AMSE JOURNALS-AMSE IIETA publication-2017-Series: Advances B; Vol. 60; N°1; pp 239-254
Submitted Jan. 2017; Revised March 15, 2017, Accepted April 15, 2017
https://doi.org/10.18280/ama_b.600115

DBIECM-an Evolving Clustering Method for Streaming Data Clustering

Kai-song Zhang, **Luo Zhong, ***Lan Tian, ****Xuan-ya Zhang, *****Lin Li

* School of Computer Science and Technology, Wuhan University of Technology, Wuhan 430070, China; School of Mechanical Engineering, Wuhan Polytechnic University, Wuhan 430023, China (zkspr@whut.edu.cn)
** School of Computer Science and Technology, Wuhan University of Technology, Wuhan 430070, China (zhongluo@whut.edu.cn)
*** School of Computer Science and Technology, Wuhan University of Technology, Wuhan 430070, China (ltian999@163.com)
**** Department of Electronic Information Engineering, Wuhan City Vocational College, Wuhan 430064, China (nike-zxy@163.com)
***** School of Computer Science and Technology, Wuhan University of Technology, Wuhan 430070, China (cathylilin@whut.edu.cn)

Abstract

To address the problem of the difficulty of traditional clustering methods to adapt to online clustering of streaming data and on the basis of the research on the evolutionary clustering method (ECM), this paper proposes a Davies-Bouldin index evolving clustering method for streaming data clustering (DBIECM). This method has improved the updating process of the clustering center and the radius of ECM and introduced the Davies-Bouldin Index (DBI) as the evaluation criterion for data classification. Compared with the traditional clustering method, DBIECM has better adaptability for stream data clustering. The experiments show that DBIECM has a better clustering effect on the evaluation criteria of the objective function value, DBI, as well as better accuracy and purity compared with ECM.

Key words

Streaming data, Online clustering, Evolving clustering, Davies-Bouldin Index

1. Introduction

As a study focus in machine learning and data mining attracting broad and intense interest, clustering analysis aims to divide the data set into different clusters according to a certain algorithm rule, which makes the difference between different classes as large as possible and that between same classes as small as possible. Clustering analysis, as a vital technology of data mining, and has a wide range of application areas such as customer analysis, biological genetic classification, network document classification and image segmentation [1].

With the development of information technology and the large amount of streaming data, traditional algorithms such as K-MEANS [2], FCM [3], BIRCH [4] etc., which deal with static data clustering problems, are required to do cluster partitioning in a given data set in the most effective manner. Traditional clustering algorithms are poor in dealing with streaming data and thus evolutionary clustering emerges at the appropriate moment in time. The evolutionary clustering method deals with online data, and the data distribution varies over time. The system requires cluster partitioning [5] at each moment of entering new data. Song et al. proposes the evolving clustering method(ECM) [6] and applies it to the dynamic evolution nerve fuzzy inference system (DENFIS) to create the fuzzy reasoning rule [7]. With the increase of input data, ECM can dynamically increase clusters, adjust the cluster center and the radius in real time. With good adaptability, this method only processes current input data and will not reprocess processed historical data. For the clustering of evolving data, Chakrabarti et al. have studied the time serialization problem of data and proposed an evolving clustering method [8] based on a smooth time frame.

The K-Means algorithm is a classic clustering method in the field of clustering analysis, which employs distance as the evaluation indicator of similarity and mean value of sample data as the cluster center. This algorithm takes multiple iterations to obtain the minimal objective function value of clustering results. The K-Means algorithm cannot solve overlapping problems without fuzziness. As the most common fuzzy clustering method which employs function value as the clustering criterion, Fuzzy C-Means (FCM) [9] uses the membership degree to measure the degree to which each sample data point belongs. Through introducing the fuzziness of the FCM method, reference [10] proposes an online evolving fuzzy clustering method(EFCM)based on ECM. Continuing online evolving clustering ability of the ECM, this method optimizes the clustering process by introducing fuzzy membership degree and has better clustering performance than ECM and FCM. Combining ECM and FCM [9], reference [11] adopts ECM to confirm the initial cluster center and optimizes it by employing FCM, thus achieving fuzzy clustering division, and finally realizes clustering of certainty classification through defuzzification. To solve the problem of ECM relying on a previously set threshold value Dthr and its sensitivity to input sequence, reference [12] presents a self-adaptive learning evolving clustering method (SALECM). Combining segmentation and fusion, this method can self-adaptively adjust clustering results without artificially setting parameters in the case where prior knowledge of data cannot be obtained. Reference [13] proposed a KNN evolving fuzzy clustering method(KEFCM), which employs the least square method to confirm the cluster center and the cluster radius, and becomes the enhanced version of traditional KNN machine learning methods.

Based on ECM and for better clustering performance, this paper improves the renewal process of the cluster center and the radius of ECM, optimizes the criteria of data classification in the clustering process and finally proposes an evolving clustering method for streaming data clustering(DBIECM). Taking the mean value of cluster center sample data as the cluster center, and the largest distance from cluster center sample data to its cluster center as the cluster radius, and DB **错误!未找到引用源。** as the current criteria of data classification, DBIECN determines the maximum radius in the clustering process by setting the value of the threshold Dthr.

2. Evolving Clustering Method

As a quick and one-pass online clustering method, ECM is used to perform dynamic clustering on streaming data with good self-adaptability, which is based on the clustering treating process of distance, and among them the cluster center represents the evolving node of the online model. In any cluster, the distances between all the sample data and the corresponding clustering center are less or equal to the preset threshold Dthr and the selection of this parameter directly affects the final clustering results.

Sample data comes from an online entered data stream during the online clustering process and the method starts from an empty cluster set. Current processed data from a newly created cluster is used to initialize the cluster center *Cc*, and the cluster radius *Ru* is initialized to zero. Some existing clusters will determine whether to update depending on the location of new data with the coming of new data. Clusters will be updated through the change of the place of the cluster center and the increase in the cluster radius. The cluster radius will not be updated when it reaches the threshold Dthr.

In this paper, the distance between two q-dimensional vector samples *x* and *y* is calculated by the formula which is defined as follows:

$$d(x,y)=\|x-y\|=\left(\sum_{i=1}^{q}|x_i-y_i|^2\right)^{\frac{1}{2}}/q^{\frac{1}{2}}. \tag{1}$$

Detailed steps of ECM are explained as follows:

Method 1 ECM

Input: sample data x_i from data streaming and the threshold Dthr.

Output: clustering results C

Step 1: Create the first cluster C_1. Take the first entered data point x_1 as the first cluster center Cc_1 and set the initial cluster radius $Ru_1=0$.

Step 2: The method is finished if all the sample data in the data streaming is disposed, otherwise, calculate the distance between current entered data x_i and all existing cluster centers Cc_j with the formula $D_{ij}=||x_i - Cc_j||$, $j=1, 2, \ldots, k$, where k stands for the number of existing cluster.

Step 3: If the distance value D_{ij} meets $D_{ij}\le Ru_j$, $j=1, 2, \ldots, k$, indicating that the current input data x_i belongs to an existing cluster C_m and $D_{im}=\|x_i-Cc_m\|=\min_{j=1,2,\cdots,k}(\|x_i-Cc_j\|)$, in this case, $x_i \in C_m$ and then neither any new cluster is created nor any existing cluster center or radius is updated. Go back to step 2, otherwise go to the next step.

Step 4: Calculate the sum of the distance D_{ij} of current input data x_i and all existing cluster centers Cc_j and the corresponding cluster radius Ru_j, that is $S_{ij}= D_{ij}+ Ru_j$, $j=1, 2, \ldots, k$. Select a cluster center C_a (The corresponding cluster center is Cc_a and the radius is Ru_a.) and make it meet $S_{ia}=D_{ia}+Ru_a=min(S_{ij})$, $j=1, 2, \ldots, k$.

Step 5: If $S_{ia}>2\times Dthr$ (Dthr is a given threshold in advance), then x_i doesn't belongs to any existing cluster, and employ the method in step 1 to create a new cluster, then go back to step 2.

Step 6: If $S_{ia}\le 2\times Dthr$, then $x_i \in C_a$ and cluster C_a is updated through removing its cluster center Cc_a and adding the cluster radius Ru_a. The updated cluster radius $Ru_{\mathrm{a}}^{new}= S_{ia}/2$ and remove the new cluster center Cc_{a}^{new} to the ligature of x_i and Cc_a. Make it meet $||Cc_{\mathrm{a}}^{new}-x_i||=Ru_{\mathrm{a}}^{new}$, and then go back to step 2.

This method has good adaptability. When streaming data continues coming in, this method only deals with current data and does not deal with processed historical data, thus saving processing time, which allows it to better fit the dynamic clustering problems of streaming data compared with the traditional clustering method. The method uses the threshold to decide whether current data belongs to any existing cluster or not, and employs the least distance to handle the classification process of current data.

3. DBIECM for streaming data clustering

3.1 Relavant Definitions

Definition 1: Cluster center Cc_j, $j=1, 2, ..., k$, also the *j*th cluster center, is equal to the mean value of all the samples in the cluster; that is:

$$Cc_j = \frac{1}{m_j}\sum_{i=1}^{m_j} x_i. \tag{2}$$

where k is the number of clusters, m_j is the number of the *jth* cluster, and x_i the sample data of the cluster.

Definition 2: Cluster radius, also the *jth* cluster radius, means the maximum distance between the cluster center of the *jth* cluster center and the sample data of the cluster. Taking the *jth* cluster as $C_j=\{x_i;\ i=1, 2, ..., m_j\}$ and the cluster center as Cc_j, then the cluster radius is:

$$Ru_j = \max_{i=1,2,\cdots,m_j}\{d(x_i, Cc_j)\} \tag{3}$$

Definition 3: The Davies-Bouldin Index (DBI) is an assessment criterion based on inner-cluster similarity and inter-cluster difference proposed by Davies and Bouldin. During the clustering process, the clustering results are measured by the degree of difference in inter cluster data and the degree of similarity in inner cluster data. The computation formula is as follows:

$$DBI = \frac{1}{k}\sum_{i=1}^{k} \max_{\substack{j\neq i \\ j=1,2,\cdots,k}} \left\{ \frac{S_i + S_j}{d(Cc_i, Cc_j)} \right\} \tag{4}$$

where $d(Cc_i, Cc_j)$ is the distance between cluster centers Cc_i and Cc_j, and S_i is the standard error of the *ith* sample data from a cluster and the corresponding cluster center Cc_i; that is, it expresses the inner-cluster similarity of the *ith* cluster. The formula is as follows:

$$S_i = \sqrt{\frac{1}{m_i}\sum_{j=1}^{m_i}\left|x_j - Cc_i\right|^2} \tag{5}$$

where m_i is the number of samples in the *ith* cluster.

For clustering results, DBI becomes smaller and the clustering quality becomes better when the similarity of the inner-cluster and the difference of the inter-cluster becomes higher.

3.2 DBIECM

In order to obtain better online clustering results for streaming data, a new online evolving clustering method DBIECM suitable for streaming data is proposed through improving ECM while retaining its advantages such as speediness, one-scan, adaptability, etc. DBIECM adopts the mean value of the cluster as the cluster center, the maximum distance between the sample data of the cluster and the cluster center as the cluster radius, and DBI as the classification criteria of current data. The threshold Dthr is set to control the maximum radius of the cluster during the clustering process. This method employs the presumption principle, Where for the classification process of current input data x_i in the method, x_i is presumed to belong to some existing cluster, and DBI of clustering results is calculated after belonging to each cluster. According to the assessment criterion “The smaller DBI, the better clustering results”, the clustering result with the smallest DBI is taken as the clustering result for handling current input data.

The detailed steps of DBIECM for streaming data is as follows.

Method 2 DBIECM for streaming data clustering

Input: sample data x_i from streaming data and the threshold Dthr.

Output: clustering result C.

Step 1: Create the first cluster C_1. Take the first input data point x_i as the first cluster center Cc_1 and set the initial cluster radius $Ru_1=0$.

Step 2: The method is finished if all the sample data from streaming data is processed. Otherwise, calculate the distance between current input data x_i and all existing cluster centers Cc_j $D_{ij}=||x_i - Cc_j||$, $j=1, 2, \ldots, k$, where k is the number of existing clusters.

Step 3: This means that current input data x_i belongs to some existing cluster C_m and $D_{im} = \|x_i - Cc_m\| = \min_{j=1,2,\cdots,k}(\|x_i - Cc_j\|)$ if there is a distance value D_{ij} and $D_{ij} \le Ru_j$, $j=1, 2, \ldots, k$. On this occasion, $x_i \in C_m$, no new cluster is created and all cluster radiuses are never be updated.

Then update the cluster center of C_m by formula (2) and go back to step 2. Otherwise, go to the next step.

Step 4: If D_{ij}>Dthr, then x_i does not belong to any existing cluster. Create a new cluster according the method in step 1 and then go back to step 2.

Step 5: If there is a cluster and $Ru_j < D_{ij} \leq Dthr, j=1, 2, ..., k$ ($Dthr$ is a given threshold in advance), then find all the clusters which satisfy $Ru_j < D_{ij} \leq Dthr, j=1, 2, ..., k$ and preserve the index of these clusters. Through formula (4), x_i is DBI_{index} value of clustering C_{index}. If $\text{index} = \arg\min(DBI_{index})$, then $x_i \in C_{index}$. Update the cluster center of C_{index} by formula (2) and update the cluster radius of C_{index} by formula (3) and go back to step 2.

DBIECM continues to maintain speediness, one process and adaptability of ECM and replaces the shortest distance criterion of ECM with DBI assessment criterion as the data classification criterion in the clustering process. which has improved and enhanced the accuracy rate in the clustering process, optimized the objective function value of clustering results and advanced the accuracy rate and purity of clustering results.

4. Experimental results and analysis

4.1 Experimental environment

Experiments in this paper are performed in a Windows 10 (64bit) operating system and the hardware configuration of the experiment station is Intel i5-2450M CPU 2.5GHz with a memory of 10 GB. The method is written in Matlab and the software tool is MATLAB R2012a.

4.2 Experimental Data Sets

To test the effectiveness of DBIECM, five standard data sets in the UCI machine learning database are used in the experimental data sets of this paper: Iris, Wine, Seeds, Glass and Breast Cancer (Details are shown in Table 1). The first row shows the data set names, and the next three rows respectively show the total number of samples, the number of attributes (including one category attribute) and the actual number of categories of corresponding data sets. Iris, Wine, Seed sand Glass are all data sets without missing value. Breast Cancer comes out of Breast Cancer Wisconsin (Original) data set, which has 699samples here 16 samples have missing value and Breast Cancer selects the remaining 683samples without missing value but have ten attributes (including category attribute) and the actual categorical number is two.

Attribute labels are added to the attributes of all data sets in Table 1 for the convenience of assessing clustering results. The category attributes of each sample in a data set are only used to assess the accuracy and the purity of the clustering method without referring to the clustering process or calculating the process of objective function value and DBI of clustering results.

Tab.1. Experimental Data Sets

Data Sets	Number of samples	Number of attributes	Number of categories
Iris	150	5	3
Wine	178	14	3
Seeds	210	8	3
Glass	214	11	6
Breast Cancer	683	10	2

4.3 The influence of the threshold on ECM and DBIECM

ECM and DBIECM are sensitive to the threshold Dthr whose value will directly affect the cluster number and the size of each cluster in the clustering results. Table 2 shows the influence of different thresholds on the clustering results of ECM and DBIECM.

Tab.2. The influence of the threshold on ECM and DBIECM

Data sets	Iris			Wine			Seeds			Glass			Breast Cancer		
Dthr	1.05	0.7	0.6	97	80	50	1.6	1.3	1.0	5.85	5.75	5.65	4.5	4	3.5
cluster number of ECM	3	5	7	3	4	6	3	4	6	6	7	7	2	4	5
cluster number of DBIECM	3	4	6	3	4	5	3	5	6	6	6	7	2	3	8

4.4 Clustering Quality

To test the effectiveness of DBIECM, four clustering performance measurements, objective function value, DBI value, accuracy [15] and purity [16] are used to access the performance of DBIECM and ECM in this paper. For convenience, the experiment simulates streaming data through reading the streaming data in data sets by row.

In clustering analysis, objective function value J, which reflects similarity and difference of clusters, usually employs a total least squared error, as expressed in the following:

$$J = \sum_{i=1}^{k} \sum_{j=1}^{m_i} (\|x_j - Cc_i\|) \tag{6}$$

where m_i is the sample number of the *ith* cluster.

Accuracy rate is a criterion to assess clustering results quality, and the formula is as follows:

$$accuracy = \frac{\sum_{j=1}^{k} a_j}{N}. \tag{7}$$

where N is the total number of samples of experimental data, and a_j is the number of samples of the *jth* cluster in clustering results which is in line with the actual cluster. Formula (7) shows that the greater the value of $a_j(j=1, 2, .., k)$ in the molecular part, the larger the number of correctly classified samples, and the greater the accuracy value, the higher the clustering results accuracy and the better the clustering quality.

Another criterion to assess the clustering quality is the average purity of clustering results. The formula for this criterion is as follows:

$$purity = \frac{1}{k}\sum_{i=1}^{k}\frac{N_i^d}{N_i}. \tag{8}$$

where k is cluster number, N_i is the sample data number of the ith cluster, and N_i^d is the sample data number of the main cluster of the *ith* cluster.

When separately calculating the DBI value, the objective function value, the accuracy and the purity of clustering results through formulas (4), (6), (7) and (8), this paper does not take the category attribute of sample data to the final calculating process. Two points have been changed for the convenience of the experiment. Firstly, remove the category attribute of experimental data sets in Table 1 to the last attribute. Secondly, improve formula (1) as follows:

$$d(x,y) = \|x-y\| = \left(\sum_{i=1}^{(q-1)}|x_i - y_i|^2\right)^{\frac{1}{2}} / (q-1)^{\frac{1}{2}}. \tag{9}$$

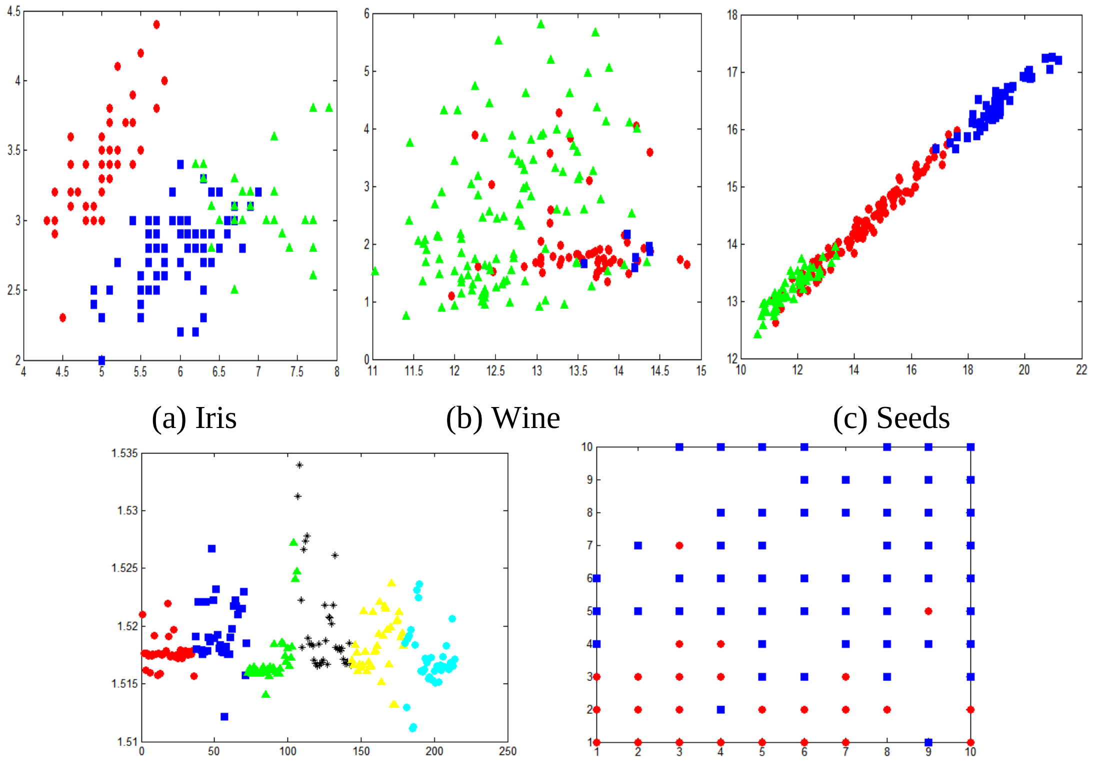
The distance formulae in this experiment are calculated by formula (9), which excludes the last attribute (category attribute) of the sample data.

The threshold Dthr is selected for the actual cluster number in Table 1 to figure out the five data sets Iris, Wine, Seeds, Glass and Breast Cancer. ECM and DBIECM are respectively employed to simulate the objective function value, the accuracy and the purity of clustering results after the clustering of streaming data as shown in Table 3.

Tab.3. Comparison of clustering results quality of ECM and DBIECM

data set	**Dthr**	**cluster number**	**ECM**				**DBIECM**			
			J	**DBI**	***accuracy***	***purity***	***J***	**DBI**	***accuracy***	***purity***
Iris	1.05	3	57.8504	0.7525	0.7333	0.8519	49.7709	0.8542	0.8733	0.9082
Wine	97	3	7147.1	1.4992	0.6124	0.7928	5820	0.8591	0.6798	0.8082
Seeds	1.6	3	158.0241	0.8956	0.7048	0.8434	131.4764	0.9638	0.8381	0.8910
Glass	5.85	6	628.1322	1.5555	0.8738	0.8736	623.9440	1.5529	0.8738	0.8742
Breast Cancer	4.5	2	2164.5	1.4285	0.6867	0.8374	1072.8	1.3449	0.9165	0.9357

It is difficult to represent the five high-dimensional experimental data sets visually. To obtain better intuitional clustering results, DBIECM is used to cluster five experimental data sets. The first two attributes of each sample data in clustering results are chosen for visualization. The results are shown in the following Figures.



(a) Iris (b) Wine (c) Seeds

(d) Glass (e) Breast Cancer

Fig.1. Clustering results of BIECM

Figure 1 shows that DBIECM has good effects on solving streaming data clustering.

To better compare the performance of ECM and DBIECM, Figure 2, 3, 4 and 5 respectively reveal the objective function value, DBI value the accuracy and the purity of the two methods.

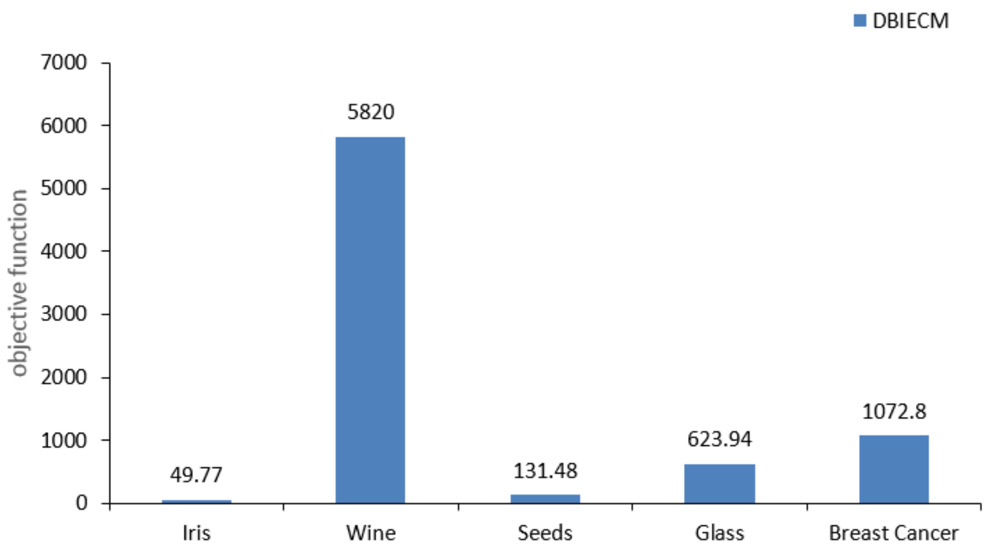


Fig.2. Comparison of objective function value of clustering results J by ECM and DBIECM

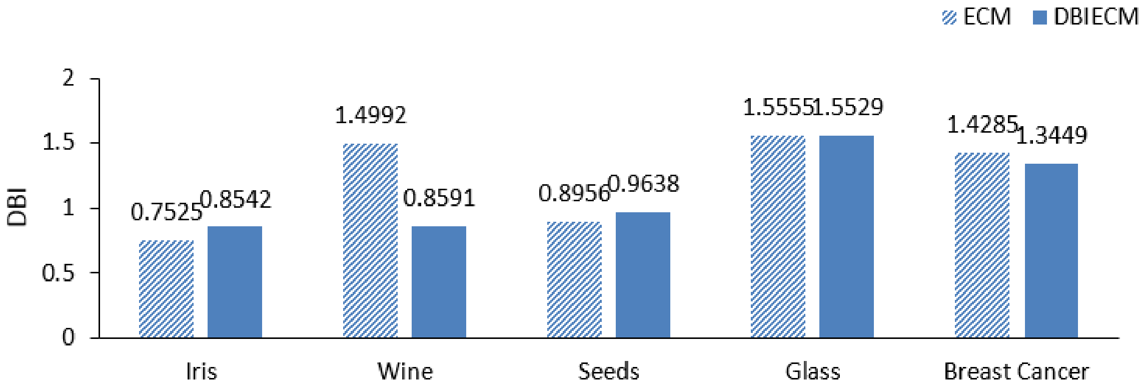


Fig.3. Comparison of DBI value of clustering results by ECM and DBIECM

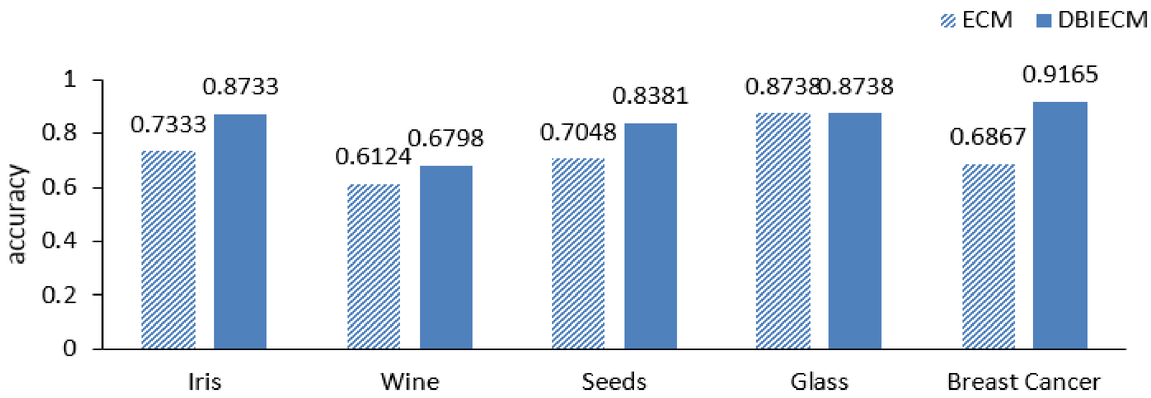


Fig.4. Comparison of accuracy of clustering results by ECM and DBIECM

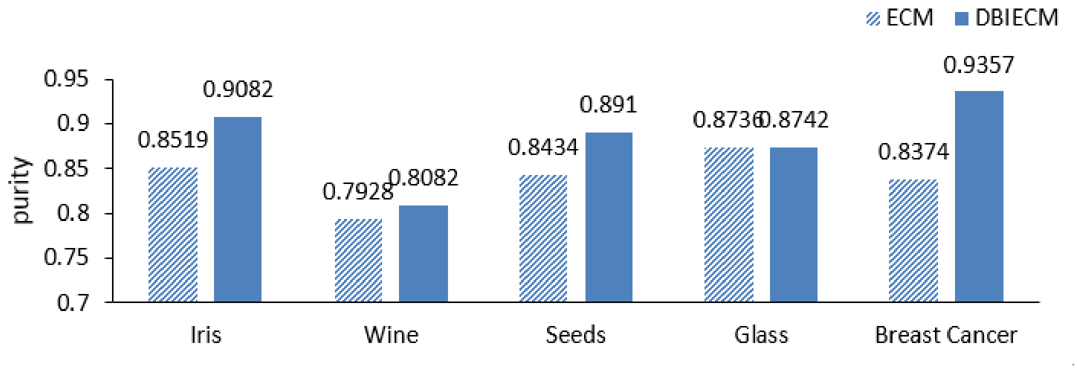


Fig.5. Comparison of purity of clustering results by ECM and DBIECM

Figure 2 shows the comparison of the objective function value of clustering results of Iris, Wine, Seeds, Glass and Breast Cancer by ECM and DBIECM. According to the clustering principle that the smaller the objective function value the better the clustering results and on the premise of choosing the same Dthr and obtaining the actual number of clusters, these five data sets separately employ DBIECM and ECM to cluster and the comparison shows that each objective function value in the clustering results by DBIECM is smaller than that by ECM. Thus we can see that, DBIECM has a better clustering effect than ECM from the assessment angle of the objective function value.

Figure 3 shows the comparison of the DBI value of clustering results of Iris, Wine, Seeds, Glass and Breast Cancer by ECM and DBIECM. According to the clustering principle that the greater the accuracy, the better the clustering results and on the premise of choosing the same DBI value and obtaining the actual number of clusters, these three data sets, Wine, Glass and Breast Cancer, separately employ DBIECM and ECM to cluster and the comparison shows that each DBI value of clustering results by DBIECM is smaller than that by ECM. Furthermore, the DBI value of Iris and Seeds by DBIECM is greater than that by ECM. Therefore, we can see that DBIECM has a slightly better effect than ECM on clustering from the assessment angle of the DBI value.

Figure 4 shows the comparison of the accuracy of clustering results of Iris, Wine, Seeds, Glass and Breast Cancer by ECM and DBIECM. According to the clustering principle that the greater the accuracy, the better the clustering results and on the premise of choosing the same DBI value and obtaining the actual number of clusters, these five data sets separately employs DBIECM and ECM to cluster and the comparison shows that the accuracy of clustering results by

DBIECM is greater than that by ECM. Thus, we can see that DBIECM is better than ECM from the assessment angle of objective function value.

Figure 5 shows the comparison of the purity of clustering results of Iris, Wine, Seeds, Glass and Breast Cancer by ECM and DBIECM. According to the clustering principle that the greater the purity the better the clustering results and on the premise of choosing the same Dthr and obtaining the actual number of clusters, these five data sets separately employ DBIECM and ECM to cluster and the comparison shows that the purity of clustering results by DBIECM is greater than that by ECM. Thus, we can see that DBIECM has a better clustering effect than ECM from the assessment angle of the purity.

The analysis above shows that for the five experimental data sets, DBICM is superior to ECM in the objective function value, and the accuracy and the purity are slightly better than ECM on DBI value. Synthetic analysis of experimental results indicate that DBIECM is much better than ECM.

5. Conclusion

This paper presents a Davies-Bouldin index evolving the clustering method DBIECM which has improved the updated process of the cluster center and radius on ECM and takes DBI as classification criterion. The introduction of DBI makes the objective function value of clustering results of DBIECM better than that of ECM and has improved upon the similarity of clustering results. Furthermore, DBIECM has a better effect on the objective function value, the accuracy and the purity when compared to ECM, and it is slightly better than ECM on the DBI value. Synthetically analyzing the four assessment criteria reveals that DBIECM has a better effect than ECM. In subsequent studies the researchers want to further optimize the parameter threshold Dthr and apply DBIECM to new research areas such as image segmentation.

Acknowledgements

We would like to thank the anonymous reviewers for their suggestions to improve this article. This work was supported by National Natural Science Foundation of China (61003130), National Science and Technology Support Program (2012BAH33F03) and Natural Science Foundation of Hubei Province (2015CFB525). Li Lin was partially supported by an award from the Wuhan Science and Technology Innovation Team Project (201307020402005).

References

1. J. Hou, W. Liu, E. Xu, H. Cui, Towards parameter-independent data clustering and image segmentation, 2016, Pattern Recognition, vol. 60, pp. 25-36.
2. J.A. Hartigan, M.A. Wong, Algorithm AS 136: A K-Means Clustering Algorithm, 1979, Applied Statistics, vol. 28, no.1, pp. 100-108.
3. Y.K. Dubey, M.M. Mushrif, FCM Clustering Algorithms for Segmentation of Brain MR Images, 2016, Advances in Fuzzy System, no. 1, pp. 1-14.
4. T. Zhang, R. Ramakrishnan, M. Livny, BIRCH: A New Data Clustering Algorithm and Its Applications, 1997, Data Mining and Knowledge Discovery, vol. 1, no. 2, pp. 141-182.
5. C. Zhang, J. Zhang, Learning on Time-Evolving Data, 2013, Chinese Journal of Computers, vol. 36, no. 2, pp. 310-316.
6. Q. Song, N. Kasabov, ECM-A Novel On-line, Evolving Clustering Method and Its Applications, 2002, M.i.posner Foundations of Cognitive Science, pp. 631-682.
7. N.K. Kasabov, Q. Song, DENFIS: Dynamic evolving neural-fuzzy inference system and its application for time-series prediction, 2002, IEEE Transactions on Fuzzy Systems, vol. 10, no. 2, pp. 144–154.
8. D. Chakrabarti, R. Kumar, A. Tomkins, Evolutionary clustering, 2011, In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 554–560.
9. E.G. Mansoori, FRBC: A Fuzzy Rule-Based Clustering Algorithm, 2011, IEEE Transactions on Fuzzy Systems, vol. 19, no. 5, pp. 960-971.
10. V. Ravi, E.R. Srinivas, N.K. Kasabov, On-line Evolving Fuzzy Clustering, 2007, International Conference on Computational Intelligence and Multimedia Applications, vol. 1, pp. 347-351.
11. G.Y. Du, S.L. Tian, F. Miao, Remote sensing image segmentation based on evolving clustering and fuzzy C-means, 2009, Application Research of Computers, vol. 26, no. 2, pp 3995-3997.
12. L. Wang, H. Sun, Evolving clustering method based on self-adaptive learning, 2016, Control and Decision, vol. 31, no. 03, pp. 423-428.
13. S. Abdulla, A. Al-Nassiri, kEFCM: kNN-Based Dynamic Evolving Fuzzy Clustering Method, 2015, International Journal of Advanced Computer Science and Applications (IJACSA), vol. 6, no. 2, pp. 5-13.

14. D.L. Davies, D.W. Bouldin, A Cluster Separation Measure, 1979, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 1, no. 2, pp. 224-227.
15. Y. Sun, Q.M. Zhu, Z.X. Chen, An iterative initial-points refinement algorithm for categorical data Clustering, 2002, Pattern Recognition Letters, vol. 23, no. 7, pp. 875-884.
16. J.Y. Chen, H.H. He, A fast density-based data stream clustering algorithm with cluster centers self-determined for mixed data, 2016, Information Sciences An International Journal, vol. 345, no. C, pp. 271-293.
17. J.C. Bezdek, R. Ehrlich, W. Full, FCM: The fuzzy c-means clustering algorithm, 1984, Computers & Geosciences, vol. 10, no. 2-3, pp. 191-203.
18. T.K. Chang, A. Talei, S. Alaghmand, L.H.C. Chua, Rainfall-runoff Modeling Using Dynamic Evolving Neural Fuzzy Inference System with Online Learning, 2016, Procedia Engineering, vol. 154, pp. 1103-1109.
19. M.J. Inácio, R.D. Maia, W.M. Caminhas, Evolving Fuzzy Classifier Based on the Modified ECM Algorithm for Pattern Classification, 2012, Springer Berlin Heidelberg, vol. 7435, pp. 612-621.
20. X.L. Xie, G. Beni, A validity measure for fuzzy clustering, 1991, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 13, no. 13, pp. 841-847.
21. A.K. Lekova, M.I. Dimitrova, Hand Gestures Recognition Based on Lightweight Evolving Fuzzy Clustering Method, 2013, Second International Conference on Image Information Processing, pp. 505-510.
22. P. Arora, Deepali, S. Varshney, Analysis of K-Means and K-Medoids Algorithm for Big Data, 2016, Procedia Computer Science, vol. 78, pp. 507-512.
23. R.G. Wei, J.G. Zhen, L.L. Bao, Study on mining big users data in the development of Hubei auto-parts enterprise, 2015, Mathematical Modelling of Engineering Problems, vol. 2, no. 4, pp. 1-6.
24. J. Xia, L. Xiao, L.P. Wan, Application of random-fuzzy probability statistics method, 2016, Mathematical Modelling of Engineering Problems, vol. 3, no. 1, pp. 19-24.
25. A.S.A. Hazmi, Z.A. Maurad, N.N.P.N. Pauzi, Z.A. Bakar, Z. Idris, Rapid evaluation of plate heat exchanger performance and fouling analysis in epoxidation of oleochemical at pilot plant scale, 2016, International Journal of Heat and Technology, vol. 34, no. 4, pp. 558-564.
26. M. Cucumo, V. Ferraro, D. Kaliakatsos, M. Mele, F. Nicoletti, Calculation model using finite-difference method for energy analysis in a concentrating solar plant with linear Fresnel

reflectors, 2016, International Journal of Heat and Technology, vol. 34, Special Issue 2, pp. S337-S345.