International Journal of Approximate Reasoning 153 (2023) 239–257

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

journal homepage: www.elsevier.com/locate/ijar

A three-way clustering method based on improved density peaks algorithm and boundary detection graph

Chen Sun, Mingjing Du*, Jiarui Sun, Kangkang Li, Yongquan Dong

School of Computer Science and Technology, Jiangsu Normal University, Xuzhou, 221116, China

ARTICLE INFO

Article history:
Received 24 October 2022
Received in revised form 5 December 2022
Accepted 6 December 2022
Available online 9 December 2022

Keywords:
Three-way clustering
Density peaks clustering
Density-based clustering
Three-way decision theory

ABSTRACT

Density Peaks Clustering (DPC) is a classic density-based clustering algorithm that has been successfully applied in various areas. However, it assigns samples based on their nearest neighbors with higher density which may lead to an error propagation problem. Besides, it can not detect fringe and overlapping samples. To handle these defects, we improve the density measurement of DPC to make it more adaptive to different shapes and varying densities. Furthermore, we extend DPC to three-way clustering which means a sample in the positive region certainly belongs to the cluster, a sample in the boundary region belongs to the cluster partially and a sample in the negative region certainly does not belong to the cluster. In this paper, we propose a three-way clustering method called TW-RDPC. It mainly consists of three steps: (1) Identify cluster centers and assign other samples based on relative Cauchy kernel density to get initial clusters. (2) Detect potential boundary samples through boundary detection graph. (3) Determine whether potential boundary samples belong to multiple clusters based on the subordinate relationship to their k nearest neighbors. In order to validate TW-RDPC, we compare it to 7 algorithms on 10 synthetic datasets and 8 real-world datasets. Experimental results indicate that TW-RDPC is competitive with the compared 7 algorithms.

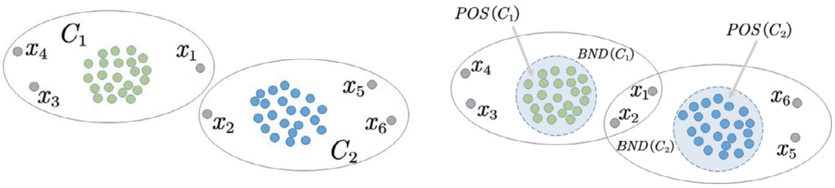
1. Introduction

Clustering is a well-known unsupervised learning technique that groups similar samples to the same clusters and dissimilar samples to different clusters [1]. The most outstanding advantage of clustering is that potential similar patterns can be found without any prior information. It has been widely used in various areas, including text mining [2], image segmentation [3], bioinformatics [4], community detections [5] and so on.

DPC is a classic density-based clustering algorithm and has obtained extensive attention [6,7], but it is limited by the error propagation problem. Many methods have been proposed to overcome the defect. For example, SNN-DPC [8] applied shared-nearest-neighbor to improve the assignment process. DPC-DLP [9] adopted a graph-based label propagation method to assign labels to remaining samples based on identified cluster centers. 3W-DPET [10] deals with the problem by extending DPC to three-way clustering. Meanwhile, three-way clustering can also identify fringe samples and overlapping samples. However, 3W-DPET may excessively assign samples to boundary regions which causes more uncertainty. Our algorithm not only solves the error propagation problem but also reduces the uncertainty of the clustering results. Besides, BPEC [11]

* Corresponding author.
E-mail addresses: 3020200309@jsnu.edu.cn (C. Sun), dumj@jsnu.edu.cn (M. Du), sunjr@jsnu.edu.cn (J. Sun), likangkang2020@jsnu.edu.cn (K. Li), tomdyq@163.com (Y. Dong).

https://doi.org/10.1016/j.ijar.2022.12.002
0888-613X/© 2022 Elsevier Inc. All rights reserved.



(a) Two-way clustering results (b) Three-way clustering results

Fig. 1. Two types of clustering results.

extends DPC to evidential clustering [12], which utilizes belief functions to form credal partitions, from which hard, fuzzy, possibilistic, and rough partitions can be derived.

As three-way clustering originates from three-way decisions, we will first introduce the two research fields. Then, we introduce the motivations and main steps of our algorithm.

1.1. Three-way decisions and three-way clustering

The three-way decision is a new theory of study proposed by Yao for complex problem analyzing and solving [13–15]. It is an extension of the binary-decision methods. Based on the idea of triadic thinking [16], a three-way decision divides a universal set into three disjoint regions and makes three types of decisions accordingly to achieve the expected results. The original goal of this theory is to provide a reasonable semantic explanation for decision-theoretic rough sets [14]. Based on the loss function values being constant or changing over time, the research of three-way decisions can be divided into static three-way decisions [17] and dynamic three-way decisions [18]. This paradigm of triadic thinking for complex problem solving and information processing fosters subsequent research topics including three-way classification [19,20], three-way attribute reduction [21], three-way cognitive computing [22], three-way conflict analysis [23], three-way clustering [24,25] etc.

In light of triadic thinking of three-way decisions, three-way clustering represents a cluster by an interval set (i.e. a pair of nested sets split by lower bound and upper bound respectively). Therefore, a cluster is made up of three regions: positive region (the samples in POS belong to the cluster certainly), boundary region (the samples in BND may be part of the cluster and may potentially belong to other clusters), negative region (the samples in NEG do not belong to the cluster certainly). Fig. 1 [26] illustrate simple examples of two-way and three-way cluster representations.

In Fig. 1 (a), C_1 and C_2 are two clusters obtained by two-way clustering. It means a sample can only belong to one cluster. Thus, some boundary information (e.g. x_1, x_2,..., x_6) can not be fully reflected. To solve the problem, as shown in Fig. 1 (b), clusters are restructured into positive regions ($POS(C_1)$, $POS(C_2)$) and boundary regions ($BND(C_1)$, $BND(C_2)$), the samples in POS certainly belong to that cluster, while the samples in BND hold a relatively loose connection to clusters (e.g. x_3, x_4, x_5, x_6). Moreover, the samples may belong to more than one cluster (e.g. x_1, x_2). Therefore, the divide-and-conquer strategy is effective in getting a more reasonable structure.

1.2. Motivation and contribution

Detecting clusters of different shapes and varying densities is a major task in density-based clustering methods. DPC is an effective density-based method promising in many applications, but it also faces some dilemmas:

- DPC's results are found to be sensitive to density measurement and large density differences across clusters tend to result in improper cluster centers and label error propagation problem [10].
- It is improper to assign some boundary samples to only one strict cluster when the sample shows nearly the same similarities to many different clusters.
- Although 3W-DPET also aims to alleviate the label error propagation problem, it tends to overly assign samples to boundary regions which may cause excessive uncertainty. Our algorithm not only tackles the label error propagation problem but also reduces the uncertainty.

In order to tackle the above mentioned defects, we improve DPC through density measurement and extend the clustering results to the three-way paradigm. The main steps can be summarized as follows:

- Firstly, we adopt Cauchy kernel density with dynamic bandwidth to get local density. Then we get the relative local density by dividing the local density by the maximum local density in the samples' k nearest neighbors. In this way, we can better adapt to different data distributions, so as to get more reasonable cluster centers and mitigate label error propagation.
- Secondly, based on the idea that boundary samples tend to have low local density and low relative density, we propose a boundary detection graph to get potential boundary samples.

- Finally, we determine whether a potential boundary sample is a fringe or overlapping sample based on the subordinate relationship to its k nearest neighbors. If its k nearest neighbors' labels are the same, the sample belongs to that one cluster. If its k nearest neighbors' labels are different, the sample belongs to clusters with these labels.

This method gets the density information and relative information of the samples in a more detailed way. The boundary detection graph utilizes the density measurements to identify potential boundary samples. Furthermore, the three-way processing procedure can also be applied to other density-based clustering methods as long as proper density estimation and thresholds are defined.

The rest of this paper is organized as follows: We review the studies related to our work in Section 2. Section 3 briefly introduces the DPC algorithm and three-way clustering representation. The details of TW-RDPC are introduced in Section 4. Section 5 presents the experimental results on datasets and sensitivity analysis. Finally, we summarize our work and future research in Section 6.

2. Related work

To relax the constraint of hard clustering, soft clustering algorithms have been applied successfully. The fuzzy C-means (FCM) [27] which assumes a cluster is represented by a fuzzy set that reflects a gradually changing boundary is the most widely used. Another effective tool for uncertain data analysis is the rough K-means algorithm (RKM) [28] which uses interval sets to represent clusters with vague and imprecise boundaries. To further enrich this field, Yu [29] proposes an evaluation-based three-way clustering model stemming from three-way decisions: Considering a pair of thresholds $(\alpha,\beta)_{\alpha\geq\beta}$ and an evaluation function $\varphi(x)$, C_i is a cluster, U is a universal set, $POS(C_i)$=$\{x\epsilon U|\varphi(x)>\alpha\}$, $BND(C_i)$=$\{x\epsilon U|\beta\leq\varphi(x)\leq\alpha\}$, $NEG(C_i)$=$\{x\epsilon U|\varphi(x)<\beta\}$. To further determine the two thresholds for overlapping clusters, Afridi et al. [30] define the between-variance and within-variance for the three regions. Ulteriorly, they take the maximization of the ratio of the former to the latter as the optimization objective to determine the thresholds. In case of missing data [31], they creatively apply game-theoretic rough sets (GTRS) for the automatic determination of thresholds. Yu [29] also proposes a three-way clustering algorithm for incomplete data by improved partial Euclidean distance modeling.

Three-way clustering has also been applied to process different types of data in different backgrounds. For incremental data, Yu, Zhang and Wang [32] put forward a two-stage three-way clustering algorithm. During the online stage, initial samples are clustered and a tree is constructed. During the off-line stage, the neighbor information is used to update the tree graph to get the three-way clusters. For multivariate time series, López-Oriona et al. [33] propose quantile-based fuzzy C-means based on the so-called metric, noise and trimmed approaches. For Multi-view data clustering, Yu et al. [34,35] propose two three-way clustering methods via decomposing similarity matrices and low-rank matrices separately. Khan et al. [36] handle this situation by low-rank sparse representation. For ensemble clustering, Yu et al. [37] propose a three-way cluster ensemble approach for large-scale data based on spark. Jiang and Zhao [38] propose a three-way clustering ensemble approach based on the shadowed set. S-M3WCE [39] is another shadowed set based multi-granular three-way ensemble clustering approach via possibilistic C-means. Wang et al. [40] propose a three-way ensemble clustering algorithm for incomplete data based on the imputation result. Meanwhile, novelty detection [41] and outlier detection [42] methods are also extended by the three-way clustering approach.

Three-way clustering has also been used to extend existing two-way clustering algorithms. Inspired by the ideas of erosion and dilation from mathematical morphology, Wang and Yao [26] propose a framework of contraction-and-expansion that extends two-way clustering to three-way clustering. Specifically, based on K-means [43], TWKM [44] uses perturbation analysis to separate the core regions from the supports. A-3WCM [45] uses cognition of distance stability to adaptively identify the cut-off threshold and weight equation based on K-means. Another three-way K-means algorithm [46] is proposed to improve the weight and the sample assignment strategy. Based on DBSCAN [47], 3W-DBSCAN [48] improves similarity measurement with a multi-dimensional distance scaling method to identify varying densities. Based on DPC [49], TWC-GS [50] uses a gravitational search strategy to adjust the thresholds automatically so that three regions can be obtained. 3W-DPET [10] utilizes evidence theory to overcome DPC's label error propagation problem. Inspired by ROBP [51]and sequential three-way decisions, Du et al. propose multistep three-way clustering [52] by progressive erosion strategy. BS3 [53] and BS3WC [54] convert hard clusters to images and define cluster blur and cluster sharp operations to get three-way clusters.

3. Preliminaries

3.1. Density peaks clustering

The DPC algorithm is mainly based on two assumptions: (1) Cluster centers are surrounded by nearby samples with low local densities. (2) The distance between cluster centers is far. Thus, for a sample x_i, we need to compute two values: local density ρ and the nearest distance to samples with higher density δ. The local density of x_i is defined as:

$$\rho_i=\sum_{j=1}^{n}\chi(d_{ij}-d_c),\qquad \chi(x)=\begin{cases}1, & x<0\\ 0, & x\geq 0\end{cases}\tag{1}$$

where n is the number of total samples, d_{ij} is the distance between x_i and x_j, d_c is the cut-off distance. It is obvious that ρ_i is equal to the number of samples distributed in the d_c radius of x_i.

Another way to compute ρ_i is through the Gaussian kernel:

$$\rho_i = \sum_{j=1}^{n} exp(-\frac{d_{ij}}{{d_c}^2}), \tag{2}$$

where d_c is used to adjust the weight degradation. An appropriate percentile is adopted to capture the only parameter d_c. δ_i is measured by calculating the minimum distance between sample x_i and other samples with higher density:

$$\delta_i = \begin{cases} \min\limits_{j:\rho_i > \rho_j} \ d_{ij}. \ if \ \exists j \ s.t. \ \rho_i > \rho_j \\ \max\limits_{j} \ d_{ij}, \ otherwise. \end{cases} \tag{3}$$

Based on ρ_i and δ_i, a decision graph is plotted to detect cluster centers with both relatively high ρ_i and δ_i. Another frequently used approach to identify cluster centers is to compute $\gamma_i = \rho_i \times \delta_i$, and select the samples with top n maximum γ_i as cluster centers where n is the number of clusters. Then, all the unlabeled samples are assigned in line with their nearest neighbors. The chains are terminated by cluster centers.

3.2. Three-way clustering representation

In this section, we elucidate the symbolic representation of three-way clustering. Suppose $U = \{x_1, x_2, ..., x_l\}$ is a finite non-empty universe of samples. In traditional clustering, $C = \{C_1, C_2, ..., C_n\}$ is a family of n clusters. While in three-way representation, an interval set is used to illustrate the clustering results: $C_i = (\underline{\mathbb{C}}, \overline{\mathbb{C}}) = (POS(C_i), POS(C_i) \cup BND(C_i))$, where $\underline{\mathbb{C}}$ is the lower bound of C_i and $\overline{\mathbb{C}}$ is the upper bound of C_i. The three regions in one cluster have four properties:

$$\begin{gathered} \text{(i)} \ POS(C_i) \cup BND(C_i) \cup NEG(C_i) = U, \\ \text{(ii)} \ POS(C_i) \cap BND(C_i) = \phi, \\ \text{(iii)} \ POS(C_i) \cap NEG(C_i) = \phi, \\ \text{(iv)} \ BND(C_i) \cap NEG(C_i) = \phi. \end{gathered}$$

The four properties mean that the three adjacent regions do not intersect with each other.

In addition, there are three properties between clusters:

$$\begin{gathered} \text{(i)} \ POS(C_i) \neq \phi, \ i = 1, 2, ..., n, \\ \text{(ii)} \ POS(C_i) \cap POS(C_j) = \phi, \ i \neq j, \\ \text{(iii)} \ \bigcup_{i=1}^{k} (POS(C_i) \cup BND(C_i)) = U. \end{gathered}$$

The three properties between clusters indicate that all positive regions are non-empty. Moreover, any two positive regions do not intersect with each other.

4. Proposed TW-RDPC algorithm

In this section, we improve DPC's density measurement through relative Cauchy kernel density, and then we propose a boundary detection graph to detect potential boundary samples. Finally, based on the k nearest neighbors, we determine whether a potential boundary sample is an overlapping sample. The three-way strategy is also applicable to other density-based clustering methods under proper density measurement.

4.1. Improved DPC algorithm

Definition 1 *(k Nearest Neighbors). The k nearest neighbors of sample x_i are defined as a set of samples x satisfy: $d(x_i, x) \leq d(x_i, x_j)$, where $d(x_i, x_j)$ is the distance between x_i and its k-th neighbor. i.e., $KNN(x_i) = \{x_i \in D | d(x_i, x) \leq d(x_i, x_j)\}$, where D is the dataset. Especially, we use $KNN_k(x_i)$ to represent the k-th nearest neighbor of x_i. Examples of KNN are shown in Fig. 2.*

When $k = 3$, the $3NN$ are the 3 green samples inside the circle with the radius r_1; when $k = 7$, the $7NN$ are the 3 green samples and 4 blue samples inside the circle with the radius r_2.

Definition 2 *(Reverse k Nearest Neighbors). The reverse k nearest neighbors of sample x_i are defined as a set of samples x that include x_i as one of its KNN, i.e., $RKNN(x_i) = \{x \in D | x_i \in KNN(x)\}$. An example of $RKNN$ is shown in Fig. 3 [55].*

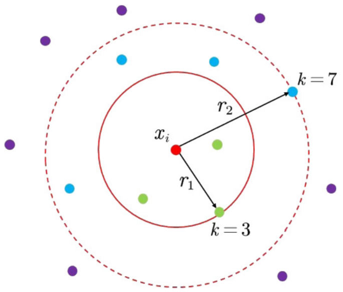


Fig. 2. Examples of KNN with $k = 3$ and $k = 7$. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

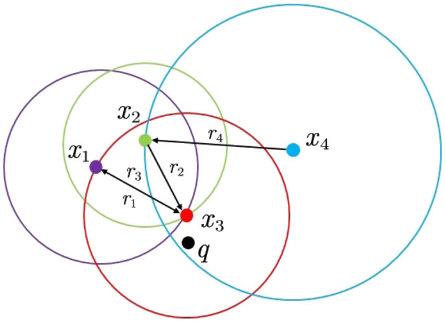


Fig. 3. Example of $RKNN$ with k=2.

The example dataset X contains 4 samples each associated with a circle covering its 2 nearest neighbors. For example, the $2NN$ of x_4 are x_2 and x_3, which are in the circle centered at x_4. Given another sample q the result of $R2NN(q)$ includes the "owners" (i.e. centers) of the circles that contain q. So, $R2NN(q) = \{x_3, x_4\}$.

Note that $x \in KNN(q)$ does not necessarily imply $x \in RKNN(q)$, and vice versa. For instance, $2NN(q) = \{x_1, x_3\}$ but x_1 does not belong to $R2NN(q)$, because $2NN(q)$ does not include q. On the other hand, although $x_4 \in R2NN(q)$, x_4 is not in $2NN(q)$.

Definition 3 *(Cauchy kernel estimation). Compared with the Gaussian distribution, the Cauchy distribution has longer tails. Meanwhile, it is more robust to non-uniformly-distributed data [51] and high-dimensional data [56] than the Gaussian kernel so we adopt Cauchy kernel estimation which is defined as:*

$$\rho_i = \sum_{x_j \in RKNN(x_i)} (\frac{||x_i - x_j||_2^2}{h^2} + 1)^{-1}, \tag{4}$$

where h is the bandwidth. We adopt a dynamic method to get h automatically in Definition 4.

Definition 4 *(Variable bandwidth kernel density). A single hard bandwidth can not adapt to varying density distribution. To capture the different bandwidths of different regions, we utilize the $RKNN$ information to dynamically get h [57], where $h = ||x_j - KNN_k(x_j)||_2$. So the kernel density is defined as:*

$$\rho_i = \sum_{x_j \in RKNN(x_i)} (\frac{||x_i - x_j||_2^2}{||x_j - KNN_k(x_j)||_2^2} + 1)^{-1}. \tag{5}$$

Definition 5 *(Relative Cauchy kernel density). Inspired by LGD [58], to further identify the relative information, we divide the Cauchy density of x_i by the maximum Cauchy density in its KNN. The relative Cauchy kernel density is defined as:*

$$\rho_i' = \frac{\rho_i}{max\{\rho_j | x_j \in KNN(x_i)\}}, \quad x_j \in RKNN(x_i). \tag{6}$$

After the local density is defined, the remaining steps are the same as DPC: we detect cluster centers and assign other samples based on their nearest neighbors with higher relative Cauchy kernel density, until a cluster center is found. The following Algorithm 1 is a summary of improved DPC:

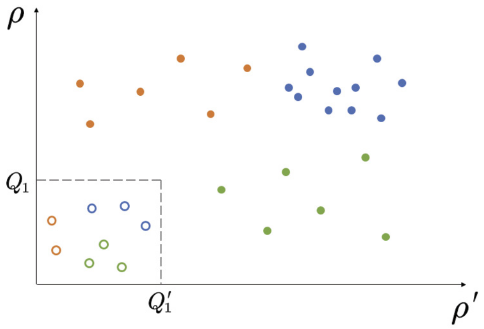


Fig. 4. Example of boundary detection graph with 3 clusters.

Algorithm 1: Improved DPC algorithm.

```
Input: A dataset $D$ and the parameter $k$
Output: Initial assigned cluster indices
Calculate distance matrix by Euclidean distance [59].
Calculate $\rho_i'$ for sample $x_i$ according to Formula (6).
Calculate $\delta_i$ for sample $x_i$ according to Formula (3).
Plot the decision graph and select cluster centers.
Assign each remaining sample to the nearest cluster center.
```

4.2. Three-way process

After all the samples are initially labeled, we start to detect potential boundary samples. A boundary sample is far from or relatively far from other samples in the cluster. Thus, it is obvious that it has both low Cauchy kernel density ρ and relative Cauchy kernel density ρ'. Firstly, we construct a boundary detection graph to detect potential boundary samples. To determine thresholds, the percentile of the total data number is adopted. Through our experiments, the quartile [60,61] of ρ and ρ' is well to detect potential boundary samples. An example of a boundary detection graph is shown in Fig. 4.

Different colors represent the initial clusters obtained by improved DPC. Solid points represent samples in positive regions and hollow points represent potential boundary samples.

After potential boundary samples are obtained, we further identify whether the potential boundary samples belong to positive regions or boundary regions based on their KNN. If the KNN of a potential boundary sample all belong to one cluster, then the potential boundary sample is assigned to the positive region of that cluster. If the KNN of a potential boundary sample belongs to different clusters, then the potential boundary samples are assigned to boundary regions of these clusters. An example of the three-way process is shown in Fig. 5.

Based on the above two strategies, the following Algorithm 2 is a summary of the three-way process, where n is the number of clusters and k is the number of nearest neighbors:

Algorithm 2: The three-way process.

```
Input: Initial cluster labels and the parameter $k$
Output: $\{POS(C_1), POS(C_2), ..., POS(C_n)\}$
and $\{BND(C_1), BND(C_2), ..., BND(C_n)\}$
Calculate $\rho_i$ and $\rho_i'$ by Formula(5) and Formula(6).
Calculate $Q_1$ and $Q_1'$ through Quartile.
if $\rho_i < Q_1$ and $\rho_i' < Q_1'$ then
    if $\{\forall x_i \mid KNN(x_i) \in C_m\}$ then
        $POS(C_m) \Leftarrow x_i \quad m \in 1, 2, ..., n.$
    else
        $\{\exists x_i \mid KNN(x_i) \in C_m\}$
        $BND(C_m) \Leftarrow x_i \quad m \in 1, 2, ..., n.$
    end
else
    Keep $x_i$ in initial clusters $C_r$ : $POS(C_r) \Leftarrow x_i$. $r \in 1, 2, ..., n.$
end
```

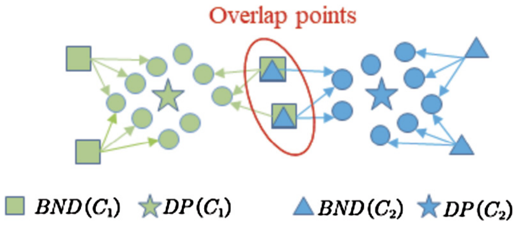


Fig. 5. Example of three-way process based on KNN with k=3.

Table 1
Datasets used in experiments.

Synthetic datasets	#Instances	#Features	#Classes
D1	87	2	3
D2	85	2	4
Zelink6	238	2	3
Compound	399	2	6
R15	600	2	15
Aggregation	788	2	7
Triangle2	1000	2	4
4C	1250	2	4
G2	1500	2	3
S1	5000	2	15
Real-world datasets	#Instances	#Features	#Classes
Iris	150	4	3
Parkinsons	195	22	2
Seeds	210	7	3
Thyroid	215	5	3
Liver	345	6	2
Pima	768	8	2
Biodeg	1055	41	2
Unbalance	6500	2	8

4.3. Complexity analysis

In this section, we analyze the time complexity of TW-RDPC. Suppose that the total number of samples is n and k is the number of nearest neighbors. Firstly we analyze the time complexity of the improved DPC algorithm (Algorithm 1). For line 1, it takes $O(n^2)$ to compute the pairwise distance matrix. Then for line 2, we need to search for the KNN of each sample to get ρ and ρ' each of them takes $O(kn)$. For line 3, like traditional DPC, it takes $O(n^2)$ to calculate δ. Thus the magnitude of the overall time complexity of the improved DPC algorithm is $O(n^2)$. Next, we analyze the time complexity of the Three-way process (Algorithm 2). After ρ and ρ' are obtained in the improved DPC algorithm, we can get Q_1 and Q_1'. Then we need to detect potential boundary samples by the judging conditions in line 3 which takes $O(n)$. Assuming the number of potential boundary samples is m, we need to search KNN to identify whether they belong to positive regions or boundary regions which takes $O(km)$. Finally, we assign all samples to their corresponding regions which takes $O(n)$. As $m \ll n$ the magnitude of the overall time complexity of the Three-way process is $O(n)$. Based on the above two parts, as $k \ll n$, the overall time complexity of TW-RDPC is $O(n^2)$.

5. Experiments and results

5.1. Experiment setup

In this section, eighteen datasets are used to evaluate the performance of TW-RDPC, including ten synthetic datasets and eight real-world datasets. For the convenience of visualization, the synthetic datasets are all two-dimensional. In order to prove the applicability of the algorithm, the real-world datasets are varying dimensions. All the synthetic datasets are from benchmark clustering datasets[1] and all the real-world datasets are from UCI Machine Learning Repository.[2] The detailed information of these datasets is shown in Table 1.

We compare the performance of TW-RDPC with seven other clustering algorithms, including CE3-kmeans [26], 3W-DPET [10], M3W [52] DPC [49], DPC-KNN [59], FCM [27] and RCM [28]. Among them, CE3-kmeans, 3W-DPET and M3W

[1] https://github.com/milaan9/Clustering-Datasets.
[2] http://archive.ics.uci.edu/ml/index.php.

Table 2
Configuration of parameters in different algorithms.

	Description	Candidates
TW-RDPC	The number of nearest neighbors	$k \in [1, 2, ..., 20]$
3W-DPET		
CE3-kmeans		
DPC-KNN		
M3W	The number of nearest neighbors	$k \in [1, 2, ..., 30]$
	The erosion levels	$L \in [1, 2, ..., 12]$
DPC	The cut-off distance in Eq. (2)	$d_c \in [0.01, 0.02, ..., 1.00]$
FCM	The fuzzy index	$m \in [1.0, 1.5, ..., 5.0]$
RKM	The approximate weight	$w_u \in [0.1, 0.2, ..., 0.5]$
	The ratio threshold	$\epsilon \in [0.1, 0.2, ..., 0.5]$

Table 3
A contingency table.

C Ω	c_1	c_2	$\cdots$	c_s	sums
ω_1	n_{11}	n_{12}	$\cdots$	n_{1n}	a_1
ω_2	n_{21}	n_{22}	$\cdots$	n_{2n}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
ω_r	n_{r1}	n_{r2}	$\cdots$	n_{rn}	a_r
sums	b_1	b_2	$\cdots$	b_s	

are state-of-the-art three-way clustering algorithms, DPC and DPC-KNN are DPC related algorithms, FCM and RKM are two representative soft clustering algorithms. All of them are widely used and successfully applied in different backgrounds.

All of these algorithms need to know the number of clusters in advance except M3W. The number of clusters is set as the true number of classes in the datasets. Additionally, these algorithms need to configure their parameters separately. TW-RDPC, 3W-DPET, DPC-KNN and CE3-kmeans all need a parameter k which describes the number of k nearest neighbors. We choose k by setting it from 1 to 25 with step 1. Then we determine k with the best performance. Especially, because of randomly selected initial center samples, CE3-kmeans get different results each running time. So we run it ten times and compute the average value as the final results. The number of k nearest neighbors and the number of erosion levels L are two parameters in M3W. We configure k from 1 to 30 with step 1 and L from 1 to 12 by increasing 1. DPC has one parameter d_c, we set d_c ranging from 0.01 to 1 with step 0.01 and determine d_c with the best performance. The maximum iteration of FCM is set as 100 by default. So, FCM needs one parameter m called fuzzy index. The value of m varies from 1.0 to 5.0 with step 0.5. Two parameters of RKM are w_u and ϵ, which respectively represent the approximate weight and the ratio threshold. We configure w_u from 0.1 to 0.5 with step 0.1 and ϵ from 0.1 to 0.5 by increasing 0.1. The configuration is summarized in Table 2. In the experiments, we use min-max normalization [62] to process all datasets.

5.2. Evaluation measures

Clustering problem is a kind of partition problem. Suppose $\Omega = \{\omega_1, \omega_2, ..., \omega_r\}$ are the clustering results by the clustering algorithm, $C = \{c_1, c_2, ..., c_n\}$ are the true clusters.

1. Purity (Pur): The general idea of purity is to divide the number of correct samples by the total number of samples, so it is also called the accuracy [63] of clustering. It is defined as:

$$Pur = (\Omega, C) = \frac{1}{N}\sum_{i=1}^{r} \max_{j} |\omega_i \cap c_j| \tag{7}$$

where N is the total number of samples, ω_i represents all samples in the ith cluster, c_j represents the true sample in the jth cluster. The purity range is $[0, 1]$, a higher purity means better clustering results.

2. Adjusted Rand Index (ARI): To mitigate the impact of random labels on RI evaluation results, ARI [64] is proposed. A contingency table (Table 3) is adopted to compute ARI:

$$ARI = \frac{\sum_i \sum_j \binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}} \tag{8}$$

where n_{ij} represents the number of intersecting samples of ω_i and c_j.

The range of ARI is $[-1, 1]$. The larger the ARI, the better the clustering results.

Table 4
Performance comparison for 7 clustering algorithms on 10 synthetic datasets.

Algorithm	Purity	ARI	NMI	Params	Purity	ARI	NMI	Params
	D1	(3 classes)			D2	(4 classes)		
TW-RDPC	**1.0000**	**1.0000**	**1.0000**	$k = 8$	**1.0000**	**1.0000**	**1.0000**	$k = 5$
CE3-kmeans	0.9034	0.8639	0.9092	$k = 12$	**1.0000**	**1.0000**	**1.0000**	$k = 6$
3W-DPET	**1.0000**	**1.0000**	**1.0000**	$k = 10$	**1.0000**	**1.0000**	**1.0000**	$k = 13$
M3W	**1.0000**	**1.0000**	**1.0000**	$k = 10$ $L = 2$	**1.0000**	**1.0000**	**1.0000**	$k = 18$ $L = 2$
DPC	**1.0000**	**1.0000**	**1.0000**	$d_c = 0.16$	0.9882	0.9679	0.9655	$d_c = 0.02$
DPC-KNN	**1.0000**	**1.0000**	**1.0000**	$k = 6$	0.9882	0.9679	0.9655	$k = 5$
FCM	0.9885	0.9610	0.9515	$m = 1.5$	0.9882	0.9679	0.9655	$m = 1.0$
RKM	0.9655	0.9053	0.8766	$w_u = 0.2$ $\epsilon = 0.2$	0.9882	0.9679	0.9655	$w_u = 0.2$ $\epsilon = 0.2$
	Zelink6	(3 classes)			Compound	(6 classes)		
TW-RDPC	**0.8824**	**0.7327**	**0.7812**	$k = 12$	**0.8997**	**0.8547**	**0.8716**	$k = 7$
CE3-kmeans	0.8517	0.6795	0.6765	$k = 14$	0.6935	0.6220	0.7414	$k = 9$
3W-DPET	0.8361	0.6572	0.6557	$k = 15$	0.8797	0.8473	0.8599	$k = 5$
M3W	0.8319	0.7299	0.6942	$k = 9$ $L = 4$	0.8722	0.8358	0.8789	$k = 17$ $L = 5$
DPC	0.8529	0.6877	0.6729	$d_c = 0.14$	0.8296	0.8327	0.8566	$d_c = 0.28$
DPC-KNN	0.8403	0.6715	0.6600	$k = 2$	0.8700	0.8090	0.8520	$k = 2$
FCM	0.8277	0.6506	0.6437	$m = 2.5$	0.6584	0.4124	0.5321	$m = 2.5$
RKM	0.8403	0.6782	0.6642	$w_u = 0.2$ $\epsilon = 0.2$	0.7920	0.7767	0.7935	$w_u = 0.2$ $\epsilon = 0.2$
	R15	(15 classes)			Aggregation	(7 classes)		
TW-RDPC	**1.0000**	**1.0000**	**1.0000**	$k = 15$	0.9987	0.9978	0.9957	$k = 6$
CE3-kmeans	0.9362	0.9324	0.9740	$k = 18$	0.8251	0.7239	0.8401	$k = 5$
3W-DPET	**1.0000**	**1.0000**	**1.0000**	$k = 15$	**1.0000**	**1.0000**	**1.0000**	$k = 7$
M3W	0.9967	0.9928	0.9942	$k = 15$ $L = 5$	0.9962	0.9910	0.9862	$k = 24$ $L = 5$
DPC	0.9967	0.9928	0.9942	$d_c = 0.02$	0.9975	0.9956	0.9942	$d_c = 0.09$
DPC-KNN	0.9967	0.9928	0.9942	$k = 5$	0.9962	0.9935	0.9896	$k = 4$
FCM	0.9050	0.8771	0.9303	$m = 2.0$	0.7893	0.6791	0.8234	$m = 1$
RKM	0.8883	0.8408	0.9282	$w_u = 0.1$ $\epsilon = 0.2$	0.7817	0.7962	0.7767	$w_u = 0.1$ $\epsilon = 0.1$
	Triangle2	(4 classes)			4C	(4 classes)		
TW-RDPC	**0.9990**	**0.9967**	**0.9946**	$k = 16$	0.7536	0.4752	0.6498	$k = 19$
CE3-kmeans	0.9973	0.9920	0.9845	$k = 15$	0.6425	0.3429	0.5035	$k = 11$
3W-DPET	0.9980	0.9933	0.9904	$k = 10$	0.6456	0.4385	0.6381	$k = 9$
M3W	0.9970	0.9900	0.9865	$k = 24$ $L = 6$	**0.8464**	**0.7088**	**0.7645**	$k = 30$ $L = 5$
DPC	0.9970	0.9900	0.9850	$d_c = 0.15$	0.7416	0.5448	0.6834	$d_c = 0.07$
DPC-KNN	0.9960	0.9867	0.9812	$k = 2$	0.7312	0.4408	0.6184	$k = 10$
FCM	0.9640	0.8990	0.8835	$m = 1.5$	0.6584	0.4124	0.5321	$m = 1.5$
RKM	0.8220	0.6506	0.7037	$w_u = 0.2$ $\epsilon = 0.2$	0.6928	0.3799	0.5336	$w_u = 0.2$ $\epsilon = 0.4$
	G2	(3 classes)			S1	(15 classes)		
TW-RDPC	**0.9973**	**0.9920**	**0.9845**	$k = 18$	**0.9973**	**0.9920**	**0.9845**	$k = 18$
CE3-kmeans	0.9478	0.9355	0.9481	$k = 14$	0.9476	0.9429	0.9754	$k = 2$
3W-DPET	0.9907	0.9724	0.9561	$k = 4$	0.9960	0.9915	0.9923	$k = 5$
M3W	0.9940	0.9831	0.9695	$k = 30$ $L = 5$	0.9630	0.9564	0.9634	$k = 15$ $L = 3$
DPC	0.9947	0.9841	0.9724	$d_c = 0.07$	0.9952	0.9897	0.9896	$d_c = 0.03$
DPC-KNN	0.9907	0.9724	0.9561	$k = 2$	0.9952	0.9897	0.9896	$k = 8$
FCM	0.9953	0.9861	0.9747	$m = 1$	0.8322	0.8183	0.9153	$m = 1.5$
RKM	0.9867	0.9608	0.9449	$w_u = 0.1$ $\epsilon = 0.2$	0.9188	0.9118	0.9610	$w_u = 0.1$ $\epsilon = 0.2$

3. Normalized Mutual Information (NMI): NMI [65] is widely used to measure the similarity of two clusters. The entropy of clusters is:

$$H(C) = -\sum_{i=1}^{k} p_i log p_i. \ where \ \ p_i = \frac{|c_i|}{N}. \tag{9}$$

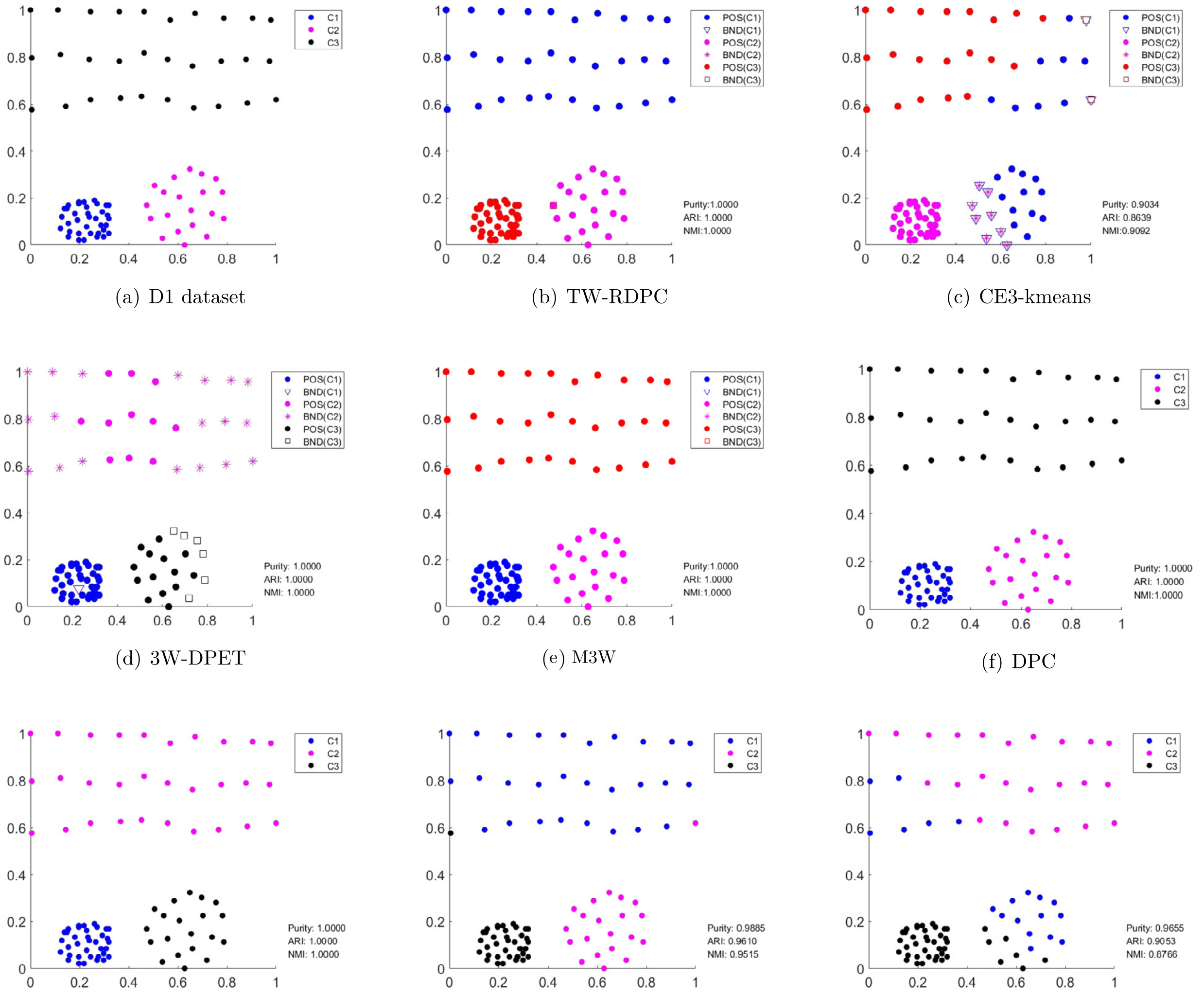
Then we compute the mutual information between predicted clusters and true clusters:

$$MI(\Omega, C) = \sum_{i=1}^{r}\sum_{j=1}^{s} p_{ij} log(\frac{p_{ij}}{p_i \times p_j}), \ \ where \ \ p_{ij} = \frac{|\omega_i \cap c_j|}{N}. \tag{10}$$

Based on the entropy and MI of clusters, we can get NMI:

$$NMI(\Omega, C) = \frac{MI(\Omega, C)}{\max(\mathrm{H}(\Omega), \mathrm{H}(\mathrm{C}))} \tag{11}$$

The range of NMI is $[0, 1]$. The larger the NMI, the more similar the clusters.



(a) D1 dataset

(b) TW-RDPC

(c) CE3-kmeans

(d) 3W-DPET

(e) M3W

(f) DPC

(g) DPC-KNN

(h) FCM

(i) RKM

Fig. 6. Clustering results on D1.

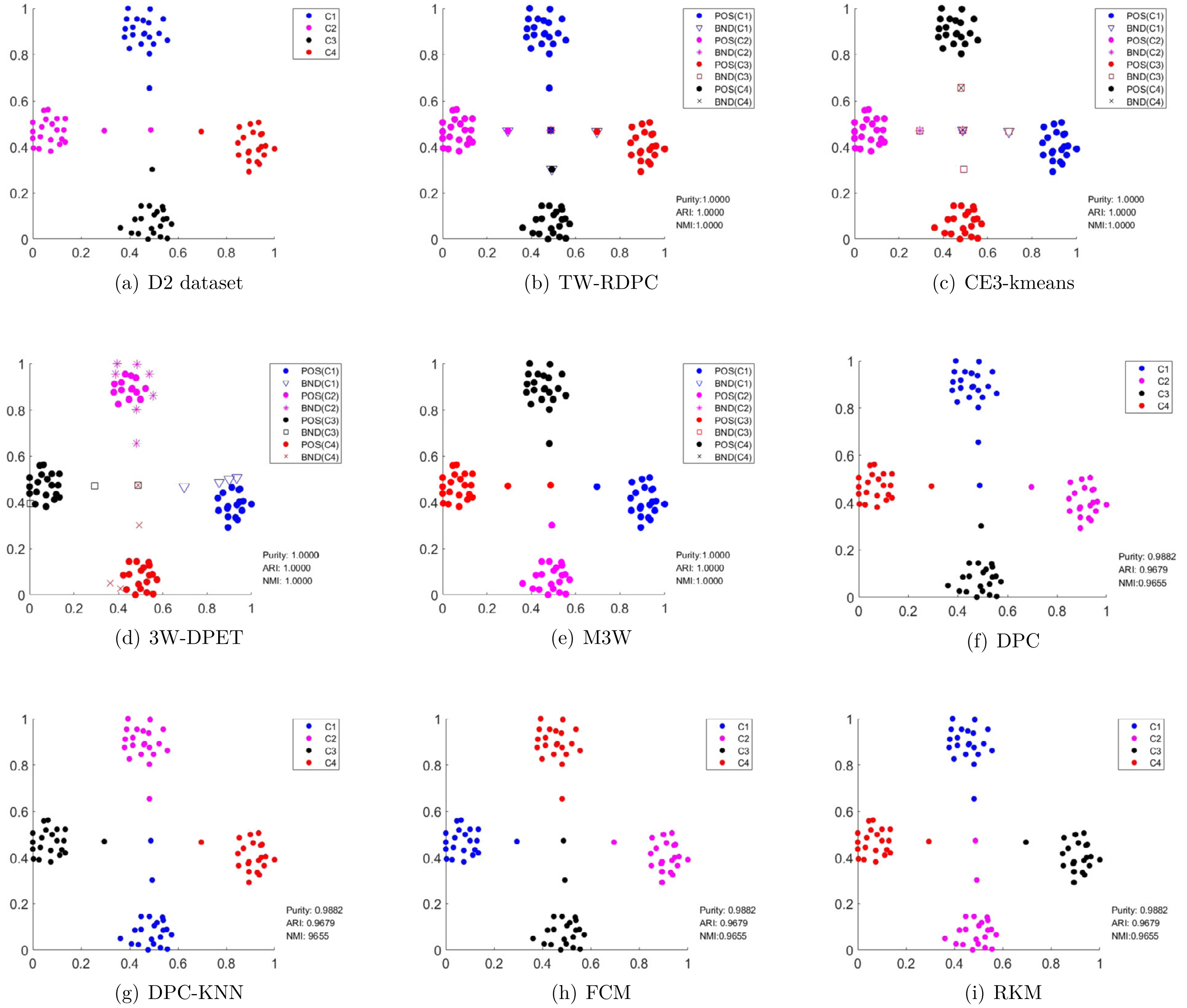
5.3. *Experimental results on synthetic data sets*

Table 4 shows the clustering results on 10 synthetic datasets, as well as the best parameter settings. The best performances for each dataset are highlighted in bold. Through the results, we can conclude that TW-RDPC achieves the best or relatively better performance in each dataset. Furthermore, the clustering results of TW-RDPC on D1, D2 and R15 are completely correct.

We visualize D1, D2, Compound, Aggregation, Triangle2 and 4C as examples to illustrate the superiority of our algorithm, which are shown in Fig. 6-11. In these results, (a) is the original distribution, (b-i) are results corresponding to each algorithm. Samples in positive regions are represented by solid dots and hollow-shaped dots mean boundary samples. The samples represented by two or more shapes are overlapping samples. Samples with the same color belong to one cluster.

D1 dataset is made up of a dense circle cluster, a relatively sparse circle cluster and a sparse wave area. The clustering results of D1 are shown in Fig. 6. TW-RDPC, 3W-DPET, M3W, DPC and DPC-KNN perfectly cluster this dataset. FCM achieves the second-best performance due to wrongly clustering two samples in the sparse region C_1. RKM and CE3-kmeans get relatively worse results because they tend to wrongly cluster samples in the fringe region to other clusters with relatively higher densities.

D2 dataset is made up of four nonadjacent dense regions and each of them is attached with a sample. An additional sample that is difficult to identify lies in the center of the four parts. The clustering results of D2 are shown in Fig. 7. TW-RDPC, CE3-kmeans and 3W-DPET can correctly cluster the sample lies in the center by considering it as the boundary overlapping sample. It is quite reasonable because its distance from the four dense regions is relatively far and holds a relatively low density. M3W also correctly clusters the sample in the center and assigns the samples to the corresponding



(a) D2 dataset

(b) TW-RDPC

(c) CE3-kmeans

(d) 3W-DPET

(e) M3W

(f) DPC

(g) DPC-KNN

(h) FCM

(i) RKM

Fig. 7. Clustering results on D2.

positive region. Instead, DPC, DPC-KNN, FCM and RKM fail to correctly cluster the sample lies in the center thus having a slightly inferior performance.

Compound dataset consists of two adjacent circle regions, a nested concentric circle-like structure and a dense irregular shape surrounded by sparse areas. The clustering results on Compound are shown in Fig. 8. TW-RDPC, CE3-kmeans, 3W-DPET, M3W, DPC-KNN and FCM can correctly distinguish the adjacent two circles. Furthermore, TW-RDPC tackles sparse adjacent fringe areas with overlapping representation for careful management. 3W-DPET, M3W and DPC can better identify the nested concentric circle-like structure, but DPC wrongly considers two adjacent circles as one cluster and thus has a lower performance. 3W-DPET considers too many samples as boundary samples leading to a lack of information in positive regions. This problem of 3W-DPET may be unacceptable in some scenarios. For the rest area, the dense irregular shape is primarily and correctly identified by TW-RDPC, M3W, DPC, DPC-KNN and RKM thus the most information is kept. Integrating the above three parts, TW-RDPC obtains the most information in the whole dataset.

Aggregation dataset consists of one crescent-shaped cluster, four circle clusters with different sizes and two connected elliptical clusters. The clustering results of Aggregation are shown in Fig. 9. 3W-DPET gets the best performance, but some samples that should belong to the positive regions are excessively divided into the boundary regions. The results of TW-RDPC, DPC and DPC-KNN are nearly perfectly right except fail to correctly cluster the samples that connect the two elliptical clusters. M3W also achieves great results except fails to detect a few connection samples between C_5 and C_6. The results of CE3-kmeans, FCM and RKM show that the error assignment problem between clusters is more serious. They excessively split a cluster or partially merge clusters to nearby clusters because of their inferior of being unable to detect clusters with arbitrary shapes.

Table 5
Performance comparison for 7 clustering algorithms on 8 real-world datasets.

Algorithm	Purity	ARI	NMI	Params	Purity	ARI	NMI	Params
	Iris	(3 classes)			Parkinsons	(2 classes)		
TW-RDPC	**0.9867**	**0.9603**	**0.9403**	$k = 13$	**0.8564**	0.4171	0.3450	$k = 5$
CE3-kmeans	0.8867	0.7150	0.7315	$k = 9$	0.6395	0.0634	0.2427	$k = 11$
3W-DPET	0.9733	0.9222	0.9144	$k = 10$	0.8103	0.3104	0.1699	$k = 3$
M3W	0.9264	0.8032	0.9315	$k = 13\ \ L = 8$	0.8267	**0.7877**	**0.7977**	$k = 11\ \ L = 7$
DPC	0.9600	0.8857	0.8623	$d_c = 0.17$	0.8513	0.3910	0.3652	$d_c = 0.06$
DPC-KNN	0.8513	0.3910	0.6352	$k = 1$	0.8513	0.3910	0.6352	$k = 1$
FCM	0.9067	0.7560	0.7550	$m = 5$	0.6872	0.1347	0.2480	$m = 1.5$
RKM	0.9467	0.8515	0.8622	$w_u = 0.2\ \ \epsilon = 0.2$	0.7385	0.1924	0.0983	$w_u = 0.1\ \ \epsilon = 0.1$
	Seeds	(3 classes)			Thyroid	(3 classes)		
TW-RDPC	**0.9476**	**0.8518**	**0.8365**	$k = 18$	**0.9628**	**0.8732**	**0.8212**	$k = 16$
CE3-kmeans	0.8924	0.7087	0.6771	$k = 4$	0.9256	0.7482	0.6982	$k = 17$
3W-DPET	0.9333	0.8148	0.7945	$k = 4$	0.7674	0.3321	0.4048	$k = 4$
M3W	0.9190	0.7796	0.7644	$k = 21\ \ L = 10$	0.8419	0.5996	0.5440	$k = 10\ \ L = 12$
DPC	0.9048	0.7455	0.7424	$d_c = 0.06$	0.7628	0.2197	0.2849	$d_c = 0.6$
DPC-KNN	0.9048	0.7430	0.7077	$k = 3$	0.7814	0.2713	0.3172	$k = 1$
FCM	0.8905	0.7056	0.6793	$m = 4$	0.8884	0.6283	0.6072	$m = 1.5$
RKM	0.9143	0.7653	0.7249	$w_u = 0.2\ \ \epsilon = 0.3$	0.9349	0.7818	0.7037	$w_u = 0.1\ \ \epsilon = 0.3$
	Liver	(2 classes)			Pima	(2 classes)		
TW-RDPC	**0.6116**	**0.0332**	**0.0272**	$k = 12$	**0.7370**	**0.1897**	**0.1337**	$k = 14$
CE3-kmeans	0.5748	0.0103	0.0038	$k = 18$	0.6823	0.1230	0.0658	$k = 9$
3W-DPET	0.5710	−0.0041	0.0071	$k = 4$	0.6484	0.0131	0.0042	$k = 7$
M3W	0.5710	−0.0001	0.0004	$k = 20\ \ L = 3$	0.6458	0.0221	0.0157	$k = 13\ \ L = 5$
DPC	0.5768	0.0121	0.0043	$d_c = 0.08$	0.6732	0.1005	0.0464	$d_c = 0.01$
DPC-KNN	0.5710	−0.0046	0.0259	$k = 6$	0.6510	0.0119	0.0052	$k = 3$
FCM	0.5159	−0.0081	0.0031	$m = 1.5$	0.6680	0.1094	0.0687	$m = 2.5$
RKM	0.5913	0.0302	0.0197	$w_u = 0.5\ \ \epsilon = 0.5$	0.7279	0.1905	0.1103	$w_u = 0.2\ \ \epsilon = 0.2$
	Biodeg	(2 classes)			Unbalances	(8 classes)		
TW-RDPC	0.6569	−0.0055	0.0157	$k = 4$	**1.0000**	**1.0000**	**1.0000**	$k = 8$
CE3-kmeans	0.5789	−0.0396	0.0503	$k = 14$	0.9480	0.9629	0.9760	$k = 4$
3W-DPET	0.6322	−0.0233	0.0222	$k = 8$	**1.0000**	**1.0000**	**1.0000**	$k = 5$
M3W	0.5033	−0.0590	0.0823	$k = 30\ \ L = 6$	0.9829	0.9989	0.9759	$k = 30\ \ L = 3$
DPC	0.6398	−0.0184	0.0196	$d_c = 0.03$	0.9845	0.9986	0.9919	$d_c = 0.02$
DPC-KNN	0.6398	−0.0184	0.0196	$k = 2$	**1.0000**	**1.0000**	**1.0000**	$k = 6$
FCM	0.6199	0.0456	0.1367	$m = 4.5$	0.7275	0.7971	0.8145	$m = 1.0$
RKM	**0.7242**	**0.2000**	**0.1615**	$w_u = 0.2\ \ \epsilon = 0.2$	0.9202	0.9194	0.8896	$w_u = 0.1\ \ \epsilon = 0.1$

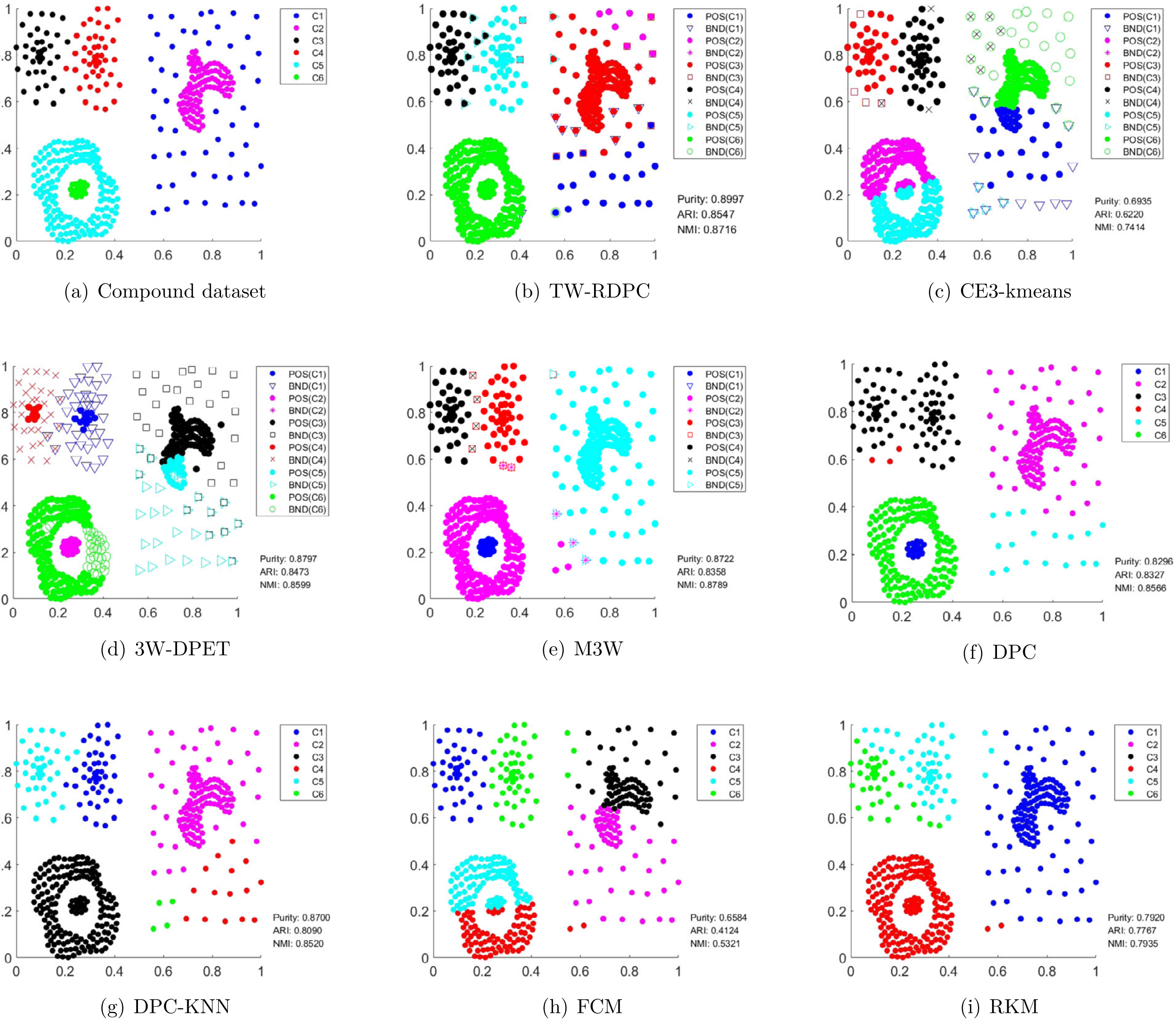
Triangle2 dataset consists of four Gaussian distributed clusters with varying variance. The clustering results of Triangles are shown in Fig. 10. The samples in the fringe regions are incorrectly clustered to different extents. TW-RDPC can correctly cluster samples in the boundary and overlapping samples to the greatest extent, followed by 3W-DPET, CE3-kmeans and M3W. DPC and DPC-KNN also get relatively great results. However, FCM does not perform well at the intersection regions. RKM gets the worst performance. It even wrongly clusters one whole cluster and unreasonably overly split the cluster at the lower left corner.

4C dataset consists of two adjacent dense circle-like regions, a dense linear region and a spare irregular region. The clustering results are shown in Fig. 11. All these algorithms can detect the two adjacent dense circle-like regions except CE3-kmeans. Meanwhile, All these algorithms can detect the spare irregular region except FCM. To the dense linear region, M3W correctly detects nearly two-thirds of this region. However, TW-RDPC, DPC and DPC-KNN can only correctly detect nearly half of this region. So, M3W achieves the best results and TW-RDPC achieves the second-best results.

In conclusion, the advantages of TW-RDPC are summarized as: Firstly, compared with 3W-DPET, it keeps as much information as possible in positive regions which enhances its usability. Secondly, compared with CE3-kmeans, FCM and RKM, it is more able to detect clusters with arbitrary shapes and varying densities and outputs stable results. Thirdly, compared with DPC, DPC-KNN, FCM and RKM, it tackles adjacent and overlapping regions more meticulously by three-way representation. To the best of our knowledge, TW-RDPC is the only three-way clustering algorithm that can represent an overlapping sample that belongs to one positive region of a cluster and belongs to boundary regions of other clusters at the same time. This representation is more reasonable in some cases for reflecting different degrees of subordination.

5.4. Experimental results on real-world data sets

Table 5 shows the clustering results on 8 real-world datasets, together with the best parameter settings. The best performances for each dataset are highlighted in bold. It is obvious that TW-RDPC gets the best or relatively better results than other algorithms in all datasets except that its performance on Biodeg is slightly inferior to RKM. But compared with RKM, TW-RDPC outputs unique certain results and needs fewer parameters. While RKM will output different results each running time because of different random initial cluster centers. The results of TW-RDPC on Iris, Seeds, Thyroid and Unbalance are completely or nearly completely correct.



(a) Compound dataset
(b) TW-RDPC
(c) CE3-kmeans
(d) 3W-DPET
(e) M3W
(f) DPC
(g) DPC-KNN
(h) FCM
(i) RKM

Fig. 8. Clustering results on Compound.

5.5. Detailed comparison with 3W-DPET

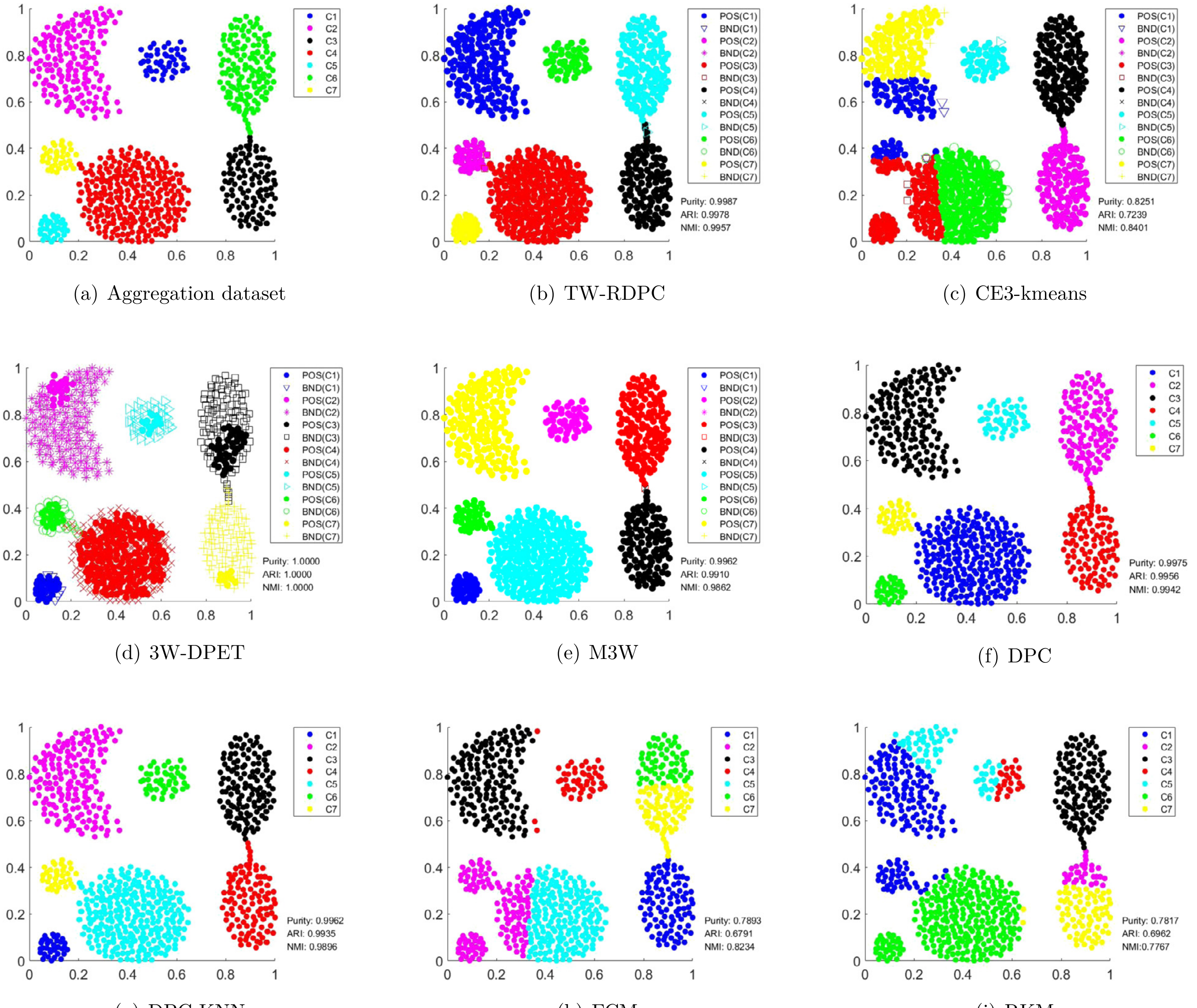
A similar approach with the name of 3W-DPET is also aimed at the error propagation problem. In this section, we will compare our algorithm with 3W-DPET and explain the advantage of our algorithm in detail.

For example, the clustering results on Aggregation and Triangle2 obtained by the two algorithms are shown in Fig. 12.

Solid points represent samples in positive regions, which means they belong to the clusters. Other samples represented by hollow shapes or * or × are in boundary regions, which means they may (or may not) belong to the clusters. Different colors represent different clusters.

Intuitively, in Fig. 12 (a) and Fig. 12 (c), 3W-DPET only assigns a few samples to positive regions (marked with red dashed lines). There still exists a large number of boundary samples that need deferral decisions. In real scenes, this means more background information is needed to mine a large number of uncertain samples. This problem of 3W-DPET commonly exists in other datasets. While in Fig. 12 (b) and Fig. 12 (d), most samples are assigned to positive regions. So, our algorithm keeps most certain information and only a few boundary samples are needed to be further determined.

Meanwhile, the two algorithms use two different ways to represent overlapping samples (marked with the green dashed line). In Fig. 12 (a) and Fig. 12 (c), the overlapping samples are all boundary samples, which is hard to reflect the different subordination to different clusters. While in Fig. 12 (b) and Fig. 12 (d), an overlapping sample may be assigned to the positive region of one cluster and be assigned to the boundary region of another cluster. The benefit of this representation is that it can reflect different levels of subordination.



(a) Aggregation dataset (b) TW-RDPC (c) CE3-kmeans

(d) 3W-DPET (e) M3W (f) DPC

(g) DPC-KNN (h) FCM (i) RKM

Fig. 9. Clustering results on Aggregation.

5.6. Sensitivity analysis

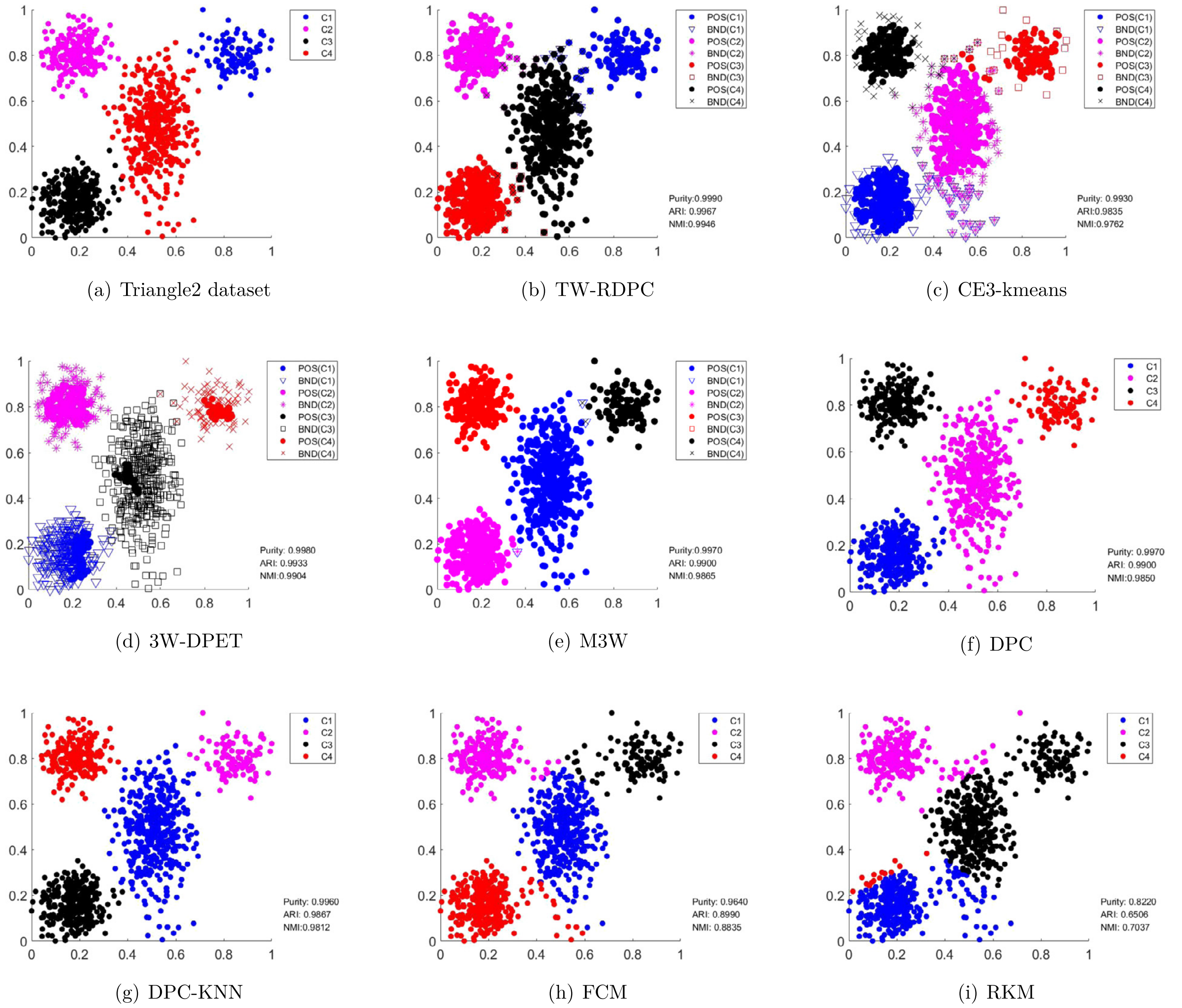
In order to validate the thresholds in our boundary detection graph, we conduct sensitivity analysis on 5 datasets. Please note that the threshold of ρ and the threshold of ρ' are the same, for the sake of convenience. Because the two thresholds are all used to partition the datasets by percentage. Further, ρ' is more tending to measure local density compared with ρ. We evaluate Purity, ARI and NMI (see Figs. 13–15) and observe whether the results of the three evaluations are stable. The thresholds range from 20% to 30% by step 1%.

Quartile achieves 4 best Purity on the 5 datasets. Especially, on the Parkinsons dataset, the purity is 0.8564, which is relatively outstanding. When using the quartile on Aggregation dataset, the purity is 0.9987, which is quite close to the best Purity 1.0000. Meanwhile, on Thyroid dataset, although the thresholds change, the Purity remains unchanged, keeping 0.8824.

Quartile achieves 3 best ARI on the 5 datasets. Especially on the Parkinsons dataset, ARI is 0.4171, which is larger than the ARI values corresponding to other thresholds. Although on Zelink6 and Aggregation datasets, the ARI is slightly lower when choosing quartile, the overall value of ARI is stable.

Quartile achieves 4 best NMI on the 5 datasets. When choosing quartile, NMI is relatively outstanding on Seeds, Zelink6 and Thyroid datasets. Meanwhile, on Aggregation dataset, the NMI is 0.9957, which is quite close to the best NMI 1.0000.

In general, Purity, ARI and NMI are very stable when the thresholds change between 20% and 30%, and the quartile achieves the best or relatively better results. So the quartile thresholds are reliable.



(a) Triangle2 dataset (b) TW-RDPC (c) CE3-kmeans

(d) 3W-DPET (e) M3W (f) DPC

(g) DPC-KNN (h) FCM (i) RKM

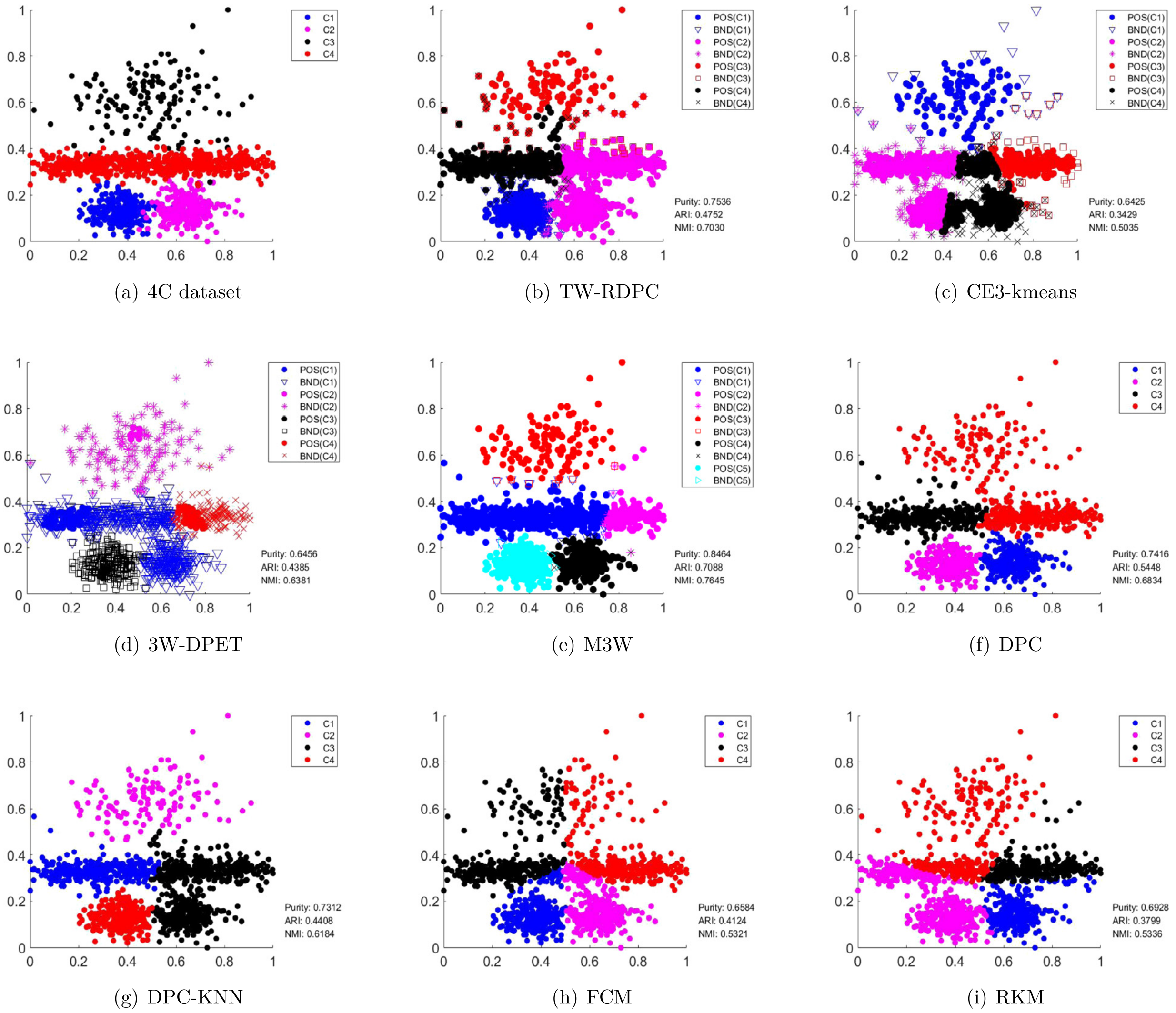
Fig. 10. Clustering results on Triangle2.

6. Conclusion

In this paper, we propose a three-way clustering method based on DPC and boundary detection graph called TW-RDPC. One contribution of TW-RDPC is that it is more adapted to varying density distribution and different shapes by improved density measurement. Another contribution of TW-RDPC is that it extends DPC to a three-way paradigm which is consistent with human cognitive thinking. The algorithm mainly includes three steps: In the first step, the proposed relative Cauchy kernel density is adopted to improve DPC's density estimation. In the second step, we create a unique boundary detection graph based on quartile threshold to identify potential boundary samples. In the third step, the KNN method is applied to assign potential boundary samples to corresponding regions. Through experiments, we compare our algorithm to 7 clustering algorithms on 10 synthetic and 8 real-world datasets to validate our algorithm. The evaluations on Purity, ARI and NMI prove our algorithm achieves satisfactory performance.

In future work, to begin with, we intend to extend the boundary detection graph to other density-based clustering algorithms and find more appropriate density measurement strategies. Furthermore, instead of using the quartile to get thresholds in the boundary detection graph, we will explore dynamic methods to automatically identify thresholds based on relative data distribution. Moreover, we will improve the three-way process to better handle boundary regions. Based on these steps, we can develop a general framework for extending traditional density-based clustering algorithms to three-way clustering methods.

Moreover, evidential clustering is a rising general framework that may extend other clustering approaches, including hard clustering and also many other soft clustering approaches like fuzzy clustering, rough clustering, etc. Especially, when



(a) 4C dataset

(b) TW-RDPC

(c) CE3-kmeans

(d) 3W-DPET

(e) M3W

(f) DPC

(g) DPC-KNN

(h) FCM

(i) RKM

Fig. 11. Clustering results on 4C.

specific conditions are satisfied, a three-way clustering corresponding to a unique extended credal partition, and vice versa. So, we will pay more attention to exploring the mutual support between the two highly related fields.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

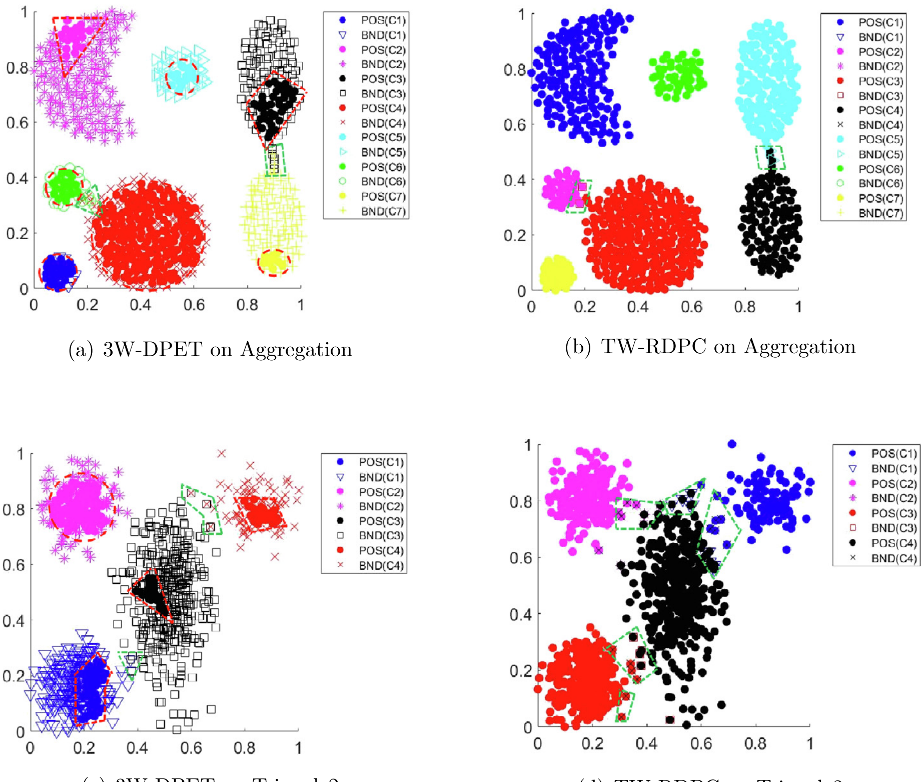
Data will be made available on request.

Acknowledgements

This work is partly funded by the National Natural Science Foundation of China (nos. 62006104, 61872168 and 62107022), and the Natural Science Foundation of the Jiangsu Higher Education Institutions (no. 20KJB520012).

Appendix A. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.ijar.2022.12.002.



(a) 3W-DPET on Aggregation

(b) TW-RDPC on Aggregation

(c) 3W-DPET on Triangle2

(d) TW-RDPC on Triangle2

Fig. 12. Comparison between 3W-DPET and TW-RDPC.

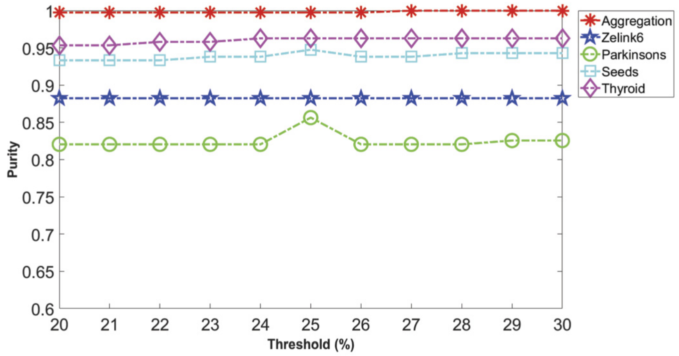


Fig. 13. Sensitivity analysis on Purity.

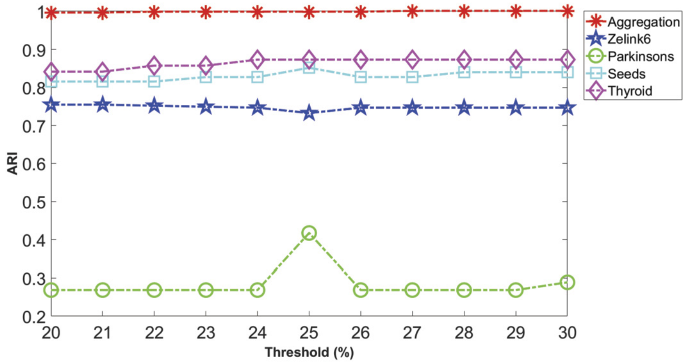


Fig. 14. Sensitivity analysis on ARI.

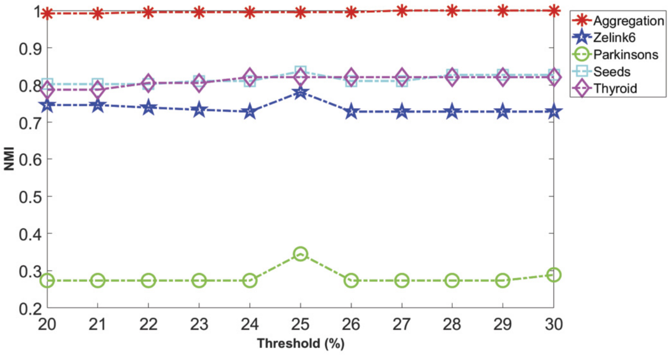


Fig. 15. Sensitivity analysis on NMI.

References

[1] L. Kaufman, P.J. Rousseeuw, Finding Groups in Data: an Introduction to Cluster Analysis, John Wiley & Sons, 2009.

[2] M. Lu, X. Zhao, L. Zhang, F. Li, Semi-supervised concept factorization for document clustering, Inf. Sci. 331 (2016) 86–98, https://doi.org/10.1016/j.ins.2015.10.038.

[3] W. Ding, S. Chakraborty, K. Mali, S. Chatterjee, J. Nayak, A.K. Das, S. Banerjee, An unsupervised fuzzy clustering approach for early screening of Covid-19 from radiological images, IEEE Trans. Fuzzy Syst. 30 (8) (2022) 2902–2914, https://doi.org/10.1109/TFUZZ.2021.3097806.

[4] H. Yu, K. Mao, J. Shi, H. Huang, Z. Chen, K. Dong, S. Yiu, Predicting and understanding comprehensive drug-drug interactions via semi-nonnegative matrix factorization, Syst. Biol. 12 (1) (2018) 101–110, https://doi.org/10.1186/s12918-018-0532-7.

[5] P. Jiao, W. Yu, W. Wang, X. Li, Y. Sun, Exploring temporal community structure and constant evolutionary pattern hiding in dynamic networks, Neurocomputing 314 (2018) 224–233, https://doi.org/10.1016/j.neucom.2018.03.065.

[6] X. Xu, S. Ding, L. Wang, Y. Wang, A robust density peaks clustering algorithm with density-sensitive similarity, Knowl.-Based Syst. 200 (2020) 106028, https://doi.org/10.1016/j.knosys.2020.106028.

[7] Y. Chen, X. Hu, W. Fan, L. Shen, Z. Zhang, X. Liu, J. Du, H. Li, Y. Chen, H. Li, Fast density peak clustering for large scale data based on kNN, Knowl.-Based Syst. 187 (2020) 104824, https://doi.org/10.1016/j.knosys.2019.06.032.

[8] R. Liu, H. Wang, X. Yu, Shared-nearest-neighbor-based clustering by fast search and find of density peaks, Inf. Sci. 450 (2018) 200–226, https://doi.org/10.1016/j.ins.2018.03.031.

[9] S.A. Seyedi, A. Lotfi, P. Moradi, N.N. Qader, Dynamic graph-based label propagation for density peaks clustering, Expert Syst. Appl. 115 (2019) 314–328, https://doi.org/10.1016/j.eswa.2018.07.075.

[10] H. Yu, L. Chen, J. Yao, A three-way density peak clustering method based on evidence theory, Knowl.-Based Syst. 211 (2021) 106532, https://doi.org/10.1016/j.knosys.2020.106532.

[11] Z. Su, T. Denoeux, BPEC: belief-peaks evidential clustering, IEEE Trans. Fuzzy Syst. 27 (1) (2019) 111–123, https://doi.org/10.1109/TFUZZ.2018.2869125.

[12] A. Campagner, D. Ciucci, T. Denœux, Belief functions and rough sets: survey and new insights, Int. J. Approx. Reason. 143 (2022) 192–215, https://doi.org/10.1016/j.ijar.2022.01.011.

[13] Y. Yao, Three-Way Decision: an Interpretation of Rules in Rough Set Theory, Proceedings of 4th International Conference on Rough Sets and Knowledge Technology, vol. 5589, Gold, Coast, Australia, 2009, pp. 642–649.

[14] Y. Yao, Three-way decisions with probabilistic rough sets, Inf. Sci. 180 (3) (2010) 341–353, https://doi.org/10.1016/j.ins.2009.09.021.

[15] Y. Yao, The superiority of three-way decisions in probabilistic rough set models, Inf. Sci. 181 (6) (2011) 1080–1096, https://doi.org/10.1016/j.ins.2010.11.019.

[16] Y. Yao, Tri-level thinking: models of three-way decision, Int. J. Mach. Learn. Cybern. 11 (5) (2020) 947–959, https://doi.org/10.1007/s13042-019-01040-2.

[17] A. Mintz, N. Geva, S.B. Redd, A. Carnes, The effect of dynamic and static choice sets on political decision making: an analysis using the decision board platform, Am. Polit. Sci. Rev. 91 (3) (1997) 553–566.

[18] Q. Zhang, G. Lv, Y. Chen, G. Wang, A dynamic three-way decision model based on the updating of attribute values, Knowl.-Based Syst. 142 (2018) 71–84, https://doi.org/10.1016/j.knosys.2017.11.026.

[19] X. Yue, J. Zhou, Y. Yao, D. Miao, Shadowed neighborhoods based on fuzzy rough transformation for three-way classification, IEEE Trans. Fuzzy Syst. 28 (5) (2020) 978–991, https://doi.org/10.1109/TFUZZ.2020.2979365.

[20] H.U. Rehman, N. Azam, J. Yao, A. Benso, A three-way approach for protein function classification, PLoS ONE (2017), https://doi.org/10.1371/journal.pone.0171702.

[21] X. Wang, P. Wang, X. Yang, Y. Yao, Attribution reduction based on sequential three-way search of granularity, Int. J. Mach. Learn. Cybern. 12 (5) (2021) 1439–1458, https://doi.org/10.1007/s13042-020-01244-x.

[22] J. Yao, Y. Yao, D. Ciucci, K. Huang, Granular computing and three-way decisions for cognitive analytics, Cogn. Comput. (2022), https://doi.org/10.1007/s12559-022-10028-0.

[23] G. Lang, D. Miao, H. Fujita, Three-way group conflict analysis based on Pythagorean fuzzy set theory, IEEE Trans. Fuzzy Syst. 28 (3) (2020) 447–461, https://doi.org/10.1109/TFUZZ.2019.2908123.

[24] H. Yu, Y. Wang, Three-way decisions method for overlapping clustering, in: Proceedings of 8th International Conference on Rough Sets and Current Trends in Computing, Chengdu, China, 2012, pp. 277–286.

[25] A. Shah, N. Azam, E. Alanazi, J. Yao, Image blurring and sharpening inspired three-way clustering approach, Appl. Intell. (2022), https://doi.org/10.1007/s10489-021-03072-0.

[26] P. Wang, Y. Yao, CE3: a three-way clustering method based on mathematical morphology, Knowl.-Based Syst. 155 (2018) 54–65, https://doi.org/10.1016/j.knosys.2018.04.029.

[27] J.C. Bezdek, R. Ehrlich, W. Full, FCM: the fuzzy c-means clustering algorithm, Comput. Geosci. 10 (2–3) (1984) 191–203, https://doi.org/10.1016/0098-3004(84)90020-7.

[28] P. Lingras, C. West, Interval set clustering of web users with rough k-means, J. Intell. Inf. Syst. 23 (1) (2004) 5–16, https://doi.org/10.1023/B:JIIS.0000029668.88665.1a.

[29] H. Yu, A framework of three-way cluster analysis, in: Proceedings of International Joint Conference on Rough Sets, Olsztyn, Poland, 2017, pp. 300–312.

[30] M.K. Afridi, N. Azam, J. Yao, E. Alanazi, A three-way clustering approach for handling missing data using GTRS, Int. J. Approx. Reason. 98 (2018) 11–24, https://doi.org/10.1016/j.ijar.2018.04.001.

[31] M.K. Afridi, N. Azam, J. Yao, Variance based three-way clustering approaches for handling overlapping clustering, Int. J. Approx. Reason. 118 (2020) 47–63, https://doi.org/10.1016/j.ijar.2019.11.011.

[32] H. Yu, C. Zhang, G. Wang, A tree-based incremental overlapping clustering method using the three-way decision theory, Knowl.-Based Syst. 91 (2016) 189–203, https://doi.org/10.1016/j.knosys.2015.05.028.

[33] Á. López-Oriona, P. D'Urso, J.A. Vilar, B. Lafuente-Rego, Quantile-based fuzzy c-means clustering of multivariate time series: robust techniques, Int. J. Approx. Reason. 150 (2022) 55–82, https://doi.org/10.1016/j.ijar.2022.07.010.

[34] J. Xiong, H. Yu, A three-way clustering algorithm via decomposing similarity matrices for multi-view data with noise, in: Proceedings of International Joint Conference on Rough Sets, Debrecen, Hungary, 2019, pp. 179–193.

[35] H. Yu, X. Wang, G. Wang, X. Zeng, An active three-way clustering method via low-rank matrices for multi-view data, Inf. Sci. 507 (2020) 823–839, https://doi.org/10.1016/j.ins.2018.03.009.

[36] G.A. Khan, J. Hu, T. Li, B. Diallo, Y. Zhao, Multi-view low rank sparse representation method for three-way clustering, Int. J. Mach. Learn. Cybern. 13 (1) (2022) 233–253, https://doi.org/10.1007/s13042-021-01394-6.

[37] H. Yu, Y. Chen, P. Lingras, G. Wang, A three-way cluster ensemble approach for large-scale data, Int. J. Approx. Reason. 115 (2019) 32–49, https://doi.org/10.1016/j.ijar.2019.09.001.

[38] C. Jiang, S. Zhao, Multi-granulation three-way clustering ensemble based on shadowed sets, Acta Electron. Sin. 49 (8) (2021) 1524.

[39] C. Jiang, Z. Li, J. Yao, A shadowed set-based three-way clustering ensemble approach, Int. J. Mach. Learn. Cybern. 13 (2022) 2558–32545, https://doi.org/10.1007/s13042-022-01543-5.

[40] P. Wang, X. Chen, Three-way ensemble clustering for incomplete data, IEEE Access 8 (2020) 91855–91864, https://doi.org/10.1109/ACCESS.2020.2994380.

[41] A. Shah, N. Azam, B. Ali, M.T. Khan, J. Yao, A three-way clustering approach for novelty detection, Inf. Sci. 569 (2021) 650–668, https://doi.org/10.1016/j.ins.2021.05.021.

[42] B. Ali, N. Azam, A. Shah, J. Yao, A spatial filtering inspired three-way clustering approach with application to outlier detection, Int. J. Approx. Reason. 130 (2021) 1–21, https://doi.org/10.1016/j.ijar.2020.12.003.

[43] J. MacQueen, Classification and analysis of multivariate observations, in: Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, 1967, pp. 281–297.

[44] P. Wang, H. Shi, X. Yang, J. Mi, Three-way k-means: integrating k-means and three-way decision, Int. J. Mach. Learn. Cybern. 10 (10) (2019) 2767–2777, https://doi.org/10.1007/s13042-018-0901-y.

[45] Q. Shen, Q. Zhang, F. Zhao, G. Wang, Adaptive three-way c-means clustering based on the cognition of distance stability, Cogn. Comput. 14 (2) (2022) 563–580, https://doi.org/10.1007/s12559-021-09965-z.

[46] K. Zhang, A three-way c-means algorithm, Appl. Soft Comput. 82 (2019) 105536, https://doi.org/10.1016/j.asoc.2019.105536.

[47] M. Ester, H.P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA, 1996, pp. 226–231.

[48] H. Yu, L. Chen, J. Yao, X. Wang, A three-way clustering method based on an improved DBSCAN algorithm, Phys. A, Stat. Mech. Appl. 535 (2019) 122289.

[49] A. Rodriguez, A. Laio, Clustering by fast search and find of density peaks, Science 344 (6191) (2014) 1492–1496, https://doi.org/10.1126/science.1242072.

[50] H. Yu, Z. Chang, G. Wang, X. Chen, An efficient three-way clustering algorithm based on gravitational search, Int. J. Mach. Learn. Cybern. 11 (5) (2020) 1003–1016, https://doi.org/10.1007/s13042-019-00988-5.

[51] M. Du, R. Wang, R. Ji, X. Wang, Y. Zhang, ROBP a robust border-peeling clustering using Cauchy kernel, Inf. Sci. 571 (2021) 375–400, https://doi.org/10.1016/j.ins.2021.04.089.

[52] M. Du, J. Zhao, J. Sun, Y. Dong, M3W: multistep three-way clustering, IEEE Trans. Neural Netw. Learn. Syst. (2022), https://doi.org/10.1109/TNNLS.2022.3208418 1–14.

[53] B. Ali, N. Azam, J. Yao, A three-way clustering approach using image enhancement operations, Int. J. Approx. Reason. 149 (2022) 1–38, https://doi.org/10.1016/j.ijar.2022.07.001.

[54] A. Shah, N. Azam, E. Alanazi, J. Yao, Image blurring and sharpening inspired three-way clustering approach, Appl. Intell. (2022), https://doi.org/10.1007/s10489-021-03072-0.

[55] D. Papadias, Y. Tao, Reverse nearest neighbor query, https://doi.org/10.1007/978-0-387-39940-9_318, 2009, pp. 2434–2438.

[56] T. Qian, J. Wang, Y. Yang, Matching pursuits among shifted Cauchy kernels in higher-dimensional spaces, Acta Sci. Math. 34 (3) (2014) 660–672, https://doi.org/10.1016/S0252-9602(14)60038-2.

[57] L. Breiman, W. Meisel, E. Purcell, Variable kernel estimates of multivariate densities, Technometrics 19 (2) (1977) 135–144.

[58] R. Li, X. Yang, X. Qin, W. Zhu, Local gap density for clustering high-dimensional data with varying densities, Knowl.-Based Syst. 184 (2019) 104905, https://doi.org/10.1016/j.knosys.2019.104905.

[59] M. Du, S. Ding, H. Jia, Study on density peaks clustering based on k-nearest neighbors and principal component analysis, Knowl.-Based Syst. 99 (2016) 135–145, https://doi.org/10.1016/j.knosys.2016.02.001.

[60] K. Spafford, J.S. Meredith, J.S. Vetter, Quartile and outlier detection on heterogeneous clusters using distributed radix sort, in: Proceedings of International Conference on Cluster Computing (CLUSTER), Austin, TX, USA, 2011, pp. 412–419.

[61] S. Goswami, A. Chakrabarti, Quartile clustering: a quartile based technique for generating meaningful clusters, CoRR, arXiv:1203.4157 [abs], 2012.

[62] P. Frnti, O. Virmajoki, V. Hautamki, Fast agglomerative clustering using a k-nearest neighbor graph, IEEE Trans. Pattern Anal. Mach. Intell. 28 (11) (2006) 1875–1881, https://doi.org/10.1109/TPAMI.2006.227.

[63] S. Ding, H. Jia, Z. Shi, Spectral clustering algorithm based on adaptive nystrom sampling for big data analysis, J. Softw. 25 (9) (2014) 2037–2049, https://doi.org/10.13328/j.cnki.jos.004643.

[64] W.M. Rand, Objective criteria for the evaluation of clustering methods, J. Am. Stat. Assoc. 66 (336) (1971) 846–850.

[65] D. Cai, X. He, J. Han, T.S. Huang, Graph regularized nonnegative matrix factorization for data representation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2011) 1548–1560, https://doi.org/10.1109/TPAMI.2010.231.