary of each cluster in the data set.

Second, we compare BDAEC with clustering algorithm DBSCAN.

There are 5931 objects (including noisy objects and interference line) in the comprehensive data set which is showed in Fig2(a). As is shown in Fig2(a), there are 6 different clusters with different sizes and shapes in the data set. And the clusters in the data set are extremely close to each other. The result of DBSCAN(*Eps*=8.1, *minpts*=28) is shown in Fig2(b)-Fig2(e), Fig2(b) is the whole boundary of the comprehensive data set, Fig2(c)-Fig2(e) is the boundary of each cluster that DBSCAN has got in the comprehensive data set. The result of BDAEC(*eps*=10.5, *cp*=0.93, *ω*=0.5, *bp*=0.4) is shown in Fig2(f)-Fig2(h), Fig2(f) is the whole boundary of the comprehensive data set, Fig1(g)-Fig1(l) is the boundary of each cluster in the comprehensive data set.

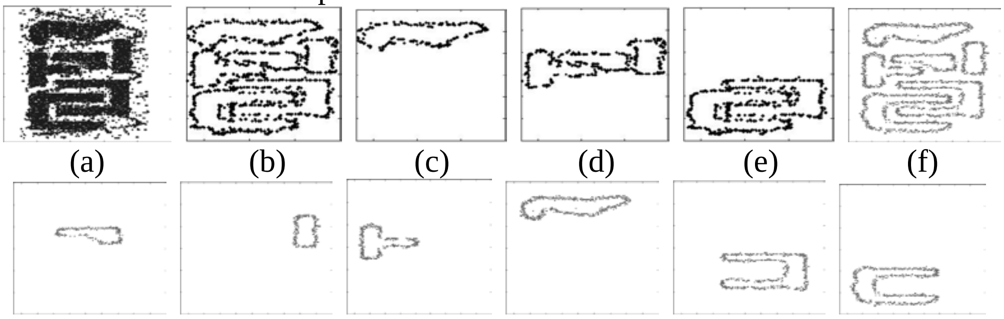

Figure 2: BDAEC compares with DBSCAN

The experiment shows: On the data set with noisy objects , interference line and all of its clusters are extremely close to each other, both DBSCAN and BDAEC can get the boundary of the whole data set. But DBSCAN can not take the appropriate parameters to get the correct cluster numbers or the cluster boundary of every cluster in the data set. BDAEC can get the right cluster numbers and the cluster boundary of each cluster in such data set.

4 Time Complexity Analysis

BDAEC algorithm has 2 phases: The first phase is to find the cluster numbers of the data set, and the time complexity of this phase is $O(cN^2)$; *c* is the cluster numbers in the data set. *N* is the number of the objects in the data set. The second phase is to find the candidate points set and the boundary of each cluster in the data set, and the time complexity of this phase is $O((2cp\text{-}cp^2)N^2/4 +aN)$; *cp* is core points percent, *a* is the average point density of the data set. In summary, the time complexity of BDAEC is $O(cN^2)$. The time complexity of BORDER is $O(kN^2)$. The time complexity of DBSCAN is $O(cN^2)$.The time complexity of BERGE is $O(N^{3/2}logN)$. So the time complexity of BDAEC is equal to DBSCAN and BORDER and bigger than BERGE.

5 Parameter Discussion

BDAEC algorithm has 4 parameters: Eps radius *eps*, core points percent *cp*, connected degree *ω*, boundary percent *bp*.The value of *eps* determines the point density of each object. *cp* means the percent of the core points in a data set. The bigger the *cp*, the less of the core points in the data set. *ω* determines the cluster numbers and the object's number in the candidate points set. *bp* determines the thickness of the boundary. The bigger of the *bp*, the thicker of the boundary.

6 Summary

This paper proposes BDAEC algorithm on the basis of DBSCAN, undirected graphs, connected degree and border degree. BDAEC can extract the boundary of each cluster and the whole data set with the function of avoiding the interference of noises in the data set. This is the feature of this algorithm. But BDAEC also has its limitation. It is only applying to the numerical data sets, and it can not applying to the categorical data sets or the mixed data sets. So how to solve the boundary detection problems on the categorical data sets and the mixed data sets are the following work of our research.

References

[1] Alex R, Alessandro L. Clustering by fast search and find of density peaks [J]. Science, Vol. 344 (2014) No. 6191, p. 1492-1496.

[2] Soumadip G, Sushanta B, Debasree S,Partha P S, A novel Neuro-fuzzy classification technique for data mining[J]. Egyptian Informatics Journal, Vol. 15 (2014) No. 3, p. 129-147.

[3] Mohamed B. A practical outlier detection approach for mixed-attribute data[J]. Expert Systems With Applications, Vol. 42 (2014) No. 22, p. 8637-8649.

[4] Qiu B Z, Wang B. Cluster boundary detection technology for categorical data[J]. Journal of Computer Applications, Vol. 32 (2012) No. 6, p. 1654-1656.

[5] Qiu B Z, Yang Y, Du X W. BRINK: An Algorithm of Boundary Points of Clusters Detection Based On Local Qualitative Factors[J]. Journal of Zhengzhou University, Vol. 33 (2012) No. 3, p. 117-121.

[6] Xia C, Hsu W, Lee M L, et al. BORDER: An efficient computation of boundary points[J]. Knowledge and Data Engineering, IEEE Transactions on, Vol. 18 (2006) No. 3, p. 289-303.

[7] Li X L, Geng P, Qiu B Z. Clustering Boundary Detection Technology for Mixed Attributes Data Set [J]. Control and Decision, Vol. 30 (2015) No. 1, p. 171-175.

[8] Ester M, Kriegel H P, Sander J, et al. A density-based algorithm for discovering clusters in large spatial databases with noise[C]. Conference on Knowledge Discovering and Data Mining. Portland, 1996, p. 226-231.

[9] Fan M, Fan H J, et al. Introduction to Data Mining [M]. Beijing: Posts &Telecom Press, 2013, p. 14-53.

[10] Yue F, Qiu B Z. Boundary Points Detecting Algorithm for Clusters in Noisy Dataset[J].Computer Engineering, Vol. 33 (2007) No. 19, p. 82-84.