Int. J. Mach. Learn. & Cyber. (2018) 9:1131–1140
DOI 10.1007/s13042-017-0636-1

ORIGINAL ARTICLE

A robust density peaks clustering algorithm using fuzzy neighborhood

Mingjing Du[1] · Shifei Ding[1,2] · Yu Xue[3]

Received: 25 September 2016 / Accepted: 9 January 2017 / Published online: 8 February 2017

Abstract The density peaks (DP) clustering approach, a novel density-based clustering algorithm, detects clusters with arbitrary shape. However, this method uses a crisp neighborhood relation to calculate local density. It cannot identify the different values of the neighborhood membership degrees of the points with respect to different distances from core point. The proposed FN-DP (fuzzy neighborhood density peaks) clustering algorithm uses fuzzy neighborhood relation to define the local density in FJP (fuzzy joint points) algorithm. The proposed algorithm integrates the speed of DP clustering algorithm with the robustness of FJP algorithm. The experimental results illustrate the superior performance of our algorithm compared with the DP clustering approach.

Keywords Clustering analysis · Density peaks clustering · Fuzzy joint points · Fuzzy neighborhood relation

1 Introduction

In unsupervised learning, as one of the most important techniques, the clustering can be used in many fields data mining, pattern recognition, document retrieval, image segmentation and so on [1]. Clustering methods are generally divided into three groups: hierarchical clustering, density/neighborhood-based clustering and partitioning/centroid-based clustering. Main procedures of agglomerative hierarchical clustering are as follow. In the first step, closer objects are merged in a cluster, and then, objects a little bit far away from the previous ones are merged in the same cluster and so forth. In centroid-based methods, however, clusters are represented by centroids which have common features of some certain clusters and may not necessarily be members of the data set. In the next step the objects are assigned to these clusters according to the similarity degrees to the centroids. In hierarchical clustering, the remoteness of objects from each other is considered, while in centroid-based methods their remoteness from the centroids is considered. Partitioning-based clustering is represented by k-means [2]. Other methods of partitioning-based clustering include k-modes and fuzzy c-means (FCM) [3–5]. Density-based clustering is based on the idea that a cluster in a data space is a contiguous region of high point density, separated from other such clusters by contiguous regions of low point density. Neither do these methods require the number of clusters as input parameters, nor do they make assumptions about the underlying density or the variance within the groups that may exist in the data. To our knowledge, density-based clustering is probably introduced for the first time by Wishart [6]. DBSCAN [7] algorithm introduces density-based clustering independently to the Computer Science Community, also proposing the use of spatial index structures to achieve a scalable clustering algorithm. To address one weakness of DBSCAN: the problem of detecting meaningful clusters in data of varying density, Ankerst et al. [8] propose a cluster analysis method based on the OPTICS algorithm computing the

✉ Shifei Ding
dingsf@cumt.edu.cn

[1] School of Computer Science and Technology, China University of Mining and Technology, Xuzhou, China

[2] Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China

[3] School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China

augmented cluster-ordering of the database objects. DENCLUE [9] proposes a notion of density-based clusters using a kernel density estimation. An algorithmic framework, called GDBSCAN [10], which generalizes the topological properties of density-based clusters, can be found in Sander et al. Some of density-based clustering methods are similar to hierarchical clustering, the main difference lying in their respective linkage criterion.

Density peaks (DP) clustering [11], a density-based algorithm, is proposed by Rodriguez and Laio. Unlike traditional density-based clustering methods, this algorithm can be considered as a combination of density-based and centroid-based. It starts by determining the centroids of clusters according to two important quantities: ρ and δ. The second step is to determine which objects to merge in a cluster. Unlike traditional density-based clustering methods, it is based on the local density of objects. All objects are in descending order according to the local density. An unclassified object is assigned to the cluster that contains a certain classified object satisfying a condition. It is the nearest of all the classified objects to the unclassified object. Similarly to other density-based methods, DP clustering algorithm is able to recognize clusters with arbitrary shape. The computation speed of this algorithm is more advantageous than traditional density-based clustering methods. In order to overcome the problem that the global structure of data is not considered, Du et al. [12] propose a density peaks clustering based on k nearest neighbors (DPC-KNN). In addition, the DP clustering performs not well when it finds some pseudo cluster centers. In order to overcome this difficulty, Liang et al. propose the 3DC clustering [13] based on the divide-and-conquer strategy and the density-reachable concept.

In real life, the boundary between clusters could not be precisely defined such that some of the objects could belong to more than one cluster with different positive degrees of membership. A classic example is the fuzzy clustering. In these methods, the Fuzzy c-Means (FCM) algorithm is perhaps the most important and widely used method. The vast majority of the research work [14–21] is based on this method. These methods suppose the fuzziness of clustering with respect to the possibility of the membership of some objects into various clusters. Nasibov and Ulutagay propose a different approach of fuzziness based on a new Fuzzy Joint Points (FJP) method [22] which perceives the neighborhood concept from a level-based viewpoint which means that the objects are considered in how much detail in construction of homogenous classes. It means that the fuzzier the objects, the more similar they are. Based on this approach, many clustering methods [23–25] are proposed.

In the case that number of clusters are known, the fuzzy clustering methods show excellent performance in specifying datasets with sphere-like shape. In these methods, FJP algorithm is robust since it uses fuzzy relation in neighborhood analysis. However, it performs poorly in terms of computation time. On the other hand, DP clustering is able to detect clusters in any shape without specifying the number of clusters. In order to be able to run correctly in a wide range of change interval could be more advantageous, in this study, the fuzzy neighborhood- density peaks(FN-DP) clustering which integrates the speed of DP clustering algorithm with the robustness of FJP algorithm is proposed.

The rest of this paper is organized as follows. In Sect. 2, the DP clustering method is mentioned and some concepts about the FJP are defined. In Sect. 3, a detailed description of FN-DP is given. In Sect. 4, experimental results are presented on synthetic data sets and real world data sets. Finally, some conclusions and the intending work are given in the last section.

2 Related work

2.1 Density peaks clustering

Let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ denote a dataset of n data objects. Each object $\mathbf{x}_i, 1 \le i \le n$ has m attributes. Thus, for each $i, 1 \le i \le n$, and for $j, 1 \le j \le m$, let $x_{i,j}$ be the j-th attribute of $\mathbf{x}_i$. The Euclidean distance $d(\mathbf{x}_i, \mathbf{x}_j)$ between any points $\mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}$ can be determined as follows:

$$d(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum_{k=1}^{m} (x_{i,k} - x_{j,k})^2 \right)^{1/2}. \tag{1}$$

Unlike DBSCAN, the DP clustering finds the cluster centers before data points are assigned. Determining the cluster centers is of vital importance to guarantee good clustering results. Because it determines the number of the clusters, and affects the assignation indirectly. In the following, we will describe the calculation of ρ and δ in much more detail.

DP represents data objects as points in a space and adopts a distance metric, such as Eq. (1), as a dissimilarity between objects. Let $D = \{d_1, d_2, \dots, d_{N_d}\}$ a set of all the distances between every two points in data set, where all the distances from smallest to greatest. $N_d = \binom{n}{2}$, where N is the number of points in the dataset. Unlike DBSCAN, the neighborhood radius is determined not by the direct value, but by the percentage. d_c indicates a percentage and is the only input parameter, which is called a cutoff distance. The neighborhood radius ε is defined as:

$$\varepsilon = d_{\lceil d^{\max} \cdot d_c \rceil}, \tag{2}$$

where $\lceil \cdot \rceil$ is the ceiling function and $d^{\max} = d_{N_d} = \max d(\mathbf{x}_i, \mathbf{x}_j)$. The method takes one parameterd_c which is a percentage.

The neighborhood set of point $\mathbf{x}_i \in \mathbf{X}$ with parameter ε (ε-neighborhood set) is as follows:

$$N(\mathbf{x}_i, \varepsilon) = \left\{ \mathbf{x}_j \in \mathbf{X} \middle| d(\mathbf{x}_i, \mathbf{x}_j) < \varepsilon \right\}. \tag{3}$$

$\mu_{\mathbf{x}_i}(\mathbf{x}_j)$ denote the membership degree of the point $\mathbf{x}_j$ to the neighborhood set of the point $\mathbf{x}_i$, as follows:

$$\mu_{\mathbf{x}_i}(\mathbf{x}_j) = \begin{cases} 1 & \text{if } d(\mathbf{x}_i, \mathbf{x}_j) < \varepsilon \\ 0 & \text{otherwise} \end{cases}. \tag{4}$$

The local density ρ_i [7] of a point $\mathbf{x}_i$ is defined as:

$$\rho_i = \sum\nolimits_j \mu_{\mathbf{x}_i}(\mathbf{x}_j). \tag{5}$$

The calculation of the delta value [7], again, is quite simple. The minimum distance between the point of $\mathbf{x}_i$ and any other points with higher density, denoted by δ_i is defined as

$$\delta_i = \begin{cases} \min\limits_{j:\rho_i<\rho_j} d(\mathbf{x}_i, \mathbf{x}_j), & \text{if } \exists j \ s.t. \ \rho_i < \rho_j \\ \max\limits_{j} d(\mathbf{x}_i, \mathbf{x}_j), & \text{otherwise} \end{cases}. \tag{6}$$

When the local density and delta values for each point have been calculated, this method identifies the cluster centers by anomalously large ρ_i and δ_i. On the basis of this idea, cluster centers always appear on the upper-right corner of the decision graph. After cluster centers have been found, the DP clustering assigns remaining points to the same cluster as its nearest neighbors with higher density.

2.2 Basic concepts about the fuzzy joint points method

Let $F(\mathfrak{R}^m)$ denote the set of m-dimensional fuzzy sets of the space $\mathfrak{R}^m$. $\mu'_A:\mathfrak{R}^m \to [0,1]$ denotes the membership function of the fuzzy set $A \in F(\mathfrak{R}^m)$.

Definition 2 A conical fuzzy point $A = (a, R) \in F(\mathfrak{R}^m)$ of the space $\mathfrak{R}^m$ is a fuzzy set with membership function (Fig. 1) [22].

$$\mu'_A(x) = \begin{cases} 1 - \frac{d(a,x)}{R} & \text{if } d(a,x) < R \\ 0 & \text{otherwise} \end{cases}, \tag{7}$$

where $a \in \mathfrak{R}^m$ is the center of fuzzy point A, and R is the radius of its support supp A, where

$$\text{supp } A = \left\{ x \in \mathfrak{R}^m \middle| \mu'_A(x) > 0 \right\}. \tag{8}$$

The α-level set of conical fuzzy point $A = (a, R)$ is calculated as

$$A_\alpha = \left\{ x \in \mathfrak{R}^m \middle| \mu'_A(x) > \alpha \right\} = \{ x \in \mathfrak{R}^m | d(a,x) < R \cdot (1 - \alpha) \}. \tag{9}$$

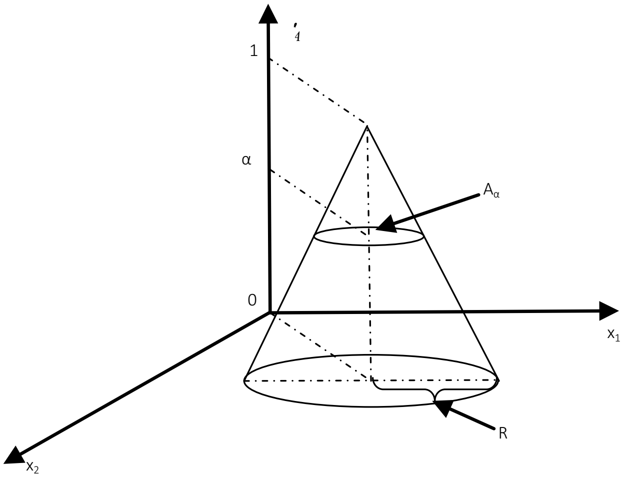


Fig. 1 Fuzzy point $A = (a, R) \in F(\mathfrak{R}^2)$ on the space $\mathfrak{R}^2$

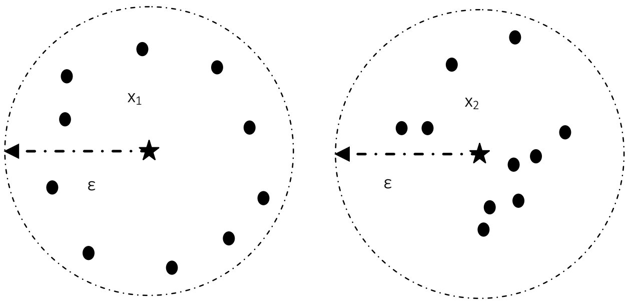


Fig. 2 Classical and fuzzy neighborhood relations

In this study, the short term "fuzzy point" is used, instead of conical fuzzy point defined in Eq. (7).

3 The proposed algorithm

Before introducing the proposed method, the fuzzy neighborhood relation is defined.

3.1 Fuzzy neighborhood relation

Points x_1 and x_2 have the same number of neighbors within $\varepsilon \leq d^{\max}$ radius (Fig. 2) [24]. There is an obvious difference between these points. It is obvious that the points x_1 and x_2 in Fig. 2 are the same according to the crisp neighborhood relation used in the DP clustering method. Because in classical case there is no difference with respect to membership degrees between points within the same neighborhood radius of core point (Fig. 3) [24]. In other words, points y_1 and y_2 have the same neighborhood membership degrees to the point x. As it is seen from Fig. 3, the membership degrees of y_1 and y_2 are both equal to 1. This study expands the neighborhood

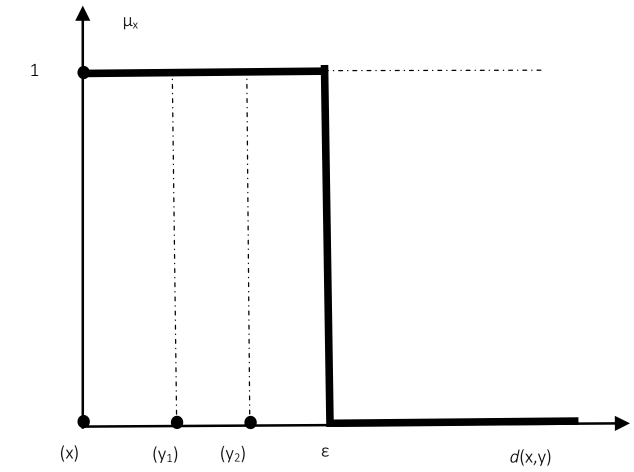


Fig. 3 The crisp neighborhood relation of the DP clustering

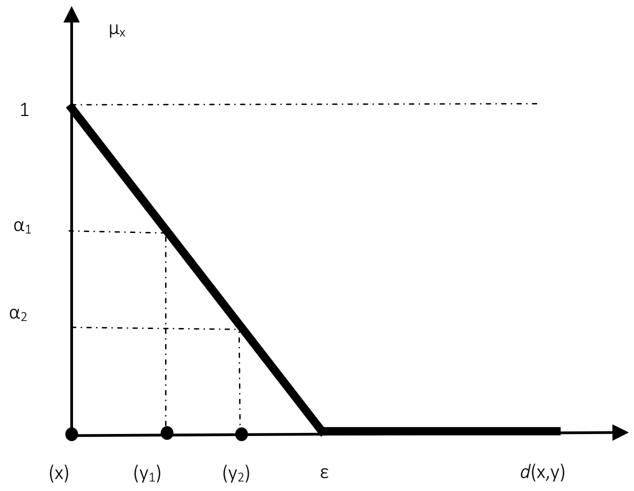


Fig. 4 The fuzzy neighborhood relation of the FN-DP clustering

set determined in Eq. (4) to the fuzzy neighborhood case. Utilizing fuzzy neighborhood function provides an advantage that there are different values of the neighborhood membership degrees of the points with respect to different distances from core point.

Note that, in order to form a fuzzy relation $\mu{:}X \times X \to [0,1]$, the idea of the membership function of the fuzzy set defined in Eq. (4) is introduced. In order to be consistent with the parameter of the original DP clustering method, the radius of the considered fuzzy points is calculated as Eq. (2) in the proposed algorithm.

Thus, such a neighborhood membership function [24] is defined as

$$\mu'_{\mathbf{x}_i}\left(\mathbf{x}_j\right) = \begin{cases} 1 - \frac{d\left(\mathbf{x}_i,\mathbf{x}_j\right)}{\varepsilon} & \text{if } d\left(\mathbf{x}_i,\mathbf{x}_j\right) < \varepsilon \\ 0 & \text{otherwise} \end{cases}. \quad (10)$$

In the previous example, if fuzzy neighborhood function is used, point x_1 will have a higher membership degree of being a core point than that of point x_2.

In neighborhood relation determined by Eq. (4), neighborhood degrees of points with varying distances to the core point will be different from each other (Fig. 3).

As it is seen from Fig. 4 [24], points y_1 and y_2 have different neighborhood membership degrees to the point x. Hence, the membership degree of y_1, i.e. α_1, is higher than the membership degree of y_2, i.e. α_2.

3.2 FN-DP

Based on the definitions given above, a new local density is redefined for fuzzy logic approach, as follows:

$$\rho_i = \sum\nolimits_j \mu'_{\mathbf{x}_i}\left(\mathbf{x}_j\right), \quad (11)$$

where μ' denotes the fuzzy neighborhood function defined by Eq. (10).

The main advantage of transformation of the DP clustering algorithm to the FN-DP clustering algorithm and using fuzzy sets theory is that the fuzzy neighborhood function that make local density more sensitive can be utilized. So the FN-DP clustering method could be more robust to the datasets various shapes and densities.

The idea of Sect. 3.1 is introduced into the DP clustering algorithm. The following algorithm is a summary of the proposed algorithm based on fuzzy neighborhood relation.

```
**Algorithm**. The proposed algorithm.
Inputs:
The samples$\mathbf{X} \in \Re^{n\times m}$
The parameter$d_c$
Outputs:
The label vector of cluster index:$\mathbf{y} \in \Re^{n\times 1}$
Method:
Step 1: Calculate distance matrix according to Eq. (1)
Step 2: Calculate $\rho_i$ for point $\mathbf{x}_i$ according to Eq. (11)
Step 3: Calculate $\delta_i$ for point $\mathbf{x}_i$ according to Eq. (6)
Step 4: Plot decision graph and select cluster centers
Step 5: Assign each remaining point to the cluster
Step 6: Return y
```

Now, the time complexity of the proposed algorithm is given. Assume that N is the number of objects in the data set. The computational complexity of the similarity matrix is $O\left(N^2\right)$. This also need $O\left(N^2\right)$ to compute the new local density. In addition, the cost of the sorting process with quick sort $O(N\log N)$. As the complexity in the assignment procedure is $O(N)$, the total time cost of the proposed algorithm is $O\left(N^2\right) + O\left(N^2\right) + O(N\log N) + O(N)\tilde{}O\left(N^2\right)$.

4 Experiments and results

In order to demonstrate the feasibility of the proposed algorithm, 12 synthetic datasets with various shapes and densities are used. To further compare FN-DP clustering algorithm based on fuzzy neighborhood relation with the original algorithm based on crisp neighborhood relation, 5 synthetic datasets are used. Moreover, by experiments on real-world datasets, the proposed method is compared with the original algorithm in terms of clustering accuracy (ACC), normalized mutual information (NMI) and adjusted Rand index (ARI) [26–31].

We conduct experiments in a desktop computer with a core i7 DMI2-Intel 3.6 GHz processor and 16 GB RAM running MATLAB 2013A. The cutoff distance d_c used in the DP clustering algorithm and the FN-DP clustering algorithm is given from 0.1 to 100% at an increment 0.1% for all 5 synthetic datasets. On the other hand, on real-world datasets, The parameter d_c is selected from the sequence {0.1% 0.5% 1% 2% 3% 4% 5% 6% 7% 8% 9% 10%} based on the clustering performance. In order to better demonstrate the feasibility of the proposed algorithm, the best results only are presented in terms of clustering accuracy.

4.1 Experiments on synthetic datasets

As shown in Fig. 5a–d, the proposed method does an excellent job in clustering datasets with spherical or ellipsoidal shape. Among them, A1, A2 and A3 datasets are large datasets with varying number of clusters. The experiment results demonstrate the robustness of the FN-DP clustering method in terms of the quantity. As shown in Fig. 5e–g, the proposed algorithm gets extraordinarily favorable performance to these three datasets with different sizes and shapes. As shown in Fig. 5h–k, the datasets S1 to S4 are two-dimensional sets with varying complexity in terms of spatial data distributions. The data sets have 5000 points around 15 clusters with a varying degrees of overlap. The performance of the proposed algorithm is perfect for data sets with varying complexity. As these experiments illustrate our algorithm is very effective in finding clusters of arbitrary shape, density, distribution and number.

4.2 Experiments for robustness

To evaluate the performances of the algorithms, "correct range percent (CRP)'' is used as an indicator to indicates the percentage of correct result range of d_c parameter to the whole [0%, 100%] interval. So the CRP criteria is calculated as follows:

$$CRP_i = \sum_j CRP_{i,j} = \sum_j \left(d_c\right)^U_{i,j} - \left(d_c\right)^L_{i,j}, \tag{12}$$

where $CRP_{i,j} = \left(d_c\right)^U_{i,j} - \left(d_c\right)^L_{i,j}$ is the j-th continuous interval of the parameter d_c, in which the algorithm can give correct results in the i-th dataset, where $\left(d_c\right)^U_{i,j}$ is the low bound and $\left(d_c\right)^L_{i,j}$ is the upper bound in the j-th continuous interval.

In addition, the strict criteria will be loosened, because even experts cannot find absolutely correct clustering results visually in some datasets. For Aggregation and R15 datasets, all cases that the accuracy is more than 99% is considered acceptable. Because of that D31 dataset has more clusters, the benchmark of the clustering accuracy is revised to 96%.

In order to show these results visually, comparisons are given as histogram (Fig. 6). The CRP values for the clusterings obtained with FN-DP are, in all the cases superior to the one obtained by DP. Flame, Twospirals and Aggregation have clusters with non-spherical shape. It is obvious that FN-DP results are significantly better than those obtained by the original algorithm for these three datasets. On R15 and D31 datasets with spherical shape, in comparisons with the DP clustering method, our method always shows a small advantage in terms of the CRP criteria. It's tempting to conclude that our algorithm does an excellent job compared with DP, when the dataset has some cluster with non-spherical shape. On the other hand, when a dataset present the non-spherical distribution, the CRP values of the clusters formed by the two methods are close. However, the proposed method performs slightly better.

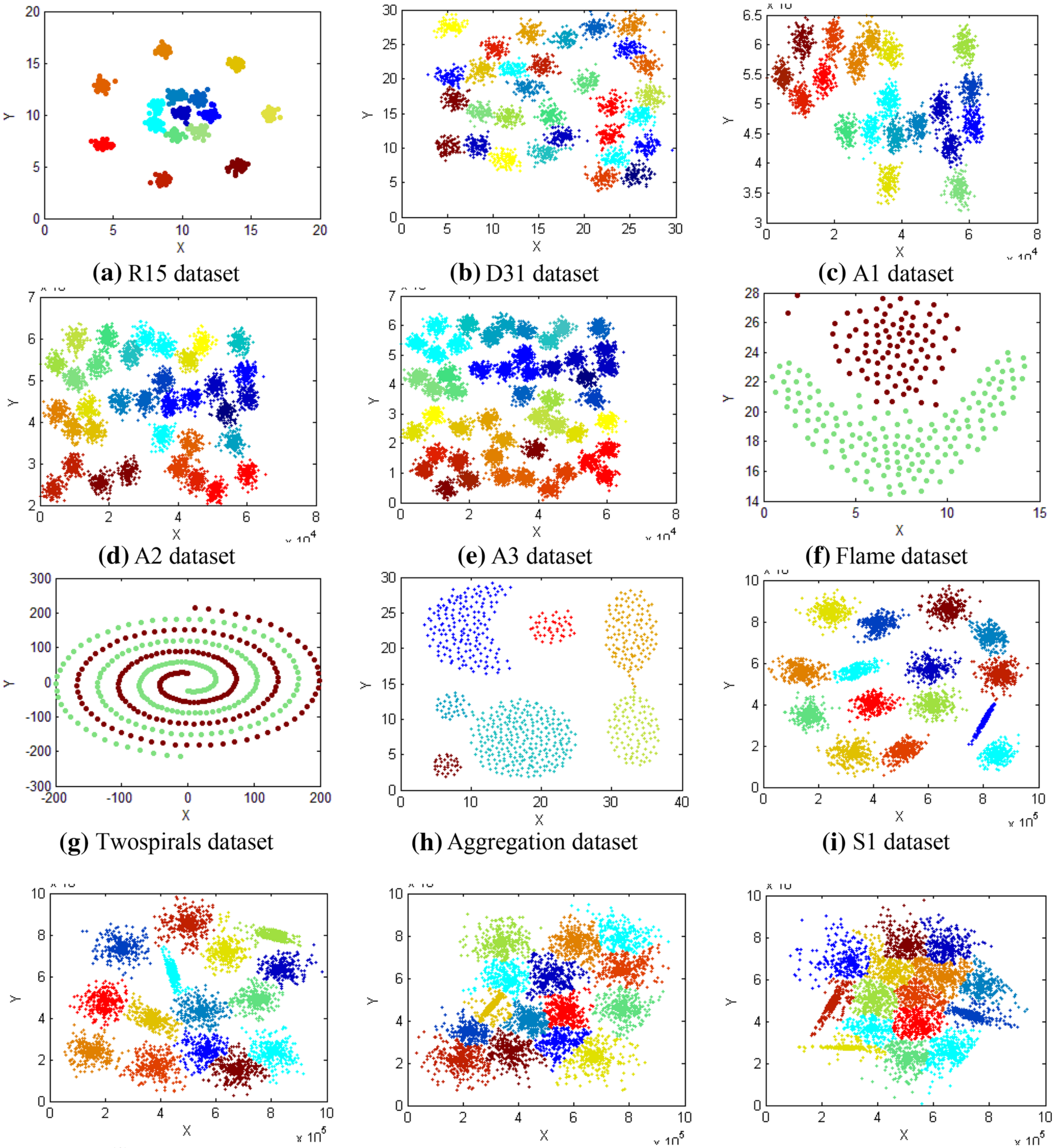
4.3 Experiments on real-world datasets

4.3.1 Real-world datasets

The real-world datasets used in the experiment also are taken from the UCI Machine Learning Repository, including Iris Plants Database (Iris), Wine Recognition Database (Wine), the Heart Disease database (Heart), Ionosphere database (Ionosphere), Wisconsin Diagnostic Breast Cancer (WDBC), Waveform Database (Waveform), Ringnorm data set (Ringnorm) and Pen-Based Recognition of Handwritten Digits (Penbased). The details of these datasets are listed in Table 1.

4.3.2 Performance on real-world datasets

Table 2 lists the clustering accuracy of our proposed algorithm and the original algorithm. In the following tables, the numbers highlighted in bold indicate that the corresponding algorithm has the best performance in terms of its corresponding evaluation. Tables 2, 3 and 4 show comparisons against DP and DBSCAN in terms of the different quality measures (ACC, NMI and ARI). In



(a) R15 dataset **(b)** D31 dataset **(c)** A1 dataset

(d) A2 dataset **(e)** A3 dataset **(f)** Flame dataset

(g) Twospirals dataset **(h)** Aggregation dataset **(i)** S1 dataset

(j) S2 dataset **(k)** S3 dataset **(l)** S4 dataset

Fig. 5 Experimental results of FN-DP on these datasets

these tables, the symbol - means that DBSCAN detects outliers on the corresponding data set. Thus, in these cases, we cannot evaluate the performance of DBSCAN using these quality measures. In addition, on Iris, Wine, Waveform and Ringnorm data sets, numbers of clusters are found accurately by DBSCAN. It only detects two clusters, one cluster, one cluster and one cluster, respectively. By comparison with the DP clustering algorithm and DBSCAN, our method obtains better clustering performance in terms of the all quality measures (ACC, NMI and ARI) on all datasets, as shown in Tables 2, 3 and 4. In the following tables, the numbers highlighted in bold

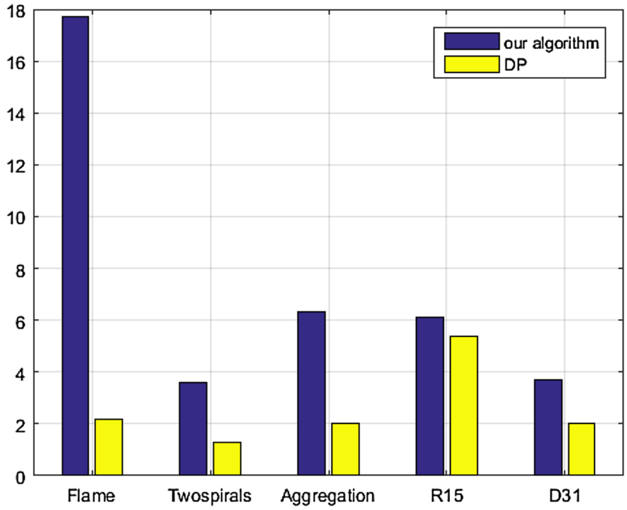


Fig. 6 Comparison of FN-DP and DP in terms of CRP

Table 1 The details of UCI data sets

Data Sets	Cluster	Dimension	Number
Iris	3	4	150
Wine	3	13	175
Heart	2	13	270
Ionosphere	2	34	351
WDBC	2	30	569
Waveform	3	21	5000
Ringnorm	2	20	7400
Penbased	10	16	10,992

Table 2 Clustering accuracy of the evaluated algorithms on real-world data sets

Data sets	Quality	FN-DP	DP	DBSCAN
Iris	ACC	**0.9667**	0.9	0.6667
	Para	d_c = 10%	d_c = 6%	$\varepsilon = 1.5 MinPts = 15$
Wine	ACC	**0.9326**	0.9157	0.3392
	Para	d_c = 6%	d_c = 4%	$\varepsilon = 4 MinPts = 10$
Heart	ACC	**0.8111**	0.8074	–
	Para	d_c = 1%	d_c = 6%	
Ionosphere	ACC	**0.7407**	0.5556	–
	Para	d_c = 0.1%	d_c = 0.5%	
WDBC	ACC	**0.8664**	0.8594	–
	Para	d_c = 4%	d_c = 3%	
Waveform	ACC	**0.5506**	0.5446	0.3392
	Para	d_c = 4%	d_c = 4%	$\varepsilon = 6 MinPts = 10$
Ringnorm	ACC	**0.5157**	0.5076	0.5049
	Para	d_c = 7%	d_c = 3%	$\varepsilon = 7.5 MinPts = 10$
Penbased	ACC	**0.7606**	0.7177	–
	Para	d_c = 0.1%	d_c = 3%	

indicate that the corresponding algorithm has the best performance in terms of its corresponding evaluation.

To explain further how to choose the parameter d_c of DP and FN-DP, we use the Heart data set as an example. Figure 7 shows the ACC values obtained by the two methods with different d_c on the Heart data set. The horizontal axis represents the parameter d_c. As explained earlier (in the beginning of Sect. 4), the parameter d_c is selected from the sequence {0.1% 0.5% 1% 2% 3% 4% 5% 6% 7% 8% 9% 10%}. The vertical axis represents the ACC value. Obviously, when we set parameter d_c to 1%, FN-DP obtains the best result (0.8111) on this data set. Nevertheless, when parameter is set to 6%, DP obtains the best result (0.8074) on the Heart data set. In this paper, the parameter d_c is selected based on the clustering performance. Thus, on the Heart data set, the parameter d_c of FN-DP is set to 1%, whereas the parameter d_c of DP is set to 6. As a result, the parameter $d_c$of the two algorithms is not the same. On other data sets, we present the best results in terms of clustering accuracy. Thus the parameter of the two algorithms may be different on each data set. Similar strategy on parameter selection has been used in some literatures [32–35]. It is interesting to note that if the parameter d_c of the two algorithms is set to 1%, the ACC value of the proposed method is slightly higher than that of DP. By contrast, if the parameter d_c of the two algorithms is set to 6%, the ACC value of the two algorithms is the same. In addition, if and only if the parameter d_c of the two algorithms is set to 7%, the ACC value of DP is slightly higher than that of FN-DP as shown in Fig. 7. In most cases, FN-DP results are significantly better than those obtained by the original algorithm on this data set. This again proves that FN-DP obtains more robust performance than the original algorithm.

Figure 8 shows the running time spent on clustering using FN-DP and DP methods. We run every algorithm 20 times on each data set and get the average. It is clear the running times of the two methods are close, within a difference of 0.0001, on the first five data sets. However, the running times of FN-DP are slightly smaller than those of DP on some larger data sets (Waveform, Ringnorm and Penbased). The reason for this result may be that MATLAB has a very strong processing capability for matrix manipulations. The code of the original algorithm does not use matrix manipulations to compute the local density. In all cases, the two methods are comparable in terms of efficiency. Our proposed method brings a boost of performance without loss of efficiency.

4.4 Experiments on image data

We test the quality of this algorithm on image data. Figure 9 shows the original images from the Berkeley database and the segmentation evaluation database [36–38].

Table 3 Normalized mutual information of the evaluated algorithms on real-world data sets

Data sets	Quality	FN-DP	DP	DBSCAN
Iris	NMI	**0.8846**	0.7869	0.5835
	Para	d_c = 10%	d_c = 6%	ε = 1.5$MinPts$=15
Wine	NMI	**0.7920**	0.7600	0.0004
	Para	d_c = 6%	d_c = 4%	ε = 4$MinPts$=10
Heart	NMI	**0.2985**	0.2898	–
	Para	d_c = 1%	d_c = 6%	
Ionosphere	NMI	**0.1395**	0.0271	–
	Para	d_c = 0.1%	d_c = 0.5%	
WDBC	NMI	**0.4540**	0.4357	–
	Para	d_c = 4%	d_c = 3%	
Waveform	NMI	**0.2620**	0.2535	0.0004
	Para	d_c = 4%	d_c = 4%	ε = 6$MinPts$=10
Ringnorm	NMI	**0.0014**	0.0004	0.0001
	Para	d_c = 7%	d_c = 3%	ε = 7.5$MinPts$=10
Penbased	NMI	**0.7377**	0.5992	–
	Para	d_c = 0.1%	d_c = 3%	

Table 4 Adjusted rand index of the evaluated algorithms on real-world data sets

Data sets	Quality	FN-DP	DP	DBSCAN
Iris	ARI	**0.9038**	0.7455	0.5681
	Para	d_c = 10%	d_c = 6%	ε = 1.5$MinPts$=15
Wine	ARI	**0.8025**	0.7562	0
	Para	d_c = 6%	d_c = 4%	ε = 4$MinPts$=10
Heart	ARI	**0.3845**	0.3753	–
	Para	d_c = 1%	d_c = 6%	
Ionosphere	ARI	**0.2231**	-0.0380	–
	Para	d_c = 0.1%	d_c = 0.5%	
WDBC	ARI	**0.5284**	0.5071	–
	Para	d_c = 4%	d_c = 3%	
Waveform	ARI	**0.2768**	0.2657	0
	Para	d_c = 4%	d_c = 4%	ε = 6$MinPts$=10
Ringnorm	ARI	**0.0009**	0.0001	0
	Para	d_c = 7%	d_c = 3%	ε = 7.5$MinPts$=10
Penbased	ARI	**0.6443**	0.5992	–
	Para	d_c = 0.1%	d_c = 3%	

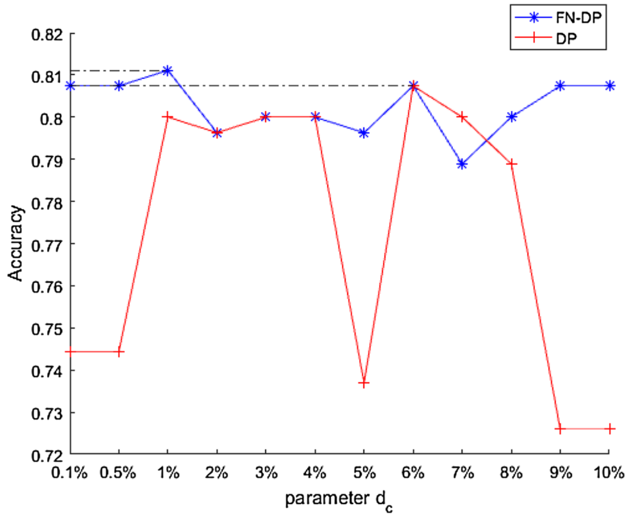


Fig. 7 The ACC values of FN-DP and DP on the Heart data set

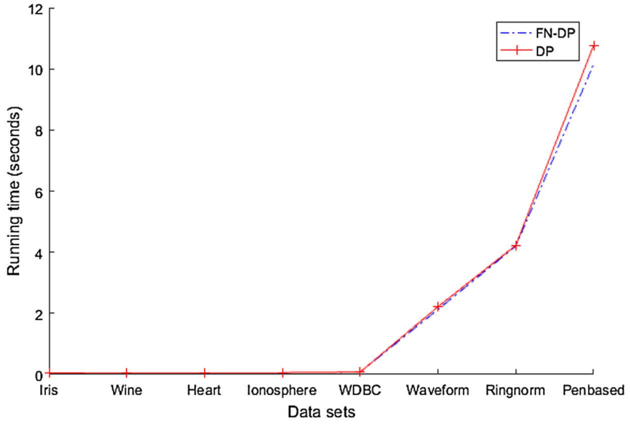


Fig. 8 Running time comparison of FN-DP and DP algorithm

Figure 10 shows image segmentation results. Unlike K-means, the number of groups is specified in advance. The number of groups may depend on the result of decision graph. Thus, FN-DP can be separated from all the other objects in the background, as shown in Fig. 10. In addition, it is also insensitive to the choice of d_c in this test. Because the same results are obtained within [0.1%, 10%] the range of d_c over these images. Experimental results show that FN-DP can be a feasible preprocessing method for image segmentation.

5 Conclusions

A new fuzzy neighborhood function is introduced into this paper and the FN-DP clustering algorithm based on this function is proposed. Experiments show that our algorithm is very effective in finding clusters with arbitrary shape, density, distribution and number. The proposed algorithm combines the speed of the DP clustering method and robustness of FJP algorithm. It is observed that our algorithm is more robust than the original algorithm to datasets with various shapes and densities. However, in this stage, the aim is not to find the optimal values of the parameters, but to prove that one can get more

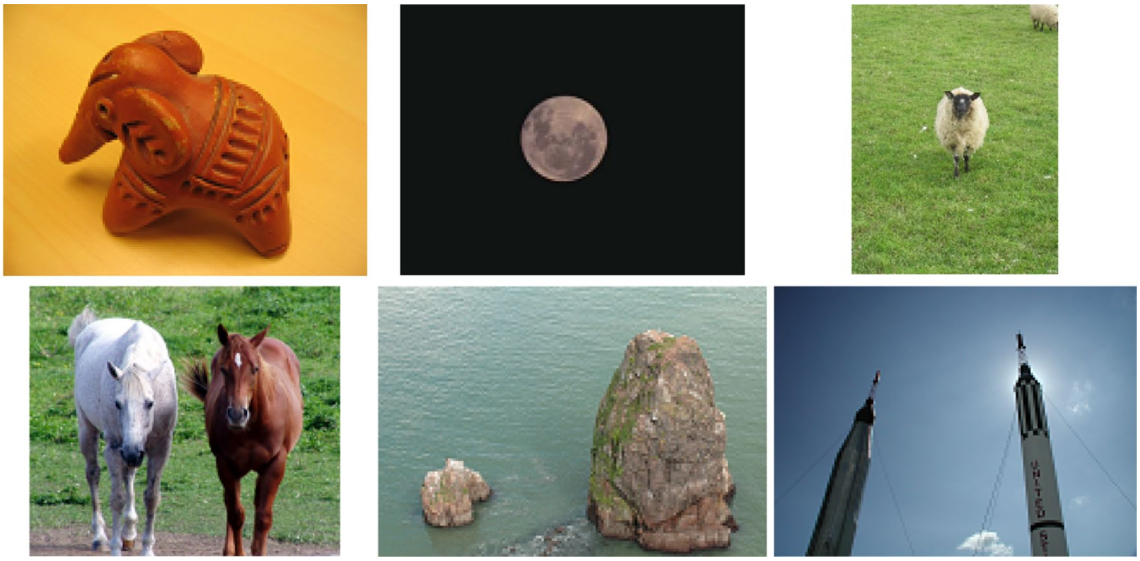


Fig. 9 The original images from the Berkeley database and the segmentation evaluation database

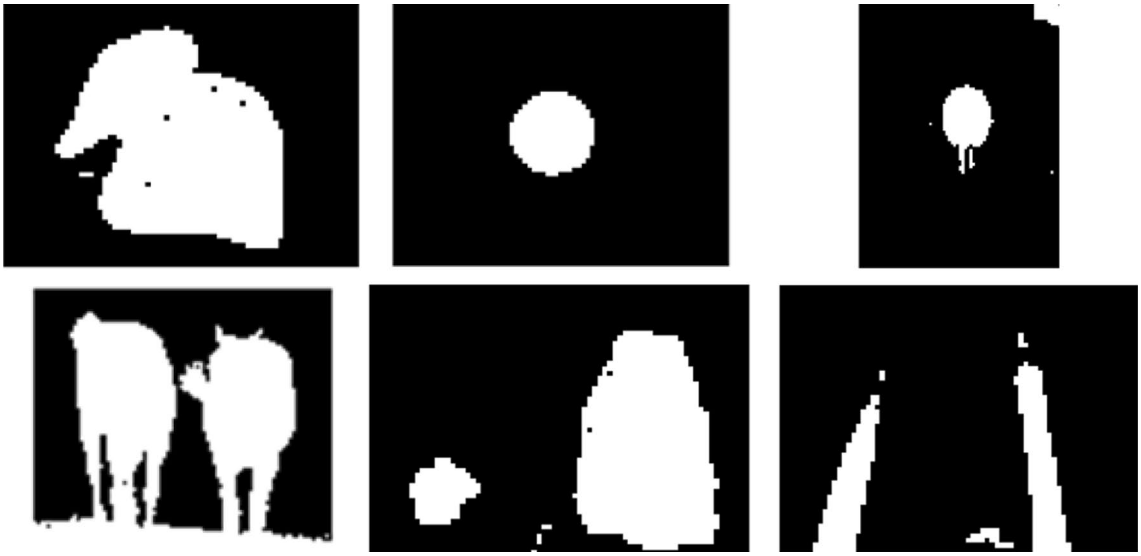


Fig. 10 Automatic image segmentation

realistic and robust results by using fuzzy neighborhood relation in the proposed algorithm instead of using crisp neighborhood relation utilized in the original algorithm. The experimental results on real-world dataset illustrate the superior performance of our algorithm compared with the DP clustering approach.

The combination of our proposed algorithm and the γ-graph displays a possibility that an automatic cluster centroid selection method is developed. FN-DP costs much time in the calculation of the similarity matrix, thus we will try to introduce the idea of the grid into our method. The cost is only associated with the number of cells. And the number of cells K is far less than the number of objects N.

Acknowledgements The work is supported by the National Natural Science Foundation of China under Grant No. 61379101 and No. 61672522, the National Basic Research Program of China under Grant No. 2013CB329502, the Priority Academic Program Development of Jiangsu Higer Education Institutions (PAPD), and the Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology (CICAEET).

References

1. Yu Z, Li L, Liu J, Zhang J, Han G (2015) Adaptive noise immune cluster ensemble using affinity propagation. IEEE Trans Knowl Data Eng 27(12):3176–3189
2. MacQueen JB (1967) Some methods for classification and analysis of multivariate observation. In: Proceedings of the fifth Berkeley symposium on mathematical statistics probability, vol 1, No. 14, pp 281–297
3. Huang Z (1998) Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Min Knowl Disc 2(3):283–304
4. Backer E, Jain AK (1981) A clustering performance measure based on fuzzy set decomposition. IEEE Trans Pattern Anal Mach Intell 1:66–75
5. Pal NR, Bezdek JC (1995) On cluster validity for the fuzzy c-means model. IEEE Trans Fuzzy Syst 3(3):370–379
6. Wishart D (1969) Mode analysis: a generalization of nearest neighbour which reduces chaining effects. In: Proceedings of the colloquium in numerical taxonomy
7. Ester M, Kriegel HP, Sander J, Xu XW (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of second international conference on knowledge discovery data mining, vol 96, No. 34, pp 226–231
8. Ankerst M, Breunig MM, Kriegel HP, Sander J (1999) OPTICS: ordering points to identify the clustering structure. In: Proceedings of ACM SIGMOD Conference, vol 28, No. 2, pp 49–60
9. Hinneburg A, Keim DA (1998) An efficient approach to clustering in large multimedia databases with noise. In: Proceedings of the fourth international conference on knowledge discovery and data mining
10. Sander J, Ester M, Kriegel HP, Xu X (1998) Density-based clustering in spatial databases: the algorithm DBSCAN and its applications. Data Mining Knowl Discov **2**(2):169–194
11. Rodriguez A, Laio A (2014) Clustering by fast search and find of density peak. Science 344(6191):1492–1496
12. Du M, Ding S, Jia H (2016) Study on density peaks clustering based on k-nearest neighbors and principal component analysis. Knowl-Based Syst 99:135–145
13. Liang Z, Chen P (2016) Delta-density based clustering with a divide-and-conquer strategy: 3DC clustering. Pattern Recognit Lett 73:52–59
14. Škrjanc I (2015) Evolving fuzzy-model-based design of experiments with supervised hierarchical clustering. IEEE Trans Fuzzy Syst 23(4):861–871
15. Yu Z, Chen H, You J, Liu J, Wong HS, Hana G, Li L (2015) Adaptive fuzzy consensus clustering framework for clustering analysis of cancer data. IEEE/ACM Trans Comput Biol Bioinform **12**(4):887–901
16. Wu CH, Ouyang CS, Chen LW, Lu LW (2015) A new fuzzy clustering validity index with a median factor for centroid-based clustering. IEEE Trans Fuzzy Syst 23(3):701–718
17. Hu L, Chan KC (2016) Fuzzy clustering in a complex network based on content relevance and link structures. IEEE Trans Fuzzy Syst 24(2):456–470
18. Wang Y, Chen L, Mei JP (2014) Incremental fuzzy clustering with multiple medoids for large data. IEEE Trans Fuzzy Syst 22(6):1557–1568
19. Zheng Y, Jeon B, Xu D, Wu QM, Zhang H (2015) Image segmentation by generalized hierarchical fuzzy C-means algorithm. J Intell Fuzzy Syst 28(2):961–973
20. Klawonn F, Kruse R, Winkler R (2015) Fuzzy clustering: more than just fuzzification. Fuzzy Sets Syst 281:272–279
21. He YL, Wang XZ, Huang JZ (2016) Fuzzy nonlinear regression analysis using a random weight network. Inf Sci 364:222–240
22. Nasibov EN, Ulutagay G (2007) A new unsupervised approach for fuzzy clustering. Fuzzy Sets Syst 158:2118–2133
23. Nasibov EN (2008) A robust algorithm for solution of the fuzzy clustering problem on the basis of the fuzzy joint points method. Cybern Syst Anal 44(1):7–17
24. Nasibov EN, Ulutagay G (2009) Robustness of density-based clustering methods with various neighborhood relations. Fuzzy Sets Syst 160:3601–3615
25. Ulutagay G, Nasibov EN (2010) Influence of transitive closure complexity in FJP-based clustering algorithms. Turkish J Fuzzy Syst 1(1):3–20
26. Wen X, Shao L, Xue Y, Fang W (2015) A rapid learning algorithm for vehicle classification. Inf Sci 295(1):395–406
27. Xia Z, Wang X, Sun X, Liu Q, Xiong N (2016) Steganalysis of LSB matching using differences between nonadjacent pixels. Multimed Tools Appl 75(4):1947–1962
28. Yu Z, Zhu X, Wong HS, You J, Zhang J, Han G (2016) Distribution-based cluster structure selection. IEEE Trans Cybern. doi:10.1109/TCYB.2016.2569529
29. Zhang Y, Sun X, Wang B (2016) Efficient Algorithm for K-barrier coverage based on integer linear programming. China Commu 13(7):16–23
30. Gu B, Sheng VS, Wang Z, Ho D, Osman S, Li S (2015) Incremental learning for ν-support vector regression. Neural Netw 67:140–150
31. Yu Z, Luo P, You J, Wong HS, Leung H, Wu S, Zhang J, Han G (2016) Incremental semi-supervised clustering ensemble for high dimensional data clustering. IEEE Trans Knowl Data Eng 28(3):701–714
32. Dong CR, Ng WWY, Wang XZ, Chan PPK, Yeung DS (2014) An improved differential evolution and its application to determining feature weights in similarity-based clustering. Neurocomputing 146:95–103
33. Huang G, Song S, Gupta JND, Wu C (2014) Semi-supervised and unsupervised extreme learning machines. IEEE Trans Cybern 44(12):2405–2417
34. Ashfaq RAR, Wang XZ, Huang JZ, Abbas H, He YL (2017) Fuzziness based semi-supervised learning approach for intrusion detection system. Inf Sci 378:484–497
35. Wang XZ, Wang YD, Wang LJ (2004) Improving fuzzy c-means clustering based on feature-weight learning. Pattern Recogn Lett 25(10):1123–1132
36. Zeng S, Yang X, Gou J, Wen J (2016) Integrating absolute distances in collaborative representation for robust image classification. CAAI Trans Intell Technol 1(2):189–196
37. Alpert S, Galun M, Brandt A, Basri R (2012) Image segmentation by probabilistic bottom-up aggregation and cue integration. IEEE Trans Pattern Anal Mach Intell 34(2):315–327
38. Ma Z, Liu Q, Sun K, Zhan S (2016) A syncretic representation for image classification and face recognition. CAAI Trans Intell Technol 1(2):173–178