Artificial Intelligence 273 (2019) 37–55

Contents lists available at ScienceDirect

Artificial Intelligence

www.elsevier.com/locate/artint

Clustering ensemble based on sample's stability

Feijiang Li [a], Yuhua Qian [a,b,*], Jieting Wang [a], Chuangyin Dang [c], Liping Jing [d]

[a] Institute of Big Data Science and Industry, Shanxi University, Taiyuan, 030006, Shanxi Province, China
[b] Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, Shanxi University, Taiyuan, 030006, Shanxi Province, China
[c] Department of Manufacture Engineering and Engineering Management, City University of Hong Kong, Hong Kong
[d] Beijing Key Laboratory of Traffic Data Analysis and Mining, Beijing Jiaotong University, Beijing, 100044, China

ARTICLE INFO

Article history:
Received 11 April 2018
Received in revised form 19 November 2018
Accepted 17 December 2018
Available online 14 February 2019

Keywords:
Clustering ensemble
Clustering analysis
Sample's stability
Ensemble learning

ABSTRACT

The objective of clustering ensemble is to find the underlying structure of data based on a set of clustering results. It has been observed that the samples can change between clusters in different clustering results. This change shows that samples may have different contributions to the detection of the underlying structure. However, the existing clustering ensemble methods treat all sample equally. To tackle this deficiency, we introduce the stability of a sample to quantify its contribution and present a methodology to determine this stability. We propose two formulas accord with this methodology to calculate sample's stability. Then, we develop a clustering ensemble algorithm based on the sample's stability. With either formula, this algorithm divides a data set into two classes: cluster core and cluster halo. With the core and halo, the proposed algorithm then discovers a clear structure using the samples in the cluster core and assigns samples in the cluster halo to the clear structure gradually. The experiments on eight synthetic data sets illustrate how the proposed algorithm works. This algorithm statistically outperforms twelve state-of-the-art clustering ensemble algorithms on ten real data sets from UCI and six document data sets. The experimental analysis on the case of image segmentation shows that cluster cores discovered by the stability are rational.

1. Introduction

Data clustering [1] is one of the most important fields in machine learning. It tries to discover the underlying structure of a data set. In general, the structure means that similar samples are assigned to the same cluster while dissimilar samples are assigned to different clusters. The lack of prior knowledge makes clustering analysis remain a very challenging problem though many clustering algorithms have been proposed in the literature. It has been accepted that a single clustering algorithm can not handle all types of data distribution effectively. Each clustering algorithm has its own strategy to discover a structure from a data set. Different algorithms or different parameters for an algorithm may lead to different clustering results. It is hard to judge which structure best matches the real distribution without supervised information. Therefore, selecting a suitable algorithm is a difficult task. To avoid this task, many pieces of research focus on integrating multiple clustering results, which is known as clustering ensemble [2]. The clustering ensemble can significantly improve the robust-

* Corresponding author at: Institute of Big Data Science and Industry, Shanxi University, Taiyuan, 030006, Shanxi Province, China.
E-mail addresses: feijiangli@email.sxu.edu.cn (F. Li), jinchengqyh@126.com (Y. Qian), jietingwang@email.sxu.edu.cn (J. Wang), mecdang@cityu.edu.hk (C. Dang), lpjing@bjtu.edu.cn (L. Jing).

https://doi.org/10.1016/j.artint.2018.12.007
0004-3702/

ness, stability, and quality of a clustering solution when compared with a single clustering algorithm. Clustering ensemble technique has been effectively utilized to handle many clustering tasks, such as categorical data [3,4], high dimensional data [5], noisy data [6], temporal data [7], feature selection [8], etc.

The clustering ensemble problem was first introduced by Strehl and Ghosh [9]. In [9], the clustering ensemble problem is described as *combining multiple clustering results of a set of objects without accessing the original features*. A clustering ensemble method should be able to combine multiple clustering results into a consistent partition which is most similar to the base clusterings without invoking the original data set. Some researchers use both the multiple clustering results and the original features as inputs to further improve clustering performance [10,11]. Clustering ensemble without accessing the original features could be applied in more fields than a method that also takes the original features as input. The fields mainly contain scenarios where the original features are unavailable, such as distributed sources of data or attributes and undisclosed data due to some secrecy reasons [12,13]. In addition, most of the clustering ensemble algorithms without invoking the original data set can be expanded to a version that invokes the original data set.

In this paper, we focus on the traditional clustering ensemble problem which does not access the original features. With this requirement, many clustering ensemble methods have been proposed. These methods utilize a lot of techniques, which include but not limited to: clustering techniques [14,3,15–17,1,6], graph technologies [18,9,19] and optimization methods [20–24]. In addition, weighted clustering ensemble and selective clustering ensemble are two approaches to further improve the performance of a clustering ensemble algorithm. These two approaches increase the effect of the base partitions which are beneficial for the following integration. Research about weighted clustering ensemble (WCE) and selective clustering ensemble (SCE) mainly focuses on exploring the characteristics of beneficial base clustering results. The characteristics mainly involve two issues, which are diversity and accuracy of base partitions. Prior research about WCE and SCE explored whether diversity or accuracy is the determining factor in the selection of base clusterings. Recently, more researchers tend to combine diversity and accuracy in the selection of a subset of base partitions.

Although the above works have obtained good performance in solving a clustering ensemble problem, there is still much room for improving the ensemble quality. The existing ensemble algorithms treat every sample equally in the construction of the underlying structure. However, given a set of clustering results, samples may have different contributions to the detection of the underlying structure. In clustering analysis, a cluster has a cluster core, which indicates robust assignment, and a cluster halo, which could be considered as noise [25]. In [26], the authors introduced profiles and motifs. Given a set of clustering results, profiles count the relative frequency that a sample occurs in a cluster and motifs are subsets of samples that have high consensus in the same cluster. However, to discover motifs, one should handle cluster correspondence discovering and cluster fusion problem, which have not solved very well [27]. In addition, it has been pointed out in the literature [28–31] that diversity and accuracy of base clusterings are important for the ensemble performance. For a set of base clustering results, diversity mainly comes from different decisions on the samples in the cluster halo, and accuracy is guaranteed by the consistent assignment of the core samples. An ensemble algorithm which differentiates between samples in a cluster core and samples in a cluster halo may generate a better solution. According to the above discussions, an interesting question arises: how can the cluster core and cluster halo be effectively discovered in a clustering ensemble problem?

To answer the question, this paper introduces sample's stability in a clustering ensemble problem. The sample's stability reflects its tendency of changing its cluster in the base clustering results. Under this stability, a cluster core should consist of the samples with high stability, whereas a cluster halo should contain the samples with low stability. With such cluster core and halo, different strategies should be adopted for handling samples in the core and halo.

The samples in a cluster core have consistent assignments from most of the base clusterings. These samples represent a clear structure which is easy to be discovered. The strategy for the cluster core samples focuses on discovering a pre-structure for guidance. For the samples in the cluster halo, there are many disagreements in the base clusterings. Therefore it is hard to generate a consistent assignment for a sample in the halo based only on the base clusterings. The discovered pre-structure will offer instructive information to the assignments of the halo samples. Now the strategy for the cluster halo samples just needs to correctly assign halo samples into the pre-structure. The assignment of a sample in the halo based on the pre-structure may be more accurate than before. Taking into account the above consideration, this paper proposes a novel clustering ensemble algorithm that integrates base clusterings from cluster core to cluster halo.

Briefly, the contributions of the paper are as follows:

- A methodology to determine the sample's stability in the clustering ensemble problem is proposed. Two versions of formulas that accord with this methodology are developed.
- A novel clustering ensemble method based on the sample's stability is proposed, which differentiates between stable samples and unstable samples. This algorithm discovers a structure based on stable samples (known as cluster core) and then gradually assigns unstable samples (known as cluster halo) to the pre-discovered structure.
- The rationality of the sample's stability that is measured by the proposed method is verified by experiments on the case of image segmentation. Eight synthetic data sets are used to show the working mechanism of the proposed clustering ensemble algorithm. In addition, experiments on benchmark data sets are conducted to show the effectiveness of the proposed algorithm.

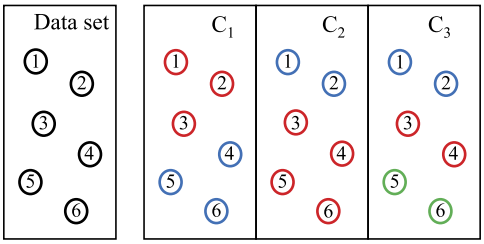

Fig. 1. An example of base clusterings.

The rest of the paper is organized as follows. The clustering ensemble problem is reviewed in section 2. In section 3, we introduce a methodology to measure the sample's stability and propose two formulas according to the methodology. In section 4, the proposed clustering ensemble method based on the sample's stability is described in detail. In section 5, experiments are conducted to show how the algorithm works and the effectiveness of the proposed algorithm. Finally, section 6 concludes the paper.

2. Clustering ensemble

In general, research about the clustering ensemble problem mainly includes three aspects: ensemble generation, ensemble selection, and ensemble integration. Suppose $X = \{x_1, x_2, \ldots, x_n\}$ is a data set with n samples. After clustering with a number of different clustering methods, a set of clustering results, $\Pi = \{C_1, C_2, \ldots, C_L\}$, will be obtained, where L is the ensemble size and indicates the number of clusterings. $C_l(x_i)$ indicates the label of x_i induced by clustering result C_l. The objective of clustering ensemble is to find a new clustering result C^* which is similar to every element in Π. In general, the ensemble selection part and ensemble integration part are solely based on a set of clustering results without invoking the original data set.

The first task in clustering ensemble is to obtain a set of base clustering results Π. Fig. 1 gives a simple example of three clustering results $\Pi = \{C_1, C_2, C_3\}$ on 5 samples.

It is well known that diversity and accuracy of the base clusterings are two key factors to the performance of ensemble techniques. To generate a more diverse and acceptable accurate ensemble set, the following strategies were suggested in the literature [32,5,31,13,33,6]:

- Different parameter settings. Base clustering results can be generated by utilizing a clustering algorithm with randomness, such as the initial centers of the k-means type of algorithms. Another important parameter is the number of clusters. It was suggested in the literature that the cluster number should be set larger than the expected cluster number [34,35,31].
- Different clustering algorithms. Each clustering algorithm has its own specific view on how to discover the underlying structure of a data set. Different clustering algorithms often generate different results. Therefore multiple clustering algorithms can be used to generate diverse base clustering results.
- Different representations of features. The representations of features mainly consist of two forms, one is data projection and the other is a subset of features. For a multi-dimensional data set, both forms of representation of features try to describe a data set from different views. Thus a set of diverse clustering results will be obtained when multiple representations of the data features are utilized.
- Weak clusterings. It has been illustrated that integrating multiple weak clusterings can generate a good ensemble solution [13]. A weak clustering is slightly better than a random partition, which can be obtained by running a clustering algorithm on a random one-dimensional projection of the data set or through splitting the data set by random hyperplanes. A set of base weak clusterings should contain higher diversity.

Inspired by the success of feature selection technique [36,37] in improving the performance of a machine learning algorithm, ensemble selection technique [38] is proposed in clustering ensemble to improve the quality of the obtained clustering results set. An ensemble selection approach selects a subset of base clustering results based on a pre-defined principle which is deemed to be beneficial for the following integration step. Research about the principles for ensemble selection mainly refers to three issues, which are listed as follows:

- Ensemble selection based on diversity. The diversity of a clustering result is generally measured by its average dissimilarity with the other clustering results in the ensemble. Diversity principle is based on a fact that low diversity limits the improvement of the ensemble performance [38]. Premier selective clustering ensemble methods mainly utilize diversity to select a subset of clustering results [31,39].
- Ensemble selection based on accuracy. The accuracy of a clustering result is generally measured by its average similarity with the other clustering results in the ensemble. Accuracy is an important element in ensemble selection stage [40]. It has been illustrated in [30] that accuracy has a positive correlation with the ensemble performance.

	C_1	C_2	C_3
x_1	Red	Blue	Blue
x_2	Red	Blue	Blue
x_3	Red	Red	Red
x_4	Blue	Red	Red
x_5	Blue	Red	green
x_6	Blue	Red	green

(a) $n \times L$ matrix

	c_1	c_2	c_3	c_4	c_5	c_6	c_7
x_1	1	0	1	0	1	0	0
x_2	1	0	1	0	1	0	0
x_3	1	0	0	1	0	1	0
x_4	0	1	0	1	0	1	0
x_5	0	1	0	1	0	0	1
x_6	0	1	0	1	0	0	1

(b) $n \times N_C$ matrix

	x_1	x_2	x_3	x_4	x_5	x_6
x_1	1	1	1/3	0	0	0
x_2	1	1	1/3	0	0	0
x_3	1/3	1/3	1	2/3	1/3	1/3
x_4	0	0	2/3	1	2/3	2/3
x_5	0	0	1/3	2/3	1	1
x_6	0	0	1/3	2/3	1	1

(c) $n \times n$ matrix

	c_1	c_2	c_3	c_4	c_5	c_6	c_7
c_1	3	0	2	1	2	1	0
c_2	0	3	0	3	0	1	2
c_3	2	0	2	0	2	0	0
c_4	1	3	0	4	0	2	2
c_5	2	0	2	0	2	0	0
c_6	1	1	0	2	0	2	0
c_7	0	2	0	2	0	0	2

(d) $N_C \times N_C$ matrix

Fig. 2. The four information matrices.

- Ensemble selection methods based on diversity and accuracy. Diversity and accuracy are two opposite objectives that are both important for the ensemble performance. Most of the recently proposed ensemble selection methods consider both of the two objectives [41,29,42–44]. These methods use a metric that combines diversity and accuracy as the selection guideline or design a complex process that takes diversity and accuracy into consideration simultaneously.

Having obtained a set of base clustering results, the last step is to generate an integrated clustering result. Many clustering ensemble methods have been proposed to discover a structure C^* from base clustering results Π. According to the different types of information matrices [45] which an ensemble method relies on, the existing methods can be classified as follows:

- Feature based approach ($n \times L$ matrix as shown in Fig. 2 (a)) [34,20,46,3,27,23,13,47]. The base clustering results set Π is actually an $n \times L$ matrix, where each column indicates a clustering result. Based on the $n \times L$ matrix, three types of methods have been proposed. The first type is finding correspondences between clustering results and transforming the problem to a classifier ensemble problem. The second type is treating clustering ensemble as the problem of clustering categorical data. Moreover, many median partition approaches are also based on this matrix [12,48,49]. These methods try to discover a clustering result which is similar to the base clustering results set based on optimization technique.
- Cluster based approach ($n \times N_C$ matrix as is shown in Fig. 2 (b)) [35,50,4,51]. If we use each column to indicate a cluster, then an $n \times N_C$ matrix will be generated, in which $N_C = k_1 + k_2 + \ldots + k_L$ and k_i is the cluster number of clustering result C_i. This matrix is binary and sparse. To generate a consensus result based on this $n \times N_C$ matrix, a general way is finding the relationships between clusters, enriching the $n \times N_C$ matrix, and utilizing an existing clustering method or a graph partition method to generate the final result.
- Sample co-association based approach ($n \times n$ matrix as shown in Fig. 2 (c)) [14–16,6,52]. Given a set of clustering results $\Pi = \{C_1, C_2, \ldots C_L\}$, the frequency that two samples x_i and x_j appear in the same cluster is calculated by:

$$p_{ij} = \frac{1}{L} \sum_{l=1}^{L} \mathbb{I}\left(C_l(x_i), C_l(x_j)\right), \tag{1}$$

where

$$\mathbb{I}\left(C_l(x_i), C_l(x_j)\right) = \begin{cases} 1, & C_l(x_i) = C_l(x_j); \\ 0, & C_l(x_i) \neq C_l(x_j). \end{cases}$$

All the pairwise frequencies will form an $n \times n$ matrix, which is called co-association matrix. This matrix reflects the relations between each pair of samples. It offers an approximate similarity matrix which can be utilized by the hierarchical type clustering algorithms. Furthermore, this matrix is the basis of many graph based clustering ensemble methods.
- Cluster intersect based approach ($N_C \times N_C$ matrix as shown in Fig. 2 (d)) [9]. Based on Π, the relationships between each pair of clusters are also reachable, such as the common samples. They form an $N_C \times N_C$ matrix. This matrix reflects the relations between clusters in different clusterings and is helpful for finding correspondences between clusters.

The above discussions about the ensemble integration methods almost touch all forms of information matrices that can be constructed by Π. Most of the existing ensemble methods treat all samples as a whole in order to discover the

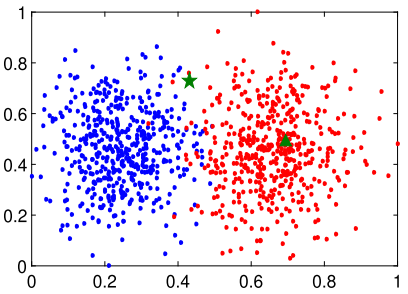

Fig. 3. 2k2d data set and two marked samples.

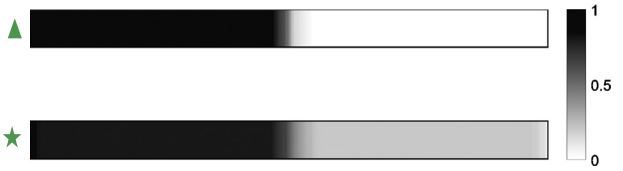

Fig. 4. Relations between the marked sample with others.

underlying structure of a data set. It is natural to conclude that different samples could have different ability to characterize the underlying structure. Thus an ensemble method that uses different strategies to handle samples in cluster core and samples in cluster halo may generate a more effective result. To design such a clustering ensemble algorithm, the cluster core should be found based on Π. To effectively address this issue, in what follows, the sample's stability is introduced to measure the degree that a sample belongs to the cluster core.

3. Sample's stability in clustering ensemble

Given a set of base clustering results, some samples remain consistently in one group, while others frequently change from one group to another. This phenomenon can be characterized by the tendency that a sample changes its group. This tendency is helpful for many tasks in clustering ensemble, such as measuring the quality of the set of base clustering results and discriminating a cluster core and a cluster halo. To quantify a sample's tendency of changing groups, we introduce a measurement named as sample's stability.

Let us first consider a simple visible artificial data set 2*d*2*k* [9], which has 1000 samples and 2 Gauss distribution clusters. Fig. 3 shows the distribution of 2d2k. In Fig. 3, two samples are singled out, which are respectively marked as triangle and star. Here, we run the random 1D k-means [13] ($k = 2$) 50 times on 2*d*2*k* to obtain diverse base clusterings. The random 1D k-means projects the data set onto a random line and uses k-means to generate a clustering result. As to a sample, it is easy to obtain the frequencies that it appears in the same clusters with others. In Fig. 4, we use gray images to respectively show the frequencies of the triangle sample and the star sample with others in the descending order, in which the black indicates the co-association frequency is 1 and white indicates the co-association frequency is 0. From Fig. 4, it is easy to find that there are many gray areas for the star sample, which indicate the changes between groups. For the triangle sample, there are many black areas and white areas, which indicate consistent assignment.

From the example, one may observe that the stability of a sample can be measured by its relationship with others. For a sample, if the co-association frequencies with others are either relatively high or relatively low, it can be deemed as a stable sample, and otherwise, it has low stability. In this section, we present a methodology to quantify such stability.

3.1. A methodology to measure the sample's stability

From the above analysis, the stability of relationship between two samples should be measured firstly. Considering the co-association frequency of two samples, if two samples x_i and x_j are assigned to the same group by all clustering results, i.e. $p_{ij} = 1$, the relationship between x_i and x_j is certain. In addition, x_i and x_j have a certain relationship if all clustering results consistently assign them to different groups, i.e. $p_{ij} = 0$. The relationship between x_i and x_j is unstable if their co-association frequency value is in $(0, 1)$. A threshold t can be learned based on the distribution of the co-association frequencies to indicate the most unstable frequency. According to the above discussion, based on a set of clustering results, the stable relation between two samples contains two aspects: (1) most of the clustering results assign them in the same cluster; (2) most of the clustering results assign them in different clusters. A stable level that is directly determined by the co-association frequency can not reflect the second aspect. Therefore, we design a determinacy function f to project the co-association frequency into a relationship stability space. To measure the determinacy of the relation of two samples, the function f should have high values at both ends but low value at t. From the above discussions, the determinacy function f is defined as follows:

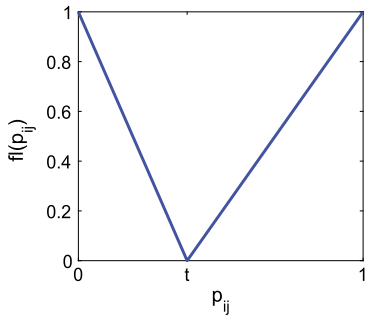

Fig. 5. The curve of fl.

Definition 1. (Determinacy function): f is a determinacy function if for arguments $p \in [0, 1]$ and with a parameter $t \in (0, 1)$, it satisfies:

(1) If $p < t$, $f'(p) < 0$; if $p > t$, $f'(p) > 0$.
(2) If $p_i < t < p_j$, and $\frac{t-p_i}{p_j-t} = \frac{t}{1-t}$, $f(p_i) = f(p_j)$.

The first condition requires that the determinacy function should obtain the minimum value on t, and obtain high value far away from t. The second condition requires that the determinacy function should be unbiased for the co-association values on both sides of t. Concretely, if two co-association values p_i and p_j are located on the same side of t, then the farther co-association value will obtain a higher determinacy value. Formally, this characteristic is shown as if $(p_i - t)(p_j - t) > 0$ and $|p_i - t| > |p_j - t|$, then $f(p_i) > f(p_j)$. This characteristic is easy to be obtained based on the first condition. In addition, the two co-association values located on the different sides of t have the following relation: if $p_i < t < p_j$ and $(1 - t)(t - p_i) > t(p_j - t)$, then $f(p_i) > f(p_j)$. This relation holds true directly based on the two conditions of the determinacy function. The above discussions indicate that the determinacy function can reflect the two aspects of a stable relation between two samples.

In clustering ensemble, for a sample, if its relations with the others are stable, this sample should have high stability. Therefore, with a determinacy function f, the stability of a sample x_i which comes from database X with n samples can be quantized by the average agreements of decisions on its relations with the other samples. Then, based on a determinacy function f, the stability of x_i, $s(x_i)$, is calculated by:

$$s(x_i) = \frac{1}{n}\sum_{j=1}^{n} f(p_{ij}). \tag{2}$$

Based on this methodology, we propose two stability measures: stability based on linear function and stability based on quadratic function.

3.2. Stability based on linear function

Satisfying the conditions of determinacy function by Definition 1, a linear function can be designed as

$$fl(p_{ij}) = \begin{cases} |\frac{p_{ij}-t}{t}|, & p_{ij} < t; \\ |\frac{p_{ij}-t}{1-t}|, & p_{ij} \geqslant t. \end{cases} \tag{3}$$

Fig. 5 shows the curve of $fl(p_{ij})$. The linear-based stability of sample x_i is quantized based on the agreement degree of its relations with the other samples, which is calculated by

$$sl(x_i) = \frac{1}{n}\sum_{j=1}^{n} fl(p_{ij}). \tag{4}$$

To learn a threshold t, in this paper, we solve a linear discriminant problem using Otsu algorithm [53]. Suppose $D = \{d_1, d_1, \cdots, d_m\}$ is a vector with m elements. A threshold $\tilde{t}$ can divide D into two groups g_0 and g_1 by:

$$g_0 = \{d_i : d_i < \tilde{t}, 1 \leqslant i \leqslant m\}, \tag{5}$$

$$g_1 = \{d_i : d_i \geqslant \tilde{t}, 1 \leqslant i \leqslant m\}. \tag{6}$$

Algorithm 1 Otsu.

INPUT: a vector $D = \{d_1, d_1, \cdots, d_m\}$
OUTPUT: threshold t^*

1: **for** $i = 1$ to m **do**
2: calculate σ_{d_i} based on Formula (7)
3: **end for**
4: $\Sigma = \{\sigma_1, \sigma_2, \cdots \sigma_m\}$
5: $i = \arg\max_{1 \leqslant i \leqslant n}(\sigma_i)$
6: $t^* = d_i$

The following task is to evaluate the performance of each threshold and select the most suitable one. A discriminant criterion which measures the between-class variance is utilized. For the two classes g_0 and g_1, the between-class variance is calculated by:

$$\sigma_t = \omega_0(\mu_0 - \mu)^2 + \omega_1(\mu_1 - \mu)^2, \tag{7}$$

where

$$\omega_0 = \frac{|g_0|}{|D|},\ \omega_1 = \frac{|g_1|}{|D|},$$
$$\mu_0 = \frac{\sum_{d_i \in g_0} d_i}{|g_0|},\ \mu_1 = \frac{\sum_{d_i \in g_1} d_i}{|g_1|},$$
$$\text{and}\quad \mu = \frac{\sum_{d_i \in D}}{|D|}.$$

In this scene, the most suitable threshold t^* should maximize Formula (7), which is:

$$t^* = \arg\max(\sigma_{\tilde{t}}). \tag{8}$$

Formula (8) can be solved by Algorithm 1. The Otsu algorithm is proposed to select a threshold from a gray-level image. In [53], the author states that the Otsu algorithm can also be applied in selecting a threshold in the scene that a histogram of some discriminative characteristic for classifying the objects is available. As a fact, the Otsu method still remains one of the most referenced threshold learning methods by now [54].

Taking the co-association matrix as the input of Otsu, we can learn a threshold t. Then, we calculate the stability of samples in Fig. 1. Intuitively, the unstable sample should be the 3*rd* sample and the most stable sample should be the 1*st* and 2*nd* samples. Based on Formula (4), the stability of each sample in Fig. 1 is given by:

$$sl_1 = 0.4444,\quad sl_2 = 0.4444,\quad sl_3 = 0.2222,$$
$$sl_4 = 0.3333,\quad sl_5 = 0.3889,\quad sl_6 = 0.3889.$$

The stability values are consistent with the visual perception.

3.3. Stability based on quadratic function

In this subsection, we consider a simple determinacy mapping function, the quadratic function, which is defined by:

$$fq(p_{ij}) = \begin{cases} \left(\frac{p_{ij}-t}{t}\right)^2, & p_{ij} < t; \\ \left(\frac{p_{ij}-t}{1-t}\right)^2, & p_{ij} \geqslant t. \end{cases} \tag{9}$$

The curve of the quadratic function fq is shown in Fig. 6.
With fq, the stability of sample x_i is calculated by:

$$sq(x_i) = \frac{1}{n}\sum_{j=1}^{n} fq(p_{ij}). \tag{10}$$

For the examples in Fig. 1, the values of samples' stability based on quadratic function are:

$$sq_1 = 0.2130,\quad sq_2 = 0.2130,\quad sq_3 = 0.0648,$$
$$sq_4 = 0.1389,\quad sq_5 = 0.1759,\quad sq_6 = 0.1759.$$

In this example, the stability values obtained by sq are consistent with the visual perception.

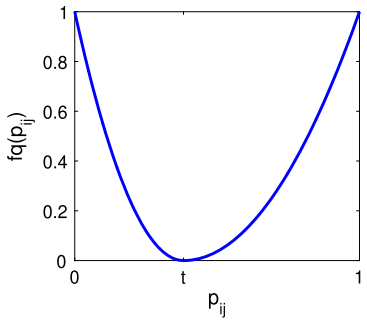

Fig. 6. The curve of fq.

Algorithm 2 Finding a cluster core and a cluster halo.

INPUT: clustering results set $\Pi(n \times L)$
OUTPUT: indices of cluster core samples O,
indices of cluster halo samples H.

1: calculating the co-association matrix $M(n \times n)$
2: **for** $i = 1$ to n **do**
3: obtain s_i^M with Formula (4) or Formula (10)
4: **end for**
5: $S^M = \{s_1^M, s_2^M, \cdots s_n^M\}$
6: $t_s \leftarrow$ Algorithm 1 (S^M)
7: $O = \{i | s_i^M > t_s\}$
8: $H = \{i | s_i^M \leq t_s\}$

The stability of each sample can be calculated based on Formula (4) or Formula (10). With these stability values, the cluster core and the cluster halo can be determined by a threshold. It is easy to discover a clear pre-structure solely based on the cluster core samples. The pre-structure can be used to guide the assignment of a hard determined halo sample. Motivated by the above results, we design in this paper a clustering ensemble algorithm based on sample's stability.

4. A clustering ensemble algorithm based on sample's stability

In this section, we present a **c**lustering **e**nsemble algorithm based on **s**ample's **s**tability (using CEs^2 for short). The main idea of CEs^2 is using a sample's stability measure to find a cluster core and a cluster halo, and handling the samples in the core and the samples in the halo differently to discover the underlying structure. The CEs^2 consists of four parts: (1) finding a cluster core and a cluster halo based on sample's stability; (2) discovering the underlying structure based on samples in the cluster core; (3) assigning samples in the cluster halo to the structure; (4) adjusting the structure to obtain a clustering solution.

4.1. Using stability to find a cluster core

Generally, in the clustering analysis, a cluster core is defined as the part with a robust assignment. An effective way to improve the robustness of a clustering algorithm is finding a right cluster core and using the cluster core to guide the assignment of samples in the cluster halo. The cluster core is often discovered according to the distribution of a data set. However, in clustering ensemble, determining the distribution of a data set is difficult due to the unknown original features of the data set. To find a cluster core, here, we will utilize the samples' stability introduced in Section 3.

Suppose that the assignment of samples in a cluster core is robust. Then different base clustering results may generate similar assignments for the cluster core samples. Thus the samples in the cluster core will have high stability. Therefore, it may be effective to use the stability to determine a cluster core. In other words, finding a cluster core with the stability assumes that the samples in a cluster core have higher stability than those samples do in a cluster halo. The stabilities of all samples $S^M = \{s_1^M, s_2^M, \ldots, s_n^M\}$ can be obtained based on Formula (4) or (10). With these stabilities, a data set can be divided into two groups according to a threshold t_s determined by Algorithm 1. The indices of samples in the cluster core and cluster halo, respectively, are given by:

$$O = \{i \mid s_i^M > t_s, i = 1, 2, \ldots, n\}, \tag{11}$$

and

$$H = \{i \mid s_i^M \leq t_s, i = 1, 2, \ldots, n\}. \tag{12}$$

Algorithm 3 Discovering core structure.

INPUT: indices of cluster core samples O,
co-association matrix M.
OUTPUT: core structure C_O^*.
1: extracting co-association matrix of cluster core M_O
2: $C_O^* \leftarrow$ HC-algorithm (M_O)

After determining the cluster core and cluster halo, the processes of handling samples in these two parts are quite different. For the cluster core, one may hope to find a clear underlying structure. With this structure, the assignments of the samples in the cluster halo can be carried out. In what follows, our suggestions for handling these two types of samples are discussed.

4.2. Discovering the structure of a cluster core

The stability of a sample is determined by the co-association matrix. In order to reduce the amount of computation, the co-association matrix should be employed to discover the structure of a cluster core. We will use the rows and columns corresponding to the samples in the cluster core to form the co-association matrix for the cluster core. In addition, the co-association matrix reflects the relationship between each pair of samples, which is beneficial for discovering a consensus clustering. Any clustering algorithms based on the co-association matrix can be applied to discover an underlying structure of cluster core. Here, we use the hierarchical clustering (HC) algorithm [55].

For a data set with n samples, the HC algorithm begins with n clusters, in which each cluster corresponds to a sample. The HC algorithm iteratively merges two most similar clusters until the number of clusters reaches the expectation. The measurement of similarity between two clusters is important to the quality of clustering result. The frequently used similarity measurements between two clusters are maximum similarity, minimum similarity, and average similarity, which correspond to three HC algorithms called single-linkage, complete-linkage, and average-linkage, respectively. The three similarity measurements are as follows:

$$d_{max}(c_i, c_j) = \min_{x \in c_i, y \in c_j} sim(x, y);$$

$$d_{min}(c_i, c_j) = \max_{x \in c_i, y \in c_j} sim(x, y);$$

$$d_{ave}(c_i, c_j) = \frac{1}{|c_i||c_j|} \sum_{x \in c_i} \sum_{x \in c_j} sim(x, y).$$

In clustering ensemble, sample features are unknown. As a compromise, p_{ij} can be used in place of the similarity between two samples x_i and x_j. Then, in clustering ensemble, the similarity between two samples will be:

$$sim(x, y) = p_{xy}.$$

The HC algorithm has two main advantages in finding the underlying structure of the cluster core. Firstly, the co-association matrix can be treated as the similarity matrix in clustering ensemble problem. This means that the input of the HC algorithm is already available, which is the main computation of the algorithm. Secondly, the HC algorithm can determine the number of clusters by the largest jump during merging clusters. During the merging process based on the co-association matrix, when the process meets a pair of clusters with the lowest similarity, it terminates.

A sample in a cluster core has very high similarity with its neighbors and very low similarity with others. Thus, each element in the co-association matrix of the cluster core is close to 0 or 1. Since the underlying structure of cluster core is clear, the three HC algorithms will discover very similar structures. In this paper, we utilize the single link algorithm to discover a structure of cluster core.

4.3. Assigning samples in a cluster halo to the structure

The task of this step is to assign the samples in the cluster halo to the discovered core structure. The core structure is expressed as a clustering result on the core samples. In addition, we have obtained the pairwise similarity matrix. Then, a direct approach to assign halo samples may be based on the relations between the halo sample and the discovered clusters. One can assign a halo sample to its nearest cluster. It should be noted that some samples in the cluster halo are far away from all core clusters. Thus determining their assignments based on only core samples may be ineffective. Here, we propose a non-parametric iterative approach to assigning the halo samples to the discovered structure.

This approach describes a spreading process which uses the samples in the cluster halo to expand the size of cluster core to the whole data gradually. This approach is realized through a two-phase iterative method: the samples which are near a core cluster are found first, and then, the cluster core are augmented by these samples. The two phases are successively executed until all samples are assigned to the cluster core.

Algorithm 4 Assigning halo samples.

INPUT: indices of cluster core samples O,
indices of cluster halo samples H,
co-association matrix M,
core structure C_O^*
OUTPUT: pre-structure C'^*

```
while |O| ≠ n do
    calculate O' with Formula (15)
    extract M_{OO'} with M, O and O'
    obtain C'* based on Formula (16)
    O = O ∪ O'
end while
```

In the first phase, to find the samples near the core cluster, the similarity between the samples in the cluster halo and the discovered core clusters are calculated. The similarity between a sample x and a cluster c can simply be measured by the similarity between the measured sample and its nearest sample in the cluster:

$$s(x, c) = s_c = \max_{y \in c} sim(x, y). \tag{13}$$

A sample's proximity to the cluster core is defined by the similarity between the sample and its nearest core cluster. For sample x, its proximity is:

$$pr_x = max\{s_{c_1}, s_{c_2}, \ldots, s_{c_{K_c}}\}, \tag{14}$$

where K_c is the number of discovered clusters in the cluster core. With Formula (14), we can obtain all the proximities of the samples in the cluster halo. With these proximities, the samples which are near the cluster core can be selected based on a threshold t_{pr} determined by Algorithm 1. The indices of selected samples are:

$$O' = \{i : pr_{x_i} > t_{pr}, i \in H\}. \tag{15}$$

Next, the selected samples are assigned to the cluster core by labeling them. The most direct method is to assign each selected sample to its nearest cluster. The assignment of a halo sample x_i can be conducted by

$$C^*(x_i) = \arg\max_{c_k, 1 \le k \le K_c} \{s_{c_1}, s_{c_2}, \ldots, s_{c_{K_c}}\}. \tag{16}$$

This process is based on the fact that the clusters in the core are far away from each other. After this assigning process is completed, the cluster core has changed. We need to select a new set of testing samples. After assigning all halo samples, the pre-structure of the data set will be discovered. Algorithm 4 shows the detailed steps of the assignment of halo samples. Due to that the halo samples are assigned gradually, the relation information in cluster halo are also fully utilized to construct the structure. In this paper, the relation information is the co-association probabilities. Actually, the above processes are all based on the pairwise relation matrix. Thus, if some semi-supervised information is available, such as must-link and must-not-link pairs, it is easy to expand the above processes to semi-supervised version through adding some constraints.

4.4. Generation of the final clustering

It is natural that the number of discovered clusters is different from the expected number. Taking the halo samples away can split a cluster into several parts. Thus, more clusters are discovered. To generate the final result, some close clusters will be merged. This task also can be conducted by the HC algorithm. Clusters are merged until the cluster number reaches the pre-defined number of clusters. Finally, a structure that describes the data distribution will be generated. The adjusting steps are shown by Algorithm 5. Sequentially executing Algorithm 2 to Algorithm 5 forms the framework of the algorithm CEs^2, which is shown as Algorithm 6.

5. Experimental analysis

In this section, we verify the performance of CEs^2. Based on the two stability measures, two CEs^2 algorithms are proposed, which are CEs^2-L using stability based on linear function and CEs^2-Q using stability based on quadratic function. The experiments consist of three parts. First, we employ the image segmentation task to verify the rationality the sample's stability measured by *sl* and *sq*. Then, eight synthetic data sets are used to show the working mechanism and the robustness of CEs^2. Finally, we compare the two CEs^2 algorithms with twelve state-of-the-art clustering ensemble algorithms on ten benchmark data sets from UCI [56] and six document data sets come with the CLUTO clustering toolkit [57].

Algorithm 5 Adjusting.

INPUT: pre-structure C'^*,
co-association matrix M,
cluster number k.
OUTPUT: clustering result C^*

$k' \leftarrow$ cluster number in C'^*
while $k' > k$ **do**
  calculate cluster similarity matrix $D(k' \times k')$ based on C'^* and M
  $(i, j) = \arg(\min(D))$
  obtain C'^* through merging c'_i and c'_j
  $k' = k' - 1$
end while

Algorithm 6 CEs^2.

INPUT: clustering results set Π, cluster number k
OUTPUT: clustering result C^*

$(O, H, M) \leftarrow$ Algorithm 2 (Π)
$C^*_0 \leftarrow$ Algorithm 3 (O, M)
$C'^* \leftarrow$ Algorithm 4 (C^*_0, O, H, M)
$C* \leftarrow$ Algorithm 5 (C'^*, M, k)

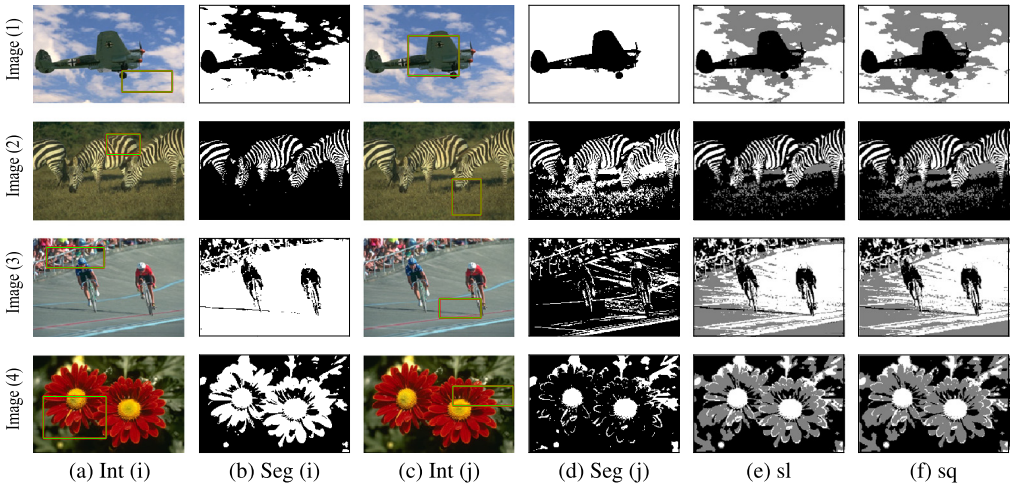

Fig. 7. Results of image segmentation experiments (1).

5.1. The rationality of the sample's stability

To visually illustrate the rationality of the proposed two sample's stability measures *sl* (Formula (4)) and *sq* (Formula (10)), we utilize the case of image segmentation. The image segmentation task is to partition an image into disjoint and homogeneous regions, which is usually fulfilled by discovering the contours of different regions. In [58], Chan and Vese proposed one of the most popular two-phase level set [59] based methods, which is just named CV method. The CV method is a particular case of the minimal partition problem, which minimizes the energy of the partition. In the CV model, the level set function is used to represent the contours. The main steps of the CV algorithm are: (1) contour initialization, (2) minimizing the energy with respect to the mean intensities in the two regions, (3) minimizing the energy with respect to the level set, (4) repeating steps (2) and (3) until convergence. It is natural that different initial contours may lead to different segmentations for an image. Thus, we run the CV method multiple times on an image to generate diverse base segmentations and discover the unstable regions based on *sl* and *sq*.

For an image, we first utilize the CV method to generate 50 segmentations with random initial contours, which are rectangles with random sizes and random locations. Then, we respectively evaluate the stability of each pixel in an image with Formula (4) and (10). Based on Algorithm 1, we divide the image into stable region and unstable region. Finally, we ensemble the segmentation results of the stable region simply using HC algorithm with $k = 2$, and draw the unstable region with gray color. With the above process, an image is shown with three colors, in which the white region and black region are the segmentation results of the stable region, and the gray region is the unstable region.

We employ The Berkeley Segmentation Dataset (500) (BSDS500) [60] to conduct this experiment. Fig. 7 and Fig. 8 show the experimental results of eight example images. In Fig. 7 and Fig. 8, (a) and (c) show two examples of random initial contour, and (b) and (d) respectively show the corresponding segmentations of (a) and (c). The unstable regions which are

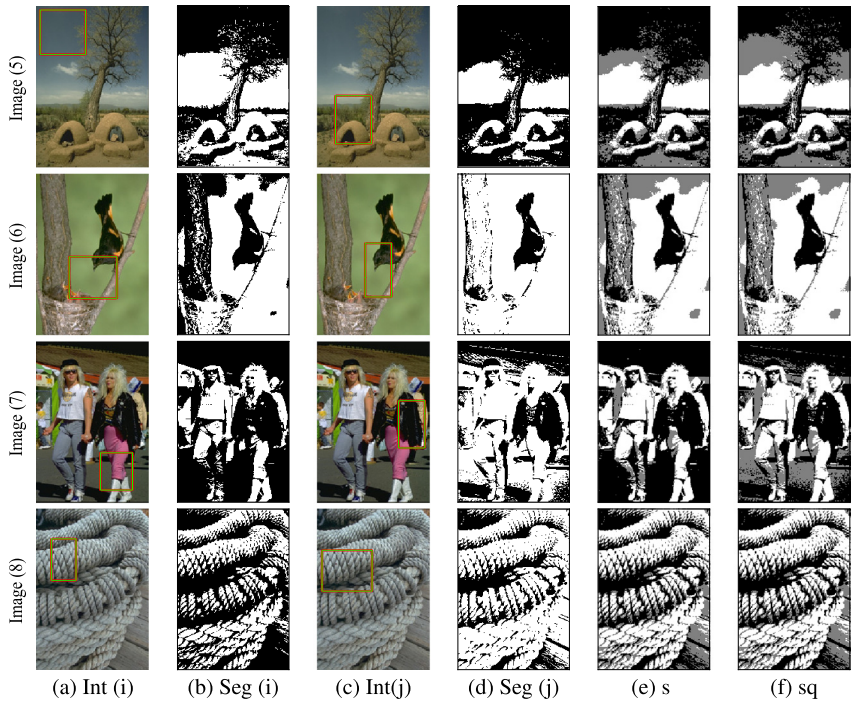

Fig. 8. Results of image segmentation experiments (2).

discovered by linear-based stability *sl* and quadratic-based stability *sq* are shown in (e) and (f), respectively. From Fig. 7 and Fig. 8, it is obvious that the CV model generates different segmentations based on different initial contours. Comparing sub-figures (e) and (f) with sub-figures (b) and (d), it can be founded that the gray regions in (e) and (f) are roughly equal to the differences between (b) and (d). Visually, the sub-figures (e) and (f) are more real and reflect the original image clearer than (b) and (b). The reasons mainly include two aspects. First, the regions which are hard to be defined are recognized as unstable region. Take for instance the image (1), the sky is recognized as the unstable region, which can be segmented in the same partition either with the cloud (Fig. 7 (b)) or with the plane (Fig. 7 (j)). The same facts occur in the images such as the road in image (3), the petals in image (4), and the trunk in image (6). Second, the borders of an object are usually recognized as unstable region, which makes the object be more stereoscopic in visual. Such as the zebra in image (2) and the ropes in image (8). In addition, the results show that the unstable regions recognized by sl and sq are very similar in most cases. While for image (2) and image (7), the sq discover more unstable regions than the sl.

5.2. Experiment on synthetic data sets

To visualize how CEs^2 works, we present the result of each step of CEs^2 on eight synthetic data sets. Table 1 summarizes the detailed information of these synthetic data sets. Fig. 9 shows the distributions of these synthetic data sets.

To generate a set of base clustering results, multiple k-means algorithms with random initial centers are utilized. As for the number of clusters k in each clustering, we follow the suggestion that k should be greater than the expected number of clusters in the literature [34,35,31], and set the cluster number in each base clustering as $k = \min\{\sqrt{n}, 50\}$ in the experiment. The size of each ensemble is set as 50 in the experiment, i.e. $L = 50$.

The results of each step for CEs^2 are shown in Fig. 10 to Fig. 17, in which (a) and (d) are the clustering results based on cluster cores, (b) and (e) are the clustering results of assigning the samples in the cluster halo, and (c) and (f) are the final clustering results of the adjusting step.

As shown in Fig. 9, samples in *Tetra* are generated by mixed Gaussian distributions. The clusters in this data set are spherical. For this data set, the traditional k-means algorithm based on Squared Euclidean distance can generate an effective result. Fig. 10 shows that CEs^2 is also able to recognize the spherical clusters in these two data sets. As shown in Fig. 10 (d), it is interesting that the samples in the center region of a cluster are recognized as belonging to the cluster halo. The reason is that when the cluster number k is larger than the expected value, the center region is divided by its surrounding clusters. In this situation, the samples in the center region of a cluster will have low stability and belong to the cluster halo.

For the data sets which are shown in Fig. 9 (b) to Fig. 9 (h), the k-means algorithm in the generation step can not handle these data very well. How CEs^2 integrates these low quality base clustering results into high quality clustering results are shown in Fig. 11 to Fig. 17. From (a) and (d) in Fig. 11 to Fig. 17, it is easy to see that effective results are generated for the samples with high stability. Given the results in (a) and (d), the predictions of the low stability samples are shown in

Table 1
Description of the eight synthetic data sets.

Data sets	N	D	K	source
Tetra	400	3	4	[61]
Noisy	200	2	2	[30]
Wingnut	1016	2	2	[61]
Flame	240	2	4	[62]
Jain	373	2	3	[63]
Target	770	2	6	[61]
Chainlink	1000	3	2	[61]
Atom	800	3	2	[61]

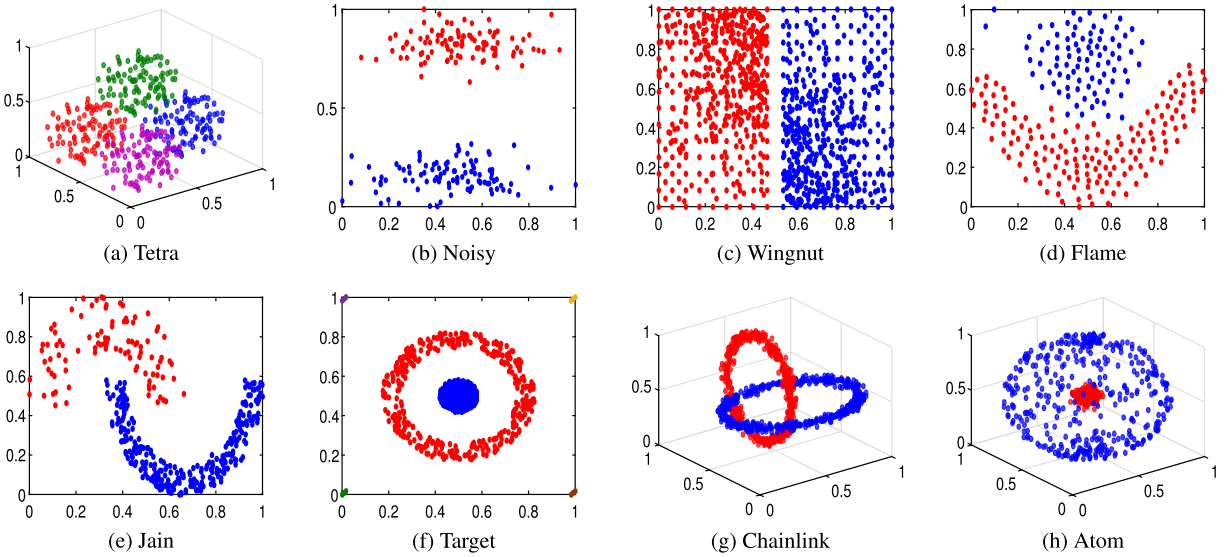

Fig. 9. The eight synthetic data sets.

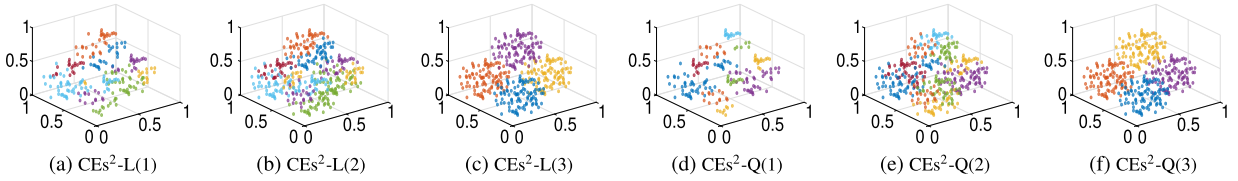

Fig. 10. Experiment on the Tetra data set.

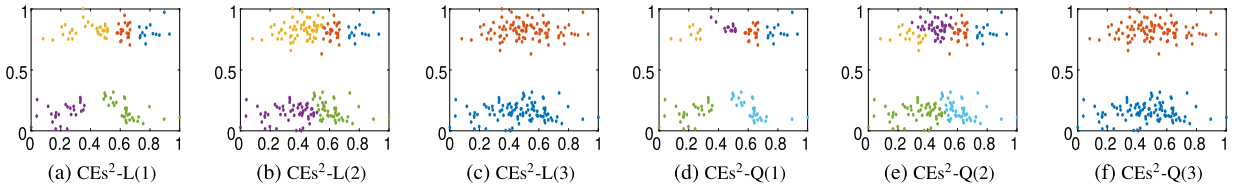

Fig. 11. Experiment on the Noisy data set.

(b) and (e) of Fig. 11 to Fig. 17. It is observed that the assigning step in CEs^2 offers more clusters than the expected, which can be deemed as segmentations of the targeted clusters. To obtain a better clustering, the adjusting step combines these clusters to attain the expected number of clusters, whose results are shown in (c) and (f) of Fig. 11 to Fig. 17. Visually, both algorithms CEs^2-L and CEs^2-Q effectively recognize the underlying structures of these synthetic data sets.

We also show the ability of other twelve clustering ensemble algorithms in handling the eight synthetic data sets. The twelve clustering ensemble algorithms include one feature-based method (Voting [47]), three cluster-based methods (WCT, WTQ, and CSM [50,4]), one cluster co-association-based methods (MCLA [9]), and seven sample co-association-based methods (CSPA [9], EAC [14], HGPA [9], PTA [22], PTGP [22], SCCE [16] and NCUT [64]). The Voting method finds cluster

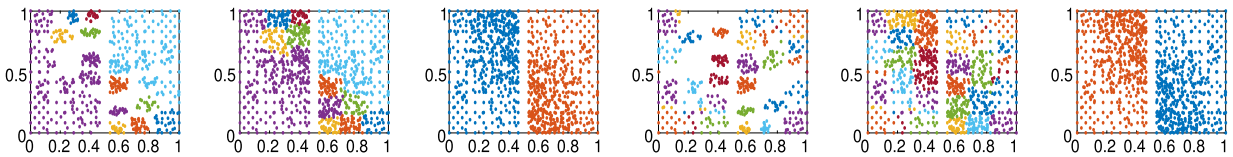

Fig. 12. Experiment on the Wingnut data set.

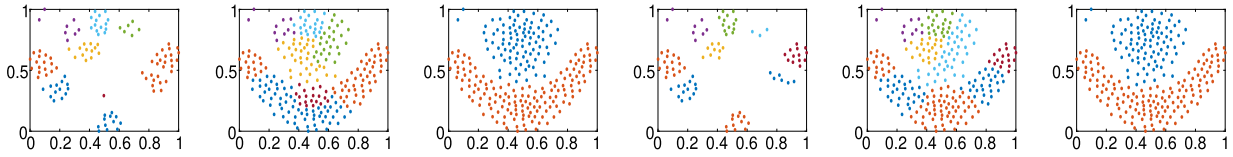

Fig. 13. Experiment on the Flame data set.

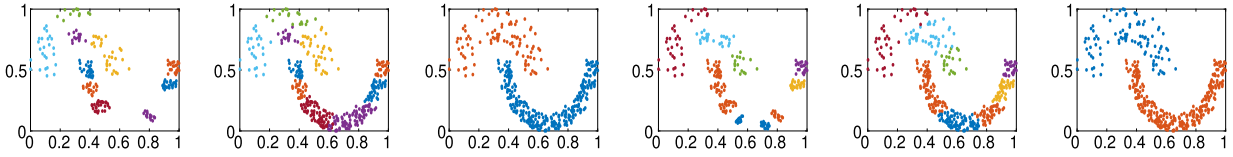

Fig. 14. Experiment on the Jain data set.

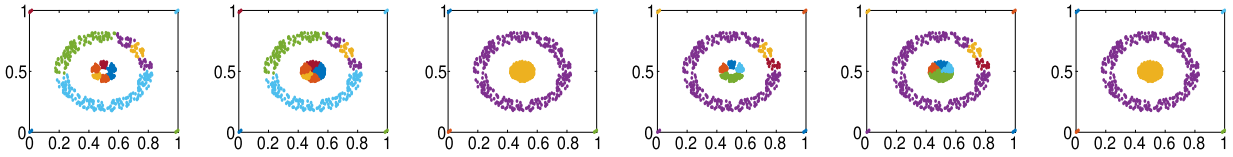

Fig. 15. Experiment on the Target data set.

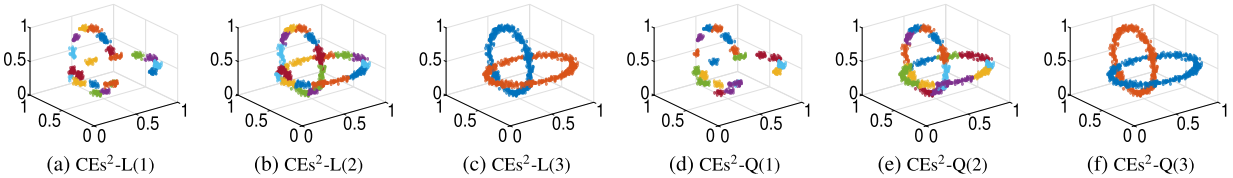

Fig. 16. Experiment on the Chainlink data set.

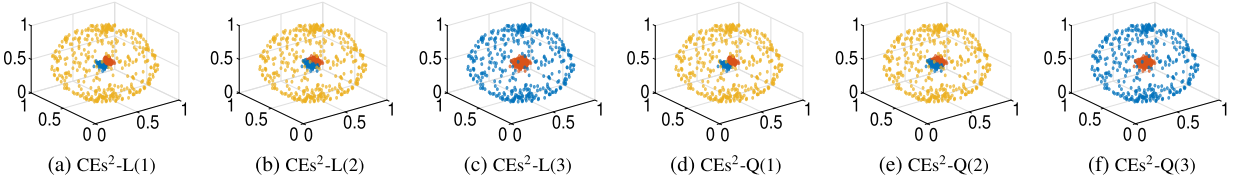

Fig. 17. Experiment on the Atom data set.

correspondences and utilizes voting strategy to generate the final result. Based on the fact that the cluster matrix is sparse and may limit the quality of data grouping, WCT, WTQ, and CSM utilize different link-based similarity measures to refine the cluster matrix and then generate a clustering result through a partition strategy. In the experiment, we use the k-means algorithm as the partition strategy. MCLA builds a cluster co-association matrix based on binary Jaccard measure and finds cluster correspondences by grouping clusters. In MCLA, each sample is assigned to its most related cluster. Based on the co-association matrix, CSPA utilizes a graph partition method METIS to generate a final result, and EAC utilizes hierarchical

Table 2
The abilities of the compared methods in handling the eight synthetic data sets.

Data	Voting	WCT	WTQ	MCLA	CSPA	EAC	HGPA	CSM	PTA	PTGP	SCCE	NCUT	CEs^2-L	CEs^2-Q
Tetra	◯	◯	◯	×	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯
Noisy	◯	×	×	◯	◯	◯	×	◯	×	◯	◯	×	◯	◯
Wingnut	◯	×	×	×	◯	◯	×	◯	◯	×	◯	×	◯	◯
Flame	×	◯	×	◯	×	×	×	×	◯	◯	◯	×	◯	◯
Jain	×	×	×	×	×	×	×	×	×	◯	◯	◯	◯	◯
Target	×	×	×	×	×	×	×	×	×	×	×	×	◯	◯
Chainlink	◯	×	×	×	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯
Atom	◯	◯	◯	◯	◯	◯	◯	×	◯	◯	◯	◯	◯	◯

Table 3
Description of the sixteen data sets.

Number	Data sets	N	D	K
1	Breast Tissue	106	9	6
2	Glass	214	9	6
3	Protein Localization Sites	272	7	3
4	Ecoli	336	7	8
5	LIBRAS Movement	360	91	15
6	User Knowledge Modeling	403	5	4
7	Cardiotocography	2126	40	10
8	Image Segmentation	2310	19	7
9	Parkinsons Telemonitoring	5875	21	42
10	Statlog Landsat Satellite	6435	36	6
11	tr23	204	5832	6
12	tr45	690	8261	10
13	tr41	878	7454	10
14	tr31	927	10128	7
15	wap	1560	8460	20
16	re1	1657	3758	25

clustering algorithm to generate a final result. HGPA is based on an $n \times n$ matrix whose elements indicate hyper-edges connecting two samples. In HGPA, clustering structure is discovered by cutting a minimum number of hyper-edges to make the hyper-graph un-connect. Based on the co-association matrix, PTA and PTGP first built a probability trajectory based similarity matrix. Then, PTA utilizes dendrogram generate a final result, while PTGP utilizes Tcut graph partition method. In SCCE, the spectral algorithm is used to solve the problem of clustering ensemble based on the co-association matrix. NCUT is an image segmentation method and it has been widely used in clustering ensemble algorithms [52,6]. The abilities of the twelve clustering ensemble algorithms in handling the eight synthetic data sets are shown in Table 2. In Table 2, symbol ◯ indicates that a clustering ensemble algorithm can correctly discover the group structure from the corresponding synthetic data, while symbol × indicates it can't. From Table 2, it is easy to find that none of the twelve compared algorithms can effectively handle all the eight synthetic data sets. Table 2 shows that the SCCE algorithms only failed in handling Target data. However, in the following section, SCCE shows a bad performance in handling the benchmark data sets.

5.3. Experiment on benchmark data sets

Ten numerical benchmark data sets from UCI and six text benchmark data sets are used in this comparison experiment. Table 3 shows the detailed information about these data sets.

To verify the performance of CEs^2, we compare the two CEs^2 algorithms (CEs^2-L and CEs^2-Q) with the twelve clustering ensemble algorithms which have been introduced in Section 5.2.

In this experiment, the ensemble size is still set as $L = 50$ and the number of clusters in each base partition is set as $k = \min\{\sqrt{n}, 50\}$. To eliminate the randomness caused by the uncertainty of ensembles, each comparison is conducted on 50 ensembles and the average estimation index values are reported.

To evaluate the performance of a clustering result, we utilize two widely used clustering estimation indexes, which are the clustering accuracy [65] and the normalized mutual information [9]. The two indexes are external criteria that measure the performance of a clustering algorithm through computing the similarity between its result and a referential clustering result. In the following experiments, the ground truth is utilized as the referential clustering. Then, we only introduced the two indices in the environment that the compared partitions have the same number of clusters, which is the true number k in each data set.

The clustering accuracy (AC) matches corresponding clusters in the compared results and reports the fraction of their common samples. Based on the overlap matrix between a clustering result C' and ground truth C, which is shown in Table 4, the AC is calculated by:

Table 4
Overlap matrix between C' and C.

$C \backslash C'$	C'_1	C'_2	$\cdots$	C'_k	Sums
C_1	n_{11}	n_{12}	$\cdots$	n_{1k}	$n_{1\cdot}$
C_2	n_{21}	n_{22}	$\cdots$	n_{2k}	$n_{2\cdot}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
C_K	n_{k1}	n_{k2}	$\cdots$	n_{kk}	$n_{k\cdot}$
Sums	$n_{\cdot 1}$	$n_{\cdot 2}$	$\cdots$	$n_{\cdot k}$	n

$$AC = \sum_{i=1}^{k} \frac{\max\{n_{ij} : \ j = 1, 2, \ldots, k\}}{n}, \tag{17}$$

where n_{ij} is the number of common samples of cluster C_i in C and cluster C'_j in C'.

The normalized mutual information (NMI) computes the information shared between two partitions, which is defined as follows:

$$NMI(\pi^b, \pi^d) = \frac{\sum_{i=1}^{k}\sum_{j=1}^{k} n_{ij} \log\left(\frac{n n_{ij}}{n_{i\cdot} n_{\cdot j}}\right)}{\sqrt{\left(\sum_{i=1}^{k} n_{i\cdot} \log\left(\frac{n_{i\cdot}}{n}\right)\right)\left(\sum_{j=1}^{k} n_{\cdot j} \log\left(\frac{n_{\cdot j}}{n}\right)\right)}}. \tag{18}$$

Both indexes are bounded between 0 and 1, in which a higher value indicates a better performance.

The values of AC and NMI in the comparison experiments are reported in Table 5 and Table 6, respectively. In Table 5 and Table 6, the last row shows the average ranks of each algorithm on the sixteen data sets. For each data set, the highest index value is double underlined, while the second highest value is marked with underline. It is easy to see from Table 5 that the proposed CEs^2 algorithms obtain the highest AC values for fourteen data sets. For twelve data sets, the two versions of CEs^2 algorithms win the first two places. For many data set, the two CEs^2 algorithms can markedly improve the AC value on many data sets. Table 6 shows that the CEs^2 algorithms get higher NMI values than the other eight algorithms on eleven data sets. From the last row in Table 5 and Table 6, it is obvious that the two CEs^2 algorithms obtain the top two ranks. The average ranks of the two CEs^2 algorithms are around 2, which indicates that the two CEs^2 algorithms consistently outperform the other algorithms on most of the data sets.

To further analyze the results reported in Table 5 and Table 6, we utilize the Friedman test to detect whether the compared algorithms are significantly different. To conduct this test, we use the Matlab function 'friedman'. Based on Table 5 and Table 6, the p-values that the test returns are 1.3026×10^{-18} and 2.1224×10^{-16}, respectively. Both the p-values are sufficiently small, which suggest that at least one pair of algorithms is significantly different. To visually show the differences of the compared algorithms, we apply Nemenyi post-hoc test [66]. The critical value of the Nemenyi test is calculated by:

$$Ne = q_\alpha \sqrt{\frac{A(A+1)}{6D}}, \tag{19}$$

where A is the number of algorithms, D is the number of data sets and $q_\alpha = 4.7427$ when the confidence level $\alpha = 0.05$. If the average rank of an algorithm is Ne different than that of another algorithm, it can be deemed that these two algorithms are significantly different. In this experiment, with Formula (19), we obtain $Ne = 7.0145$. Fig. 18 and Fig. 19 show the results of the Nemenyi test. In Fig. 18 and Fig. 19, the horizontal axis corresponds to the fourteen algorithms and the vertical axis corresponds to the value of average ranks. For each algorithm, its average rank is shown by a red point and its confidence interval is shown by a blue line whose length is 7.0145. The black dotted line shows the up confidence level of CEs^2-L, which obtains a high rank in the two CEs^2 algorithms. From Fig. 18 and Fig. 19, it is easy to see that the two CEs^2 algorithms obtain much higher average ranks than the other algorithms. Concretely, CEs^2-L and CEs^2-Q are significantly different than CSPA, EAC, SCCE, and NCUT.

6. Conclusion

Clustering ensemble is an effective approach to solve data clustering problem. Many clustering ensemble algorithms have been proposed during the past decade, most of which treat each data samples equally. However, given a set of clustering results, the frequencies that samples changing between clusters are different, which means the contributions of different samples to the detection of the underlying structure should be different. In this paper, we have introduced sample's stability to reflect this difference, and have proposed a methodology to calculate this stability. Based on the sample's stability, we have proposed a novel clustering ensemble algorithm (CEs^2). This algorithm takes different processes to handle the samples in cluster core and the samples in cluster halo, which are divided based on the sample's stability. To verify the rationality of the sample's stability, we have applied it on the case of image segmentation. The results visually show that recognizing the unstable regions, the segmentation results are very encouraging. Experimental analysis on eight synthetic

Table 5
The index AC from ten clustering ensemble methods for the sixteen data sets.

Data	Voting	WCT	WTQ	CSM	MCLA	CSPA	EAC	HGPA	PTA	PTGP	SCCE	NCUT	CEs^2-L	CEs^2-Q
1	0.6283	0.5179	0.5151	0.5198	0.6028	0.5590	0.4976	0.5009	0.4816	0.4943	0.2406	0.4349	0.7113	0.6943
2	0.4220	0.4530	0.4196	0.4383	0.4432	0.4213	0.4290	0.4292	0.4776	0.4731	0.3570	0.3353	0.4995	0.4923
3	0.7489	0.8066	0.8443	0.8706	0.7581	0.7428	0.5151	0.7233	0.8149	0.8729	0.5160	0.4518	0.8965	0.8899
4	0.4638	0.5016	0.5186	0.5208	0.4808	0.4862	0.4552	0.5101	0.5638	0.4976	0.3987	0.2957	0.5326	0.5530
5	0.4432	0.4129	0.4040	0.4188	0.4531	0.4208	0.4496	0.4440	0.4407	0.4247	0.2794	0.3519	0.6328	0.6367
6	0.5444	0.4965	0.5306	0.5063	0.5437	0.5264	0.3900	0.5329	0.4603	0.5661	0.3279	0.5395	0.5482	0.5603
7	0.6199	0.6151	0.6377	0.6466	0.6248	0.6170	0.4650	0.6710	0.9905	0.9540	0.2761	0.8267	1.0000	0.9882
8	0.7061	0.5434	0.5759	0.5645	0.7041	0.6999	0.3535	0.6579	0.6463	0.6607	0.1457	0.6382	0.7292	0.7302
9	0.4857	0.5129	0.5068	0.5156	0.4885	0.4601	0.4899	0.3677	0.5107	0.5442	0.3306	0.4904	0.5458	0.5567
10	0.6119	0.5722	0.6283	0.5511	0.6160	0.5852	0.3116	0.4065	0.7380	0.6549	0.2438	0.5654	0.7508	0.7437
11	0.4359	0.5005	0.5213	0.5092	0.4373	0.4375	0.4137	0.4467	0.4512	0.4556	0.4074	0.4402	0.5306	0.5193
12	0.5406	0.5863	0.5604	0.5841	0.5403	0.5428	0.5356	0.5125	0.5305	0.5467	0.5251	0.4478	0.6110	0.6236
13	0.5306	0.5278	0.5370	0.5204	0.5353	0.4398	0.5200	0.5551	0.5755	0.5798	0.5521	0.5044	0.5800	0.5888
14	0.5038	0.4831	0.4742	0.4862	0.5077	0.3154	0.4780	0.5190	0.5106	0.5064	0.5101	0.4983	0.5483	0.5351
15	0.4267	0.4437	0.4827	0.4873	0.4809	0.4600	0.4429	0.4574	0.4314	0.4157	0.4035	0.4262	0.5306	0.5342
16	0.3965	0.3804	0.3772	0.3960	0.4008	0.3649	0.3509	0.3546	0.3983	0.3999	0.3739	0.3649	0.4187	0.4208
ave rank	8	8.1875	7.5	7.0625	6.6875	9.5	11.1875	8.3125	6.3125	5.6875	12.3125	10.875	1.6875	1.6875

Table 6
The index NMI from ten clustering ensemble methods for the sixteen data sets.

Data	Voting	WCT	WTQ	CSM	MCLA	CSPA	EAC	HGPA	PTA	PTGP	SCCE	NCUT	CEs^2-L	CEs^2-Q
1	0.5401	0.5261	0.5171	0.5201	0.5314	0.4843	0.4611	0.4680	0.4976	0.4860	0.1435	0.3308	0.5528	0.5347
2	0.2968	0.3452	0.3027	0.3260	0.3122	0.3209	0.3264	0.3363	0.3412	0.3575	0.0601	0.1571	0.4126	0.4022
3	0.4573	0.6670	0.6876	0.7139	0.4675	0.4757	0.3413	0.5115	0.6271	0.7113	0.0251	0.0353	0.7283	0.7214
4	0.5214	0.5611	0.5580	0.5557	0.5318	0.5128	0.5434	0.5383	0.5817	0.5505	0.0457	0.2213	0.5643	0.5729
5	0.5926	0.5786	0.5768	0.5869	0.5916	0.5582	0.6059	0.5782	0.5849	0.5724	0.1366	0.4955	0.6313	0.6208
6	0.3789	0.3146	0.3802	0.3328	0.3799	0.3678	0.1403	0.3731	0.2534	0.3808	0.0413	0.3911	0.3755	0.3841
7	0.7824	0.7695	0.8012	0.8048	0.7849	0.7677	0.5752	0.8242	0.9942	0.9336	0.0591	0.9068	1.0000	0.9940
8	0.6597	0.5730	0.6140	0.6073	0.6586	0.6511	0.3693	0.6069	0.6538	0.6382	0.0374	0.6255	0.6553	0.6698
9	0.6878	0.6812	0.6812	0.6825	0.6849	0.6621	0.6858	0.5412	0.6905	0.6799	0.0632	0.6754	0.6889	0.6912
10	0.5312	0.5202	0.5599	0.5181	0.5345	0.5407	0.2964	0.2562	0.6039	0.5584	0.0488	0.5328	0.5679	0.5907
11	0.2982	0.2898	0.3232	0.3005	0.3022	0.2153	0.2778	0.3222	0.3184	0.3198	0.2965	0.2647	0.3346	0.3366
12	0.5452	0.5272	0.5159	0.5352	0.5450	0.4736	0.5373	0.5287	0.5394	0.5314	0.5210	0.4155	0.5546	0.5564
13	0.5679	0.5667	0.5792	0.5611	0.5671	0.4611	0.5436	0.6018	0.5802	0.5775	0.5683	0.5020	0.5859	0.5916
14	0.4773	0.4355	0.4531	0.4602	0.4756	0.3127	0.4496	0.4980	0.4061	0.4066	0.3940	0.3804	0.4807	0.4815
15	0.5398	0.5428	0.5823	0.5936	0.5669	0.5892	0.5750	0.5873	0.5846	0.5666	0.5600	0.5577	0.6124	0.6123
16	0.4900	0.4869	0.4822	0.4843	0.4903	0.4782	0.4658	0.4520	0.4819	0.4799	0.4819	0.4705	0.4889	0.4915
ave rank	7	8.625	7	7.1875	6.5625	10.4375	9.8125	8	5.6875	7	12.4375	10.9375	2.5	1.8125

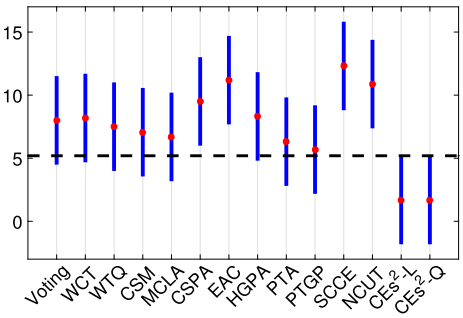

Fig. 18. Nemenyi test based on Table 5. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

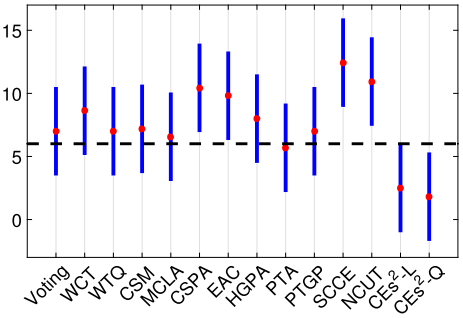

Fig. 19. Nemenyi test based on Table 6.

data sets shows how CEs^2 works, and experimental analysis on ten UCI data sets and six document data sets demonstrate the superior performance of CEs^2. In addition, the sample's stability could be effective for measuring the quality of a set of base clustering results. Therefore, stability can also be utilized to select clustering results, which is known as the selective clustering ensemble. In general, a selective clustering ensemble algorithm only integrates the selected clustering results. However, the discarded clustering results may offer useful information. It could be interesting to design a method that differentiates between selected clustering results and unselected clustering results.

Acknowledgements

This work was supported by National Key R&D Program of China (No. 2018YFB1004300), National Natural Science Foundation of China (Nos. 61802238, 61672332, 61432011, U1435212, 61773050, 61872226, 61802238, and 618822601), Program for the Outstanding Innovative Teams of Higher Learning Institutions of Shanxi, Program for the San Jin Young Scholars of Shanxi, Natural Science Foundation of Shanxi Province (Grant No. 201701D121052), and Innovation Program for Postgraduate Education of Shanxi (2018BY005). It was partially supported by CityU: 101113 of Hong Kong SAR Government.

References

[1] R. Xu, D. Wunsch, Survey of clustering algorithms, IEEE Trans. Neural Netw. 16 (3) (2005) 645–678.
[2] S. Vega-Pons, J. Ruiz-Shulcloper, A survey of clustering ensemble algorithms, Int. J. Pattern Recognit. Artif. Intell. 25 (03) (2011) 337–372.
[3] Z. He, X. Xu, S. Deng, A cluster ensemble method for clustering categorical data, Inf. Fusion 6 (2) (2005) 143–151.
[4] N. Iam-On, T. Boongeon, S. Garrett, C. Price, A link-based cluster ensemble approach for categorical data clustering, IEEE Trans. Knowl. Data Eng. 24 (3) (2012) 413–425.
[5] L. Jing, K. Tian, J.Z. Huang, Stratified feature sampling method for ensemble clustering of high dimensional data, Pattern Recognit. 48 (11) (2015) 3688–3702.
[6] Z. Yu, L. Li, J. Liu, J. Zhang, G. Han, Adaptive noise immune cluster ensemble using affinity propagation, IEEE Trans. Knowl. Data Eng. 27 (12) (2015) 3176–3189.
[7] Y. Yang, K. Chen, Temporal data clustering via weighted clustering ensemble with different representations, IEEE Trans. Knowl. Data Eng. 23 (2) (2011) 307–320.
[8] H. Elghazel, A. Aussem, Unsupervised feature selection with ensemble learning, Mach. Learn. 98 (1–2) (2015) 157–180.
[9] A.L. Strehl, J. Ghosh, Cluster ensembles — a knowledge reuse framework for combining multiple partitions, J. Mach. Learn. Res. 3 (3) (2003) 583–617.
[10] S. Vegapons, J. Correamorris, J. Ruizshulcloper, Weighted partition consensus via kernels, Pattern Recognit. 43 (8) (2010) 2712–2724.
[11] Z. Yu, H. Wong, J. You, G. Yu, G. Han, Hybrid cluster ensemble framework based on the random combination of data transformation operators, Pattern Recognit. 45 (5) (2012) 1826–1837.
[12] A. Gionis, H. Mannila, P. Tsaparas, Clustering aggregation, ACM Trans. Knowl. Discov. Data 1 (1) (2007) 4.
[13] A. Topchy, A.K. Jain, W.F. Punch, Clustering ensembles: models of consensus and weak partitions, IEEE Trans. Pattern Anal. Mach. Intell. 27 (12) (2005) 1866–1881.
[14] A.L. Fred, A.K. Jain, Combining multiple clusterings using evidence accumulation, IEEE Trans. Pattern Anal. Mach. Intell. 27 (6) (2005) 835–850.

[15] J. Hu, T. Li, H. Wang, H. Fujita, Hierarchical cluster ensemble model based on knowledge granulation, Knowl.-Based Syst. 91 (2016) 179–188.
[16] S. Huang, H. Wang, D. Li, Y. Yang, T. Li, Spectral co-clustering ensemble, Knowl.-Based Syst. 84 (2015) 46–55.
[17] J. Wu, H. Liu, H. Xiong, J. Cao, J. Chen, K-means-based consensus clustering: a unified view, IEEE Trans. Knowl. Data Eng. 27 (1) (2015) 155–169.
[18] D. Huang, J. Lai, C.-D. Wang, Ensemble clustering using factor graph, Pattern Recognit. 50 (2016) 131–142.
[19] X.Z. Fern, C.E. Brodley, Solving cluster ensemble problems by bipartite graph partitioning, in: Proceedings of the Twenty-First International Conference on Machine Learning, ACM, 2004, p. 36.
[20] C. Claudio, R. Giovanni, Consensus clustering based on a new probabilistic rand index with application to subtopic retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 34 (12) (2012) 2315–2326.
[21] L. Du, Y.-D. Shen, Z. Shen, J. Wang, Z. Xu, A self-supervised framework for clustering ensemble, in: Proceedings of the International Conference on Web-Age Information Management, Springer, 2013, pp. 253–264.
[22] D. Huang, J.-H. Lai, C.-D. Wang, Robust ensemble clustering using probability trajectories, IEEE Trans. Knowl. Data Eng. 28 (5) (2016) 1312–1326.
[23] Z. Lu, Y. Peng, J. Xiao, From comparing clusterings to combining clusterings, in: Proceedings of the Twenty-Third National Conference on Artificial Intelligence, 2008, pp. 665–670.
[24] V. Singh, L. Mukherjee, J. Peng, J. Xu, Ensemble clustering using semidefinite programming with applications, Mach. Learn. 79 (1–2) (2010) 177–200.
[25] A. Rodriguez, A. Laio, Clustering by fast search and find of density peaks, Science 344 (6191) (2014) 1492–1496.
[26] B.J. Jain, The mean partition theorem in consensus clustering, Pattern Recognit. 79 (2018) 427–439.
[27] F. Li, Y. Qian, J. Wang, J. Liang, Multigranulation information fusion: a Dempster-Shafer evidence theory-based clustering ensemble method, Inf. Sci. 378 (2017) 389–409.
[28] C. Domeniconi, M. Alrazgan, Weighted cluster ensembles: methods and analysis, ACM Trans. Knowl. Discov. Data 2 (4) (2009) 17.
[29] X.Z. Fern, W. Lin, Cluster ensemble selection, Stat. Anal. Data Min. 1 (3) (2008) 128–141.
[30] L.I. Kuncheva, D.P. Vetrov, Evaluation of stability of k-means cluster ensembles with respect to random initialization, IEEE Trans. Pattern Anal. Mach. Intell. 28 (11) (2006) 1798–1808.
[31] L.I. Kuncheva, S.T. Hadjitodorov, Using diversity in cluster ensembles, in: Proceedings of the 2004 IEEE International Conference on Systems, Man and Cybernetics, vol. 2, IEEE, 2004, pp. 1214–1219.
[32] B. Fischer, J.M. Buhmann, Bagging for path-based clustering, IEEE Trans. Pattern Anal. Mach. Intell. 25 (11) (2003) 1411–1415.
[33] F. Yang, X. Li, Q. Li, T. Li, Exploring the diversity in cluster ensemble generation: random sampling and random projection, Expert Syst. Appl. 41 (10) (2014) 4844–4866.
[34] H.G. Ayad, M.S. Kamel, Cumulative voting consensus method for partitions with variable number of clusters, IEEE Trans. Pattern Anal. Mach. Intell. 30 (1) (2008) 160–173.
[35] H.G. Ayad, M.S. Kamel, On voting-based consensus of cluster ensembles, Pattern Recognit. 43 (5) (2010) 1943–1953.
[36] A.L. Blum, P. Langley, Selection of relevant features and examples in machine learning, Artif. Intell. 97 (1) (1997) 245–271.
[37] Y. Qian, J. Liang, W. Pedrycz, C. Dang, Positive approximation: an accelerator for attribute reduction in rough set theory, Artif. Intell. 174 (9–10) (2010) 597–618.
[38] X.Z. Fern, C.E. Brodley, Random projection for high dimensional data clustering: a cluster ensemble approach, in: Proceedings of the Twentieth International Conference on International Conference on Machine Learning, 2003, pp. 186–193.
[39] S.T. Hadjitodorov, L.I. Kuncheva, L.P. Todorova, Moderate diversity for better cluster ensembles, Inf. Fusion 7 (3) (2006) 264–275.
[40] F.J. Duarte, A.L.N. Fred, A. Lourenco, M.F. Rodrigues, Weighting cluster ensembles in evidence accumulation clustering, in: Proceedings of the 2005 Portuguese Conference on Artificial Intelligence, 2007, pp. 159–167.
[41] E. Akbari, H.M. Dahlan, R. Ibrahim, H. Alizadeh, Hierarchical cluster ensemble selection, Eng. Appl. Artif. Intell. 39 (2015) 146–156.
[42] J. Jia, X. Xiao, B. Liu, L. Jiao, Bagging-based spectral clustering ensemble selection, Pattern Recognit. Lett. 32 (10) (2011) 1456–1467.
[43] F. Li, Y. Qian, J. Wang, C. Dang, B. Liu, Cluster's quality evaluation and selective clustering ensemble, ACM Trans. Knowl. Discov. Data 12 (5) (2018) 60.
[44] P. Rastin, R. Kanawati, A multiplex-network based approach for clustering ensemble selection, in: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, IEEE, 2015, pp. 1332–1339.
[45] N. Iam-On, T. Boongoen, Comparative study of matrix refinement approaches for ensemble clustering, Mach. Learn. 98 (1–2) (2015) 269–300.
[46] E. Dimitriadou, A. Weingessel, K. Hornik, A combination scheme for fuzzy clustering, Int. J. Pattern Recognit. Artif. Intell. 16 (07) (2002) 901–912.
[47] Z.-H. Zhou, W. Tang, Clusterer ensemble, Knowl.-Based Syst. 19 (1) (2006) 77–83.
[48] V. Filkov, S. Skiena, Integrating microarray data by consensus clustering, Int. J. Artif. Intell. Tools 13 (04) (2004) 863–880.
[49] L. Franek, X. Jiang, Ensemble clustering by means of clustering embedding in vector spaces, Pattern Recognit. 47 (2) (2014) 833–842.
[50] N. Iam-On, T. Boongoen, S. Garrett, C. Price, A link-based approach to the cluster ensemble problem, IEEE Trans. Pattern Anal. Mach. Intell. 33 (12) (2011) 2396–2409.
[51] Y. Qian, F. Li, J. Liang, B. Liu, C. Dang, Space structure and clustering of categorical data, IEEE Trans. Neural Netw. Learn. Syst. 27 (10) (2016) 2047–2059.
[52] C. Zhong, X. Yue, Z. Zhang, J. Lei, A clustering ensemble: two-level-refined co-association matrix with path-based transformation, Pattern Recognit. 48 (8) (2015) 2699–2709.
[53] N. Otsu, A threshold selection method from gray-level histograms, Automatica 11 (285–296) (1975) 23–27.
[54] M. Sezgin, B. Sankur, Survey over image thresholding techniques and quantitative performance evaluation, J. Electron. Imaging 13 (1) (2004) 146–168.
[55] S.C. Johnson, Hierarchical clustering schemes, Psychometrika 32 (3) (1967) 241–254.
[56] M. Lichman, UCI machine learning repository, http://archive.ics.uci.edu/ml, 2013.
[57] M. Steinbach, G. Karypis, V. Kumar, et al., A comparison of document clustering techniques, in: Proceedings of the World Text Mining Conference Workshop, Boston, 2000, pp. 525–526.
[58] T.F. Chan, L.A. Vese, Active contours without edges, IEEE Trans. Image Process. 10 (2) (2001) 266–277.
[59] S. Osher, J.A. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations, J. Comput. Phys. 79 (1) (1988) 12–49.
[60] P. Arbelaez, M. Maire, C.C. Fowlkes, J. Malik, Contour detection and hierarchical image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (5) (2011) 898–916.
[61] A. Ultsch, Clustering with SOM: U*C, in: Proceedings of the 5th Workshop on Self-Organizing Maps, vol. 2, 2005, pp. 75–82.
[62] L. Fu, E. Medico, FLAME, a novel fuzzy clustering method for the analysis of DNA microarray data, BMC Bioinform. 8 (1) (2007) 3.
[63] A.K. Jain, M.H. Law, Data clustering: a user's dilemma, in: Proceedings of the International Conference on Pattern Recognition and Machine Intelligence, Springer, 2005, pp. 1–10.
[64] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.
[65] Y. Yang, An evaluation of statistical approaches to text categorization, Inf. Retr. 1 (1–2) (1999) 69–90.
[66] J. Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (2006) 1–30.