International Journal of Machine Learning and Cybernetics (2020) 11:545–552
https://doi.org/10.1007/s13042-019-00993-8

ORIGINAL ARTICLE

Improved fuzzy C-means algorithm based on density peak

Xiang-yi Liu[1] · Jian-cong Fan[1,2,3] · Zi-wen Chen[1]

Received: 7 May 2019 / Accepted: 25 July 2019 / Published online: 31 July 2019

Abstract
Fuzzy C-means (FCM) algorithm is a fuzzy clustering algorithm based on objective function compared with typical "hard clustering" such as k-means algorithm. FCM algorithm calculates the membership degree of each sample to all classes and obtain more reliable and accurate classification results. However, in the process of clustering, FCM algorithm needs to determine the number of clusters manually, and is sensitive to the initial clustering center. It is easy to generate problems such as multiple clustering iterations, slow convergence speed and local optimal solution. To address those problems, we propose to combine the FCM algorithm and DPC (Clustering by fast search and find of density peaks) algorithm. First, DPC algorithm is used to automatically select the center and number of clusters, and then FCM algorithm is used to realize clustering. The comparison experiments show that the improved FCM algorithm has a faster convergence speed and higher accuracy.

Keywords Fuzzy C-means algorithm · Density peak · Clustering

1 Introduction

Clustering [1–4] is a process of dividing the data object set into multiple groups and clusters, which recognizes different groups (clusters) underlying data. Clustering is different from classification. Classification needs class labels while there are no labels for clustering [5–10]. Currently, clustering analysis has been widely used in many areas [11–14], such as business intelligence, image pattern recognition, web search, biology and security, and so forth. Traditional clustering algorithms are mainly categorized as partitioning clustering, hierarchical clustering, density-based clustering, grid-based clustering, and model-based clustering. The k-means algorithm [15] is a classical partition-based clustering method. The k-means algorithm is simple in principle, easy to implement and fast in convergence. However, by adopting iterative algorithm, k-means algorithm can only obtain local optimal solution. In addition, the value of parameter k in k-means algorithm needs to be given in advance, and the initial clustering center has a great impact on the clustering results. DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm [16] is a representative density-based clustering algorithm. Different from partitioning and hierarchical clustering, DBSCAN algorithm defines clusters as areas with higher density than the remainder of the data set, which can divide areas with sufficient density into clusters and find clusters of arbitrary shapes in spatial databases. BIRCH (Balanced Iterative Reducing and Clustering Using Hierarchies) algorithm [17] is based on hierarchical clustering, and it used the limited memory resources to complete the high quality of the clustering of large datasets. BIRCH, however, does not work well if the clusters aren't spherical because it uses the concept of radius or diameter to control the boundaries of the clusters. The spectral clustering algorithm [18] is based on the spectral theory of graph theory. It transforms the clustering problem into the optimal partition problem of graph. The spectral clustering can overcome the shortcomings of some classical clustering algorithms, which ensures that the result converges to the global optimal solution, and has a good application prospect for data clustering.

On the other hand, clustering analysis is generally classified into two types: hard clustering, and soft clustering

✉ Jian-cong Fan
fanjiancong@sdust.edu.cn

1 College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao, China

2 Provincial Key Laboratory for Information Technology of Wisdom Mining of Shandong Province, Shandong University of Science and Technology, Qingdao, China

3 Provincial Experimental Teaching Demonstration Center of Computer, Shandong University of Science and Technology, Qingdao, China

[19–21]. Traditional clustering analysis is a hard partition, in other words, every object to be identified is strictly divided into a certain class with distinct boundaries. Most of the algorithms mentioned above are hard clustering. In order to improve the accuracy of clustering algorithm, fuzzy mathematical method [22–24] is introduced into clustering analysis. The concept of fuzzy clustering analysis was firstly proposed by Ruspini [25]. It generally refers to the construction of fuzzy matrix based on the object's properties of the object itself and the determination of clustering relationship based on certain membership degree. By means of fuzzy mathematics, the fuzzy relationship between samples is quantitatively determined, resulting in objective and accurate clustering. Fuzzy c-means algorithm [26, 27] is one of fuzzy clustering algorithms that are widely used. However, there are some underlying drawbacks of FCM algorithm. One of the issues is that the number of clusters that is determined artificially is sensitive to the initial clustering center. Also FCM algorithm is easy to generate problems such as multiple clustering iterations, slow convergence speed and local optimal solution. Many algorithms have been proposed to improve the FCM algorithm. Geweniger [28] combined the median c-means algorithm with the fuzzy c-means approach to improve the accuracy of this algorithm. Xue Zhenxia [29] presented a fuzzy rough semi-supervised outlier detection approach with the help of some labeled samples and fuzzy rough C-means clustering. This method introduces an objective function, which minimizes the sum squared error of clustering results and the deviation from known labeled examples as well as the number of outliers. As a result, better clustering results for normal points and better accuracy for outlier detection can be achieved. Zexuan [30] introduced a novel adaptive method to compute the weights of local spatial in the objective function, the new adaptive fuzzy clustering algorithm is allowing the suppression of noise and helping to resolve classification ambiguity. Fritz Heinrich [31] proposed several fuzzy clustering methods which are able to handle the presence of noise. Lai et al. [32] presented a rough k-means clustering algorithm by minimizing the dissimilarity to solve the divergence problem of the original approaches that the cluster centers may not be converged to their final positions. Wang [33] presented a rough-set [34, 35] based measurement for the membership degree of fuzzy C-means algorithm, and take the advantage of the positive region set and the boundary region set of rough set. In this paper, the FCM algorithm is combined with DPC algorithm. As a density-based clustering algorithm, DPC can get the cluster center while using less parameters than other clustering algorithms. Experiment results and analysis demonstrate that the proposed algorithms can effectively solve the problem that the FCM algorithm is sensitive to the initial cluster center and improve the accuracy of the algorithm.

2 Priliminaries

In this section, we introduce the standard DPC algorithm and the original fuzzy C-means Algorithm. Through this section we can understand these basic notions and descriptions.

2.1 DPC algorithm

In 2014, Alex Rodriguez and Alessandro Laio [36] proposed a clustering algorithm named clustering by fast search and find of density peaks. The advantage of this algorithm [37, 38] is that it requires fewer parameters, is insensitive to noise, and can find clusters with arbitrary shapes and dimensions. The clustering algorithm is divided into two steps. The first step is to calculate the local density and distance of samples according to the parameters input by users, and find the clustering center, which is also called density peak. Then, the appropriate clustering center is selected from the samples according to the decision graph. In the second step, the remaining samples are allocated to the cluster where the nearest and densest samples are located.

The algorithm has its basis with the assumptions that cluster centers are surrounded by neighbors with lower local density. They are also at a relatively large distance between any points with a higher local density. So, for each data point i, compute two quantities: its local density ρ_i and its distance δ_i from points of higher density. Both of these quantities depend only on the distance dis_{ij} between the data points. The definition of the local density ρ_i and distance δ_i is shown in Eqs. (1) and (2), respectively:

$$\rho_i = \sum_j \chi(dis_{ij} - d_c) \tag{1}$$

In the above equation, dis_{ij} is the distance between sample i and j, and d_c is the truncation distance, $\chi(\mathrm{x})\begin{cases} 1, & x < 0 \\ 0, & x \geq 0 \end{cases}$.

$$\delta_i = \min_{j:\rho_j > \rho_i} (dis_{ij}) \tag{2}$$

For the highest density point, the distance is $\delta_i = \max_j(dis_{ij})$.

In addition, Alex Rodriguez and Alessandro Laio also proposed a method to calculate local density using gaussian kernel function, as shown in Eq. (3):

$$\rho_i = \sum_{j \neq i} e^{-\left(\frac{dis_{ij}}{d_c}\right)^2} \tag{3}$$

By comparing Eq. (1) and (3), it is easy to know that Eq. (1) is a discrete value and Eq. (3) is a continuous value. Therefore, the latter is less likely to collide, which means different data points have the same local density value. In Eqs. (2), the distance between the sample and its nearest one with a higher local density ρ is δ. DPC algorithm uses the local density ρ and distance δ to construct decision graph and choose samples with large values both in ρ and δ as a clustering center. Then the algorithm allocates the rest of the sample j to the cluster with the highest local density on the nearest sample.

2.2 Fuzzy C-means algorithm

FCM algorithm is an unsupervised learning method based on the objective function optimization, which realizes the partition of a given dataset through the iterative optimization of the objective function [27]. FCM algorithm studies the clustering problem by means of fuzzy mathematics. The clustering result is a numerical value representing the membership degree of each data point to the clustering center.

Let dataset $X = \{X_1, X_2, \ldots, X_n\}$ is a collection of n data, where each sample X_i has p features. We divide the n samples into C fuzzy groups, and set the central matrix of the dataset as $V = \{V_1, V_2, \ldots, V_C\}$. The objective function defining FCM is:

$$J(U,V) = \sum_{i=1}^{C} \sum_{j=1}^{n} (u_{ij})^m (d_{ij})^2 \tag{4}$$

where $U = (u_{ij})$ is an $n \times c$ dimension membership matrix, and u_{ij} represents the membership between V_i and X_j. $d_{ij} = x_j - V_i$ is the Euclidean distance between the sample j and the cluster center i. m $(m > 1)$ is the fuzzy exponent in the algorithm, usually set as 2. Equation (4) also needs to meet the following conditions:

$$\begin{cases} \sum\limits_{i=1}^{C} u_{ij} = 1, & j = 1,2,\ldots,n; \\ 0 \le u_{ij} \le 1, & i = 1,2,\ldots,C; \quad j = 1,2,\ldots,n; \\ 0 < \sum\limits_{j=1}^{n} u_{ij} < n, & i = 1,2,\ldots,C; \end{cases} \tag{5}$$

Finally, the minimum objective function $J(U, V)$ of FCM algorithm was obtained through iterative optimization, and U and V can be obtained as follows:

$$u_{ij} = \begin{cases} \dfrac{1}{\sum_{k=1}^{C} \frac{d_{ij}}{d_{kj}}^{\frac{1}{m-2}}} \\ 1 \end{cases} \tag{6}$$

$$V_i = \frac{\sum_{j=1}^{n} (u_{ij})^m x_j}{\sum_{j=1}^{n} (u_{ij})^m} \tag{7}$$

3 DP-FCM algorithm

3.1 The improvement of DP-FCM algorithm

FCM algorithm needs to set the cluster centers and cluster number artificially, and is sensitive to the initial cluster centers. The algorithm is easy to generate problems such as multiple clustering iterations, slow convergence, local optimal solution and poor stability. Therefore, we propose a new density peak-based FCM algorithm (DP-FCM).

In DP-FCM, the first step is to determine the centers of the clusters based on the two parameters: ρ_i and δ_i of dataset. We define the density distance index φ_i:

$$\varphi_i = \rho_i \delta_i \tag{8}$$

Traverse the n sample points and find the φ_i values of all the points. Sort the density distance values in descending order and extract the first z points. Calculate an average density distance:

$$\varphi_{ave} = \frac{1}{z} \sum_{i=1}^{z} \varphi_i \tag{9}$$

It is easy to know that the points of the density distance value show a downward trend. The points with larger value maintain a higher possibility to be the clustering center. When $\varphi_i > \varphi_{ave}$, we specify that C_i at that point is a clustering center. Then, the clustering center is selected according to the decision graph drawn [39].

The second step of DP-FCM is to put the selected clustering center point into the FCM algorithm and obtain the clustering result.

3.2 The steps of DP-FCM algorithm

DP-FCM algorithm
Step1: Determine the initial cluster center
Input:
Sample dataset: $X = \{X_1, X_2, \ldots, X_n\}$
Output:
Clustering center: C_i
1.1 Calculate the values of the sample local density ρ_i and distance δ_i;
1.2 Calculate the density distance value φ_i and sort the φ_i values in descending order;
1.3 Select the z samples with the largest density distance and calculate φ_{ave};
1.4 Select the points that $\varphi_i > \varphi_{ave}$ as a cluster of k center points, and select it in the decision graph;
Step2: Execute the FCM algorithm
Input:
Number of clustering centers: k;
Clustering center matrix: $C = (C_1, C_2, \ldots, C_k)$;
Maximum number of iterations: L;
Output:
The clustering results of DP-FCM algorithm;
2.1 In the algorithm, determine the fuzzy index=2, the termination iteration threshold ε. And the condition for termination iteration is $\varepsilon > 0$;
2.2 Initialize the membership matrix according to the distance from each sample data to the clustering center;
2.3 Calculate the value of the objective function;
2.4 Use Equation (6) to update the membership matrix;
2.5 Use Equation (7) to calculate and update the fuzzy clustering center;
2.6 Calculate the objective function again. If the error of the two objective functions is smaller than ε or the maximum number of iterations exceeds L, exit. Otherwise, return to step 2.4.

Through the analysis of the above steps, we can see that for a dataset with n samples, the time complexity of the algorithm mainly comes from the local density of ρ_i, distance δ_i and FCM algorithm time complexity. If the Eq. (1) is used to calculate the local density of ρ_i, we need to look for samples that are less than d_c away from sample i, the worst time complexity is $O(n^2)$. If Eq. (3) is used to calculate the local density of ρ_i, we need to calculate the sum of weighted values of distances between each sample i with other samples, its time complexity is $O(n^2)$. The time complexity of the distance of δ_i is $O\left(n^2\right)$. The time complexity of FCM algorithm is $O(npCL)$, where n is the number of samples in the dataset, p is the dimension of the dataset, C is the number of clusters, and L is the number of iterations. Therefore, the total time of algorithm complexity is $O(2n^2 + npCL)$.

4 Experimental results

In order to test the effect of DP-FCM algorithm, we use several classical artificial datasets and real datasets in UCI repository. We also compare our DP-FCM algorithm with FCM algorithm, k-means algorithm, DBSCAN algorithm and NMF(Non-negative Matrix Factorization)-based model. Among them, k-means algorithm and DBSCAN algorithm are classical algorithms in the clustering algorithm based on partition and the clustering algorithm based on density, respectively. NMF algorithm is representative work in clustering. FCM algorithm is dependent on its initial cluster center and has poor stability. The optimized DP-FCM algorithm can solve these problems well and improve the effectiveness of the algorithm.

4.1 The datasets and evaluation indexes of experiment

Table 1 lists the experimental datasets, including 6 artificial datasets and 10 real datasets from UCI (Table 2).

Clustering performance measurement is also known as the validity indexes of clustering. To evaluate the performance, we apply metrics of Accuracy, ARI (Adjusted Rand index) and NMI (Normalized Mutual Information) [40]. These metrics are widely used in the fields of information retrieval and statistics to evaluate the quality of

Table 1 Description of the classical artificial datasets

Dataset	Size	Attribute	Cluster
R15	600	2	15
D31	3100	2	31
Square3	1000	2	4
Shapes	1000	2	4
Long1	1000	2	2
Twenty	1000	2	20

Table 2 Description of the real datasets

Dataset	Size	Attribute	Cluster
Jain	373	2	2
Aggregation	788	2	7
Flame	240	2	2
Art	300	4	3
Compound	399	2	6
Seeds	210	7	3
Wingnut	1016	2	2
Iris	150	4	3
Pima	768	8	2
Wine	178	13	3

clustering results. The range of the three evaluation metrics is between 0 and 1. And the larger the value is, the better the clustering effect is.

Specifically, Accuracy measures the percentage of correctly classified data points in the clustering solution over the pre-defined class labels. The Accuracy is calculated as:

$$Accuracy = \sum_{i=1}^{k} \frac{\max(C_i|L_i)}{|X|} \tag{10}$$

where C_i is the set of instances in the ith cluster, L_i is the class labels for all instances in the ith cluster. And $\max(C_i|L_i)$ is the number of instances with the majority label in the ith cluster (e.g. if label l appears in the ith cluster more often than any other labels, then $\max(C_i|L_i)$ is the number of instances in C_i with the lable l). $|X|$ is the number of elements in dataset X.

ARI (Adjusted Rand Index) takes into account the number of instances that exist in the same cluster and different clusters. The expected value of such a validation measure is not zero when comparing partitions. ARI is defined as:

$$ARI = \frac{n_{11} + n_{00}}{n_{11} + n_{10} + n_{01} + n_{00}} \tag{11}$$

where n_{11} number of pairs of instances that are in the same cluster. n_{00} Number of pairs of instances that are in different clusters. n_{10} Number of pairs of instances that should be in the same cluster in A, but in different clusters in B. n_{01} Number of pairs of instances that should be in different clusters in A, but in the same cluster in B.

NMI (Normalized Mutual Information) is a measure of the interdependencies between variables.

$$\mathrm{NMI} = \frac{I(X;Y)}{\sqrt{H(X)H(Y)}} \tag{12}$$

X and Y are the random variables. $I(X;Y)$ represents the mutual information of two variables. H_X is the entropy of X. They are defined as follows:

$$I(X;Y) = \sum_{x,y} p(x,y) \log\left(\frac{p(x,y)}{p(x)p(y)}\right) \tag{13}$$

$$H_X = \sum_{i=1}^{n} p(x_i)I(x_i) = \sum_{i=1}^{n} p(x_i)\log\frac{1}{p(x_i)} = -\sum_{i=1}^{n} p(x_i)\log p(x_i) \tag{14}$$

$$H_Y = \sum_{i=1}^{n} p(y_i)I(y_i) = \sum_{i=1}^{n} p(y_i)\log\frac{1}{p(y_i)} = -\sum_{i=1}^{n} p(y_i)\log p(y_i) \tag{15}$$

4.2 Analysis of experimental results on the artificial datasets

Figures 1, 2, 3, 4, 5 and 6 shows the clustering results of DP-FCM algorithm on six different artificial datasets. The datasets including D31, R15, Square3, Long1, Shapes and Twenty, which are described in Table 1. The coordinates of x-axis and y-axis represent two-dimensional plane. By

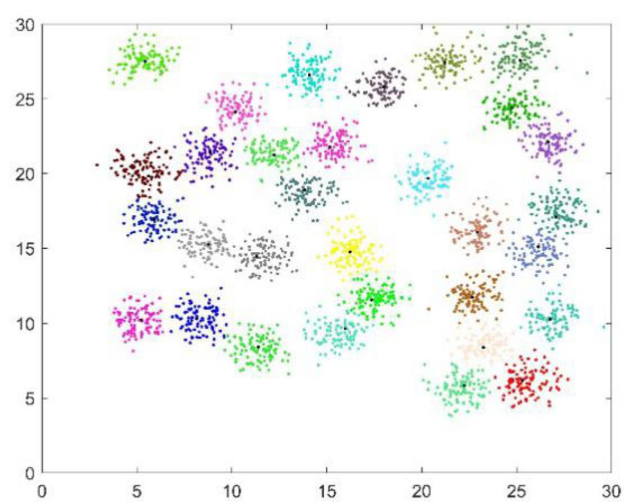

Fig. 1 Clustering result of D31

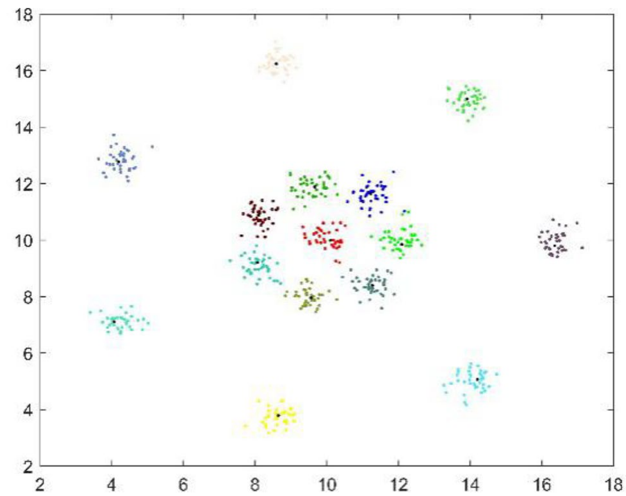

Fig. 2 Clustering result of R15

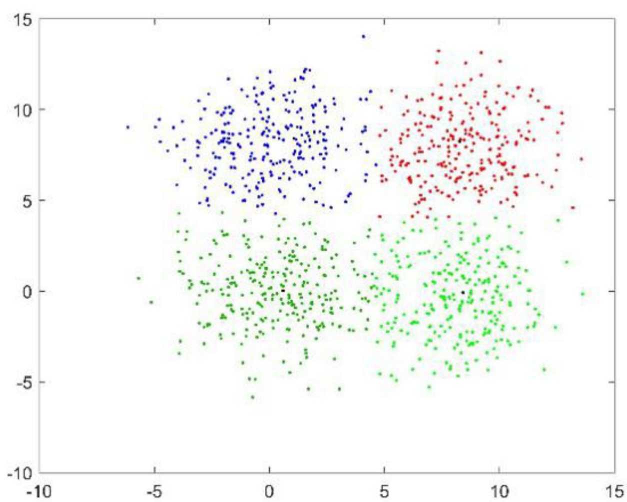

Fig. 3 Clustering result of Square3

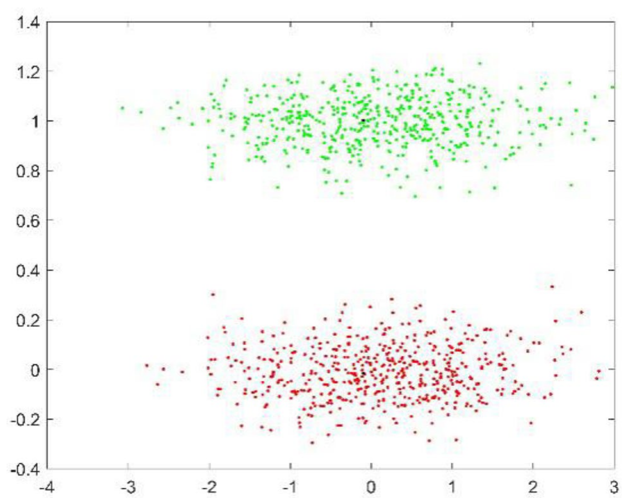

Fig. 4 Clustering result of Long1

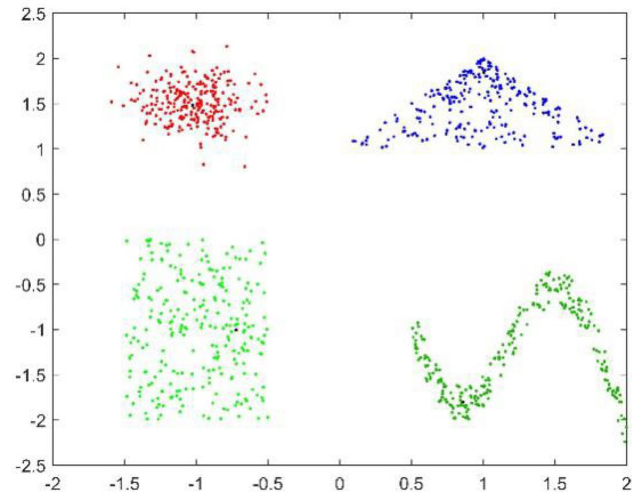

Fig. 5 Clustering result of Shapes

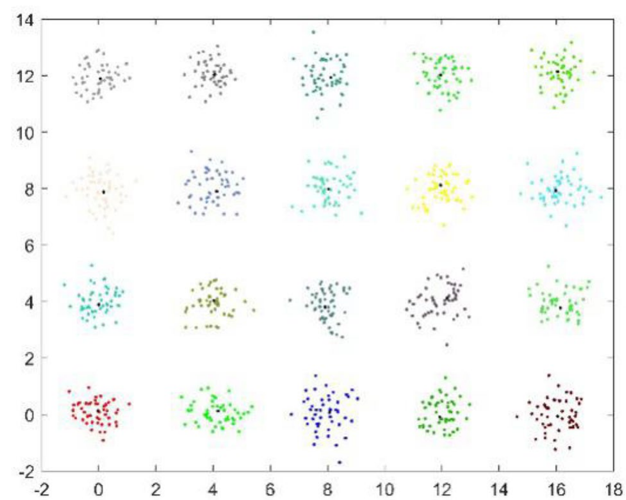

Fig. 6 Clustering result of Twenty

mapping the results to a two-dimensional plane, we can see that DP-FCM algorithm has a significant lift on the clustering of balanced datasets.

4.3 Analysis of experimental results on the real datasets in UCI

The experimental results on the real datasets in UCI are shown in Tables 3, 4 and 5. The tables show Accuracy, ARI(Adjusted Rand index), NMI(Normalized Mutual Information) of clustering results for each algorithm respectively. The experimental results are percentages, and the bolded values in the three tables represent the best experimental results.

Table 3 The accuracy comparison of clustering algorithms

Dataset	DP-FCM (%)	FCM (%)	K-means (%)	DBSCAN (%)	NMF (%)
Jain	**84.99**	74.80	79.36	80.70	76.14
Aggregation	**92.26**	79.82	80.25	88.83	80.96
Flame	84.58	85.00	82.92	**92.50**	84.17
Art	**91.67**	91.00	76.67	61.33	88.67
Compound	**65.91**	61.23	65.41	60.55	49.62
Seeds	87.27	**89.52**	89.04	62.85	87.14
Wingnut	**98.43**	94.09	97.24	96.15	93.90
Iris	**91.33**	90.00	74.67	63.33	88.67
pima	65.10	**65.89**	65.88	65.10	51.56
Wine	**70.79**	68.54	56.74	67.32	61.24

Table 4 The ARI comparison of clustering algorithm

Dataset	DP-FCM (%)	FCM (%)	K-means (%)	DBSCAN (%)	NMF (%)
Jain	**48.48**	30.04	34.19	25.63	27.74
Aggregation	**83.62**	73.44	76.57	78.67	64.70
Flame	47.62	48.80	43.12	**75.62**	47.62
Art	73.03	**74.34**	72.34	57.86	72.34
Compound	**53.57**	49.35	50.91	36.42	37.71
Seeds	64.29	**71.66**	71.03	48.75	66.35
Wingnut	**92.69**	77.06	89.27	80.50	77.40
Iris	**74.64**	72.94	71.63	68.50	73.02
pima	**9.32**	8.04	3.97	3.14	2.55
Wine	**37.21**	35.39	34.47	11.75	23.57

Table 5 The NMI comparison of clustering algorithm

Dataset	DP-FCM (%)	FCM (%)	K-means (%)	DBSCAN (%)	NMF (%)
Jain	**48.76**	35.71	35.31	26.72	34.42
Aggregation	**84.60**	84.38	85.40	82.15	80.18
Flame	41.43	44.22	40.50	**78.80**	41.43
Art	70.56	**74.50**	73.50	56.70	73.45
Compound	**71.13**	69.59	69.97	65.55	55.28
Seeds	61.51	69.49	71.01	56.92	63.63
Wingnut	**86.56**	66.85	81.80	83.25	67.24
Iris	**74.96**	70.21	74.19	72.15	74.82
pima	**4.05**	3.37	1.73	2.15	3.46
Wine	38.33	36.67	**42.65**	8.37	26.66

5 Summary and conclusion

FCM algorithm needs to determine the number of clusters manually and is sensitive to the initial clustering center. To solve this problem, we propose an improved FCM algorithm based on density peak: DP-FCM algorithm. This algorithm uses the density peak of data points to optimize the selection of the initial clustering center, which reduces the number of iterations, improves the convergence speed and avoids falling into the local optimal solution, and effectively solves the shortcomings of the original algorithm. In the experiment, 16 datasets in UCI were used to analyze the algorithm from three aspects, accuracy, ARI, and NMI. The experimental results show that DP-FCM algorithm is superior to the comparative FCM algorithm, K-means algorithm, DBSCAN algorithm and NMF algorithm, which shows DP-FCM an effective clustering algorithm.

Acknowledgements We would like to thank the anonymous reviewers for their valuable comments and suggestions. This work is supported by Shandong Provincial Natural Science Foundation of China under Grant ZR2018MF009, The State Key Research Development Program of China under Grant 2017YFC0804406, National Natural Science Foundation of China under Grant 91746104, the Special Funds of

Taishan Scholars Construction Project, and Leading Talent Project of Shandong University of Science and Technology.

References

1. Bailey KD (1994) Numerical taxonomy and cluster analysis. In: Typologies and taxonomies. Sage, California, issue 102, pp 34–57
2. Meilă Marina (2003) Comparing clusterings by the variation of information. Learning theory and kernel machines. Lect Notes Comput Sci 2777:173–187
3. Zhang Y, Li ZM, Zhang H, Yu Z, Lu TT (2018) Fuzzy c-means clustering-based mating restriction for multiobjective optimization. Int J Mach Learn Cybern 9:1609–1621
4. Ma HF, Zhang D, Jia MHZ, Lin XH (2019) A term correlation based semi-supervised microblog clustering with dual constraints. Int J Mach Learn Cybern 10:679–692
5. Wang Xizhao, Xing Hong-Jie, Li Yan et al (2015) A study on relationship between generalization abilities and fuzziness of base classifiers in ensemble learning. IEEE Trans Fuzzy Syst 23(5):1638–1654
6. Wang Ran, Wang Xizhao, Kwong Sam, Chen Xu (2017) Incorporating diversity and informativeness in multiple-instance active learning. IEEE Trans Fuzzy Syst 25(6):1460–1475
7. Wang Xizhao, Wang Ran, Chen Xu (2018) Discovering the relationship between generalization and uncertainty by incorporating complexity of classification. IEEE Trans Cybernet 48(2):703–715
8. Wang X, Zhang T, Wang R (2019) Non-iterative deep learning: incorporating restricted Boltzmann machine into multilayer random weight neural networks. IEEE Trans Syst Man Cybern Syst 49(7):1299–1380
9. Lin JCW, Yang L, Fournier-Viger P, Hong TP (2018) Mining of skyline patterns by considering both frequent and utility constraints. Eng Appl Artif Intell 77:229–238
10. Fournier-Viger P, Lin JCW, Kiran RU, Koh YS, Thomas R (2017) A survey of sequential pattern mining. Data Sci Pattern Recognit 1(1):54–77
11. Yang S, Han Y, Zhang X (2012) Kernel sparse representation for image classification and face recognition. Comput Vis ECCV 6314:1–14
12. Han JW, Kamber M, Pei J (2011) Data mining: concepts and techniques, 3rd edn. Morgan Kaufmann, Waltham, MA
13. Lim TS, Loh WY, Shih YS (2000) A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. Mach Learn J 40:203–228
14. Fan JC, Niu ZH, Liang YQ, Zhao ZY (2016) Probability model selection and parameter evolutionary estimation for clustering imbalanced data without sampling. Neurocomputing 211:172–181
15. Kanungo T, Mount DM, Netanyahu NS, Piatko CD, Silverman R, Wu AY (2002) An efficient k-means clustering algorithm: analysis and implementation. IEEE Trans Pattern Anal Mach Intell 24(7):881–892
16. Sander J, Ester M, Kriegel HP, Xu XW (1998) Density-based clustering in spatial databases: the algorithm GDBSCAN and its applications. Data Min Knowl Discov 2(2):169–194
17. Zhang T, Ramakrishnan R, Livny M (1996) BIRCH: an efficient data clustering method for very large databases. In: Proceedings of the 1996 ACM SIGMOD international conference on management of data. pp 103–114
18. Arias-Castro E, Chen G, Lerman G (2011) Spectral clustering based on local linear approximations. Electron J Stat 5:1537–1587
19. Xie XL, Beni G (1991) A validity measure for fuzzy clustering. IEEE Trans Pami 13(13):841–847
20. Li Y, Fan J, Pan J-S, Mao G, Wu G (2019) A novel rough fuzzy clustering algorithm with a new similarity measurement. J Internet Technol 20(4):
21. Fan J (2015) OPE-HCA: an optimal probabilistic estimation approach for hierarchical clustering algorithm. Neural Comput Appl 8:20–25. https://doi.org/10.1007/s00521-015-1998-5
22. Kosko B (1994) Fuzzy systems as universal approximators. IEEE Trans Comput 43(11):1329–1333
23. Chen Chien-Ming, Xiang Bin, Liu Yining, Wang King-Hang (2019) A secure authentication protocol for internet of vehicles. IEEE Access 7(1):12047–12057
24. Chen C-M, Xiang B, Wang K-H, Yeh K-H, Wu T-Y (2018) A robust mutual authentication with a key agreement scheme for session initiation protocol. Appl Sci 8(10):1789
25. Ruspini EH (1969) A new approach to clustering. Inf Control 15(1):22–32
26. Dunn JC (1973) A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. J Cybern 3(3):32–57
27. Bezdek JC (1981) Pattern recognition with fuzzy objective function algorithms. Adv Appl Pattern Recognit 22(1171):203–239
28. Geweniger T, Zülke D, Hammer B, Villmann T (2010) Median fuzzy c-means for clustering dissimilarity data. Neurocomputing 73:1109–1116
29. Xue Z, Shang Y, Feng A (2010) Semi-supervised outlier detection based on fuzzy rough C-means clustering. Math Comput Simul 80:1911–1921
30. Ji Z, Sun Q, Xia D (2011) A modified possobilistic fuzzy c-means clustering algorithm for bias field estimation and segmentation of brain MR image. Comput Med Imaging Graph 35:383–397
31. Fritz H, García-Escudero LA, Mayo-Iscar A (2013) Robust constrained fuzzy clustering. Inf Sci 245:38–52
32. Lai JZC, Juan EYT, Lai FJC (2013) Rough clustering using generalized fuzzy clustering algorithm. Pattern Recognit 46:2538–2547
33. Wang ZH, Fan JC (2018) A rough-set based measurement for the membership degree of fuzzy C-means algorithm. In: Proceedings of SPIE-the international society for optical engineering, 3rd international workshop on pattern recognition
34. Pawlak Z (1982) Rough sets. Int J Comput Inf Sci 11(5):341–356
35. Fan JC, Li Y, Tang LY, Wu GK (2018) RoughPSO: rough set-based particle swarm optimisation. Int J Bio-inspired Comput 12:245–253
36. Rodriguez A, Laio A (2014) Clustering by fast search and find of density peaks. Science 344(6191):1492–1496
37. Liu R, Wang H et al (2018) Shared-nearest-neighbor-based clustering by fast search and find of density peaks. Inf Sci 450:200–226
38. Bie R, Mehmood R, Ruan S et al (2016) Adaptive fuzzy clustering by fast search and find of density peaks. Pers Ubiquitous Comput 20(5):785–793
39. Zahn CT (1971) Graph-theoretical methods for detecting and describing gestalt clusters. IEEE Trans Comput 20(1):68–86
40. Fahad A, Alshatri N, Tari Z et al (2014) A survey of clustering algorithms for big data: taxonomy and empirical analysis. IEEE Trans Emerg Top Comput 2(3):267–279