Soft Computing
https://doi.org/10.1007/s00500-018-3221-y

METHODOLOGIES AND APPLICATION

Quasi-cluster centers clustering algorithm based on potential entropy and t-distributed stochastic neighbor embedding

Xian Fang[1] · Zhixin Tie[1] · Yinan Guan[1] · Shanshan Rao[1]

Abstract
A novel density-based clustering algorithm named QCC is presented recently. Although the algorithm has proved its strong robustness, it is still necessary to manually determine the two input parameters, including the number of neighbors (k) and the similarity threshold value (α), which severely limits the promotion of the algorithm. In addition, the QCC does not perform excellently when confronting the datasets with relatively high dimensions. To overcome these defects, firstly, we define a new method for computing local density and introduce the strategy of potential entropy into the original algorithm. Based on this idea, we propose a new QCC clustering algorithm (QCC-PE). QCC-PE can automatically extract optimal value of the parameter k by optimizing potential entropy of data field. By this means, the optimized parameter can be calculated from the datasets objectively rather than the empirical estimation accumulated from a large number of experiments. Then, t-distributed stochastic neighbor embedding (tSNE) is applied to the model of QCC-PE and further brings forward a method based on tSNE (QCC-PE-tSNE), which preprocesses high-dimensional datasets by dimensionality reduction technique. We compare the performance of the proposed algorithms with QCC, DBSCAN, and DP in the synthetic datasets, Olivetti Face Database, and real-world datasets respectively. Experimental results show that our algorithms are feasible and effective and can often outperform the comparisons.

Keywords Data clustering · Quasi-cluster centers clustering · Potential entropy · Optimal parameter · t-distributed stochastic neighbor embedding

1 Introduction

The purpose of clustering is that dividing the objects into different clusters or classes according to the similarity of sample data. Clustering technology has been widely used in many fields: pattern recognition (Horn and Gottlieb 2002), image processing (Liew and Yan 2003; Li and Shen 2010), and machine learning (Wu 2014). The traditional clustering methods could be roughly grouped into five categories: hierarchical clustering, partition-based clustering, density-based clustering, grid-based clustering, and model-based clustering (Omran et al. 2007; Xu and Tian 2015).

Communicated by V. Loia.

✉ Zhixin Tie
tiezx@zstu.edu.cn

Xian Fang
xianfangfx@163.com

[1] School of Information Science and Technology, Zhejiang Sci-Tech University, Hangzhou, Zhejiang, China

The basic idea of hierarchical clustering is to establish a hierarchical relationship of all data points based on the hierarchical tree structure, there are two ways to realize it: bottom-up and top-down. The former supposes that each object stands for an individual cluster at the beginning; then, the most similar two clusters are merged into a new cluster loops until the last one is left. The latter is the opposite process. BIRCH (Zhang et al. 1996, 1997; Madan and Dana 2015), ROCK (Guha et al. 1999; Dutta et al. 2005), and Chameleon (Karypis et al. 1999) are the representatives of this sort of method. Hierarchical clustering does not require the number of clusters to be specified in advance and can handle isolated and noise data well, but the complexity of time and space is too high to be suitable for large dataset. Partition-based clustering regards the center of data points as the center of the corresponding cluster, and the quality of clustering would be gradually improving through attempting to move data objects from one cluster to others employing iterative relocation technique. K-means (Macqueen 1967) and K-medoids (Park and Jun 2009) are the

Published online: 11 May 2018

two most famous ways of this kind of clustering algorithm. Partition-based clustering has relatively low time complexity and high computing efficiency in general. However, it does not suitable for non-convex datasets and sensitive to the outliers. Except that, the number of clusters needs to be preset. The core idea of density-based clustering is that the data in the region with high density of the data space are considered to belong to the same cluster. There are some representatives: DBSCAN (Ester et al. 1996; Kumar and Reddy 2016), OPTICS (Ankerst et al. 1999), Mean-shift (Comaniciu and Meer 2002), and DP (Rodriguez and Laio 2014; Du et al. 2016; Mehmood et al. 2016). Density-based clustering can correctly cluster the non-spherical-shape datasets, but it always produces clustering results with low quality when the density of data space is not even. Grid-based clustering is based on the idea that the object space is quantized into a finite number of cells, thereby forming a grid structure so that all the clustering operations are carried out in this grid structure. STING (Wang et al. 1997) and CLIQUE (Agrawal et al. 1998) are the well-known branches of such clustering. The advantage of grid-based clustering can be attributed to its great efficiency; namely, the algorithm performance is independent of the number of objects and just depends on the number of units on each dimension in the data space. There are also many disadvantages, like being sensitive to choosing parameters, unable to deal with irregularly distributed data, and curse of dimensionality. Model-based clustering assumes a model for each class and then looks for the objects which conform to the model. It tries to get the best fit between the given data and a mathematical model. These models are divided into the model based on probabilistic, which includes GMM (Rasmussen 2000), COBWEB (Fisher 1987), and the model based on neural network, which includes SOM (Kohonen 1998), ART (Carpenter and Grossberg 1987, 1990). However, it is usually difficult to find the model or distribution of real datasets before clustering.

All of the above clustering algorithms possess their own strong points and applicable fields, but the inefficiency in complex manifold datasets has becoming general character. Aiming at clustering data with arbitrary shape excellently, a novel density-based clustering algorithm was proposed, called QCC (Huang et al. 2017). In comparison with the performance of DP, DAAP, and DBCAN algorithms, the efficiency and robustness of the algorithm is manifested. However, QCC still has some defects, and we enumerate the most significant two aspects.

First, two parameters of QCC need to be set on the basis of one's empirical experience. For the same datasets, different users estimate the parameters and the results of evaluation may be varied. Even the same user will also be subject to inconsistent evaluation under different external conditions. It therefore becomes a challenge for this clustering algorithm to get optimal parameters. To solve this problem, we propose a QCC-PE algorithm to automatically search for the optimal QCC's parameter k from the original datasets by using the potential entropy.

Second, traditional clustering algorithms suffer two main problems all the time when clustering in high-dimensional datasets (Tomasev et al. 2014). On the one hand, the high-dimensional datasets have a large number of irrelevant attributes, which makes the possibility of the existence of clusters in all the dimensions becomes almost zero. On the other hand, the distribution of the data in the high-dimensional space is sparser than which in the low-dimensional space, in which it is common phenomenon that the distances between the data are almost equal. Thus, it is not advisable to purely build clusters based on the distance in high-dimensional space, while the traditional QCC calculates local density of each point relying entirely on the simple Euclidean distance between points. To allow QCC algorithm to apply in high-dimensional space data, we further bring forward a method called QCC-PE-tSNE, which is based on t-distributed stochastic neighbor embedding (tSNE).

We test the proposed algorithms in the synthetic datasets, Olivetti Face Database, and real-world datasets compared with QCC and some other outstanding clustering algorithms including DBSCAN and DP. The experimental results show that the proposed algorithms are very competitive with the comparisons. The rest of this paper is organized as follows. Section 2 introduces the related works including introducing principal concept of the QCC algorithm, the potential entropy, and the tSNE. Section 3 presents QCC-PE and QCC-PE-tSNE algorithms in detail. Section 4 gives the experimental results and comparative analysis. Finally, conclusions are drawn in Sect. 5.

2 Related works

2.1 QCC clustering algorithm

QCC clustering algorithm mainly relies on the two ideas: The density of a cluster center is the highest in its k nearest neighborhood or reverse k nearest neighborhood, and clusters are divided by sparse regions. This method requires two important parameters to be manually set in advance: One is the number of neighbors, denoted by k, and it plays the role of computing local density and determining quasi-cluster centers; the other is the similarity threshold value, denoted by α, and it is the threshold for merging similar classes. These two parameters are determined in practice through the cumulative experience of many experiments. Specifically, quasi-cluster centers number inclines to decrease as the value of parameter k increases, and larger values of parameter α often result in more clustering results. In the following, we will describe the computation of the local density of each point ρ_i and the sim-

ilarity between clusters $sim(c_i, c_j)$, which are closely related by two corresponding parameters.

Suppose that there are point sets $X = \{x_1, x_2, \ldots, x_n\}$. Most density-based clustering algorithms, such as DBSCAN and DP, calculate ρ_i according to Eq. (1).

$$\rho_i = \sum_{j=1}^{n} \chi(d_{ij} - d_c) \tag{1}$$

where $\chi(x) = 1$ if $x < 0$ and $\chi(x) = 0$ otherwise; d_{ij} denotes the Euclidean distance between the point x_i and x_j. d_c is a cutoff distance.

From the point of getting more precise neighborhoods density distribution, QCC proposes to take both the k nearest neighbor (KNN) and reverse k nearest neighbor (RKNN) into account. Eq. (2) gives the calculation method of density using KNN in QCC.

$$\rho_i = \frac{1}{Dist_k(x_i)} \tag{2}$$

where $Dist_k(x_i)$ is the distance $d(x_i, o)$ between x_i and o in X, such that: At least k objects $o' \in X/\{x_i\}$ satisfy $d(x_i, o') \leq d(x_i, o)$ and at most k-1 objects $o' \in X/\{x_i\}$ satisfy $d(x_i, o') < d(x_i, o)$. KNN and RKNN are used to determine whether the definition of quasi-cluster center is satisfied. The definition of $KNN(x_i)$ and $RKNN(x_i)$ is shown as follows:

$$KNN(x_i) = \{x_j \mid (x_i, x_j) \leq Dist_k(x_i)\} \tag{3}$$

$$RKNN(x_i) = \{x_j \mid (x_i, x_j) \leq Dist_k(x_j)\} \tag{4}$$

When x_i satisfies the following condition, x_i is called a quasi-cluster center.

$$\forall x_j \in KNN(x_i) \; or \; RKNN(x_i), \rho(i) \geq \rho(j) \tag{5}$$

The computation of $sim(c_i, c_j)$ is quite easy that is defined as the radio of the number of points in $c_i \bigcap c_j$ and the value of k as follows:

$$sim(c_i, c_j) = \frac{c_i \bigcap c_j}{k} \tag{6}$$

The following algorithm is a summary of the QCC.

Algorithm 1 : QCC algorithm

Input: The dataset (X), the number of neighbor of each point (k), and the minimum similarity between cluster (α)
Output: The final cluster results $C = \{c_1, c_2, \ldots, c_M\}$
1: Calculate the density of each point x_i according to Eq. (2)
2: Calculate the $KNN(x_i)$ and $RKNN(x_i)$ according to Eqs. (3–4)
3: Search quasi-clustering center according to Eq. (5)
4: Calculate the similarity matrix $sim(c_i, c_j)$ between the clusters according to Eq. (6)
5: Merge all initial clusters those $sim(c_i, c_j) > \alpha$
6: Return C

2.2 Potential entropy

Objects in certain areas are usually interrelated and interacting, and each object often has a propagation radius. In physics, this argument is described by "field". The physical volume which only has the size without the direction is called scalar field, and the physical volume which have the both size and direction is called vector field. Potential field is a scalar field and always described by potential function. Obviously, each data object in the spatial domain contributes to the potential function. The potential is relatively strong in relatively intensive regions of data. In contrast, it is relatively weak in the data sparse regions (Barbieri et al. 2014; Wang et al. 2016; Zang et al. 2017). For the point sets X, the potential field function is defined as Eq. (7)

$$\varphi_i = \sum_{j=1}^{n} m_j \cdot K\left(\frac{d_{ij}}{\sigma}\right) \tag{7}$$

where σ is an impact factor which is used to control the influence range, m_j is the mass of x_j, and $K(x)$ is a unit potential function.

The uncertainty of data (i.e., the degree of chaos in the system) is represented by the topological potential entropy. The greater the entropy is, the greater the uncertainty is. In the data field, if the potential value of a data object is equal to other data objects, the uncertainty of the original data distribution is the largest, that is, the entropy is the largest. If the potential value on the data object position is extremely asymmetric, the uncertainty is minimal, that is, the entropy is minimal. Let the potential of every point in the field be $\{\varphi_1, \varphi_2, \ldots, \varphi_n\}$ and the potential entropy H be:

$$H = -\sum_{i=1}^{n} \frac{\varphi_i}{Z} \log\left(\frac{\varphi_i}{Z}\right) \tag{8}$$

where $Z = \sum_{i=1}^{n} \varphi_i$ is a normalization factor.

2.3 t-distributed stochastic neighbor embedding

As a classical dimensionality reduction algorithm, the tSNE algorithm adopts a nonlinear dimensionality reduction method, which is distinctly different from linear dimensionality reduction algorithms such as principal component analysis (PCA) and linear discriminant analysis (LDA). The tSNE algorithm (Van der Maaten and Hinton 2008; Van der Maaten 2014; Gisbrecht et al. 2015) defines a probability distribution model over pairs of high-dimensional objects

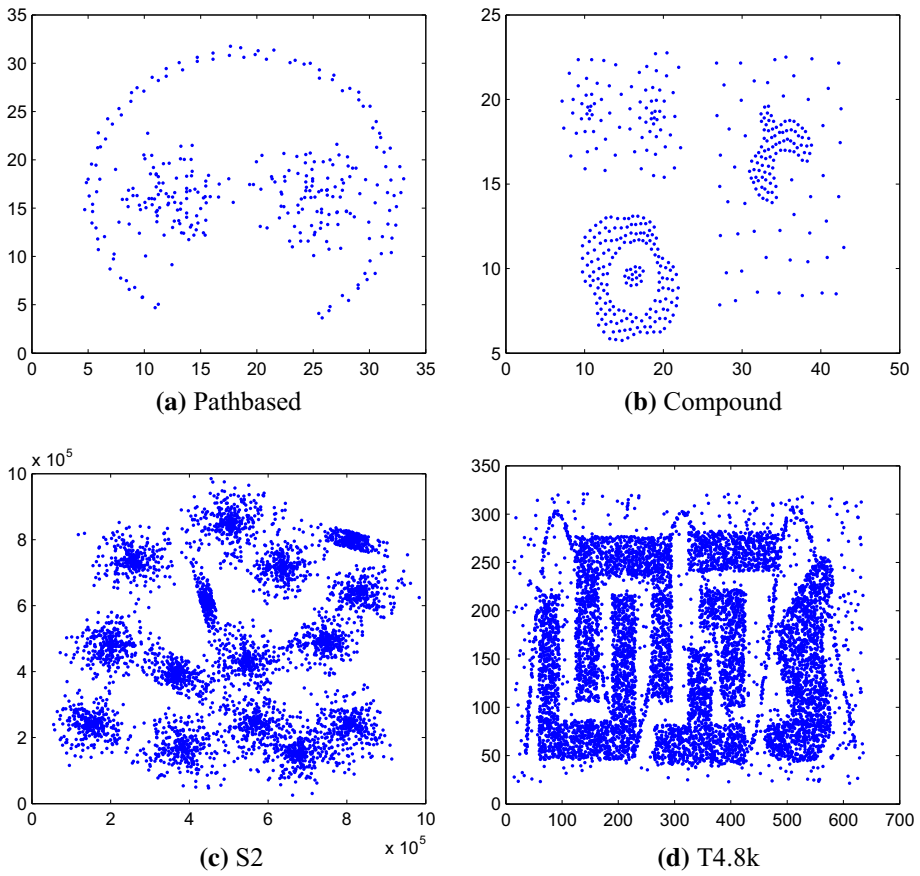


Fig. 1 Four synthetic datasets

in such a way that similar objects have a high probability of being picked, while dissimilar points have an extremely small probability of being picked. Let $X = \{x_1, x_2, \ldots, x_n\}$ denote the data points in the high-dimensional space and $Y = \{y_1, y_2, \ldots, y_n\}$ denote the corresponding embeddings in the low-dimensional space; the similarity between x_i and x_j meets the Gaussian distribution as follows:

$$p_{ij} = \frac{\exp(- \parallel x_i - x_j \parallel^2 / 2\delta^2)}{\sum_{k \neq l} \exp(- \parallel x_k - x_l \parallel^2 / 2\delta^2)} \tag{9}$$

where δ represents the variance of the Gauss distribution.

The tSNE also constructs a similar probability distribution over the points in the low-dimensional map, and the similarity between y_i and y_j is modeled by a student t-distribution with one degree of freedom as follows:

$$q_{ij} = \frac{(1+ \parallel y_i - y_j \parallel^2)^{-1}}{\sum_{k \neq l} (1+ \parallel y_k - y_l \parallel^2)^{-1}} \tag{10}$$

and it minimizes the Kullback–Leibler divergence between the two distributions with respect to the locations of the points in the map. The tSNE gets the optimal low-dimensional representation $C(Y)$ by minimizing the following cost function using the gradient descent method.

$$C(Y) = KL(P \parallel Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{11}$$

In data preprocessing, tSNE can always simplify the high-dimensional complex data into characteristic data with the highest principal component. $C(Y)$ will represent several feature vectors that map from a high-dimensional space to a low-dimensional space, and it is also regarded as simplified data for clustering analysis.

3 The proposed algorithm

The proposed QCC-PE gives another option for the local density computation so that we can find the optimized parameter according to the principle of potential energy. The new method of computing local density is proposed taking all the points scattered in space into account on the basis of QCC that weakens the weight of the first k point to the result, thereby reducing the consequences of local differences and taking more attention to the connection between the global points. The formula is as follows:

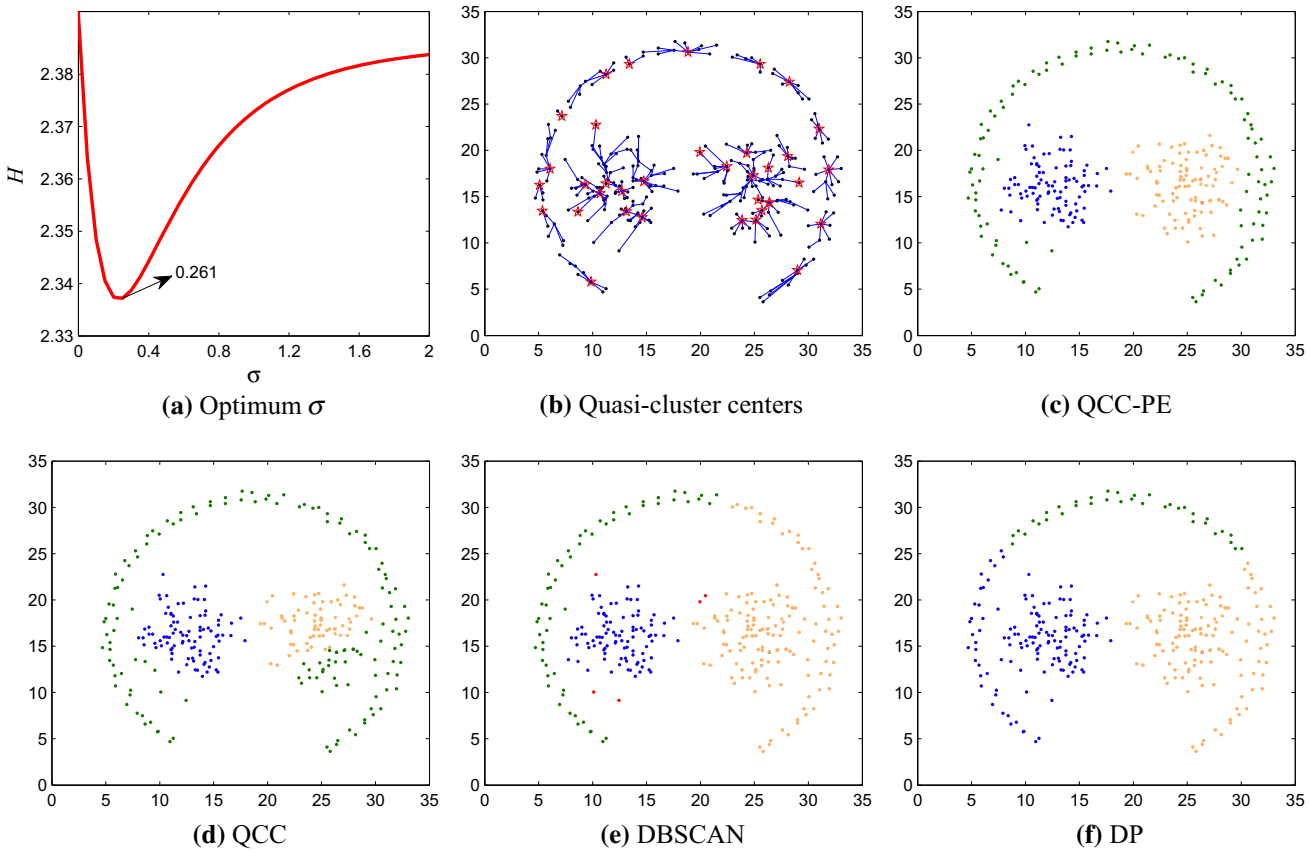


Fig. 2 Clustering result of QCC-PE, QCC, DBSCAN, and DP on dataset Pathbased

$$\rho_i = \sum_{j=1}^{n} \frac{1/Dist_k(x_i)}{d_{ij}}, \; j \neq i \tag{12}$$

In Eq. (7), if $K(x)$ is chosen as reciprocal function, namely the function value $K(x)$ is equal to the reciprocal of the argument x, and let $m_j = 1$, then the potential φ_i of each point is calculated as Eq. (13):

$$\varphi_i = \sum_{j=1}^{n} \frac{\sigma_i}{d_{ij}}, \; j \neq i \tag{13}$$

Let

$$\sigma_i = 1/Dist_k(x_i), \tag{14}$$

then Eq. (13) is same as Eq. (12), and the potential of data field is same as the local density of each point in QCC. In this case, $1/Dist_k(x_i)$ can be calculated in the same way that is used to optimize the impact factor of data field, namely σ. Optimal value of the parameter k can be easily obtained if σ is solved. As for σ, the optimization problem becomes a minimization problem of the single variable function $H(\sigma)$, that is, $\min H(\sigma)$. In view of the time cost of the node topological potential in the iterative computation, the optimization interval can be approximately estimated, and the optimization value is accurately searched. In space, the entropy of potential energy can be used to describe the degree of density and sparsity of points. Data points with larger potential entropy are located in the dense region and vice versa. So, through the behavior of embarking on the potential entropy, the optimal parameter can be extracted from raw datasets. The following algorithm is a summary of the proposed QCC-PE.

Algorithm 2 : QCC-PE algorithm

Input: The dataset (X), the minimum similarity between cluster (α)
Output: The final cluster results $C = \{c_1, c_2, \ldots, c_M\}$
1: Calculate the optimal value of σ with potential entropy according to Eq. (13)
2: Obtain optimal value of the parameter k according to Eq. (14)
3: Calculate the density of each point x_i according to Eq. (12)
4: Calculate the $KNN(x_i)$ and $RKNN(x_i)$ according to Eq. (3–4)
5: Search quasi-clustering center according to Eq. (5)
6: Calculate the similarity matrix $sim(c_i, c_j)$ between the clusters according to Eq. (6)
7: Merge all initial clusters those $sim(c_i, c_j) > \alpha$
8: Return C

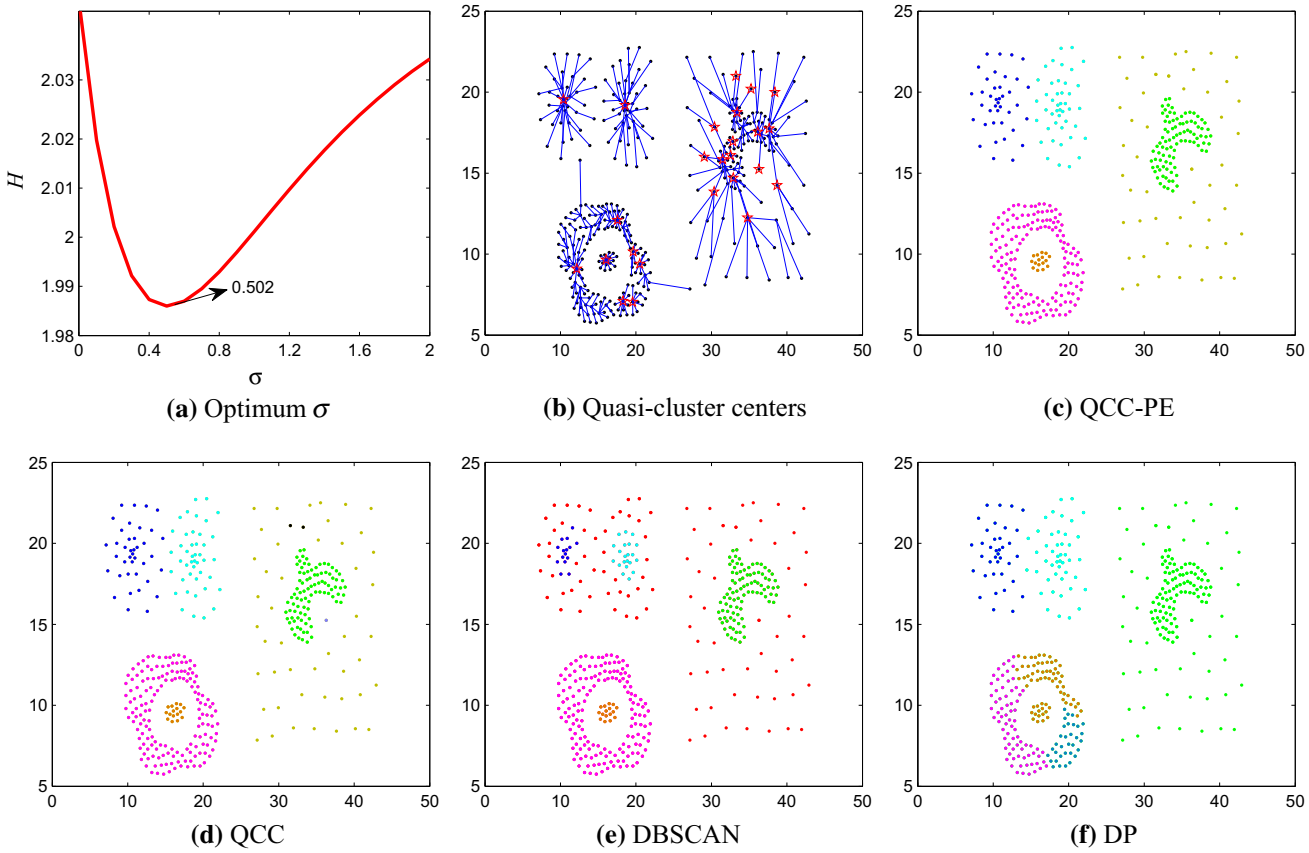


Fig. 3 Clustering result of QCC-PE, QCC, DBSCAN, and DP on dataset Compound

With the aim of clustering high-dimensional datasets, we bring forward QCC-PE-tSNE based on the proposed QCC-PE, which embeds high-dimensional data into low-dimensional space by using t-distributed stochastic neighbor embedding (tSNE) in the first step of clustering. After reducing dimensionality of tSNE, complex datasets containing multiple attributes will be represented only by a small number of attributes that best characterize the potential characteristics of the data. The following algorithm is a summary of the proposed QCC-PE-tSNE.

Algorithm 3 : QCC-PE-tSNE algorithm

Input: The dataset (X), the minimum similarity between cluster (α)
Output: The final cluster results $C = \{c_1, c_2, \ldots, c_M\}$
1: Calculate joint probability distribution in high-dimensional space according to Eq. (9)
2: Calculate joint probability distribution in low-dimensional space according to Eq. (10)
3: Achieve the optimal low-dimensional representation $C(Y)$ optimizing Eq. (11)
4: Calculate the optimal value of σ with potential entropy according to Eq. (13)
5: Obtain optimal value of the parameter k according to Eq. (14)
6: Calculate the density of each point x_i according to Eq. (12)
7: Calculate the $KNN(x_i)$ and $RKNN(x_i)$ according to Eqs. (3–4)
8: Search quasi-clustering center according to Eq. (5)
9: Calculate the similarity matrix $sim(c_i, c_j)$ between the clusters according to Eq. (6)
10: Merge all initial clusters those $sim(c_i, c_j) > \alpha$
11: Return C

4 Experiments and results

The experiments contain three main parts: experiments on synthetic datasets, experiments on Olivetti Face Database, and experiments on real-world datasets. The first part is conducted to verify the validity and power of the proposed QCC-PE algorithm in two-dimensional space, while the second part is conducted to further verify the performance improvement of QCC-PE than the original algorithm. The third part is conducted to prove the superiority of the proposed QCC-PE-tSNE algorithm in high-dimensional space.

4.1 Experiments on synthetic datasets

In order to verify the validity and power of the proposed QCC-PE clustering algorithm, we first compare QCC-PE with QCC, DBSCAN, and DP on synthetic datasets. We conduct experiments on four representative datasets which

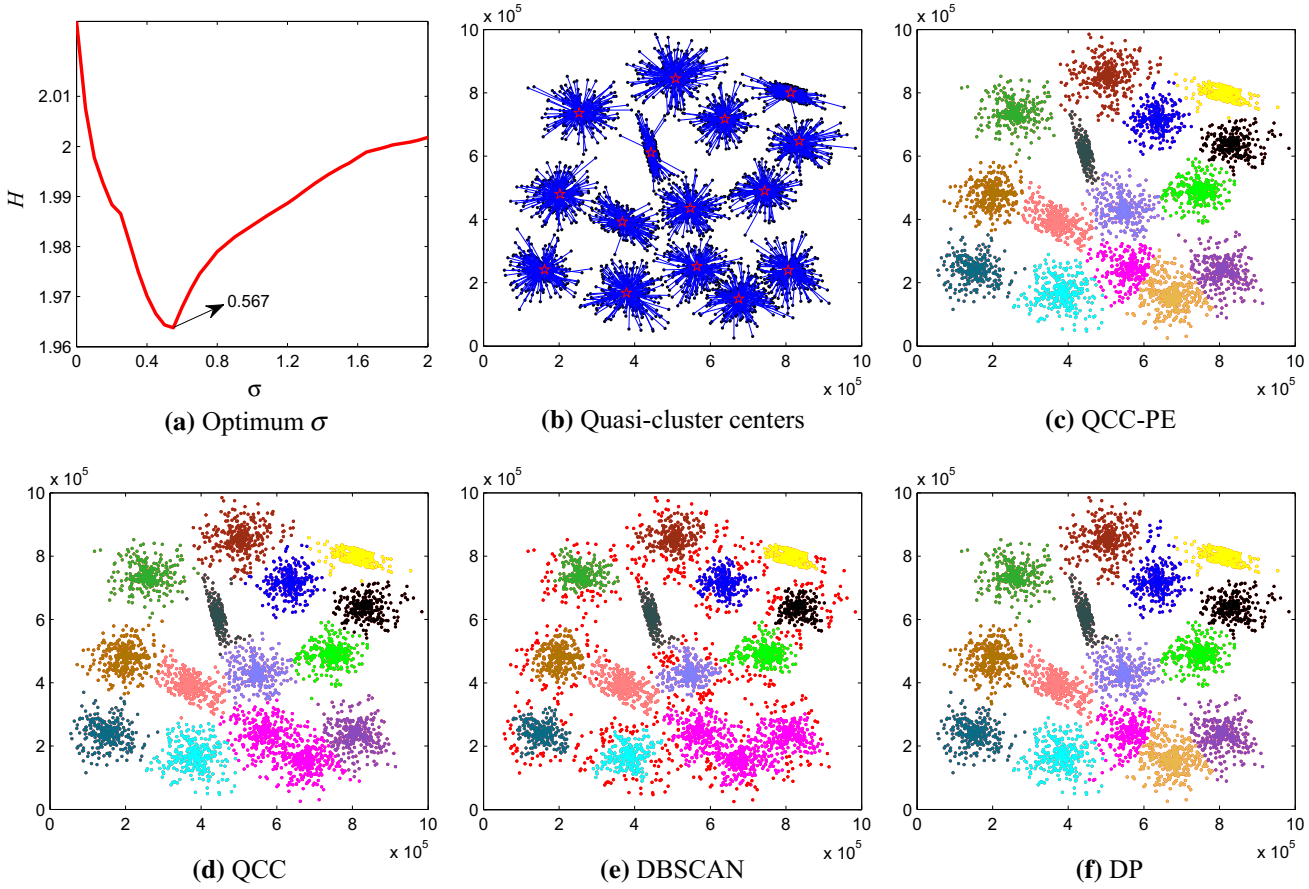


Fig. 4 Clustering result of QCC-PE, QCC, DBSCAN, and DP on dataset S2

are illustrated in Fig. 1. The dataset Pathbased (Chang and Yeung 2008) is generated by two nearly symmetric spherical classes and a manifold class that surround them. The dataset Compound (Zahn 1971) has different sizes, shapes, and densities in terms of the six manifold classes. The dataset S2 (Du et al. 2016) is composed of 15 spherical classes distributed in space. The dataset T4.8k (Cassisi et al. 2013) consists of six high-density manifold classes with some noise points.

Figures 2, 3, 4, and 5 show the clustering results of QCC-PE on the dataset Pathbased, the dataset Compound, the dataset S2, the dataset T4.8k, respectively. For the purposes of comparison, the original QCC algorithm, DBSCAN algorithm, and DP algorithm are used in experiments simultaneously, which are also based on the local density.

Figure 2 shows the clustering results of each approach on the dataset Pathbased. For QCC-PE, the value of optimum parameter σ is 0.261, which is obtained by adopting the method of potential entropy. We also set many other parameters manually relied on the experience after many experiments, such as QCC-PE ($\alpha = 0.4$), QCC ($k = 6$, $\alpha = 0.4$), DBSCAN ($Eps = 1.8$, $MinPts = 4$), DP (right number of clusters: $rn = 3$). From Fig. 2, we can see that QCC-PE produces lots of quasi-clustering centers. However, after merging, the final number of clusters plummeted to three and the clustering result is consistent with human eye observation to a certain extent. QCC is also successfully clustered into three categories, but it is obvious that there are much more errors on clustering points. Although DBSCAN is also clustered into three classes, it divides a large amount of points that should have belonged to one class into two classes, and at the same time, a large number of points which are not supposed to belong to a class are classified as same class. In addition, some normal points are regarded as noise points via this method. DP results in bad clustering results on the premise that the correct number of clusters is set.

Figure 3 shows the clustering results of each approach on the dataset Compound. For QCC-PE, the value of the optimum parameter σ is 0.502, which is obtained by adopting the method of potential entropy. We also set many other parameters manually relied on the experience after many experiments, such as QCC-PE ($\alpha = 1.4$), QCC ($k = 9, \alpha = 1.4$), DBSCAN ($Eps = 1$, $MinPts = 3$), DP (right number of clusters: $rn = 6$). From Fig. 3, we can see that the quasi-clustering centers of QCC-PE algorithm are not too much. After further merging, the final number of clusters is reduced to six and there was no mistake at all. The clustering number

of QCC is eight. DBSCAN clusters data into five categories and treats many normal data as noise points. Although the number of DP clusters is set into right six classes, the result of clustering in manifold dataset is still unsatisfactory.

Figure 4 shows the clustering results of each approach on the dataset S2. For QCC-PE, the value of the optimum parameter σ is 0.567, which is obtained by adopting the method of potential entropy. We also set many other parameters manually relied on the experience after many experiments, such as QCC-PE ($\alpha = 5$), QCC ($k = 60, \alpha = 5$), DBSCAN ($Eps = 14000, MinPts = 10$), DP (right number of clusters: $rn = 15$). From Fig. 4, we can see that the quasi-clustering center number of QCC-PE algorithm is 15, which is equal to the actual number of clusters. After merging, the final number of clusters did not change. The results show that QCC-PE and DP perform better than the other two algorithms whose number of clusters is incorrect.

Figure 5 shows the clustering results of each approach on the dataset T4.8k. For QCC-PE, the value of the optimum parameter σ is 0.448, which is obtained by adopting the method of potential entropy. We also set many other parameters manually relied on the experience after many experiments, such as QCC-PE ($\alpha = 6$), QCC ($k = 80, \alpha = 6$), DBSCAN ($Eps = 5, MinPts = 12$), DP (right number of clusters: $rn = 6$). From Fig. 5, we can see that all of four algorithms obtain the right number of clusters, that is, six clusters. What needs to be pointed out here is that QCC-PE need not to manually set parameter k but automatically obtain that by the optimization algorithm. Even in this case, its quality of clustering results is better than or equal to that of QCC. DBSCAN can detect out the noise points, but some normal points are treated as noise points. By DP, there are four classes that are incorrectly clustered, and only two classes can be considered correct.

From the analysis mentioned above and the contrast of the pictures, we can draw the following conclusions: Although DP does not need to set parameters and be implemented easily, the number of clusters needs to be determined subjectively according to the decision graph. For spherical datasets, DP can achieve excellent clustering results. But the effect of DP is unsatisfactory when dealing with manifold datasets. DBSCAN has a strong ability to handle datasets with outliers or noises. However, it does not perform well on manifold datasets with various densities among clusters. Furthermore, it also is not good in the complex manifold class having deceptive characteristics. QCC, by contrast, outperforms generally the previous two algorithms and can be better applied to arbitrary shape cluster, and it requires two param-

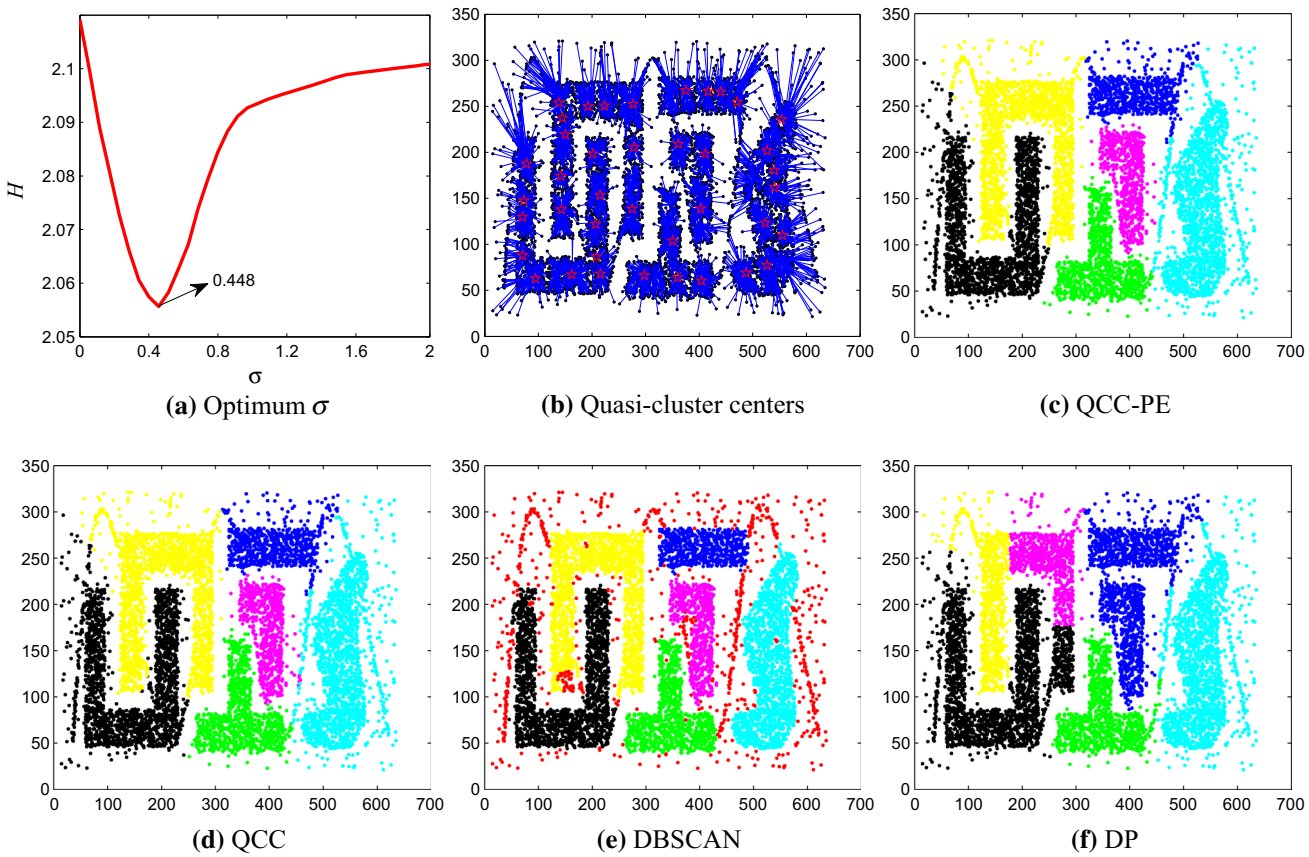


Fig. 5 Clustering result of QCC-PE, QCC, DBSCAN, and DP on dataset T4.8k

Fig. 6 Olivetti Face Database

eters to be manually set yet. Therefore, the superiority of this algorithm is still not fully embodied subject to the restriction of the choice of optimal parameter. QCC-PE can make up for this deficiency, and it is very effective in finding clusters of arbitrary shape, density, distribution, and number. Obviously, the effect of QCC-PE algorithm is the best in all these algorithms as a whole.

4.2 Experiments on Olivetti Face Database

Olivetti Face Database contains a series of face pictures, which is the most important test data resources in the field of machine learning and artificial intelligence. The database has 400 existing pictures from 40 people with each of ten pictures. The size of each picture is 92×112. Like Huang et al. (2017), we also selected the 100 same pictures of them as the sample of the experiment which includes ten classes, the raw images as shown in Fig. 6.

Figure 7 shows the clustering results of the QCC-PE, QCC, DBSCAN, and DP algorithms on Olivetti Face Database, respectively. The gray image in the picture represents the noise points detected by various algorithms. We can find through comparative research that seven of the ten classes are successfully clustered by the QCC-PE. In the rest three classes, there are two classes that both have an image that is considered as noise, and there is one class that is identified into two classes. QCC as well as QCC-PE has the same class of identification, and it is eleven classes. In general speaking, it treats more image as noise than QCC-PE. DBSCAN only detects eight clusters, and only three of eight classes are absolutely right. There are a lot of noise spots that are wrongly detected. For DP, we select ten points as cluster centers through decision graph, but only one class conforms to the facts.

Although it is revealed here that QCC-PE is the most prominent followed by QCC, DBSCAN, and DP in turn, the Olivetti Face Database should actually be ten classes rather than 11 classes that are clustered by QCC-PE. QCC-PE, to some extent, has improvement over QCC, but still remains a lot of room to continue to enhance.

4.3 Experiments on real-world datasets

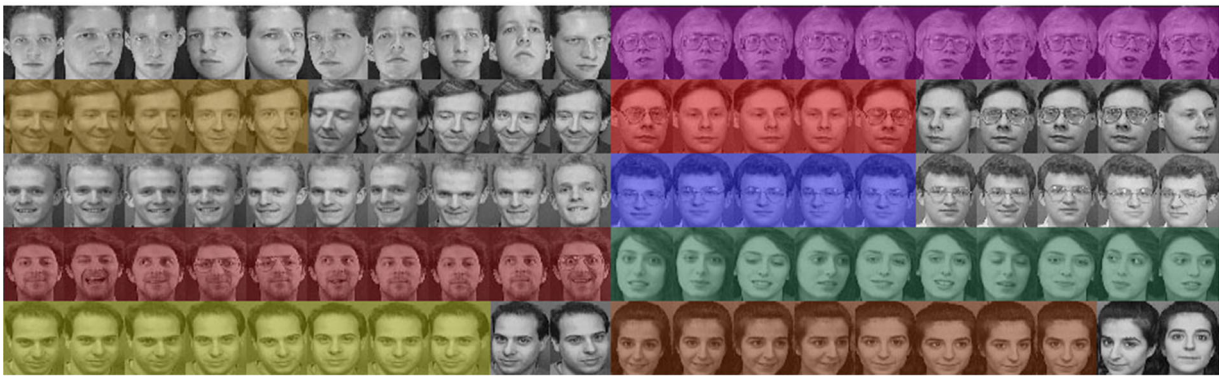
The UCI database is a resource library for the empirical analysis of machine learning algorithms and is also the most impactful means of testing the performance of classification algorithms and clustering algorithm. The material used in the experiment is eight typical real-world datasets come from the UCI repository, which include Iris, Seeds, Heart, Image segmentation, Waveform, Parkinsons, Sonar, and Libras movement. The details are shown in Table 1.

To quantitatively evaluate the performance of clustering algorithms of QCC-PE-tSNE, QCC-PE, QCC, DBSCAN, and DP, the following four indexes are introduced (Xie et al. 2016; Ding et al. 2017, 2018): Adjusted Rand Index (ARI), Adjusted Mutual Information (AMI), Accuracy (Acc), and F_1 Score (F_1). The range of ARI is $[-1, 1]$, and the range of AMI, Acc, and F_1 is $[0, 1]$. The closer the value of the four indexes is to 1, the closer the clustering result is to the real class grouping.

Table 2 shows the ARI, AMI, Acc, and F_1 of various clustering algorithms on real-world datasets. The symbol "–" means there is no value for that entry, and the best results are portrayed in boldface. From Table 2, we can reach that QCC-PE performs best in relatively low-dimensional spaces such as datasets Iris and Seeds. With the growth of the number of dimensions, the performance of QCC-PE becomes worse, and the clustering results cannot even be obtained on the 60 dimensional spaces such as Sonar dataset. QCC-PE-tSNE gives an overall highest performance compared with others in relatively high-dimensional spaces such as datasets Image segmentation, Waveform, Parkinsons, Sonar, and Libras movement. DBSCAN own an best performance on the Heart dataset, in which it is only a little better than QCC-PE-tSNE in terms of F_1.

(a) QCC-PE

(b) QCC



(c) DBSCAN

(d) DP

Fig. 7 Clustering results of QCC-PE, QCC, DBSCAN, and DP on Olivetti Face Database

Table 1 Details of real-world datasets

Datasets	Number	Dimension	Cluster
Iris	150	4	3
Seeds	210	7	3
Heart	270	13	2
Image segmentation	2310	19	7
Waveform	5000	21	3
Parkinsons	195	23	2
Sonar	208	60	2
Libras movement	360	91	15

The reasons about experiment results could be analyzed as follows: Compared to QCC-PE-tSNE, QCC-PE does not lose any essential information in the relatively low-dimensional space, which causes the QCC-PE to grasp the original features of the data, but QCC-PE-tSNE cannot. As the dimensions of datasets increase, QCC-PE is difficult or unable to deal with extremely complex datasets directly limited by the curse of dimensionality, whereas QCC-PE-tSNE not only avoids this terrible situation, but also pulls useful information as much as possible which leads to advantages gradually appearing in high-dimensional space. DBSCAN has acceptable stability and is especially suitable for small- and medium-size datasets. DP fails to cluster on datasets of Sonar and Heart, due to the fact that only one cluster center was clearly distinguished by the decision graph and second cluster center could not be found in those databases.

5 Conclusions

This paper proposes a QCC-PE clustering algorithm, which focuses on the global relationship between all points on the basis of QCC and weakens the weight of k nearest

Table 2 Performance comparison of QCC-PE-tSNE, QCC-PE, QCC, DBSCAN, and DP on real-world datasets

Algorithms	Iris				Seeds			
	ARI	AMI	Acc	F_1	ARI	AMI	Acc	F_1
QCC-PE-tSNE	0.713	0.725	0.865	0.925	0.605	0.664	0.828	0.889
QCC-PE	**0.938**	**0.954**	**0.977**	**0.963**	**0.934**	**0.922**	**0.933**	**0.909**
QCC	0.907	0.903	0.921	0.914	0.852	0.885	0.915	0.817
DBSCAN	0.732	0.775	0.893	0.805	0.686	0.644	0.881	0.871
DP	0.720	0.767	0.887	0.824	0.734	0.717	0.900	0.807
Algorithms	Heart				Image segmentation			
	ARI	AMI	Acc	F_1	ARI	AMI	Acc	F_1
QCC-PE-tSNE	0.667	0.626	0.815	0.797	**0.582**	**0.686**	**0.733**	**0.830**
QCC-PE	0.591	0.534	0.626	0.754	0.372	0.412	0.422	0.688
QCC	0.584	0.568	0.624	0.659	–	–	–	–
DBSCAN	**0.724**	**0.682**	**0.834**	**0.803**	0.227	0.435	0.441	0.534
DP	–	–	–	–	0.550	0.651	0.684	0.626
Algorithms	Waveform				Parkinsons			
	ARI	AMI	Acc	F_1	ARI	AMI	Acc	F_1
QCC-PE-tSNE	**0.617**	0.495	**0.864**	**0.802**	**0.531**	**0.453**	**0.863**	**0.788**
QCC-PE	0.241	**0.498**	0.674	0.593	0.012	0.118	0.323	0.547
QCC	–	–	–	–	–	–	–	–
DBSCAN	–	–	–	–	0.225	0.205	0.672	0.468
DP	0.268	0.318	0.568	0.487	0.027	0.201	0.610	0.509
Algorithms	Sonar				Libras movement			
	ARI	AMI	Acc	F_1	ARI	AMI	Acc	F_1
QCC-PE-tSNE	**0.425**	**0.564**	**0.668**	**0.774**	**0.563**	**0.512**	**0.587**	**0.751**
QCC-PE	–	–	–	–	–	–	–	–
QCC	–	–	–	–	–	–	–	–
DBSCAN	0.197	0.242	0.578	0.433	0.154	0.408	0.350	0.425
DP	–	–	–	–	0.214	0.390	0.361	0.496

neighbor in computing local density. To this end, a new method for calculating density is designed. QCC-PE can automatically determine optimal parameter k using potential entropy. Dedicated to applying to high-dimensional datasets, we incorporate the idea of dimensionality reduction which is based on t-distributed stochastic neighbor embedding and further propose QCC-PE-tSNE to improve QCC-PE. The experimental results on considerable amount of datasets demonstrate that the proposed algorithms achieve gratifying results and exhibit a promising performance advantage.

Acknowledgements The authors would like to express their sincere thanks to the editor and the anonymous reviewers for their valuable and insightful comments. This work is supported by the National Natural Science Foundation of China (NSFC) (No. 61170110) and Zhejiang Provincial Natural Science Foundation of China (No. LY13F020043).

Compliance with ethical standards

Conflict of interest The authors declare that there are no conflicts of interest regarding the publication of this paper.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

Agrawal R, Gehrke J, Gunopulos D, Raghavan P (1998) Automatic subspace clustering of high dimensional data for data mining applications. In: Proceedings 1998 ACM sigmod international conference on management of data, vol 27, pp 94–105

Ankerst M, Breunig MM, Kriegel HP, Sander J (1999) OPTICS: ordering points to identify the clustering structure. In: Proceedings on 1999 ACM sigmod international conference on management of data, vol 28, pp 49–60

Barbieri F, Mazzoni A, Logothetis NK, Panzeri S, Brunel N (2014) Stimulus dependence of local field potential spectra: experiment versus theory. J Neurosci 34(44):14589–14605

Carpenter GA, Grossberg S (1987) A massively parallel architecture for a self-organizing neural pattern recognition machine. Comput Vis Gr Image Process 37(1):54–115

Carpenter GA, Grossberg S (1990) ART 3: hierarchical search using chemical transmitters in self-organizing pattern recognition architectures. Neural Netw 3(2):129–152

Cassisi C, Ferro A, Giugno R, Pigola G, Pulvirenti A (2013) Enhancing density-based clustering: parameter reduction and outlier detection. Inf Syst 38(3):317–330

Chang H, Yeung DY (2008) Robust path-based spectral clustering. Pattern Recogn 41(1):191–203

Comaniciu D, Meer P (2002) Mean shift: a robust approach toward feature space analysis. IEEE Trans Pattern Anal Mach Intell 24(5):603–619

Ding SF, Du MJ, Sun TF, Xu X, Xue Y (2017) An entropy-based density peaks clustering algorithm for mixed type data employing fuzzy neighborhood. Knowl-Based Syst 133:294–313

Ding SF, Jia HJ, Du MJ, Xue Y (2018) A semi-supervised approximate spectral clustering algorithm based on HMRF model. Inf Sci 429:215–228

Du MJ, Ding SF, Jia HJ (2016) Study on density peaks clustering based on k-nearest neighbors and principal component analysis. Knowl-Based Syst 99:135–145

Dutta M, Mahanta AK, Pujari AK (2005) QROCK: a quick version of the ROCK algorithm for clustering of categorical data. Pattern Recogn Lett 26(15):2364–2373

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 international conference on knowledge discovery and data mining, vol 96, pp 226–231

Fisher DH (1987) Knowledge acquisition via incremental conceptual clustering. Mach Learn 2(2):139–172

Gisbrecht A, Schulz A, Hammer B (2015) Parametric nonlinear dimensionality reduction using kernel t-SNE. Neurocomputing 147:71–82

Guha S, Rastogi R, Shim K (1999) ROCK: a robust clustering algorithm for categorical attributes. In: Proceedings of the 15th international conference on data engineering, pp 512–521

Horn D, Gottlieb A (2002) Algorithm for data clustering in pattern recognition problems based on quantum mechanics. Phys Rev Lett 88(1):1–4

Huang JL, Zhu QS, Yang LJ, Cheng DD, Wu QW (2017) QCC: a novel clustering algorithm based on quasi-cluster centers. Mach Learn 106(3):337–357

Karypis G, Han EH, Kumar V (1999) Chameleon: hierarchical clustering using dynamic modeling. Computer 32(8):68–75

Kohonen T (1998) The self-organizing map. Neurocomputing 21(1):1–6

Kumar KM, Reddy ARM (2016) A fast DBSCAN clustering algorithm by accelerating neighbor searching using groups method. Pattern Recogn 58:39–48

Li YL, Shen Y (2010) An automatic fuzzy c-means algorithm for image segmentation. Soft Comput 14(2):123–128

Liew AW, Yan H (2003) An adaptive spatial fuzzy clustering algorithm for 3-D MR image segmentation. IEEE Trans Med Imaging 22(9):1063–1075

Macqueen J (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, vol 1, pp 281–297

Madan S, Dana KJ (2015) Modified balanced iterative reducing and clustering using hierarchies (m-BIRCH) for visual clustering. Pattern Anal Appl 19:1–18

Mehmood R, Zhang G, Bie R, Dawood H, Ahmad H (2016) Clustering by fast search and find of density peaks via heat diffusion. Neurocomputing 208:210–217

Omran MGH, Engelbrecht AP, Salman A (2007) An overview of clustering methods. Intell Data Anal 11(6):583–605

Park HS, Jun CH (2009) A simple and fast algorithm for k-medoids clustering. Expert Syst Appl 36(2):3336–3341

Rasmussen CE (2000) The infinite gaussian mixture model. Adv Neural Inf Process Syst 12:554–560

Rodriguez A, Laio A (2014) Clustering by fast search and find of density peaks. Science 344(6191):1492–1496

Tomasev N, Radovanovic M, Mladenic D, Lvanovic M (2014) The role of hubness in clustering high-dimensional data. IEEE Trans Knowl Data Eng 26(3):739–751

Van der Maaten LJP (2014) Accelerating t-SNE using tree-based algorithms. J Mach Learn Res 15(1):3221–3245

Van der Maaten LJP, Hinton G (2008) Visualizing data using t-SNE. J Mach Learn Res 9(11):2579–2605

Wang SL, Wang DK, Li CY, Li Y, Ding GY (2016) Clustering by fast search and find of density peaks with data field. Chin J Electron 25(3):397–402

Wang W, Yang J, Muntz RR (1997) STING: a statistical information grid approach to spatial data mining. In: International conference on very large data bases, Inc, pp 186–195

Wu YC (2014) A top-down information theoretic word clustering algorithm for phrase recognition. Inf Sci 275:213–225

Xie J, Gao H, Xie W, Liu X, Grant PW (2016) Robust clustering by detecting density peaks and assigning points based on fuzzy weighted k-nearest neighbors. Inf Sci 354:19–40

Xu DK, Tian YJ (2015) A comprehensive survey of clustering algorithms. Ann Data Sci 2(2):165–193

Zahn CT (1971) Graph-theoretical methods for detecting and describing gestalt clusters. IEEE Trans Comput 100(1):68–86

Zang WK, Ren LY, Zhang WQ, Liu XY (2017) Automatic density peaks clustering using DNA genetic algorithm optimized data field and Gaussian process. Int J Pattern Recognit Artif Intell 31(8):1750023

Zhang T, Ramakrishnan R, Livny M (1996) BIRCH: an efficient data clustering method for very large databases. Acm Sigmod Record 25(2):103–114

Zhang T, Ramakrishnan R, Livny M (1997) BIRCH: a new data clustering algorithm and its applications. Data Min Knowl Disc 1(2):141–182