Adaptive Clustering by Fast Search and Find of Density Peaks

Yuanyuan Chen[1], Lina Ge[1,2(✉)], Guifen Zhang[1], and Yongquan Zhou[1,2]

[1] School of Artificial Intelligence, Guangxi Minzu University, Nanning 530006, Guangxi, People's Republic of China
66436539@qq.com

[2] Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis, Nanning 530006, People's Republic of China

Abstract. Clustering by fast search and find of density peaks is a new density-based clustering algorithm, which is widely used in various fields owing to its simplicity and efficiency, unique parameters, and recognition of arbitrary shape clusters. However, when selecting the cluster center requires human participation, which makes the clustering result to be subjectively affected by the operator, thus reducing the availability of clustering and interrupting the fluency of the algorithm. In this study, to eliminate artificial participation in the selection of cluster centers, a weighted decision measurement slope change method is proposed to select cluster centers, and the F-Measure, ARI, and AMI of the algorithm are tested in the UCI and synthetic datasets. Experimental results show that the proposed algorithm addresses the limitation of human participation in the selection of cluster centers and improves the clustering performance of the algorithm.

Keywords: Clustering algorithm · Clustering by fast search and find of density peaks (DPC) · Cluster centers · Decision metrics

1 Introduction

Cluster analysis is one of the key technologies in data mining, and its main idea is to divide the dataset into different clusters so that the data in the same cluster are more similar and the data similarity between different clusters is low. Cluster analysis is widely used in image processing [1], social sciences [2], biomedicine [3], and other fields [4]. Classical clustering algorithms are divided into five types: division-based (such as K-Means [5]), hierarchical (such as BIRCH [6]), density-based (such as DBSCAN [7]), grid-based (such as STING [8]), and model-based.

Rodríguez and Laio proposed a fast search and find density peak clustering in 2014 that identifies arbitrary-shaped clusters, and is easy to understand, and does not require iteration [9]. The algorithm calculates the local density of a data point and the distance between the point and the point with a higher density and the closest point, generates a decision map, selects the cluster center based on the decision map, and assigns the non-cluster center point to the cluster with the highest density and closest point. Although

D.-S. Huang et al. (Eds.): ICIC 2022, LNAI 13395, pp. 802–813, 2022.
https://doi.org/10.1007/978-3-031-13832-4_65

DPC is simple and efficient, it has some disadvantages: (1) the value of the cutoff distance of the input parameter is selected empirically; (2) the algorithm adopts a one-step allocation strategy for the allocation strategy of the non-clustered center point; if the cluster center point is selected incorrectly, the subsequent data point allocation will also be incorrect; and (3) the cluster center point needs to be artificially selected.

In response to these shortcomings, many studies have been conducted to improve the DPC. Liang et al. [10] introduced Chameleon to DPC and proposed a clustering algorithm that requires only one discrete parameter, which realizes automatic detection of the algorithm. To reduce the computational complexity of the DPC, Xu et al. [11] proposed a new sparse search strategy to improve the similarity measure between data points. To reduce the influence of parameters on clustering results, the density-sensitive similarity was used, and a density cluster index was proposed for selecting cluster centers [12]. Xu et al. [13] introduced graph theory ideas to DPC by selecting cluster centers through the graphical connectivity of corners and centroids.

In this study, adaptive clustering by fast search and find density peaks, referred to as AdDPC, is proposed to address the problem that the density peak clustering algorithm requires human participation in the selection of cluster centers. AdDPC introduces weighted thinking and uses weighted decision measurement changes to select cluster centers to avoid the influence of artificial subjective thoughts on the clustering results.

The remainder of this paper is organized as follows: the second section introduces the original clustering by fast search and finds density peaks; the third section describes the AdDPC proposed in this paper; the fourth section analyses the results of the experiment; and the fifth section provides conclusions and future work.

2 Clustering by Fast Search and Find of Density Peaks

Clustering by fast search and find of density peaks(DPC) is based on the following two assumptions: (1) the cluster center is surrounded by low-density neighbor data points, and (2) the cluster center is sufficiently distance from another data point with a higher density. The main steps of DPC are divided into three stages: calculating the local density and distance, selecting cluster centers, and allocating the remaining data points.

For dataset $D = \{x_1, x_2, \ldots, x_n\}$, the DPC preprocesses the input dataset and calculates the Euclidean distances between data points, thus generating a distance matrix. The algorithm calculates the local density ρ_i of the data point x_i according to Eqs. (1) or (2), and its distance δ_i to the higher density data points is calculated using Eq. (3).

$$\rho_i = \sum_j \chi\left(d_{ij} - d_c\right), \chi(a) = \begin{cases} 1, a < 0 \\ 0, otherwise \end{cases} \tag{1}$$

$$\rho_i = \sum_j \exp(-\frac{d_{ij}^2}{d_c^2}) \tag{2}$$

where d_c is the cut-off distance. In the selection method, the value of d_c is the average number of near neighbors of the data points, which is approximately 1–2% of the total number of data points in the entire data set. d_{ij} is the Euclidean distance from the data

point x_i to the data point x_j. As can be noted from Eq. (1), the local density of the data points is the number of data points whose distance from the data point x_i is smaller than the cut-off distance. Equation (2) was used to calculate the local density for small-scale datasets (datasets with a total number of data points less than 6000).

$$\delta_i = \begin{cases} \min\limits_j \left(d_{ij}\right), \rho_j > \rho_i \\ \max\limits_j \left(d_{ij}\right), other \end{cases} \tag{3}$$

Distance δ_i of the data point x_i is calculated using Eq. (3), that is, the shortest distance between the point and other points of higher density in the dataset; if the point is already the highest density point, its distance δ_i is its maximum distance to other points.

After calculating the local density and distance of the data points, the DPC enters the cluster center selection stage. There are two methods for selecting the cluster center of the DPC (1) Decision-making diagram method. The decision-making diagram method generates a decision graph based on the local density and distance, with the local density as the x-axis and the distance as the y-axis, and then manually selects the best cluster center according to the decision graph. As shown in Fig. 1 (b), is the decision diagram corresponds to Fig. 1 (a), the number of data points in the figure represents the local density of the data point sorting, number 1 is the data point with the largest local density, and number 28 is the data point with the smallest local density. The rules for selecting cluster centers based on the decision graph are as follows. Select the data points in the upper-right corner of the decision graph that have both large local density values and distance values as the cluster centers.

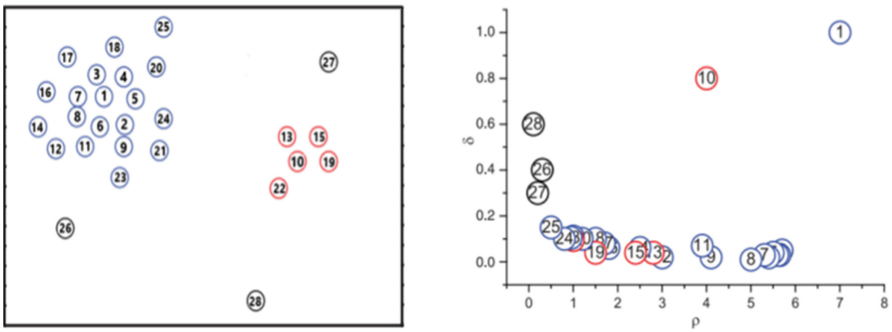
As can be seen from Fig. 1 (a), the local density values of data points 1 and 10 are higher, and the distance from other data points with higher densities is farther away; thus, they are suitable as cluster centers; data points 26, 27, and 28 are free from the data class cluster and are therefore treated as noise points; the rest of the data points are non-clustered center points.

From Fig. 1(b), it can be seen that the DPC divides the points, and the cluster center points are distributed in the upper right corner of the decision map, that is, the data points with large local density values and distant distance values are used as cluster center points; data points that are close to the δ axis and farther away from the ρ axis have smaller local density values and larger distance values, making them suitable as outliers; the remaining data points are close to the ρ axis, have small distances and relatively large local density values, and are divided into ordinary data points in the class cluster.

(2) Formulation method. The formulation method was proposed by Rodríguez and Laio, considering that if the decision graph of the dataset cannot be used to distinguish the cluster center point with the naked eye, the decision measurement γ is generated according to Eq. (4). Then, the decision measurement γ is sorted in descending order, and the data point corresponding to the first k values is selected as the cluster center.

$$\gamma_i = \rho_i \times \delta_i \tag{4}$$

After the cluster center is selected, the DPC assigns the remaining data points to the cluster of classes that are closest to the point and have a high local density.



(a) (b)

Fig. 1. Data distribution plot and decision diagram [9]; (a) Data distribution plot; (b) Decision diagram

3 Adaptive Clustering by Fast Search and Find of Density Peaks

From the introduction of the second section, it can be noted that when selecting clustering center points, the DPC has two schemes: the decision diagram method and the formula method. If the decision diagram method is used to select the cluster center point, then the clustering of the algorithm needs to be artificially involved, and in the process of intercepting the cluster center point, has a certain subjectivity. In Fig. 2, points with large local density values and large distance values are difficult to determine, and manual selection may lead to incorrect selection of the number of cluster center points, resulting in a poor clustering effect. Figure 3(b) illustrates the incorrect clustering result, in which, DPC selects four clustering center points, while Fig. 3(a) shows the standard clustering result.

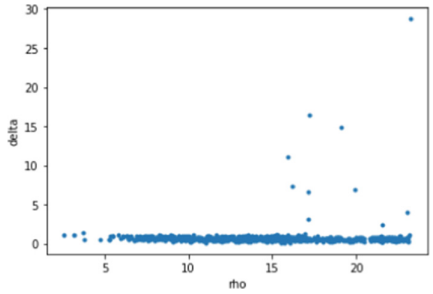


Fig. 2. Decision diagram of the Aggregation dataset

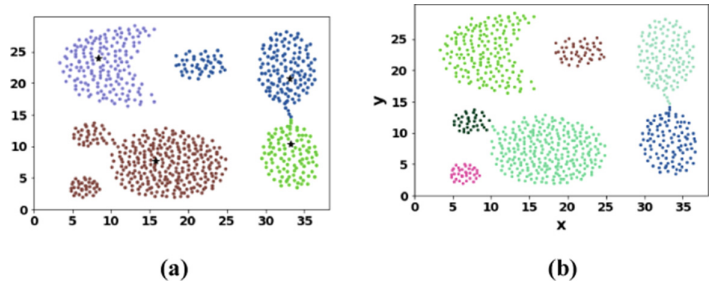


Fig. 3. Clustering result on Aggregation; (a) Standard clustering results; (b) Incorrect clustering result of DPC

To overcome the limitation that DPC requires human participation when selecting the center point of the cluster, this study uses the formula method to select the cluster center to achieve the adaptive selection of the cluster center.

To eliminate the influence of different orders of magnitude on the data, this study normalizes the local density ρ and distance δ. According to Eq. (4), the decision measure of the data point is calculated, and the decision measurement γ_i is normalized to obtain γ_i^*. Then, γ_i^* is sorted in descending order, and the first 50 dots are used to draw a descending sorting diagram, as shown in Fig. 4 (Fig. 4, 5, and 6 use the Aggregation dataset as an example, which has 788 data points and contains seven class clusters).

As can be observed from Fig. 4, the change in γ_i^* decreases from rapid to flat, and there are multiple inflection points in the graph; therefore, it is difficult to rely on the descending sorting plot of γ_i^* to determine the number of cluster center points. To solve this problem, this study proposes the use of a slope to represent the downtrend of γ_i^* values, as shown in Eq. (5).

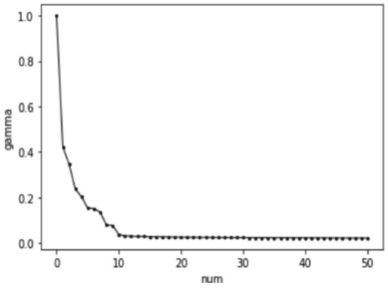


Fig. 4. γ_i^* descending sort graph

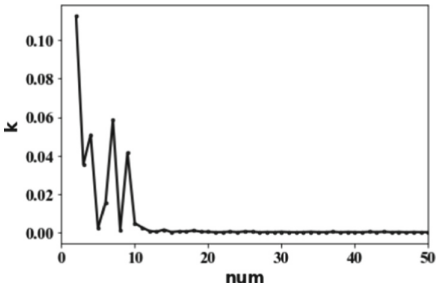


Fig. 5. Slope trend graph

$$k_i = \frac{\gamma_i^* - \gamma_{i+1}^*}{\gamma_{max}^* - \gamma_{min}^*}, (i = 1, 2, \ldots, 50) \tag{5}$$

The slope trend plot generated according to Eq. (5) is shown in Fig. 5, which shows that if the most varied point is selected as the demarcation point, the cluster center point may be selected incorrectly. From the comprehensive comparison of Fig. 4 and Fig. 5, it

can be observed that the first few values of γ_i^* are large, and the jump is relatively strong. To reduce the influence of this on the selection of clustered center points, a weighted idea is introduced, as shown in Eq. (6), where η is a weighted factor. $\eta = 1.001$.

$$kit_i = (i - \eta)k_i, i = 1, 2, \ldots, 50 \tag{6}$$

$$\gamma_m^* = arg\max_i(kit_i) \tag{7}$$

If γ_m^* satisfies Eq. (7), it becomes the point with the largest slope variation and can be used as the dividing point between clustered and non-clustered center points; specifically, the data point corresponding to $\gamma_1^*, \gamma_2^*, \ldots, \gamma_m^*$ is the cluster center point, and the number of cluster center points is m. Figure 6 shows a cluster center point discriminant plot generated from the calculation in Eq. (6). As can be observed from the plot, it is the maximum value; thus, the cluster center point of the dataset is 7.

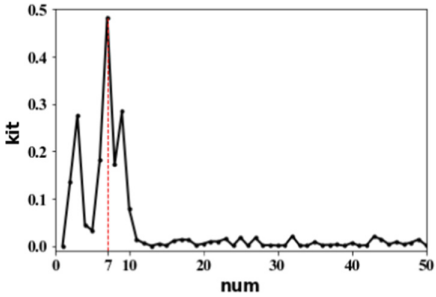


Fig. 6. Cluster center point discriminant plot of Aggregation

Figure 7 shows the γ_i^* descending sorting plot and clustered center point discriminant plot of the Spiral dataset. The total number of data points in the Spiral dataset is 312, including 3 class clusters. As can be observed in Fig. 7(b), the number of cluster center is 3, which is the same as the number of real class clusters in the dataset. If the number of cluster center is determined according to Fig. 7(a), the number of γ_i^* tends to be stable is selected, that is, the number of cluster center points is selected as 5, and the number of cluster center points is selected incorrectly, which leads to incorrect cluster results.

Figure 8 illustrates the γ_i^* descending sorting plot and cluster center point discriminant plot of the S2 dataset. The S2 dataset contains 15 class clusters, and the total number of data points is 5000. From Fig. 8(b), it is clear that the number of cluster center points is 15, which is the same as the real number of class clusters in the dataset. Thus, the scheme for determining the number of cluster center points according to Eq. (6) is suitable for most datasets.

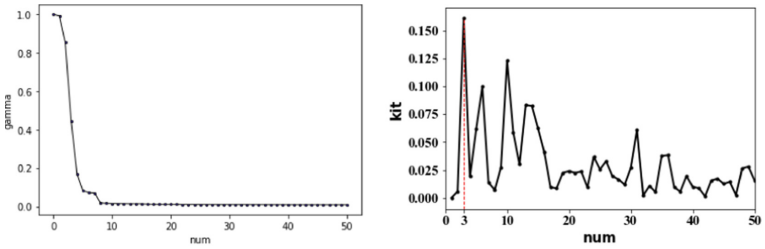


Fig. 7. Plot of the Spiral dataset; (a) γ_i^* descending sort graph; (b) Cluster center point discriminant plot.

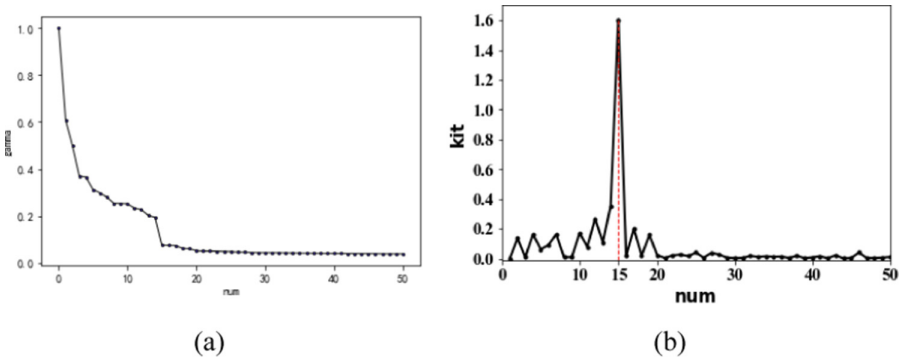


Fig. 8. Plot of the S2 dataset; (a) γ_i^* descending sort graph; (b) Cluster center point discriminant plot.

The major steps of AdDPC:

Input: dataset D; cutoff distance d_c.

Output: cluster result.

Step1: Use data preprocessing to calculate the Euclidean distance matrix between data points.

Step2: Calculate ρ using Eq. (2), calculate δ using Eq. (3), and normalize and generate decision diagrams.

Step3:Calculate γ using Eq. (4) and use the normalization process to obtains γ^*, sorting γ^* in descending order, calculate the slope change rate of γ^* using Eq. (6), and generate a cluster center point discriminant graph.

Step4: Calculate the dividing point between the cluster center points and non-cluster center points using Eq. (7). The data points corresponding to $\gamma_1^*, \gamma_2^*, \ldots, \gamma_m^*$ are used as the cluster center points, and m is used as the number of class clusters.

Step5: Assign the remaining data points to the class cluster that has the highest local density and is closest to them.

4 Experiments and Results

To prove the performance of the proposed algorithm, the dataset in Table 1 is used for experiments and compared with the DPC, DBSCAN, and K-Means. Further, the F-M [22], ARI(Adjusted Rand Index) [23], and AMI(Adjusted Mutual Information) [24] of each algorithm are tested. These three indicates are commonly used to judge the quality of clustering. The larger the value is, the better the clustering effect is.

Table 1. Tested dataset

Type	Name	Size	Dimension	Cluster	Source
Synthetic	Aggregation	788	2	7	[14]
	Spiral	312	2	3	[15]
	D31	3100	2	31	[16]
	Asymmetric	1000	2	5	[17]
	S2	5000	2	15	[18]
Real	Waveform	5000	21	3	[19]
	Seeds	210	7	3	[20]
	Libras-movement	360	91	15	[21]

Figure 9 and Fig. 10 illustrate the results of the clusters of the four algorithms on the Aggregation and the Spiral datasets, respectively; the color of the figure indicates the data points that are divided into clusters of the same class. The black rectangles in Fig. 9(a) and Fig. 10(a) represent the algorithm's selection of clustering center points; the stars in Fig. 9(b) and Fig. 10(b) represent the center points of clustering; the black dots in Fig. 9(c) and Fig. 10(c) represent noise points; and in Fig. 9(d) and Fig. 10(d), the black triangles represent the cluster center point.

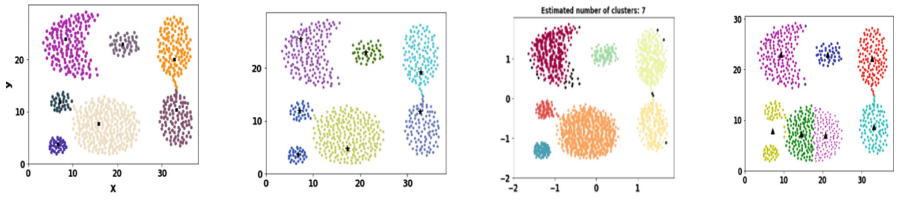
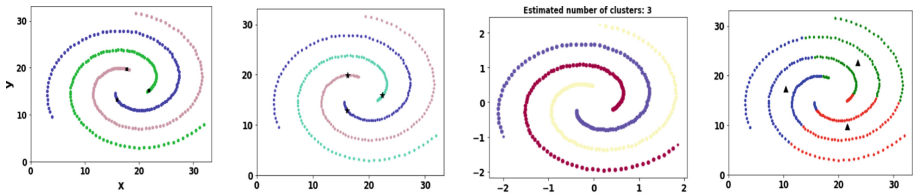


Fig. 9. Cluster result of Aggregation. (a)AdDPC; (b)DPC; (c)DBSCAN; (d)K-Means.

As shown in Fig. 9, all four algorithms can determine the correct number of class clusters; however, the K-Means clustering algorithm is erroneous because it identifies two cluster center points in the same cluster and groups two different clusters into one when identifying the cluster center point. Although DBSCAN does not have evident

clustering errors, it erroneously identifies some non-clustered center points as noise points when searching for noise points, which reduces the availability of the algorithm. Both AdDPC and DPC can cluster correctly, and the cluster availability is high, indicating that AdDPC is more accurate in the selection of cluster center points.



(a) (b) (c) (d)

Fig. 10. Cluster result of the Spiral. (a)AdDPC; (b)DPC; (c)DBSCAN; (d)K-Means.

As shown in Fig. 10, in addition to the K-Means, the other three algorithms are capable of correct clustering. K-Means divides the dataset into three parts and takes the center of each part as the cluster center point. The selection of the cluster center point leads to poor availability of clustering, which indicates that even if the correct number of class clusters k is entered, K-Means cannot effectively process the non-convex dataset. The comparisons in Fig. 10(a) and Fig. 10(b) show that for the Spiral dataset, the cluster center points selected by AdDPC are closer to the end of each cluster, and the local density of the data points is larger and more reasonable than that of the other algorithms.

Figures 11, 12, and 13 are the comparative charts of cluster evaluation indicators of AdDPC, DPC, DBSCAN and K-Means respectively, in Table 1. From the comparison of the three graphs, it can be noted that the improved DPC has better indicator values on the six datasets than the other three algorithms. Among the indicators of the Aggregation dataset, the indicators of AdDPC and DPC are higher than those of DBSCAN and K-Means. In addition, the index values of AdDPC are slightly higher than those of DPC, whereas those of K-Means are the smallest. AdDPC has the best clustering effect on the Aggregation dataset is, whereas K-Means exhibits the worst performance.

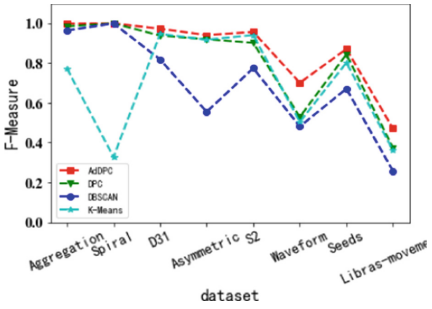


Fig. 11. F-Measure on eight datasets

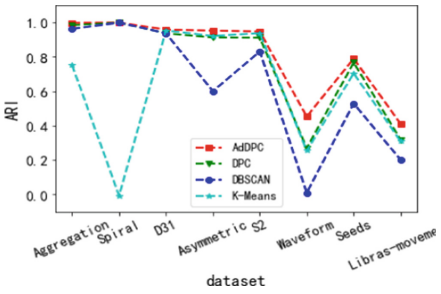


Fig. 12. ARI on eight datasets

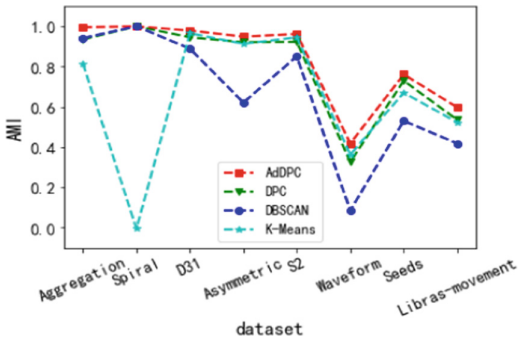


Fig. 13. AMI on eight datasets

By comparing the clustering index values of the Spiral dataset by the algorithms in the figure, it can be noted that the indicator values of AdDPC, DPC, and DBSCAN reached the optimal value, whereas K-Means had the worst indicators on the dataset. By comparing the clustering result graph in Fig. 10, it can be noted that on the Spiral dataset, the AdDPC, DPC and DBSCAN algorithms can not only find the correct number of cluster center points, but also that the values of each indicator are optimal. In particular, these three algorithms achieve the best clustering performance on this dataset; however, the clustering center points selected by the algorithms are different.

The D31 dataset contained 3100 data points and 31 high-density spherical clusters. Among the indicators in the dataset, the AdDPC indicators were the optimal values of the four algorithms; As can be observed in the F-Measure indicator graph, the values of DPC and K-Means almost coincide, while the values of DBSCAN are the smallest. In the ARI comparison, the values of DPC and DBSCAN coincide, while the values of K-Means are the smallest in this case. Overall, AdDPC exhibited the best clustering effect on the D31 dataset.

The Asymmetric dataset has a total of 1000 data points and contains five classes. On this dataset, DBSCAN divides many data points incorrectly during clustering, resulting in the worst clustering effect on both sides. Compared with the DPC and K-Means, the AdDPC in this study is better in terms of the distribution of some boundary points and has the best index values.

The S2 dataset contained 5000 data points and 15 categories. DBSCAN identifies more data points as noise points, and as can be noted from the data of the three index charts on this dataset, DBSCAN has the smallest index value and the worst performance. The index value of the AdDPC algorithm is the best.

The Waveform dataset is a dataset with high dimensions and a large total amount of data, containing three types of data samples. The three performance index values of AdDPC are better than those of the other three algorithms, as indicated by the data in the figure, implying that AdDPC has a better clustering performance on this dataset. DBSCAN has the smallest the value in the F-Measure, while K-Means exhibits the worst performance in ARI. By comparing the AMI, it can be seen that DPC and K-Means have the lowest AMI in this dataset, indicating the worst performance. Overall, of the four

algorithms for clustering on this dataset, AdDPC performed the best, and the remaining three algorithms each had advantages.

The Seeds dataset contained three types of wheat seed information, each described by seven geometric parameters of the seeds. When the performance indicators F-Measure, ARI, and AMI of the algorithm are compared, it can be shown that AdDPC is superior to the other three algorithms, and DBSCAN has the worst clustering performance on this dataset in terms of the overall index.

The Libras-movement dataset contained 360 data points in 15 clusters, each containing 24 data points. In this dataset, the amount of data in each class cluster makes calculating local densities for the algorithm more complex. On this dataset, the clustering index values of AdDPC are higher than those of other algorithms in Figs. 11, 12, and 13. Furthermore, the index values of the other algorithms have been significantly improved; that is, on this dataset, AdDPC has the best indicator performance.

Based on the above experimental clustering result graphs and the comparison of cluster evaluation index values, it can be demonstrated that the clustering effect of AdDPC proposed in this study is the best overall when compared to K-Means, DBSCAN, and DPC. In addition, AdDPC can correctly identify a reasonable clustering center point, which reduces the randomness effect of human participation in the selection of cluster center points.

5 Conclusion and Future Work

This study proposes adaptive clustering by fast search and find of density peaks. Compared with the DPC algorithm, this algorithm does not require the artificial selection of clustering centers and does not interrupt the continuity of the algorithm. First, the shortcomings of the DPC are analyzed, and an improved scheme is proposed. The weighting factor is introduced to calculate and weight the slope change rate of the decision measurement so that the algorithm can adaptively select the cluster center point and verify it experimentally. The experimental results show that the clustering performance of AdDPC is improved, and the problem of human participation in the selection of cluster center points by DPC is solved.

In future work, the improvement scheme of the allocation strategy for non-clustered center points should be further studied. In addition, the problem of varying cut-off distances affecting the clustering results for different datasets also should be further investigated.

Acknowledgments. This work was supported by the National Natural Science Foundation of China under Grant 61862007.

References

1. Ma, S., Guo, P., You, H., et al.: An image matching optimization algorithm based on pixel shift clustering RANSAC. Inf. Sci. **562**, 452–474 (2021)
2. Du, Z., Luo, H., Lin, X., et al.: A trust-similarity analysis-based clustering method for large-scale group decision-making under a social network. Information Fusion **63**, 13–29 (2020)

3. Hassan, B.A., Rashid, T.A., Hamarashid, H.K.: A novel cluster detection of COVID-19 patients and medical disease conditions using improved evolutionary clustering algorithm star. Comput. Biol. Med. **138**, 104866 (2021)
4. Yan, M., Chen, Y., Hu, X., et al.: Intrusion detection based on improved density peak clustering for imbalanced data on sensor-cloud systems. J. Syst. Architect. **118**, 102212 (2021)
5. Hartigan, J.A., Wong, M.A.: Algorithm AS 136: A k-means clustering algorithm. J. Royal Statistical Society. Series C (Applied Statistics) **28**(1), 100–108 (1979)
6. Zhang, T., Ramakrishnan, R., Livny, M.: BIRCH: a new data clustering algorithm and its applications. Data Min. Knowl. Disc. **1**(2), 141–182 (1997)
7. Ester, M., Kriegel, H.P., et al.: A density based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the 2nd International Conference on Knowledge Discovering in Databases and Data Mining (KDD-96), pp. 226–232 (1996)
8. Wang, W., Yang, J., Muntz, R.: STING: A statistical information grid approach to spatial data mining. Vldb. **97**, 186–195 (1997)
9. Rodriguez, A., Laio, A.: Clustering by fast search and find of density peaks. Science **344**(6191), 1492 (2014)
10. Liang, Z., Chen, P.: An automatic clustering algorithm based on the density-peak framework and Chameleon method. Pattern Recogn. Lett. **150**, 40–48 (2021)
11. Xu, X., Ding, S., Wang, Y., et al.: A fast density peaks clustering algorithm with sparse search. Inf. Sci. **554**, 61–83 (2021)
12. Xu, X., Ding, S., Wang, L., et al.: A robust density peaks clustering algorithm with density-sensitive similarity. Knowl.-Based Syst. **200**, 106028 (2020)
13. Xu, T., Jiang, J.: A Graph Adaptive Density Peaks Clustering algorithm for automatic centroid selection and effective aggregation. Expert Syst. Appl. **195**, 116539 (2022)
14. Gionis, A., Mannila, H., Tsaparas, P.: Clustering aggregation. Acm Trans. Knowledge Discovery from Data **1**(1), 4 (2007)
15. Hong, C., Yeung, D.Y.: Robust path-based spectral clustering. Pattern Recogn. **41**(1), 191–203 (2008)
16. Veenman, C.J., Reinders, M.J.T., Backer, E.: A maximum variance cluster algorithm. Pattern Analysis Machine Intelligence IEEE Trans. on **24**(9), 1273–1280 (2002)
17. Rezaei, M., Fränti, P.: Can the number of clusters be determined by external indices? IEEE Access **8**, 89239–89257 (2020)
18. Fränti, P., Virmajoki, O.: Iterative shrinking method for clustering problems. Pattern Recogn. **39**(5), 761–775 (2006)
19. Breiman, L., Friedman, J., Stone, C.J., Olshen, R.A.: Classifification and Regression Trees, CRC Press (1984)
20. Charytanowicz, M., Niewczas, J., Kulczycki, P., et al.: Complete gradient clustering algorithm for features analysis of x-ray images. Information Technologies in Biomedicine. Springer, Berlin, Heidelberg, pp. 15–24 (2010) https://doi.org/10.1007/978-3-642-13105-9_2
21. Dias, D.B., Madeo, R.C.B., Rocha, T., et al.: Hand movement recognition for brazilian sign language: a study using distance-based neural networks. In: 2009 International Joint Conference on Neural Networks. IEEE pp. 697–704 (2009)
22. Sasakl, Y.: The truth of the F-measure. Teach Tutor mater **1**(5), 1–5 (2007)
23. Hubert, L., Arabie, P.: Comparing partitions. J. Classif **2**(1), 193–218 (1985)
24. Vinh, N.X., Epps, J., Bailey, J.: Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. J. Machine Learning Res. **11**, 2837–2854 (2010)