Hindawi
Mathematical Problems in Engineering
Volume 2020, Article ID 4724150, 11 pages
https://doi.org/10.1155/2020/4724150

Research Article

Automatic Determination of Clustering Centers for "Clustering by Fast Search and Find of Density Peaks"

Xiangqiang Min,[1,2] Yi Huang,[1,2] and Yehua Sheng [1,2]

[1]*Key Laboratory of Virtual Geographic Environment, Ministry of Education of PRC, Nanjing Normal University, Nanjing, China*
[2]*Jiangsu Center for Collaborative Innovation in Geographical Information Resource Development and Application, Nanjing, China*

Correspondence should be addressed to Yehua Sheng; shengyehua163@163.com

Received 8 November 2019; Revised 13 February 2020; Accepted 5 March 2020; Published 9 April 2020

Academic Editor: Weifeng Pan

Dividing abstract object sets into multiple groups, called clustering, is essential for effective data mining. Clustering can find innate but unknown real-world knowledge that is inaccessible by any other means. Rodriguez and Laio have published a paper about a density-based fast clustering algorithm in *Science* called CFSFDP. CFSFDP is a highly efficient algorithm that clusters objects by using fast searching of density peaks. But with CFSFDP, the essential second step of finding clustering centers must be done manually. Furthermore, when the amount of data objects increases or a decision graph is complicated, determining clustering centers manually is difficult and time consuming, and clustering accuracy reduces sharply. To solve this problem, this paper proposes an improved clustering algorithm, ACDPC, that is based on data detection, which can automatically determinate clustering centers without manual intervention. First, the algorithm calculates the comprehensive metrics and sorts them based on the CFSFDP method. Second, the distance between the sorted objects is used to judge whether they are the correct clustering centers. Finally, the remaining objects are grouped into clusters. This algorithm can efficiently and automatically determine clustering centers without calculating additional variables. We verified ACDPC using three standard datasets and compared it with other clustering algorithms. The experimental results show that ACDPC is more efficient and robust than alternative methods.

1. Introduction

In data mining, clustering is the process of dividing abstract object sets into multiple groups. The groups are formed such that the objects in a group are more similar to each other than they are to objects in other groups [1–5]. Clustering is an important research hotspot in data mining because it is very important for revealing inherent, latent, and unknown knowledge or rules in the real world. It is widely applied in a variety of fields, such as intelligent computing, information retrieval, biology, psychology, and economics [6–9]. However, with the current rapid growth in data volume and data diversity and with limited prior knowledge of data (such as categories or class labels), effective clustering is a challenging task. Therefore, more efforts are being made to exploring efficient and effective clustering algorithms.

Rodriguez and Laio [1] published the novel clustering algorithm "clustering by fast search and find of density peaks" (CFSFDP) in *Science*. The CFSFDP algorithm must calculate only two variables: local density and minimum density-based distance, and then it draws a decision graph according to both of them. Because of the high local densities and large minimum density-based distances of clustering centers, potential clustering centers can be identified from the decision graph using visual judgment. The remaining objects are then grouped into clusters according to certain rules. Compared with other clustering algorithms, the advantages of CFSFDP are as follows:

(1) It is simple, fast, and efficient, needing to calculate only two variables to do clustering

(2) It is not necessary to do an iterative calculation of objective functions in the clustering process

(3) It can be done on datasets of various numbers and densities

However, one obvious imperfection remains in CFSFDP that needs to be resolved. The problem stems from the dependence on visual judgment in determining clustering centers. In most cases, clustering centers can be correctly identified in this manner. But for datasets with a large volume of data or complex decision graphs, it is difficult to identify the correct clustering centers using CFSFDP.

To solve this problem, this paper integrated the comprehensive metrics of CFSFDP with the distances between potential clustering centers to detect clustering centers synthetically and automatically, after which the clustering process is continued. Because the proposed algorithm is an improved version of CFSFDP, we call it automatic cluster density peak clustering (ACDPC).

The rest of this paper is organized as follows: in Section 2, the related works on clustering and CFSFDP clustering center detection are analyzed and reviewed. In Section 3, the ACDPC algorithm is demonstrated based on CFSFDP. In Section 4, we describe the experimental results and results of testing the performance of ACDPC using three standard test datasets. Section 5 concludes the paper.

2. Related Work

2.1. Data Object Clustering Methods. Researchers have put great effort into devising and proposing highly efficient clustering algorithms. That work can be divided into seven categories:

(1) Partition-based algorithms, such as k-means [10] and k-medoids [11]. The main idea of this kind of clustering approach is that for datasets containing n objects, given the number of clusters k ($k \leq n$), the datasets are divided into k clusters by continuously optimizing certain object partitioning criteria. Partition-based algorithms are simple and efficient, but the number of clusters needs to be known in advance, and the algorithms are sensitive to the selection of initial clustering centers.

(2) Hierarchical-based algorithms [12, 13] construct a cluster tree based on data objects and then seek optimal clustering results by iteratively splitting or aggregating. Hierarchical-based algorithms are simple and efficient, but their executive processes are easily affected, and the terminating condition is difficult to determine.

(3) Density-based algorithms, such as DBSCAN [14, 15], can cluster datasets with convex shapes and noisy objects, but it is difficult to determine the density threshold. CFSFDP is eminent among the density-based clustering algorithms. CFSFDP calculates only two variables (local density and minimum density-based distance), but the determination of clustering centers is done by visual judgment and manual selection. Therefore, for datasets with complex decision graphs, it is difficult to correctly identify clustering centers.

(4) Graph-based algorithms [16, 17] first construct a graph according to the characteristics of the dataset and then divide the graph into a series of subgraphs based on set rules. Each subgraph is then regarded as a cluster. However, the problems of "neck" and "chain" in the clustering process are unsolved.

(5) Model-based algorithms [18, 19] use a given mathematical model to fit datasets and then group objects into several clusters. However, the clustering results are sensitive to the parameters of the mathematical models, and it is difficult for model-based approaches to identify clusters with different shapes and densities.

(6) Grid-based algorithms [20, 21], similar to density-based algorithms, do clustering on grid merging and segmenting, but they are not suitable for clusters with different densities.

(7) Hybrid clustering algorithms such as ensemble clustering [22–24] combine at least two kinds of the clustering algorithms mentioned above to get higher quality clustering results. Also, ensemble clustering algorithms using various strategies [25–34] to break through the limitations of base clustering algorithms have been increasingly popular in recent years. But these kinds of algorithms may have high time complexity.

2.2. Detection of Clustering Centers Based on CFSFDP. To solve the problem of identifying clustering centers for CFSFDP, researchers have also proposed various algorithms to automatically detect clustering centers. Integrating the local density and hierarchical-based algorithms, Xu et al. proposed a linear fitting method to identify potential clustering centers, which turned out to have high efficiency [35]. But when clusters are highly overlapped, the number of identified clustering centers may be higher than the correct number of clusters. Rong et al. also combined the local density approach with an improved hierarchical clustering algorithm to improve the clustering process [36], but if clusters of dataset overlap are higher, incorrect clusters may be produced. Ding et al. proposed DPC-GEV and DPC-CI to automatically identify clustering centers based on the generalized extreme value and Chebyshev's inequality, respectively [37], but this method cannot be applied to datasets with high overlapping. Chen analyzed and extracted the information of data objects using the normal distribution theory, excluded the abnormal objects, and then identified the clustering centers [38]. Again, for datasets with a high degree of overlap, the clustering result may be less than ideal. Liang and Chen integrated the divide-and-conquer strategy and the density-reachable ideas of DBSCAN to determine clustering centers [39]. However, an inappropriate cutoff distance may result in a misidentification of clustering centers. Wang and Song proposed a clustering algorithm (STClu) to automatically identify clustering centers based on comprehensive metrics γ following the long-tailed distribution [40]. However, when clusters have similar numbers

of objects and distribute as a regular grid, the number of clustering centers identified may be lower than the actual number of clusters.

3. Principles and Algorithm of ACDPC

Here, the algorithm of ACDPC is described based on CFSFDP.

3.1. Theories of ACDPC. Dataset $S = \{X_i\}_{i=1}^{N}$ where N is the total number of objects in the dataset S:

$$d(i, j) = \left\| X_i - X_j \right\|, \tag{1}$$

where $d(i, j)$ is the distance from the object X_i to X_j in S, and the objects X_i and X_j have 2-D or higher-dimensional features.

The local density ρ_i of any object X_i in S is defined as

$$\rho_i = \sum_j \exp\left(-\frac{d(i, j)^2}{d_c^2}\right), \tag{2}$$

where d_c is the cutoff distance, which is represented as

$$d_c = d_{[N_d * \frac{p}{100}]}, \tag{3}$$

where $N_d = \begin{pmatrix} N \\ 2 \end{pmatrix}$ $d_c = d_{[N_d * (p/100)]} \in D = [d_1, d_2, d_3, \ldots, d_{N_d}]$, d, sorted in ascending order, is the set of the distance between every two objects in S, $N_d * (p/100)$ is a subscript of $d_{[N_d * (p/100)]}$, $[*]$ is the ceiling function, and P is the percent of the total number of objects in the dataset. The value of p varies from 1% to 2%.

The minimum density-based distance δ_i is defined as the minimum distance between the object X_i and any other higher density objects:

$$\delta_i = \min_{j:\, \rho_j > \rho_i} d\left(X_i, X_j\right). \tag{4}$$

For objects with the highest local density, the minimum density-based distance is defined as max $(d(X_i, X_j))$.

A decision graph can be constructed for each object X_i in S, after calculating the local density and the minimum density-based distance. According to the large size of both ρ and δ values of clustering centers, potential clustering centers are identified by observing the decision graph.

The comprehensive metric γ_i of the object X_i is defined as

$$\gamma_i = \rho_i \cdot \delta_i. \tag{5}$$

Because ρ and δ values are large for clustering centers, their corresponding γ values are also large. Conversely, γ values of nonclustering centers are small. Therefore, there are large gaps between clustering centers and nonclustering centers. Generally, clustering centers can be detected by observing the decision graph and comprehensive metrics sorting figures. But for large datasets or complex decision graphs, it is difficult to select clustering centers. To solve this problem, this paper proposes an algorithm to automatically detect clustering centers.

Because comprehensive metrics γ and distances between potential clustering centers are always related [41], they can be integrated to automatically identifying clustering centers.

$\{T_i\}_{i=1}^{N}$ is the subscript of the descending order of $\{\gamma_i\}_{i=1}^{N}$:

$$\gamma_{T_1} \geq \gamma_{T_2} \geq \cdots \geq \gamma_{T_N}. \tag{6}$$

Let Dm_{T_i} represent the minimum density-based distance of an undetermined clustering center X_{T_i}:

$$\mathrm{Dm}_{T_i} = \delta_{T_i} - 0.7 * d_c. \tag{7}$$

We improved the algorithm proposed by Zhao [41, 42] to recognize the clustering centers. Discriminant distance φ_{T_i} is defined as

$$\varphi_{T_i} = \frac{\sqrt{5} - 1}{2} \cdot \left(\left[(1/2)\left(\delta_{T_1} + \delta_{T_2} - 2 * \mathrm{dc}\right) + \frac{\sum_{i>2}\left(\gamma_{T_{(i-1)}} / \left(3/2 * \gamma_{T_{(i-2)}}\right)\right) * \left(\delta_{T_{i-1}} - d_c - \ln\left(\delta_{T_{i-1}}\right)\right)]}{(i-1)} \right), \tag{8}$$

where δ_{T_i} is the minimum density-based distance according to γ ,sorted in descending order, and i is the number of determinate clustering centers.

The discriminant condition of clustering centers is defined as [41]

$$\mathrm{Dm}_{T_i} > \varphi_{T_i}. \tag{9}$$

If X_{T_i} can follow the discriminant condition, it is defined as the clustering center. The objects with the largest and second-largest γ are first defined as the clustering centers.

$C = \{C_j\}_{j=1}^{M}$ is the set of M clustering centers, where $M < N$. First, X_{T1} is defined as an intrinsic member of C. Second, if object X_{T_i} follows equation (9), it will be added to C. Otherwise, the identification of clustering centers is terminated. Third, the number of clustering centers M is output. Finally, the remaining objects are clustered in Algorithm 1.

3.2. The Process of the ACDPC Algorithm. The detailed algorithm is described in pseudocode as follows:

4. Experiment and Discussion

4.1. Datasets. Standard clustering datasets were used to evaluate the effectiveness and robustness of ACDPC. These 2-D datasets come from http://people.sissa.it/laio/Research/Res_clustering.Php and http://cs.joensuu.fi/sipu/datasets/. The details of these datasets are as follows:

Input: datasets $S = \{X_i\}_{i=1}^{N}$, parameter P;
Output: clustering result;
(1) RhoSet = Ø, DeltaSet = Ø, and GammaSet = Ø;
//Part 1: Metric extraction
(2) distanceMatrix = DistanceFunction (S); //Calculate distance according to equation (1);
(3) Calculate the cutoff distance dc according to equation (3);
(4) RhoSet = F_ρ (distanceMatrix, d_c); //Calculate ρ
(5) DeltaSet = F_δ (distanceMatrix, Rhoset); //Calculate δ
(6) GammaSet = Rhoset·DeltaSet; //$\gamma = \rho \cdot \delta$
//Part 2: clustering center identification
(7) γ_{T_i} = sort (GammaSet, "descend"); //Sort GammaSet in descending order to get a set of ordered statistics γ, $\{T_i\}_{i=1}^{N}$ indicates the subscript of GammaSet in descending order
(8) Calculate Dm_{Ti} according to equation (6);
(9) Calculate the discrimination distance φ_{T_i} according to equation (8);
(10) **while** $i > 1$ **do**
(11) **If** ($Dm_{T_{(i+1)}} > \varphi_{T_{(i+1)}}$)
{
(12) $M = i$;
}
(13) **Else**
{
(14) **Break**;
}
(15) **end**
(16) Identify the objects corresponding to $\{X_{T1}, X_{T2}, ..., X_{T(i)}\}$ as the clustering centers $\{C_1, C_2, ..., C_M\}$, and label C_i as i;
//Part 3: Object clustering
(17) **for** $i = 1$ to N **do**
(18) **if** X_i is unlabeled then
(19) Mark X_i the with label of its nearest neighbor with higher ρ;
(20) **end**
(21) **end**
(23) return;

Algorithm 1: ACDPC.

4.1.1. S Sets. Made up of four subsets (*S*1, *S*2, *S*3, and *S*4) with different degrees of overlap. Each subset contained 5,000 objects and 15 Gaussian clusters. A larger degree of overlap makes it more difficult to identify clustering centers; therefore, the *S* sets can be used to evaluate the clustering performance of ACDPC on datasets with differing degrees of overlap.

4.1.2. Shape Sets. Made up of five subsets (Aggregation, R15, D31, Five-Gaussian, and Spiral). Aggregation contained 788 objects and 7 Gaussian clusters, D31 contained 3,100 objects and 31 Gaussian clusters, R15 contained 600 objects and 15 Gaussian clusters, Five-Gaussian contained 4,000 objects and 5 Gaussian clusters, and Spiral contained 312 objects and 3 Gaussian clusters. Because these subsets contained clusters with various shapes, proximities, orientations, and densities, they could be used to evaluate the performance of ACDPC in identifying clustering centers and clustering accuracy for complex datasets.

4.1.3. Birch1. A dataset contained 100 Gaussian clusters and 100,000 objects. The clustering centers were arranged in 10×10 regular grids, and the number of objects in each cluster was almost equal. To improve the experimental efficiency, nine clusters were selected, with the nine clustering centers arranged in 3×3 regular grids. This kind of dataset was used to evaluate the efficiency of ACDPC for regular distributions with a similar number of objects in each cluster.

The correct clustering centers and object labels of the above datasets and subsets were known in advance, except for the object labels of Five-Gaussian.

4.2. Experimental Results and Analysis. The ACDPC algorithm in Section 3.2 was implemented in *C*++, and three groups of datasets were loaded to test. The experimental environment was Windows 10 64 bit running on an Intel Core i7-4770 CPU, with 8 GB of memory and a 1 TB hard disk. To assess the clustering performance of ACDPC intuitively, in this paper, the clustering results are shown with 2-D figures. Small circles of various colors are used in the figures to indicate that objects belong to different clusters. The p-value of the cutoff distance d_c was uniformly set to 2%. The same datasets were clustered by CFSFDP, STClu, DPC-CI, and DBSCAN to compare their performances. For DPC-CI, the parameter ε was set to optimal value 2.

4.2.1. Results and Analysis on S Sets. Clustering results by ACDPC on the S sets are shown in Figure 1. Figures 1(a) and

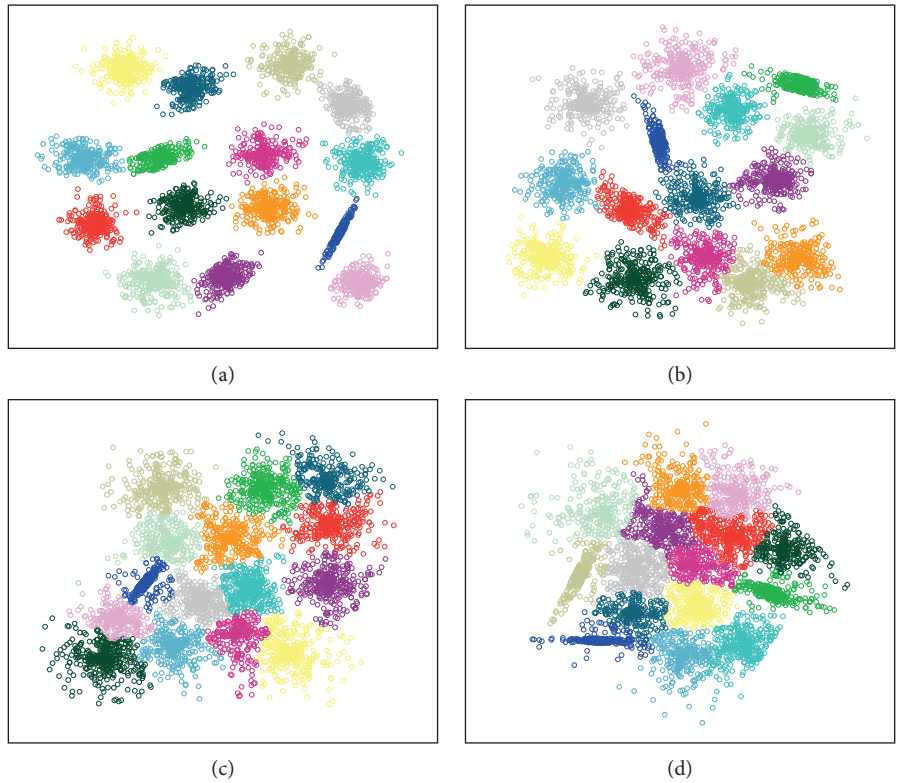

Figure 1: The results of ACDPC for S sets: (a) Subset $S1$, (b) Subset $S2$, (c) Subset $S3$, and (d) Subset $S4$.

1(b) show that for data subsets $S1$ and $S2$ with less overlap, correct clustering centers can be effectively identified. Figures. 1(c) and 1(d) show that for data subsets $S3$ and $S4$, with greater overlap, ACDPC can still recognize the correct clusters. From the corresponding decision graphs (shown in Figure 2), it can be seen that the first 15 objects with larger ρ-δ were distant from other objects, so these were selected as the clustering centers by CFSFDP. However, as the degree of clustering overlap increased, some clustering center objects were closer to other objects in the decision graphs, which created difficulties for CFSFDP in identifying the correct clustering centers. Moreover, the cluster results for STClu and DBSCAN show that the correct clustering centers could also be identified. For DPC-CI, it could find the number of clusters for subsets $S1$, $S2$, and $S3$, but it failed for $S4$.

Clustering centers were identified by five different algorithms on S sets, and the statistics of the number of clustering centers are shown in Table 1. There, we can see that ACDPC, CFSFDP, STClu, and DBSCAN were all able to identify the correct clustering centers. For DPC_CI, except for the subset $S4$, it also could correctly find the clustering centers.

To quantitatively analyze the clustering results on S sets, the clustering accuracy of the five algorithms is displayed in the right-hand columns of Table 1. We can see that the clustering accuracy of ACDPC gradually decreased from $S1$ to $S4$ as the degree of clustering overlap increased. Because the process of grouping objects of ACDPC and DPC-CI was the same as that of CFSFDP, their clustering accuracy was the same when correct clustering centers could be detected. Also, if the clustering centers could not be correctly identified, the clustering accuracy was not calculated in this paper. Moreover, the clustering accuracy by STClu was consistent with that of ACDPC. For DBSCAN, however, the clustering accuracy was not ideal, especially for subsets of $S3$ and $S4$.

4.2.2. Results and Analysis on Shape Sets. Figure 3 shows the results of ACDPC on shape sets. The figure shows that ACDPC could recognize the correct clusters for the five subsets. But if we used CFSFDP to cluster the same subsets, we got different results. The decision graphs of these subsets (Figure 4) show that the clustering centers were easily detected with CFSFDP for the subsets Spiral and R15.

However, the decision graph of the Aggregation subset (Figure 4(a)) shows that the number of clustering centers could be 7 or 8–10. According to the decision graph of the subset D31 (Figure 4(b)), the number of clustering centers was less than 31, which obviously was not consistent with the correct number of clusters. The decision graph of the subset Five-Gaussian (Figure 4(d)) shows that the number of clustering centers could be either 5 or 6. Because CFSFDP detects clustering centers based on visual judgment, bad results may be archived for a complex dataset.

To analyze the clustering performance of ACDPC on complex datasets, STClu, DPC-CI, and DBSCAN were also used to detect clustering centers. The results show that STClu and DBSCAN were able to correctly identify the clustering centers for the five subsets. Moreover, DPC-CI failed on subsets R15 and Five-Gaussian.

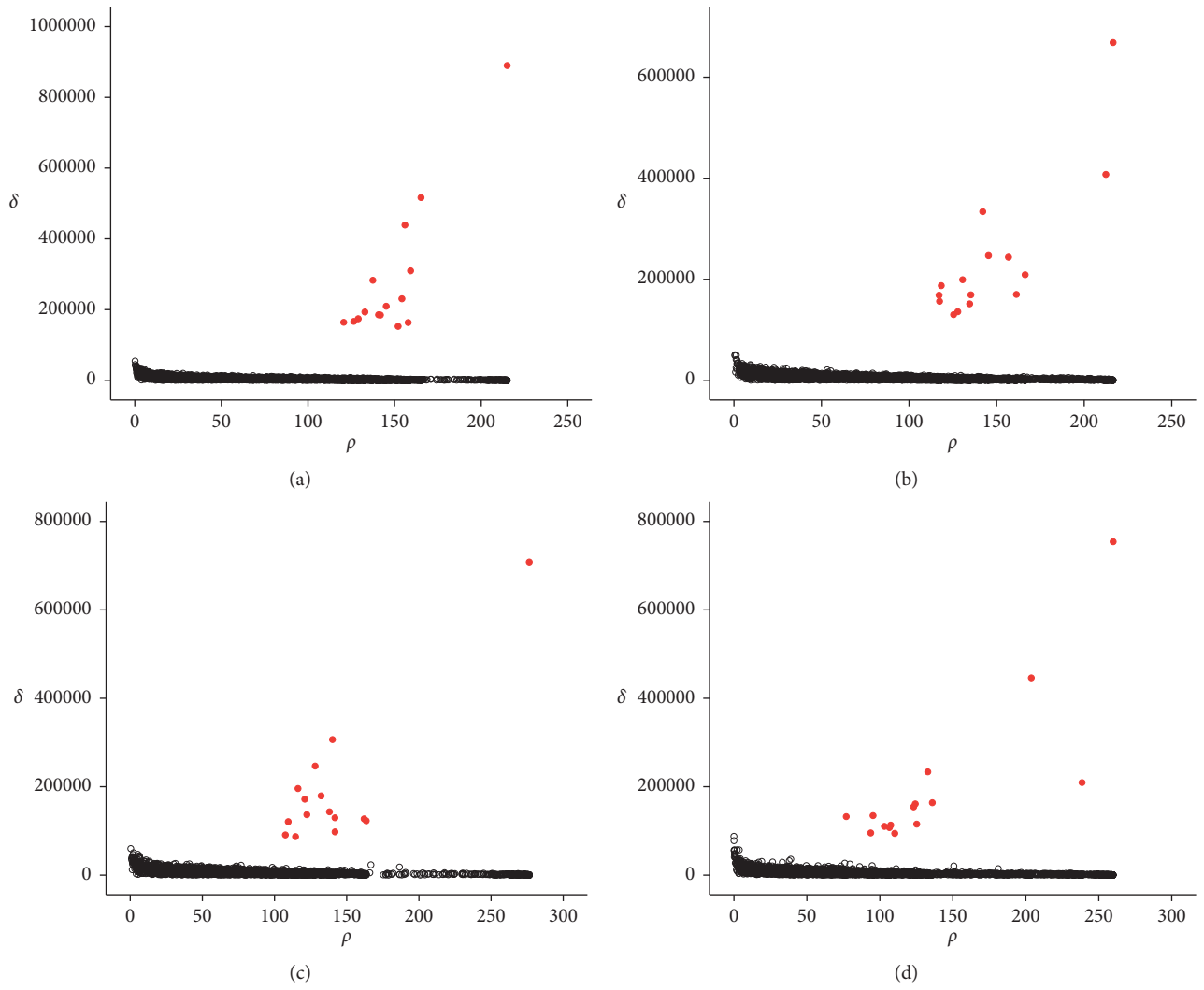

Figure 2: Decision graphs for S sets, (a) Subset $S1$, (b) Subset $S2$, (c) Subset $S3$, and (d) Subset $S4$.

Table 1: Number of clustering centers and accuracy for various algorithms on S sets.

Algorithm	Subset							
	$S1$		$S2$		$S3$		$S4$	
	Clustering centers	Accuracy (%)	Clustering centers	Accuracy (%)	Clustering centers	Accuracy (%)	Clustering centers	Accuracy (%)
ACDPC	**15**	**99.7**	**15**	97.2	**15**	**90.5**	**15**	**87.2**
CFSFDP	**15**	**99.7**	**15**	97.2	**15**	**90.5**	**15**	**87.2**
STClu	**15**	99.0	**15**	**98.4**	**15**	90.0	**15**	**87.2**
DPC-CI	**15**	**99.7**	**15**	97.2	**15**	**90.5**	16	—
DBSCAN[a]	**15**	96.6	**15**	82.1	**15**	72.6	**15**	52.1

[a]The parameters of DBSCAN (S1 : Eps = 30,000, MinPts = 20; S2 : Eps = 25,000, MinPts = 23; S3 : Eps = 23,000, MinPts = 24; S4 : Eps = 20,700, MinPts = 35).

To compare the performance of ACDPC, CFSFDP, STClu, DPC-CI, and DBSCAN in identifying clustering centers for shape sets, we listed the results in Table 2. Table 2 shows that ACDPC, STClu, and DBSCAN could accurately determine the clustering centers of all five subsets, but neither CFSFDP nor DPC-CI could well recognize the clustering centers on shape sets.

Clustering accuracy was also calculated for these five algorithms, with the results shown in the accuracy columns of Table 2. Because the object labels of the subset Five-Gaussian were unknown, its accuracy is omitted. Table 2 shows that for the remaining subsets, the clustering accuracy with ACDPC was higher than 98%. Therefore, ACDPC achieved good clustering results for datasets containing arbitrary shapes,

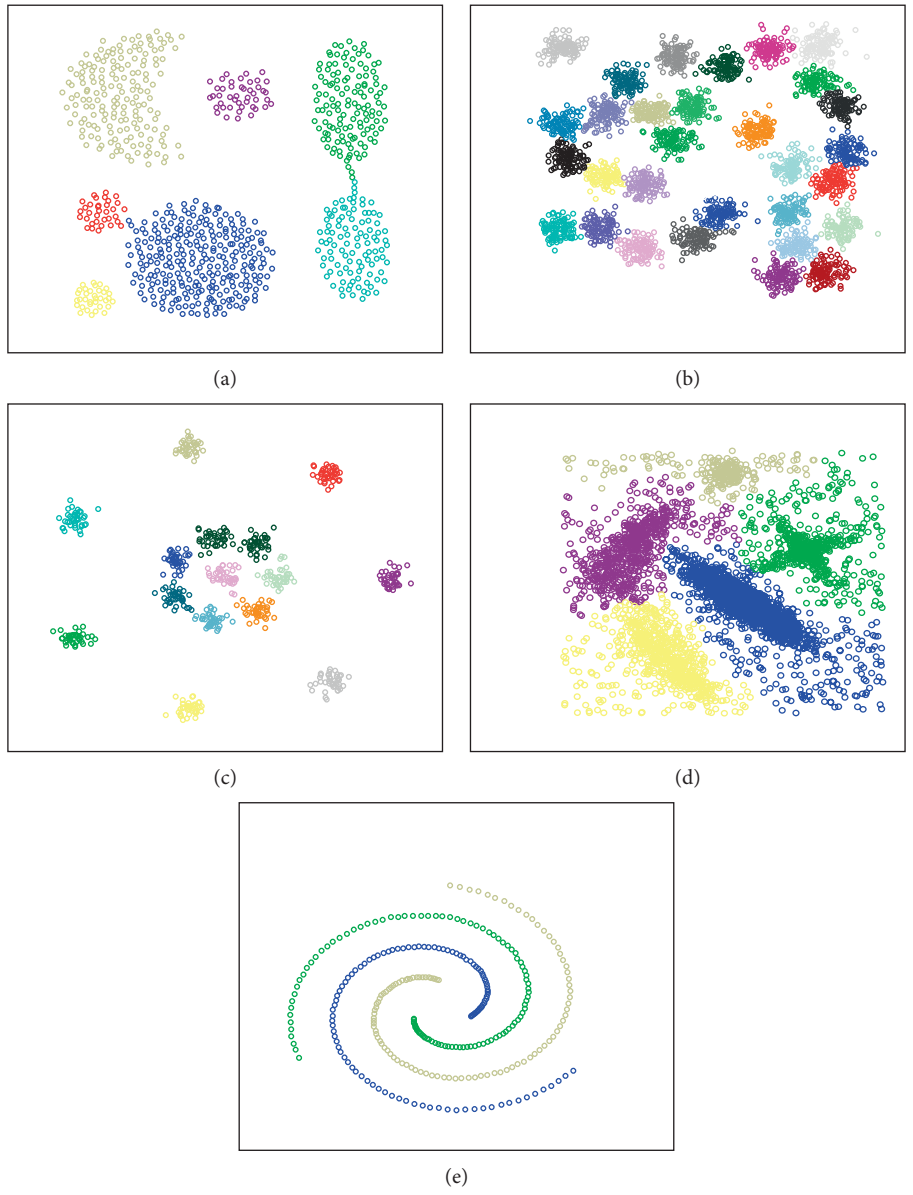

Figure 3: Results of ACDPC for shape sets: (a) Subset Aggregation, (b) Subset D31, (c) Subset R15, (d) Subset Five-Gaussian, and (e) Subset Spiral.

proximity, orientation, and density. The clustering accuracies for the other four subsets using STClu and DBSCAN were also high, but for the subsets Aggregation, R15, and D31, they were nonetheless lower than those of ACDPC. In sum, the overall experimental results on the four groups of data subsets show that the clustering accuracy of ACDPC was superior to that of STClu and DBSCAN.

4.2.3. Results and Analysis on Birch1. Figure 5 shows the clustering results for ACDPC on the dataset Birch1. Figure 5 shows that nine clusters could be identified. From the decision graph (Figure 6), the number of clustering centers could also be correctly detected. The number of identified clustering centers with these five algorithms is shown in Table 3. The results show that ACDPC, CFSFDP, DPC-CI, and DBSCAN can determine the correct clustering centers. However, the number of clustering centers identified by STClu is 3, which is much lower than the correct number.

Table 3 shows the clustering accuracy for the five different algorithms. The clustering accuracy with ACDPC was 98.4%, showing that ACDPC can not only find clustering centers correctly but also achieve high clustering accuracy. The clustering accuracy of DBSCAN was 84.4%, but still poorer than ACDPC.

In conclusion, it is apparent that ACDPC performs better than the other algorithms on three test cases. Except for the DBSCAN algorithm, none of them were able to identify all clustering centers in the three datasets. However, a problem with the DBSCAN algorithm is that the parameters Eps and MinPts were difficult to find. In

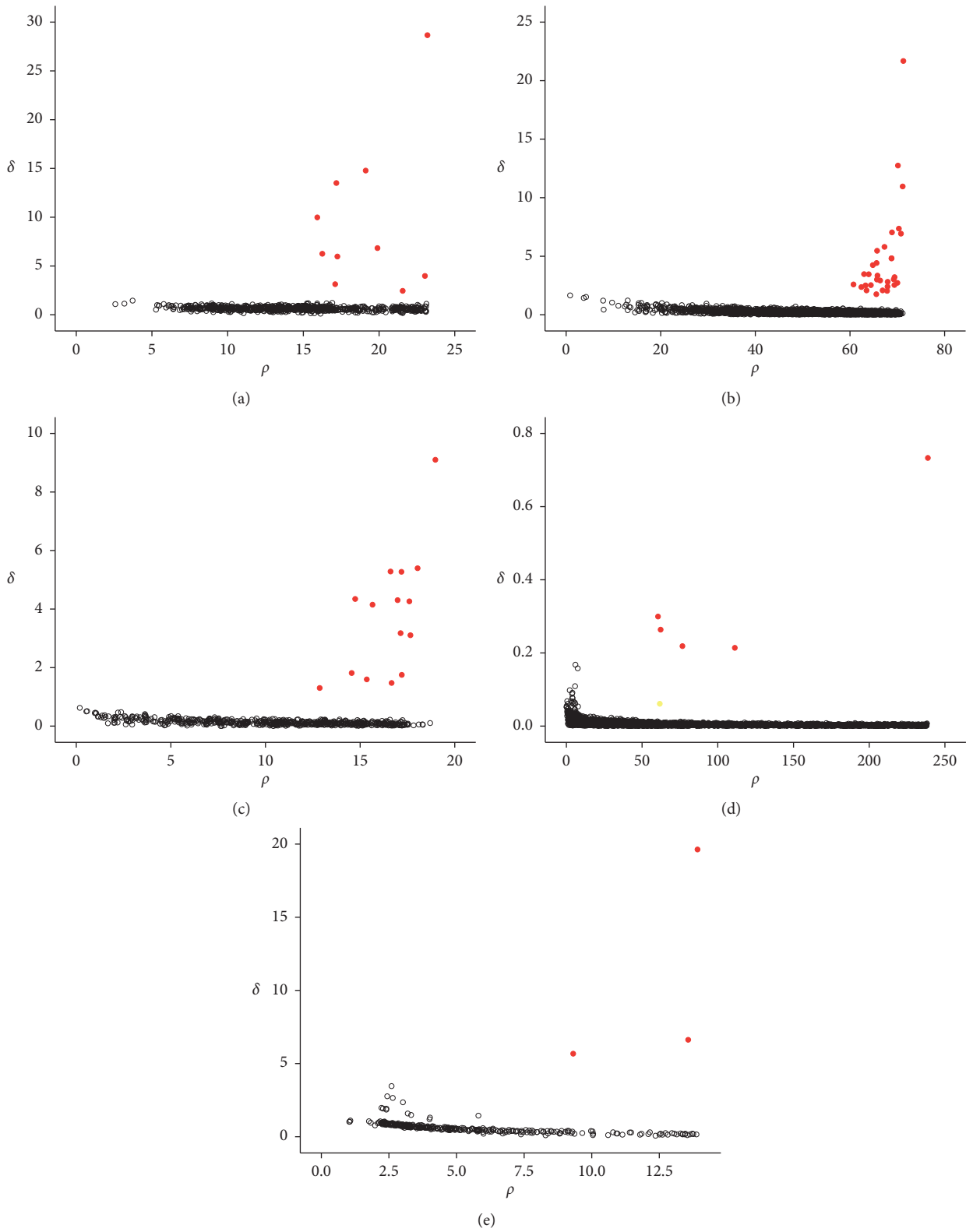

Figure 4: Decision graphs for shape sets, (a) Subset Aggregation, (b) Subset D31, (c) Subset R15, (d) Subset Five-Gaussian, and (e) Subset Spiral.

clustering accuracy, the allocation strategy of ACDPC was the same as that of both CFSFDP and DPC-CI, so their performance was equal. Moreover, the clustering accuracy of ACDPC was superior to that of STClu on the shape sets, and approximately the same on the other two groups of datasets. It was better than DBSCAN in three datasets. In

Table 2: Number of clustering centers and accuracy for various algorithms on shape sets.

Algorithm	Subset									
	Aggregation		R15		D31		Five-Gaussian		Spiral	
	Clustering centers	Accuracy (%)	Clustering centers	Accuracy (%)	Clustering centers	Accuracy (%)	Clustering centers	Accuracy (%)	Clustering centers	Accuracy (%)
ACDPC	**7**	**99.9**	**15**	**99.7**	**31**	**98.4**	**5**	—	**3**	**100**
CFSFDP	{7,8,9,10}	—	**15**	**99.7**	28	—	{5, 6}	—	**3**	**100**
STClu	**7**	98.5	**15**	90.3	**31**	93.5	**5**	—	**3**	**100**
DPC-CI	**7**	**99.9**	13	—	**31**	**98.4**	1	—	**3**	**100**
DBSCAN[a]	**7**	96.7	**15**	95.9	**31**	90.4	**5**	—	**3**	**100**

[a]The parameters of DBSCAN (Aggregation: Eps = 1.8, MinPts = 15; R15 : Eps = 0.3, MinPts = 5; D31 : Eps = 0.9, MinPts = 35; Five-Gaussian: Eps = 0.04, MinPts = 35; and Spiral: Eps = 2.2, MinPts = 5).

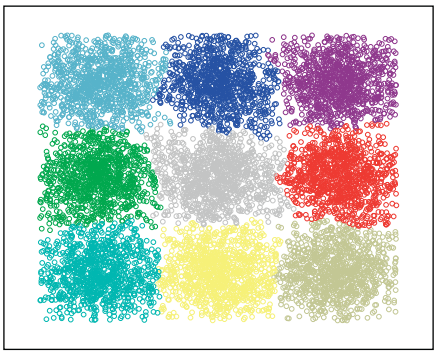

Figure 5: Results of ACDPC on Birch1.

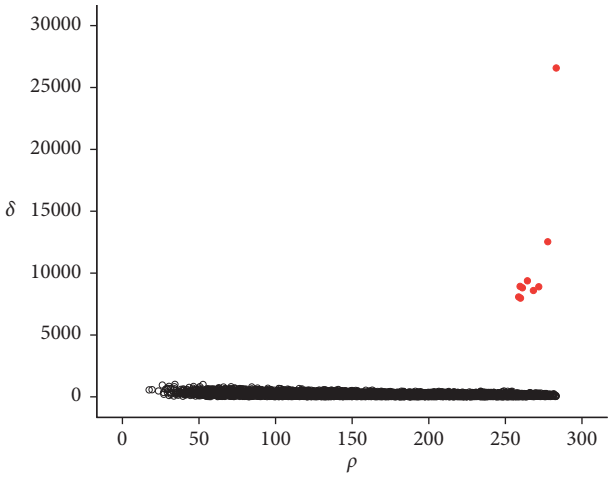

Figure 6: Decision graph on Birch1.

Table 3: Number of clustering centers and accuracy for various algorithms on the Birch1 dataset.

Algorithm	Clustering centers	Accuracy (%)
ACDPC	**9**	**98.4**
CFSFDP	**9**	**98.4**
STClu	3	—
DPC-CI	**9**	**98.4**
DBSCAN[a]	**9**	84.4

Note. Parameter of DBSCAN (Birch1: Eps = 11 000, MinPts = 50).

addition, the time complexity of the ACDPC algorithm is approximately $O(N^2)$, where N is the number of objects.

To further analyze the experimental results in term of accuracy, we use the paired t-test [27, 43] (with $p < 0.05$) to evaluate the statistical significance of the differences between proposed method and state-of-the-art methods, the results are shown in Table 4. In the average row, the performances (in terms of the accuracy) of different algorithms averaged over all 9 datasets (except Five-Gaussian) have been presented. In

Table 4: The performance of the different state-of-the-art algorithms and the proposed method in terms of accuracy on 9 datasets.

Name	CFSFDP	STClu	DPC-CI	DBSCAN	ACDPC
Average (%)	93.8	87.8	94.6	85.9	96.8
W-d-l	2-7-0	3-6-0	2-7-0	8-1-0	—

Table 5: Number of clustering centers with varied p in ACDPC.

Dataset	Subset	Clustering centers		
		p = 1, (%)	p = 1.5 (%)	p = 2 (%)
S Sets	*S*1	**15**	**15**	**15**
	*S*2	**15**	**15**	**15**
	*S*3	**15**	**15**	**15**
	*S*4	**15**	**15**	**15**
Shape sets	Aggregation	8	8	**7**
	R15	**15**	**15**	**15**
	D31	**31**	**31**	**31**
	Five-Gaussian	**5**	**5**	**5**
	Spiral	**3**	**3**	**3**
Birch1	—	**9**	**9**	**9**

the second row, a triple b-e-w indicates the number of that ACDPC method is better-than/equal-to/worse-than other methods. The results reported in Table 4 indicate the proposed method outperforms the state-of-the-art methods.

4.3. Sensitivity to Parameters of ACDPC. The parameter p in equation (3) to calculate the cutoff distance d_c affects the calculation of local density ρ. Furthermore, p will influence the determination of clustering centers. To make the experiment more convincing, in Section 4.1, we uniformly set the p-value to 2%. To further analyze ACDPC's performance in detecting clustering centers, we tried various p-values. For a dataset with N objects, the default value of p follows with CFSFDP in the range of 1% to 2%. Therefore, the experiments were run with p set to 1.5% and 1%.

Table 5 shows that ACDPC can determine the number of correct clustering centers of these datasets when p is set to 1.5% or 1%, except for the Aggregation subset. The main reason for the exception is that the number of objects in each cluster and the densities among clusters are greatly different in this subset, so the selection of p-value must be more rigorous. But for the varied ranges of p from 1.75% to 2.25%, the number of correct clustering centers can be detected. From the results, we found that ACDPC was robust when p was set within a suitable range, and the clustering centers could be automatically and effectively identified for various kinds of datasets.

5. Conclusions

To overcome the requirement to manually detect clustering centers in CFSFDP, this paper proposes an automatic determination algorithm called ACDPC based on the combination of comprehensive metrics and distances between potential clustering centers. Through comprehensive experiments and comparison with some classic and state-of-the-art algorithms, the ACDPC algorithm was demonstrated to be effective and robust. ACDPC can correctly determinate clustering centers for datasets with various densities, shapes, or distributions, and the clustering accuracy is excellent.

We also tested the performance of ACDPC for high-dimensional datasets (Dim 32) and real-world datasets (the Olivetti face data). Because the number of clustering centers and the clustering accuracy with ACDPC were the same as those of the other algorithms, the results of those experiments are not shown in this paper.

As further work, we will explore the following aspects: (1) the time complexity of the ACDPC algorithm is still high; we would like to reduce it. (2) The calculation of local density, whether by ACDPC or CFSFDP, does not apply well to datasets whose clusters have convex shapes; this needs further improvement. (3) We would like to improve the object-allocating strategy to get better clustering results.

Data Availability

The data used to support the findings of this study are included within the article. These data sets can be accessed from http://people.sissa.it/laio/Research/Res_clustering.Php and http://cs.joensuu.fi/sipu/datasets/.).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Key Fund of National Natural Science Foundation of China (grant no. 41631175) and the National Key Research and Development Program of China (grant no. 2017YFB0503500).

References

[1] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[2] R. Liu, H. Wang, and X. Yu, "Shared-nearest-neighbor-based clustering by fast search and find of density peaks," *Information Sciences*, vol. 450, pp. 200–226, 2018.

[3] H. Parvin and B. Minaei-Bidgoli, "A clustering ensemble framework based on elite selection of weighted clusters," *Advances in Data Analysis and Classification*, vol. 7, no. 2, pp. 181–208, 2013.

[4] M. Mojarad, H. Parvin, S. Nejatian, and V. Rezaie, "Consensus function based on clusters clustering and iterative fusion of base clusters," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 27, no. 1, pp. 97–120, 2019.

[5] H. Alizadeh, M. Yousefnezhad, and B. M. Bidgoli, "Wisdom of Crowds cluster ensemble," *Intelligent Data Analysis*, vol. 19, no. 3, pp. 485–503, 2015.

[6] Y.-W. Chen, D.-H. Lai, H. Qi, J.-L. Wang, and J.-X. Du, "A new method to estimate ages of facial image for large database," *Multimedia Tools and Applications*, vol. 75, no. 5, pp. 2877–2895, 2016.

[7] J. Zhong, P. W. Tse, and Y. Wei, "An intelligent and improved density and distance-based clustering approach for industrial survey data classification," *Expert Systems with Applications*, vol. 68, pp. 21–28, 2017.

[8] Y. Shi, Z. Chen, Z. Qi, F. Meng, and L. Cui, "A novel clustering-based image segmentation via density peaks algorithm with mid-level feature," *Neural Computing and Applications*, vol. 28, no. 1, pp. 29–39, 2017.

[9] B. Wu and B. M. Wilamowski, "A fast density and grid based clustering method for data with arbitrary shapes and noise," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 1620–1628, 2017.

[10] M. A. Masud, J. Z. Huang, C. Wei, J. Wang, I. Khan, and M. Zhong, "I-nice: a new approach for identifying the number of clusters and initial cluster centres," *Information Sciences*, vol. 466, pp. 129–151, 2018.

[11] D. Yu, G. Liu, M. Guo, and X. Liu, "An improved K-medoids algorithm based on step increasing and optimizing medoids," *Expert Systems with Applications*, vol. 92, pp. 464–473, 2018.

[12] F. Murtagh and P. Contreras, "Algorithms for hierarchical clustering: an overview, II," *WIREs Data Mining and Knowledge Discovery*, vol. 7, no. 6, p. e1219, 2017.

[13] D. Cheng, Q. Zhu, J. Huang, Q. Wu, and L. Yang, "A local cores-based hierarchical clustering algorithm for data sets with complex structures," *Neural Computing and Applications*, vol. 31, no. 11, pp. 8051–8068, 2019.

[14] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," *Kdd*, vol. 96, no. 34, pp. 226–231, 1996.

[15] D. Luchi, A. Loureiros Rodrigues, and F. Miguel Varejão, "Sampling approaches for applying DBSCAN to large datasets," *Pattern Recognition Letters*, vol. 117, pp. 90–96, 2019.

[16] C. Zhong, D. Miao, and R. Wang, "A graph-theoretical clustering method based on two rounds of minimum spanning trees," *Pattern Recognition*, vol. 43, no. 3, pp. 752–766, 2010.

[17] K. V. Bhaskar, K. T. Reddy, and S. Sumalatha, "Clustering of graphs using divisive hierarchical approach," *International Journal of Computer Science and Information Security*, vol. 14, no. 1, p. 57, 2016.

[18] D. McParland and I. C. Gormley, "Model based clustering for mixed data: clustMD," *Advances in Data Analysis and Classification*, vol. 10, 2016.

[19] M. Fop and T. B. Murphy, "Variable selection methods for model-based clustering," *Statistics Surveys*, vol. 12, pp. 18–65, 2018.

[20] J. Chen, X. Lin, Q. Xuan, and Y. Xiang, "FGCH: a fast and grid based clustering algorithm for hybrid data stream," *Applied Intelligence*, vol. 49, no. 4, pp. 1228–1244, 2019.

[21] S. Dong, J. Liu, Y. Liu, L. Zeng, C. Xu, and T. Zhou, "Clustering based on grid and local density with priority-based expansion for multi-density data," *Information Sciences*, vol. 468, pp. 103–116, 2018.

[22] E. Akbari, H. Mohamed Dahlan, R. Ibrahim, and H. Alizadeh, "Hierarchical cluster ensemble selection," *Engineering Applications of Artificial Intelligence*, vol. 39, pp. 146–156, 2015.

[23] M. Mojarad, S. Nejatian, H. Parvin, and M. Mohammadpoor, "A fuzzy clustering ensemble based on cluster clustering and iterative Fusion of base clusters," *Applied Intelligence*, vol. 49, no. 7, pp. 2567–2581, 2019.

[24] A. Bagherinia, B. Minaei-Bidgoli, M. Hossinzadeh, and H. Parvin, "Elite fuzzy clustering ensemble based on clustering diversity and quality measures," *Applied Intelligence*, vol. 49, no. 5, pp. 1724–1747, 2019.

[25] M. Yousefnezhad, A. Reihanian, D. Zhang, and B. Minaei-Bidgoli, "A new selection strategy for selective cluster ensemble based on Diversity and Independency," *Engineering Applications of Artificial Intelligence*, vol. 56, pp. 260–272, 2016.

[26] M. Yousefnezhad, S. J. Huang, and D. Zhang, "WoCE: a framework for clustering ensemble by exploiting the wisdom of crowds theory," *IEEE Transactions on Cybernetics*, vol. 48, no. 2, pp. 486–499, 2017.

[27] F. Rashidi, S. Nejatian, H. Parvin, and V. Rezaie, "Diversity based cluster weighting in cluster ensemble: an information theory approach," *Artificial Intelligence Review*, vol. 52, no. 2, pp. 1341–1368, 2019.

[28] H. Parvin and B. Minaei-Bidgoli, "A clustering ensemble framework based on selection of fuzzy weighted clusters in a locally adaptive clustering algorithm," *Pattern Analysis and Applications*, vol. 18, no. 1, pp. 87–112, 2015.

[29] A. Nazari, A. Dehghan, S. Nejatian, V. Rezaie, and H. Parvin, "A comprehensive study of clustering ensemble weighting based on cluster quality and diversity," *Pattern Analysis and Applications*, vol. 22, no. 1, pp. 133–145, 2019.

[30] D. Huang, C. D. Wang, and J. H. Lai, "Locally weighted ensemble clustering," *IEEE Transactions on Cybernetics*, vol. 48, no. 5, pp. 1460–1473, 2017.

[31] D. Huang, J. H. Lai, and C. D. Wang, "Robust ensemble clustering using probability trajectories," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 5, pp. 1312–1326, 2015.

[32] D. Huang, J. Lai, and C.-D. Wang, "Ensemble clustering using factor graph," *Pattern Recognition*, vol. 50, pp. 131–142, 2016.

[33] S.-o. Abbasi, S. Nejatian, H. Parvin, V. Rezaie, and K. Bagherifard, "Clustering ensemble selection considering quality and diversity," *Artificial Intelligence Review*, vol. 52, no. 2, pp. 1311–1340, 2019.

[34] H. Alizadeh, B. Minaei-Bidgoli, and H. Parvin, "To improve the quality of cluster ensembles by selecting a subset of base clusters," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 26, no. 1, pp. 127–150, 2014.

[35] J. Xu, G. Wang, and W. Deng, "DenPEHC: density peak based efficient hierarchical clustering," *Information Sciences*, vol. 373, pp. 200–218, 2016.

[36] Z. Rong, Z. Yong, F. Shengzhong, and L. Nurbol, "A novel hierarchical clustering algorithm based on density peaks for complex datasets," *Complexity*, vol. 2018, Article ID 2032461, 8 pages, 2018.

[37] J. Ding, X. He, J. Yuan, and B. Jiang, "Automatic clustering based on density peak detection using generalized extreme value distribution," *Soft Computing*, vol. 22, no. 9, pp. 2777–2796, 2018.

[38] C. Jinyin, L. Xiang, Z. Haibing, and B. Xintong, "A novel cluster center fast determination clustering algorithm," *Applied Soft Computing*, vol. 57, pp. 539–555, 2017.

[39] Z. Liang and P. Chen, "Delta-density based clustering with a divide-and-conquer strategy: 3DC clustering," *Pattern Recognition Letters*, vol. 73, pp. 52–59, 2016.

[40] G. Wang and Q. Song, "Automatic clustering via outward statistical testing on density metrics," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 8, pp. 1971–1985, 2016.

[41] J. Zhao, J. Sun, Y. Zhai, Y. Ding, C. Wu, and M. Hu, "A novel clustering-based sampling approach for minimum sample set in big data environment," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 32, no. 2, pp. 1–20, 2018.

[42] D. Li and Y. Du, *Artificial Intelligence with uncertainty*, CRC Press, Boca Raton, FL, USA, 2017.

[43] T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural Computation*, vol. 10, no. 7, pp. 1895–1923, 1998.