Applied Mathematics and Computation 381 (2020) 125280

Contents lists available at ScienceDirect

Applied Mathematics and Computation

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

Combining attribute content and label information for categorical data ensemble clustering

Liqin Yu[a], Fuyuan Cao[a,b,*], Xingwang Zhao[a], Xiaodan Yang[a], Jiye Liang[a,b]

[a] Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China
[b] Collaborative Innovation Center of Big Data Mining and Intelligence Technology, Shanxi University, Taiyuan 030006, China

ARTICLE INFO

Article history:
Received 3 June 2019
Revised 3 April 2020
Accepted 5 April 2020

Keywords:
Ensemble clustering
Information matrix
Original data information
Label information

ABSTRACT

Ensemble clustering has been attracting increasing attention in recent years, because it is able to combine multiple base clusterings (ensemble members) into a more robust clustering. It mainly consists of two parts, generating multiple ensemble members and finding a final partition. The construction of the information matrix plays an important role for finding a final partition. In general categorical data ensemble clustering framework, most existing information matrices are constructed only relying on label information of ensemble members without considering original information of data sets. To solve this problem, a new ensemble clustering framework for categorical data is proposed, in which the information matrix considers label information and original data information together, and is instantiated into the ALM matrix in this paper. The ALM matrix takes account of not only the distribution of attribute content in each ensemble member, but also the relationship among ensemble members based on the distribution. To simplicity, the *k*-means technique is used to cluster the ALM matrix and form a new ensemble clustering algorithm. The experimental results have shown the benefits of the ALM matrix by comparing the proposed algorithm with other ensemble clustering algorithms.

1. Introduction

Clustering analysis as an unsupervised learning method has been widely used in many real fields [1]. It aims to partition a given data set into a certain number of homogeneous groups, i.e, clusters. Many typical algorithms such as *k*-means [2], DBSCAN [3] have been proposed to cluster numerical data [4–6]. For categorical data, these algorithms can not be applied to cluster them directly because their attribute values are discrete and unorder.

To cluster categorical data, some algorithms have been proposed such as *k*-modes type algorithms [7–13], ROCK [14] and COOLCAT [15]. However, in most cases each categorical data clustering algorithm only can be applied to discover a kind of data structure and not performs best for all data. For a given categorical data set, different algorithms may produce different clustering results. Even though the same algorithm is used, results may also have a difference because of different parameters. Therefore, it is very difficult to find the most suitable clustering algorithm for a given categorical data set.

* Corresponding author at: Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China.
E-mail addresses: 1367545521@qq.com (L. Yu), cfy@sxu.edu.cn (F. Cao), zhaoxw84@163.com (X. Zhao), 784235753@qq.com (X. Yang), ljy@sxu.edu.cn (J. Liang).

https://doi.org/10.1016/j.amc.2020.125280
0096-3003/

Ensemble clustering has emerged as a powerful tool to discover the groups of a given categorical data set [16–19]. Compared with a single algorithm, ensemble clustering has better robustness, because multiple clusterings are combined to find a final partition. The multiple clusterings combined are referred as a set of base clusterings and each base clustering is also called as an ensemble member. Apparently, ensemble clustering consists of two parts, generating multiple ensemble members and finding a final partition. In the second part, the information matrix plays an important role, so its construction is of profound significance in ensemble clustering. The information matrix that describes the relationships between objects and clusters is the most widely used, because it is more simple and intuitive, and can be clustered by more kinds of algorithms.

Despite notable success of this kind of information matrix has been achieved in ensemble clustering, original information of data set has not been considered in general ensemble clustering framework. Most existing information matrices consider only label information of ensemble members. That is to say, after obtaining the set of base clusterings, the information matrix can be directly or indirectly obtained according to label information of ensemble members. That means the construction of the matrix only relies on the set of base clusterings. It is likely to find a final partition by the information matrix that is inconsistent with the original data structure.

To solve this problem, we propose a new ensemble clustering framework for categorical data, in which the information matrix is improved by considering original data information and label information together, and is instantiated into the ALM matrix. For clarity, we summarize the main contributions of this paper as follows.

- We propose a new ensemble clustering framework for categorical data, in which original data information and label information are combined to construct the information matrix. For the new framework, different methods can be proposed to combine original data information and label information, the set of base clusterings can be generated by different methods, and the information matrix can be clustered by different algorithms. The new framework can be instantiated into many ensemble clustering algorithms according to different requirements.
- We construct a new matrix ALM by instantiating information matrix of the new framework. The ALM matrix is constructed by combining attribute content and label information, considering not only the distribution of attribute content in each ensemble member, but also the relationship among ensemble members based on the distribution.

The rest of this paper is organized as follows. Related work is reviewed in Section 2. A new ensemble clustering framework for categorical data is proposed in Section 3. The ALM matrix is instantiated in Section 4. Experimental results on real data sets are reported in Section 5. The paper is concluded in Section 6.

2. Related work

Many methods have been proposed to construct an information matrix. To our best knowledge, there are three main types. The first kind is constructed as the relationships between objects and ensemble members. Each object is described by a vector formed by its labels in each ensemble members. The label-assignment matrix (LM) [20] is the most typical case. This type of matrix is generally dealt with to find a final partition by two ways. One is to cluster it by some categorical clustering algorithms. Another is to relabel objects that allows the homogeneous labels to be established from heterogeneous clustering results.

The second kind is constructed as the relationships between objects and objects. The pairwise similarity matrix (PM) [21] is the most typical case. It records co-occurrence statistics among objects in ensemble members. For objects that are not in the same cluster in all ensemble members, their similarities are not represented well. Therefore, Connected-Triple based similarity (CTS) matrix and SimRank based similarity (SRS) matrix are constructed to solve this problem [22]. Although the accuracy of the final partition has been improved, the two matrices are highly expensive to obtain [20] and the technique computing the similarities between objects, SimRank [23], is inapplicable to large data sets. For this type of matrix, some similarity-based algorithms (e.g., hierarchical clustering) can be used to find a final partition [24,25].

The third kind is constructed as the relationships between objects and clusters. The most typical case is the binary cluster-association matrix (BM) [26]. It records memberships of objects in clusters. If an object belongs to a cluster, the corresponding value is written as 1. Otherwise, it is written as 0. In this matrix, many potential relationships have not been discovered. So a link-based similarity algorithm was proposed, by which the relationships between objects and clusters that objects not belongs to are discovered and the refined cluster-association matrix (RM) is constructed [20]. This type of matrix can be clustered by traditional numeric and categorical algorithms, also can be processed by graph clustering algorithms like METIS [27] and SPEC [28].

Among the three main types, the third is the most widely used. Despite notable success of existing information matrices has been achieved in ensemble clustering, the original data information has not been considered in general ensemble clustering framework. Most existing information matrices only consider label information of ensemble members that makes the final partition sensitive to ensemble members. An example is listed to show the difficulties encountered by the existing information matrices. Suppose that $X = \{x_1, x_2, x_3, x_4, x_5, x_6\}$ is a categorical data set, it is described by three attributes $\{A_1, A_2, A_3\}$, as Table 1. The data set X can be clustered to generate a set of base clusterings $\Pi = \{\pi_1, \pi_2, \pi_3\}$, where $\pi_1 = \{c_1^1, c_2^1\}$, $\pi_2 = \{c_1^2, c_2^2, c_3^2\}$, and $\pi_3 = \{c_1^3, c_2^3\}$, consisting of 2, 3, 2 clusters respectively. The distribution of objects in each cluster is shown in Table 2. According to the set of base clusterings Π, the information matrices BM and RM can be obtained in Tables 3 and 4, without considering the information of original data set X.

Table 1
The data X.

objects	A_1	A_2	A_3
x_1	B	E	H
x_2	C	E	N
x_3	C	E	H
x_4	B	E	N
x_5	D	E	N
x_6	C	E	P

Table 2
A set of base clusterings of X.

members	*clusters*	*objects*
π_1	c_1^1	$\{x_1, x_4\}$
	c_2^1	$\{x_2, x_3, x_5, x_6\}$
π_2	c_1^2	$\{x_1, x_4\}$
	c_2^2	$\{x_2, x_3, x_6\}$
	c_3^2	$\{x_5\}$
π_3	c_1^3	$\{x_1, x_3\}$
	c_2^3	$\{x_2, x_4, x_5, x_6\}$

Table 3
The BM.

	c_1^1	c_2^1	c_1^2	c_2^2	c_3^2	c_1^3	c_2^3
x_1	1	0	1	0	0	1	0
x_2	0	1	0	1	0	0	1
x_3	0	1	0	1	0	1	0
x_4	1	0	1	0	0	0	1
x_5	0	1	0	0	1	0	1
x_6	0	1	0	1	0	0	1

Table 4
The RM.

	c_1^1	c_2^1	c_1^2	c_2^2	c_3^2	c_1^3	c_2^3
x_1	1.00	0.49	1.00	0.49	0.20	1.00	0.85
x_2	0.49	1.00	0.49	1.00	0.38	0.85	1.00
x_3	0.49	1.00	0.49	1.00	0.38	1.00	0.85
x_4	1.00	0.49	1.00	0.49	0.20	0.85	1.00
x_5	0.49	1.00	0.20	0.38	1.00	0.85	1.00
x_6	0.49	1.00	0.49	1.00	0.38	0.85	1.00

The information matrices are generally clustered to find a final partition, in which each column is taken as a feature. Therefore, it is supposed that the distance between two objects in the information matrix is as close as possible to their distance in the original data. We use $\Delta(x_i, x_j)$ to represent Euclidean distance between x_i and x_j. It is shown that the distance between x_2 and x_6 is computed as $\Delta(x_2, x_6) = 0$ in the BM and RM. However, in Table 1, $x_2 = (C, E, N)$, $x_6 = (C, E, P)$. Apparently, they are two different vectors, so their distance should not be 0. Again, in Table 1, $x_5 = (D, E, N), x_4 = (B, E, N), x_3 = (C, E, H)$. Obviously, x_5 is more similar to x_4 compared with x_3. But $\Delta(x_5, x_4) = \Delta(x_5, x_3)$ in the BM, and $\Delta(x_5, x_4) > \Delta(x_5, x_3)$ in the RM. They don't match the facts. Above all, the distances between objects in the BM and RM may be inconsistent with their distance in the original data because the information matrices only consider label information of ensemble members. In this paper, we propose a new ensemble clustering framework, in which original data information is added to improve the information matrix, and the matrix is instantiated into the ALM matrix.

3. A new ensemble clustering framework for categorical data

Different with numeric data, attribute values of categorical data are discrete and unorder. For a given categorical data set, its a general description is illustrated as follows. Suppose that $X = \{x_1, x_2, \ldots, x_n\}$ is a set of n objects described by m categorical attributes $\{A_1, A_2, \ldots, A_m\}$, where $x_i = (x_{i1}, x_{i2}, \ldots, x_{im})$ and x_{is} denotes the attribute value of x_i on A_s. Suppose that V^s represents the domain values of X on the attribute A_s. Obviously, $\bigcup_{i=1}^{n} x_{is} = V^s$. And for $\forall x_{ps}, x_{qs} \in V^s$, $x_{ps} = x_{qs}$ or $x_{ps} \neq x_{qs}$.

A new ensemble clustering framework for categorical data set is proposed, as shown in Fig. 1. Given a categorical data set X, it can be clustered to generate a set of base clusterings $\Pi = \{\pi_1, \pi_2, \ldots, \pi_M\}$, where $\pi_j = \{C_1^j, C_2^j, \ldots, C_{k_j}^j\}(1 \le j \le M)$

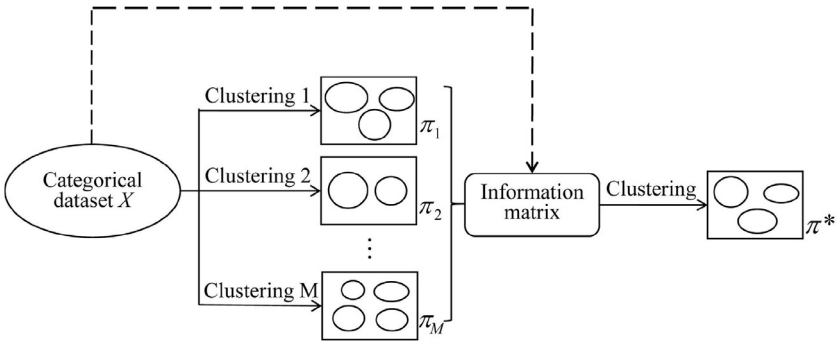

Fig. 1. The new ensemble clustering framework for categorical data.

denotes the jth base clustering or ensemble member, and k_j represents the number of clusters in the jth ensemble member. Then, the information matrix is constructed by combining the original data set X and the set of base clusterings Π, in which original data information and label information of ensemble members are considered together. The final partition π^* is found by clustering the information matrix.

In the new ensemble clustering framework, original data information is added to construct the information matrix, which has not been considered in general ensemble clustering framework (see dotted line in Fig. 1). For the new framework, different methods can be proposed to combine original data information and label information of ensemble members, different methods can be used to generate a set of base clusterings [7,29,30], and the information matrix can be clustered by different algorithms [2,4]. The new framework can be instantiated into many ensemble clustering algorithms according to different requirements. In this paper, we propose a method to combine original data information and label information of ensemble members, and the information matrix is instantiated into the ALM matrix.

4. The construction of the ALM matrix

To find a better final partition, the information matrix is supposed to consider the distribution of each ensemble member and the relationships among ensemble members together. For the ALM matrix, the attribute content is used in the two process to achieve the adding of original data information. The ALM matrix describes the similarities between objects and clusters. For a given object $x_i (x_i \in X)$ and a cluster $C_t^j (1 \le j \le M, 1 \le t \le k_j)$, we call the ensemble member $\pi_j (C_t^j \in \pi_j)$ as the current ensemble member. In the member π_j, the similarity between x_i and C_t^j is not only decided by C_t^j but also related to the relationship between x_i and $C_u^j (C_u^j \in \pi_j, C_u^j \neq C_t^j)$. Furthermore, in each ensemble member $\pi_r (1 \le r \le M)$, x_i can be allocated to a cluster $C_u^r (1 \le u \le k_r)$. The relationship between $C_u^r (x_i \in C_u^r, 1 \le r \le M, 1 \le u \le k_r)$ and C_t^j also affect the similarity between x_i and C_t^j. Therefore, the similarity can be obtained based on two parts: the current ensemble member (the current-member-based similarity) and all ensemble members (the all-members-based similarity).

4.1. The current-member-based similarity

To consider the information of original data set, for a given object, the importance of its attribute values in a cluster can be used to describe the current-member-based similarity between it and the cluster. The more important its attribute values are in a cluster, the higher the similarity between it and the cluster is. Therefore, the current-member-based similarity between an object and a cluster can be defined as follows.

Definition 1. Suppose that X is a data set described by m categorical attributes, it can be clustered to generate a set of base clusterings $\Pi = \{\pi_1, \pi_2, \ldots, \pi_M\}$, where $\pi_j = \{C_1^j, C_2^j, \ldots, C_{k_j}^j\}$ $(1 \le j \le M)$. $\forall x_i \in X$, the current-member-based similarity between x_i and C_t^j can be defined as

$$cur\_sim(x_i, C_t^j) = \frac{1}{m} \sum_{s=1}^{m} \omega(x_{is} | C_t^j), \tag{1}$$

where $t \in \{1, 2, \ldots, k_j\}$ and $\omega(x_{is}|C_t^j)$ represents the importance of x_{is} in C_t^j.

How to obtain the importance of attribute values in a cluster. In a given ensemble member, if an attribute value has higher frequency in a cluster and has lower frequency in the other clusters in the ensemble member, the attribute value is very important to the cluster [31]. Based on this insight, the importance of attribute values in clusters is defined as follows.

Definition 2. Suppose that X is a data set described by m categorical attributes, it can be clustered to generate a set of base clusterings $\Pi = \{\pi_1, \pi_2, \ldots, \pi_M\}$, where $\pi_j = \{C_1^j, C_2^j, \ldots, C_{k_j}^j\}$ $(1 \le j \le M)$. $\forall v \in V^s$, the importance of the attribute value v

in $C_t^j (t \in \{1, 2, \ldots, k_j\})$ can be defined as

$$\omega(v|C_t^j) = \frac{\sum_{\forall x_l \in C_t^j} \delta(v, x_{ls})}{|C_t^j|} \times f_{\pi_j}(v), \tag{2}$$

where

$$\delta(x, y) = \begin{cases} 1, & if \quad x == y, \\ 0, & otherwise, \end{cases}$$

and $|C_t^j|$ represents the number of objects in the cluster C_t^j. $f_{\pi_j}(v)$ represents the measurement of uncertainty of v in all clusters for π_j, and the formula is given as follows.

$$f_{\pi_j}(v) = 1 - \frac{-1}{\log k_j} \times \sum_{t=1}^{k_j} p(v|C_t^j) \log(p(v|C_t^j)), \tag{3}$$

where

$$p(v|C_t^j) = \frac{\sum_{\forall x_l \in C_t^j} \delta(v, x_{ls})}{\sum_{t=1}^{k_j} \sum_{\forall x_l \in C_t^j} \delta(v, x_{ls})}. \tag{4}$$

Obviously, $\sum_{\forall x_l \in C_t^j} \delta(v, x_{ls})$ in Eq. (2) is the number of the attribute value v in the cluster C_t^j. Therefore, $\frac{1}{|C_t^j|} \sum_{\forall x_l \in C_t^j} \delta(v, x_{ls})$ denotes the frequency of the attribute value v in the cluster C_t^j. If the frequency is higher, the attribute value v becomes more important in the cluster C_t^j. Moreover, $\sum_{t=1}^{k_j} \sum_{\forall x_l \in C_t^j} \delta(v, x_{ls})$ in Eq. (4) denotes the number of the attribute value v in all clusters of π_j. So the entropy $-\sum_{t=1}^{k_j} p(v|C_t^j) \log(p(v|C_t^j))$ in Eq. (3) is the probability distribution of v in all clusters of π_j. The bigger the entropy is, the closer the probability distribution of v in each cluster is. If the entropy is smaller and the frequency of v in C_t^j is higher, the frequency of v in the other clusters $C_u^j (C_u^j \in \pi_j, C_u^j \neq C_t^j)$ is smaller. That is, the attribute value v is very important to the given cluster C_t^j. As the range of the entropy is in $[0, \log k_j]$, we add a normalization factor $\frac{1}{\log k_j}$ in Eq. (3).

4.2. The all-members-based similarity

Given an object $x_i \in X$, in each ensemble member $\pi_r (1 \leq r \leq M)$, it can be allocated to a cluster $C_u^r (1 \leq u \leq k_r)$. When we compute the similarity between x_i and a given cluster C_t^j, if the distribution of $C_u^r (x_i \in C_u^r, 1 \leq r \leq M, 1 \leq u \leq k_r)$ and C_t^j is very similar, the similarity between x_i and C_t^j also becomes higher. Therefore, the similarity between $C_u^r (x_i \in C_u^r, 1 \leq r \leq M, 1 \leq u \leq k_r)$ and C_t^j is used to represent the all-members-based similarity between the object x_i and the cluster C_t^j, which is defined as follows.

Definition 3. Suppose that X is a data set described by m categorical attributes, it can be clustered to generate a set of base clusterings $\Pi = \{\pi_1, \pi_2, \ldots, \pi_M\}$, where $\pi_j = \{C_1^j, C_2^j, \ldots, C_{k_j}^j\}$ $(1 \leq j \leq M)$. $\forall x_i \in X$, the all-members-based similarity between x_i and C_t^j can be defined as

$$all\_sim(x_i, C_t^j) = \frac{1}{M} \sum_{r=1}^{M} s(C^r(x_i), C_t^j), \tag{5}$$

where $t \in \{1, 2, \ldots, k_j\}$. $C^r(x_i)$ denotes the cluster that x_i belongs to in π_r. $s(C^r(x_i), C_t^j)$ denotes the similarity between $C^r(x_i)$ and C_t^j.

How to compute the similarity between two clusters. To consider the information of original data sets, we use the distributions of attribute values in two clusters to describe the similarity. In the k-mw-modes algorithm [32] for clustering matrix-object data, the distance between two matrix-objects can be taken as a reference to compute the similarity between two clusters, because we can consider a matrix-object described by multiple records as a cluster. The similarity between two clusters is defined as follows.

Definition 4. Suppose that $C_t^j = \{x_1, x_2, \ldots, x_p\}$, $C_u^r = \{x_1, x_2, \ldots, x_q\}$ are two clusters described by m attributes, where $j, r \in \{1, 2, \ldots, M\}$, $1 \leq t \leq k_j$, $1 \leq u \leq k_r$. Let $V_{C_t^j}^s = (x_{1s}, x_{2s}, \ldots, x_{ps})'$, $V_{C_u^r}^s = (x_{1s}, x_{2s}, \ldots, x_{qs})'$ denote the values on the sth attribute of the two clusters. The similarity between C_t^j and C_u^r is defined as

$$s(C_t^j, C_u^r) = 1 - \frac{1}{m} \sum_{s=1}^{m} d(V_{C_t^j}^s, V_{C_u^r}^s), \tag{6}$$

Table 5
The ALM.

	c_1^1	c_2^1	c_1^2	c_2^2	c_3^2	c_1^3	c_2^3
x_1	0.3205	0.0159	0.3745	0.0387	0.0118	0.2804	0.0159
x_2	0.0235	0.2612	0.0152	0.3065	0.0149	0.0242	0.2451
x_3	0.0164	0.2259	0.0529	0.3303	0.0120	0.2737	0.0287
x_4	0.3222	0.0272	0.3098	0.0159	0.0147	0.0174	0.2002
x_5	0.0227	0.1000	0.0147	0.0176	0.2698	0.0136	0.3005
x_6	0.0156	0.3238	0.0152	0.4011	0.0149	0.0242	0.1046

where

$$d(V^s_{C_t^j}, V^s_{C_u^r}) = \frac{1}{2}\sum_{v\in V^s_{C_t^j}\cup V^s_{C_u^r}}\left|\frac{\sum_{i=1}^{p}\delta(v,x_{is})}{p} - \frac{\sum_{l=1}^{q}\delta(v,x_{ls})}{q}\right|,$$

and $|\cdot|$ represents the absolute value of a value. $\delta(\cdot,\cdot)$ is computed as the definition in Eq. (2).

Obviously, $\sum_{i=1}^{p}\delta(v,x_{is})$ denotes the number of the attribute value v in X_s, so $\frac{1}{p}\sum_{i=1}^{p}\delta(v,x_{is})$ is the frequency of v in X_s. For the given vectors $V^s_{C_t^j}, V^s_{C_u^r}$, v is from the set of their domain values. $|\frac{1}{p}\sum_{i=1}^{p}\delta(v,x_{is}) - \frac{1}{q}\sum_{l=1}^{q}\delta(v,x_{ls})|$ denotes the difference of the frequency of v in $V^s_{C_t^j}, V^s_{C_u^r}$. The smaller the difference is, the more similar the two vectors $V^s_{C_t^j}, V^s_{C_u^r}$ are. The differences of all domain values in $V^s_{C_t^j}, V^s_{C_u^r}$ are used to depict the distance $d(V^s_{C_t^j}, V^s_{C_u^r})$ between $V^s_{C_t^j}$ and $V^s_{C_u^r}$. As $0 \le d(V^s_{C_t^j}, V^s_{C_u^r}) \le 2$, we add a normalization factor $\frac{1}{2}$ in Eq. (6). We have $d(V^s_{C_t^j}, V^s_{C_u^r}) = 2$ when $V^s_{C_t^j} \cap V^s_{C_u^r} = \emptyset$.

4.3. The similarities between objects and clusters

According to Eqs. (1) and (5), the similarity between an object and a cluster is computed from two different viewpoints. The definition of the similarity between them can be given as follows.

Definition 5. Suppose that X is a data set described by m categorical attributes, it can be clustered to generate a set of base clusterings $\Pi = \{\pi_1, \pi_2, \ldots, \pi_M\}$, where $\pi_j = \{C_1^j, C_2^j, \ldots, C_{k_j}^j\}$ $(1 \le j \le M)$. $\forall x_i \in X$, the similarity between x_i and $C_t^j (1 \le t \le k_j)$ can be defined as

$$sim(x_i, C_t^j) = cur\_sim(x_i, C_t^j) \times all\_sim(x_i, C_t^j). \tag{7}$$

As $0 \le cur\_sim(x_i, C_t^j) \le 1$, $0 \le all\_sim(x_i, C_t^j) \le 1$, so $0 \le sim(x_i, C_t^j) \le 1$. In the similarity $sim(x_i, C_t^j)$, both the relationships based on the current ensemble member π_j and all ensemble members Π are considered. In each viewpoint, the information of original data is applied. With the similarity, the new information matrix ALM can be constructed in which each entry is the similarity between an object and a cluster.

The example in Section 2 is used to illustrate the construction of the ALM matrix. According to Tables 1 and 2, for the original data set X, we can compute the importance of its each attribute value in each cluster by Eq. (2). For example, the importance of the attribute value N in C_2^1 can be computed as

$$\begin{aligned}\omega(N|C_2^1) &= \tfrac{2}{4} \times f_{\pi_1}(N)\\ &= \tfrac{1}{2} \times (1 - \tfrac{-1}{\log 2} \times (\tfrac{1}{3}\log\tfrac{1}{3} + \tfrac{2}{3}\log\tfrac{2}{3}))\\ &\approx 0.0409.\end{aligned}$$

By this way, we can compute $\omega(C|C_2^1) = 0.75$, $\omega(E|C_2^1) \approx 0.0817$. Afterwards, the current-member-based similarity between x_2 and C_2^1 can be computed as $cur\_sim(x_2, C_2^1) = \frac{1}{3}(\omega(C|C_2^1) + \omega(E|C_2^1) + \omega(N|C_2^1)) \approx 0.2909$ by Eq. (1).

Furthermore, we can compute the similarity between any two clusters by Eq. (6). For C_2^1 and C_2^2, the set of their attribute values on A_1 are $\{C, D\}$, $\{C\}$ respectively, so their distance on A_1 can be computed as $\frac{1}{2}(|\frac{3}{4} - \frac{3}{3}| + |\frac{1}{4} - \frac{0}{3}|) = \frac{1}{4}$. Similarly, their distances on A_2, A_3 can be obtained as 0, $\frac{1}{6}$. Therefore, the distance between C_2^1 and C_2^2 can be obtained as $\frac{1}{3}(\frac{1}{4} + 0 + \frac{1}{6}) = \frac{5}{36}$. That is to say, $s(C_2^1, C_2^2) = 1 - \frac{5}{36} = \frac{31}{36}$. Similarly, $s(C_2^1, C_2^1) = 1$, $s(C_2^1, C_3^3) = \frac{5}{6}$. As x_2 is allocated into C_2^1, C_2^2, C_3^3 in π_1, π_2, π_3 respectively, the all-members-based similarity between x_2 and C_2^1 can be computed as $all\_sim(x_2, C_2^1) = \frac{1}{3}(s(C_2^1, C_2^1) + s(C_2^1, C_2^2) + s(C_2^1, C_2^3)) \approx 0.8981$ by Eq. (5).

Above all, the similarity between x_2 and C_2^1 can be computed as $sim(x_2, C_2^1) = cur\_sim(x_2, C_2^1) \times all\_sim(x_2, C_2^1) \approx 0.2612$ by Eq. (7). Table 5 is the ALM matrix constructed by the above way.

Compared with the BM and RM matrix, the relationships between objects in the ALM matrix may be closer to the relationships in original data set. For example, in Table 1, there are two different vectors, $x_2 = (C, E, N)$, $x_6 = (C, E, P)$. According to Section 2, $\Delta(x_2, x_6) = 0$ in the BM and RM that doesn't match the facts. In the ALM, their Euclidean distance is computed as $\Delta(x_2, x_6) = 0.21$. Again, $x_5 = (D, E, N), x_4 = (B, E, N), x_3 = (C, E, H)$. Obviously, x_5 is more similar to x_4 compared with x_3.

Table 6
The details of nine data sets.

Data sets	Number of objects	Number of attributes	Number of attribute values	Number of clusters
Zoo	101	16	36	7
Lymph	148	18	59	4
Soybean	307	35	132	19
PTumor	339	17	42	22
CVotes	435	16	48	2
BCancer	683	9	89	2
Mush	8124	22	117	2
Balance	625	4	20	3
Dema	357	34	189	6

But $\Delta(x_5, x_4) = \Delta(x_5, x_3)$ in the BM, and $\Delta(x_5, x_4) > \Delta(x_5, x_3)$ in the RM according to Section 2. They don't match the facts. However, in the ALM, $\Delta(x_5, x_4) < \Delta(x_5, x_3)$. That is because the BM and RM only consider label information of ensemble members. The ALM improves them by adding the attribute information of original data set, so a better partition may be obtained by the ALM. An algorithm of constructing the ALM is designed in Algorithm 1.

Algorithm 1 An algorithm of constructing the ALM.

Input:
- X: a categorical data set of n objects $\{x_1, x_2, \ldots, x_n\}$;
- Π: a set of base clusterings, $\Pi = \{\pi_1, \pi_2, \ldots, \pi_M\}$, and $\pi_j = \{C_1^j, C_2^j, \ldots, C_{k_j}^j\}(1 \le j \le M)$;

Output:- *ALM*: the new matrix that combines label information and attribute content;
Method:
$ALM = \emptyset$;
for $j = 1$ *to* M **do**
  $S = zeros(n, k_j)$;
  for $t = 1$ *to* k_j **do**
    for $i = 1$ *to* n **do**
      Compute the current-member-based similarity $cur\_sim(x_i, C_t^j)$ between x_i and C_t^j by Eq.(1);
      Compute the all-members-based similarity $all\_sim(x_i, C_t^j)$ between x_i and C_t^j by Eq.(5);
      $S(i, t) = cur\_sim(x_i, C_t^j) \times all\_sim(x_i, C_t^j)$;
    end for
  end for
  $ALM = [ALM, S]$;
end for
Return *ALM*.

The time complexity is analyzed as follows. For each attribute, the computation complexity of the similarity between two clusters is $\mathcal{O}(|V'|)$, and the computation complexity for the importance of attribute values in a cluster is $\mathcal{O}(k)$, where $|V'| = max\{|V^s|, 1 \le s \le m\}$, $k = max\{k_j, 1 \le j \le M\}$. Therefore, the time complexity of the algorithm is $\mathcal{O}(nmMk(|V'| + k))$ ($|V'| \gg k$), and can be simplified as $\mathcal{O}(nmMk|V'|)$.

5. Experiments on real data sets

In this section, we conduct some experiments on nine real data sets to validate the benefits of the ALM. Firstly, the details of nine data sets are given. Secondly, three evaluation indexes are introduced. Then, experimental setup is shown. Finally, comparison results of some ensemble clustering algorithms are reported.

5.1. Data sets

To evaluate the benefits of the ALM, some experiments are conducted on nine real data sets, Zoo, Lymphography, Soybean, Primary Tumor, Congressional Votes, Breast Cancer, Mushroom, Balance and Dematology. They all can be downloaded from UCI [33] For simplicity, they are simplified as Zoo, Lymph, Soybean, PTumor, CVotes, BCancer, Mush, Balance and Dema respectively. The details are listed in Table 6.

5.2. Evaluation indexes

We used the following three external criteria: (1) accuracy (AC), (2) adjusted rand index (ARI) [34], (3) normalized mutual information (NMI) [35] to measure the similarity between two partitions of objects in a given data set.

Table 7
The contingency table.

	C_1	C_2	$\cdots$	$C_{k'}$	*Sums*
P_1	n_{11}	n_{12}	$\cdots$	$n_{1k'}$	p_1
P_2	n_{21}	n_{22}	$\cdots$	$n_{2k'}$	p_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
P_k	n_{k1}	n_{k2}	$\cdots$	$n_{kk'}$	p_k
Sums	c_1	c_2	$\cdots$	$c_{k'}$	n

Let X be a categorical data set, $C = \{C_1, C_2, \ldots, C_{k'}\}$ be a clustering result of X, $P = \{P_1, P_2, \ldots, P_k\}$ be a real partition of X. The overlap between C and P can be summarized in a contingency table shown in Table 7, where n_{ij} denotes the number of objects in common between P_i and C_j, $n_{ij} = |P_i \bigcap C_j|$. p_i and c_j are the number of objects in P_i and C_j, respectively.

The three evaluation indexes are defined as follows:

$$AC = \frac{1}{n} \max_{j_1 j_2 \cdots j_k \in S} \sum_{i=1}^{k} n_{ij_i},$$

$$ARI = \frac{\sum_{ij} C_{n_{ij}}^2 - [\sum_i C_{p_i}^2 \sum_j C_{c_j}^2]/C_n^2}{\frac{1}{2}[\sum_i C_{p_i}^2 + \sum_j C_{c_j}^2] - [\sum_i C_{p_i}^2 \sum_j C_{c_j}^2]/C_n^2},$$

$$NMI = \frac{\sum_{i=1}^{k}\sum_{j=1}^{k'} n_{ij} log(\frac{n_{ij} n}{p_i c_j})}{\sqrt{\sum_{i=1}^{k} p_i log(\frac{p_i}{n}) \sum_{j=1}^{k'} c_j log(\frac{c_j}{n})}},$$

where $n_{1j_1^*} + n_{2j_2^*} + \cdots + n_{kj_k^*} = \max_{j_1 j_2 \cdots j_k \in S} \sum_{i=1}^{k} n_{ij_i}$ $(j_1^* j_2^* \ldots j_k^* \in S)$ and $S = \{j_1 j_2 \ldots j_k : j_1, j_2, \ldots, j_k \in \{1, 2, \ldots, k\},\ j_i \neq j_t \text{ for } i \neq t\}$ is a set of all permutations of $1, 2, \ldots, k$. In addition, we consider that the higher the values of *AC, ARI, NMI* are, the better the clustering solution is.

5.3. Experimental setup

In this experiment, in order to cluster the ALM, we apply the k-means technique into the new ensemble clustering framework, and form a new ensemble clustering algorithm named as KALM. To validate the benefits of the ALM, the same clustering technique, k-means, is used to cluster the BM [26] and the RM [20]. The corresponding ensemble clustering algorithms are called as KBM and KRM. As KALM, KBM and KRM only have a difference on the information matrix, the benefits of the ALM can be shown by comparing these three ensemble clustering algorithms. Moreover, the proposed algorithm KALM is also compared with other ensemble clustering algorithms, including CSPA [36], HGPA [36], MCLA [36], SEC [37].

To generate a set of base clusterings, the k-modes algorithm [7] is applied in this experiment. In this process, Full-space method and Sub-space method are used. They are described as follows.

Full-space method: For a given categorical data set, multiple different ensemble members can be generated by the k-modes algorithm with random initial centers. In order to introduce an artificial instability to the k-modes, we employ two ways to obtain the number of clusters in each ensemble member [20]: (i) Fixed-k, $k = K$ (where K is the number of real clusters), (ii) Random-k, $k \in \{2, 3, \ldots, \lceil\sqrt{n}\rceil\}$.

Sub-space method: In this method, a given categorical data set is clustered on its different subspaces to generate multiple ensemble members [21]. Similar to the study of [38], for a given data set of n objects described by m attributes, a subspace described by q attributes can be generated by $q = q_{\min} + \lfloor \alpha(q_{\max} - q_{\min}) \rfloor$, where $\alpha \in [0, 1]$ is an uniform random variable, $q_{\min}$ and $q_{\max}$ represent the lower and upper bounds of the subspace respectively. They usually are set to 0.75m and 0.85m. We select q attributes sequently from m attributes. In each selection, we randomly select the hth attribute of m attributes as an attribute in the subspace, $h = \lfloor 1 + \beta m \rfloor$, and $\beta \in [0, 1)$ is also an uniform random variable. The k-modes algorithm is still used to obtain a set of base clusterings, and the two ways of obtaining the number of clusters are still used.

In this experiment, we run the k-modes algorithm 10 times to generate a set of base clusterings containing 10 ensemble members for the full-space and sub-space methods. Then, those compared ensemble clustering algorithms are executed 30 times and the average values are taken as the final results to guarantee the credibility of experiments. To avoid the influence of the set of base clusterings, the above process is repeated 30 times.

5.4. Experimental results

The comparison results of the seven algorithms on the nine data sets are shown inTables 8–10. The values of evaluation indexes obtained by these algorithms are ranked for each data set, the most highest value getting the rank 1, the second

Table 8
Comparison results of the ensemble algorithms on AC.

	Full-space(i)							Full-space(ii)						
Datasets	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC
Zoo	0.7945(1)	0.6728(5)	0.6730(4)	0.5812(7)	0.5902(6)	0.7211(2)	0.6809(3)	0.8000(1)	0.6780(4)	0.6769(5)	0.5805(6)	0.5753(7)	0.7063(2)	0.6791(3)
Lymph	0.4532(4)	0.4543(3)	0.4565(2)	0.4189(6)	0.3226(7)	0.4417(5)	0.4559(1)	0.4996(1)	0.4637(5)	0.4835(2)	0.4351(6)	0.3910(7)	0.4721(3)	0.4663(4)
Soybean	0.6177(1)	0.4951(3)	0.5458(2)	0.4417(7)	0.4792(6)	0.4833(5)	0.4840(4)	0.6102(1)	0.4626(3)	0.4995(2)	0.4100(6)	0.4278(5)	0.1964(7)	0.4592(4)
PTumor	0.3128(1)	0.3029(5)	0.3072(3)	0.2793(6)	0.2673(7)	0.3056(4)	0.3115(2)	0.3053(2.5)	0.3045(4)	0.3053(2.5)	0.2652(5)	0.2565(6)	0.2544(7)	0.3099(1)
CVotes	0.8657(1)	0.8622(3.5)	0.8615(5)	0.8530(6)	0.5333(7)	0.8622(3.5)	0.8633(2)	0.8757(1)	0.8664(4)	0.8750(2)	0.8518(6)	0.8684(3)	0.8535(5)	0.8467(7)
BCancer	0.9460(1)	0.9226(4)	0.9270(3)	0.8097(6)	0.5038(7)	0.9281(2)	0.9065(5)	0.9395(1)	0.8290(6)	0.9283(2)	0.8141(7)	0.8380(4)	0.8929(3)	0.8376(5)
Mush	0.8546(1)	0.7851(3)	0.7597(5)	0.7556(6)	0.5052(7)	0.8038(2)	0.7725(4)	0.7269(3)	0.6693(5)	0.7221(4)	0.7815(1)	0.5236(7)	0.7470(2)	0.6684(6)
Balance	0.4356(1)	0.4204(5)	0.4347(2)	0.4022(7)	0.4025(6)	0.4235(3)	0.4232(4)	0.4572(1)	0.4297(2)	0.4237(4)	0.4034(7)	0.4039(6)	0.4193(5)	0.4264(3)
Dema	0.7047(1)	0.6621(2)	0.6272(6)	0.6276(5)	0.5350(7)	0.6538(4)	0.6595(3)	0.7178(1)	0.6710(4)	0.6742(3)	0.6426(7)	0.6481(6)	0.7149(2)	0.6675(5)
AvgR	**1.33**	3.72	3.56	6.22	6.67	3.39	3.11	**1.39**	4.11	2.94	5.67	5.67	4.25	4.38
	Sub-space(i)							Sub-space(ii)						
Datasets	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC
Zoo	0.7944(1)	0.6786(5)	0.6831(4)	0.5772(7)	0.5932(6)	0.6968(2)	0.6837(3)	0.8031(1)	0.6961(4)	0.7035(2)	0.5904(6)	0.5677(7)	0.6929(5)	0.6973(3)
Lymph	0.4624(1)	0.4450(4)	0.4424(5)	0.4239(6)	0.3240(7)	0.4560(2)	0.4491(3)	0.4881(1)	0.4452(5)	0.4693(2)	0.4232(6)	0.3725(7)	0.4487(3)	0.4484(4)
Soybean	0.6160(1)	0.4917(3)	0.5392(2)	0.4440(7)	0.4779(6)	0.4846(4)	0.4813(5)	0.6093(1)	0.4680(3)	0.5007(2)	0.4064(6)	0.4202(5)	0.2187(7)	0.4613(4)
PTumor	0.3050(1)	0.2863(4)	0.2925(3)	0.2667(6)	0.2559(7)	0.2815(5)	0.2928(2)	0.2996(1)	0.2905(4)	0.2932(3)	0.2616(5)	0.2466(7)	0.2516(6)	0.2935(2)
CVotes	0.8676(1)	0.8618(3)	0.8609(5)	0.8530(6)	0.5333(7)	0.8616(4)	0.8638(2)	0.8744(1)	0.8696(3)	0.8697(2)	0.8540(5)	0.8515(7)	0.8577(4)	0.8517(6)
BCancer	0.9446(1)	0.9023(3)	0.8056(6)	0.8248(5)	0.5044(7)	0.9310(2)	0.8842(4)	0.9404(1)	0.8334(5)	0.9017(3)	0.8225(7)	0.8408(4)	0.9178(2)	0.8289(6)
Mush	0.8496(1)	0.7864(3)	0.7690(6)	0.7778(5)	0.5052(7)	0.8135(2)	0.7835(4)	0.7215(4)	0.6710(5)	0.7278(3)	0.7438(2)	0.5267(7)	0.7447(1)	0.6668(6)
Balance	0.4507(2)	0.4272(5)	0.4603(1)	0.4101(6)	0.4033(7)	0.4432(3)	0.4296(4)	0.4641(1)	0.4422(3)	0.4564(2)	0.4145(6)	0.4143(7)	0.4234(5)	0.4415(4)
Dema	0.6746(1)	0.6535(3)	0.6237(5)	0.6133(6)	0.5281(7)	0.6569(2)	0.6516(4)	0.7106(2)	0.6682(5)	0.6701(4)	0.6407(6)	0.6036(7)	0.7199(1)	0.6705(3)
AvgR	**1.11**	3.67	4.11	6	6.78	2.89	3.44	**1.44**	4.11	2.56	5.44	6.44	3.78	4.22

Table 9
Comparison results of the ensemble algorithms on ARI.

	Full-space(i)							Full-space(ii)						
Datasets	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC
Zoo	0.7606(1)	0.6092(5)	0.6121(4)	0.4636(7)	0.4640(6)	0.6543(2)	0.6182(3)	0.7688(1)	0.6090(5)	0.6149(3)	0.4592(6)	0.4533(7)	0.6409(2)	0.6135(4)
Lymph	0.1312(1)	0.1201(2)	0.1094(5)	0.1014(6)	0.0134(7)	0.1110(4)	0.1152(3)	0.1697(1)	0.1255(4)	0.1485(2)	0.1198(6)	0.0771(7)	0.1428(3)	0.1216(5)
Soybean	0.4408(1)	0.3564(3)	0.3850(2)	0.2933(7)	0.3170(6)	0.3261(5)	0.3441(4)	0.4372(1)	0.3237(3)	0.3448(2)	0.2553(5)	0.2549(6)	0.0527(7)	0.3173(4)
PTumor	0.1318(1)	0.1240(3)	0.1270(2)	0.1084(6)	0.0991(7)	0.1202(5)	0.1234(4)	0.1251(1)	0.1200(3)	0.1207(2)	0.0987(5)	0.0860(6)	0.0197(7)	0.1184(4)
CVotes	0.5338(1)	0.5237(3.5)	0.5217(5)	0.4975(6)	0.0023(7)	0.5237(3.5)	0.5270(2)	0.5636(1)	0.5468(3)	0.5618(2)	0.4942(7)	0.5424(4)	0.4988(6)	0.5063(5)
BCancer	0.7947(1)	0.7266(4)	0.7447(2)	0.3832(6)	-0.0013(7)	0.7325(3)	0.6809(5)	0.7717(1)	0.5189(5)	0.7671(2)	0.3969(7)	0.4566(6)	0.6253(3)	0.5324(4)
Mush	0.5320(1)	0.3992(3)	0.3562(5)	0.2784(6)	0.0000(7)	0.4151(2)	0.3760(4)	0.2292(3)	0.1394(5)	0.2145(4)	0.3350(1)	0.0034(7)	0.2883(2)	0.1384(6)
Balance	0.0301(2)	0.0277(3)	0.0250(6)	0.0254(5)	0.0318(1)	0.0230(7)	0.0269(4)	0.0400(2)	0.0250(4.5)	0.0217(7)	0.0258(3)	0.0250(4.5)	0.0286(2)	0.0243(6)
Dema	0.6418(1)	0.5660(2)	0.5122(5)	0.5036(6)	0.4080(7)	0.5279(4)	0.5469(3)	0.6581(1)	0.5768(4)	0.5869(3)	0.5145(7)	0.5728(5)	0.6085(2)	0.5580(6)
AvgR	**1.11**	3.17	4	6.11	6.11	3.94	3.56	**1.33**	4.06	3	5.22	5.83	3.78	4.89
	Sub-space(i)							Sub-space(ii)						
Datasets	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC
Zoo	0.7630(1)	0.6181(5)	0.6340(2)	0.4558(7)	0.4677(6)	0.6240(3)	0.6214(4)	0.7731(1)	0.6334(4)	0.6482(2)	0.4584(6)	0.4487(7)	0.6233(5)	0.6360(3)
Lymph	0.1392(1)	0.1192(3)	0.0947(6)	0.1128(5)	0.0140(7)	0.1208(2)	0.1172(4)	0.1613(1)	0.1168(4)	0.1368(2)	0.1132(5)	0.0624(7)	0.1296(3)	0.1130(6)
Soybean	0.4419(1)	0.3519(3)	0.3816(2)	0.2908(7)	0.3087(6)	0.3253(5)	0.3380(4)	0.4405(1)	0.3287(3)	0.3448(2)	0.2493(5)	0.2465(6)	0.0724(7)	0.3188(4)
PTumor	0.1260(1)	0.1124(3)	0.1162(2)	0.0985(5)	0.0858(7)	0.0922(6)	0.1069(4)	0.1189(1)	0.1082(3)	0.1116(2)	0.0920(5)	0.0778(6)	0.0153(7)	0.1038(4)
CVotes	0.5395(1)	0.5227(3)	0.5206(5)	0.4976(6)	0.0023(7)	0.5219(4)	0.5283(2)	0.5596(1)	0.5510(2)	0.5460(3)	0.5007(6)	0.4964(7)	0.5110(5)	0.5120(4)
BCancer	0.7896(1)	0.6892(3)	0.4796(5)	0.4236(6)	-0.0012(7)	0.7407(2)	0.6318(4)	0.7747(1)	0.5318(4)	0.7060(2)	0.4218(7)	0.4640(6)	0.7001(3)	0.5088(5)
Mush	0.5139(1)	0.3816(3)	0.3458(5)	0.3260(6)	0.0000(7)	0.4220(2)	0.3710(4)	0.2167(4)	0.1400(5)	0.2266(3)	0.2947(1)	0.0048(7)	0.2867(2)	0.1360(6)
Balance	0.0424(1)	0.0357(3)	0.0335(5)	0.0316(6)	0.0287(7)	0.0402(2)	0.0338(4)	0.0438(1)	0.0375(3)	0.0387(2)	0.0347(5)	0.0334(6)	0.0313(7)	0.0358(4)
Dema	0.6104(1)	0.5551(2)	0.5131(5)	0.4961(6)	0.3995(7)	0.5318(4)	0.5382(3)	0.6530(1)	0.5783(4)	0.5816(3)	0.5167(7)	0.5200(6)	0.6151(2)	0.5684(5)
AvgR	**1**	3.11	4.11	6	6.78	3.33	3.67	**1.33**	3.56	2.33	5.22	6.44	4.56	4.56

Table 10
Comparison results of the ensemble algorithms on NMI.

	Full-space(i)							Full-space(ii)						
Datasets	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC
Zoo	0.8472(1)	0.7679(5)	0.7735(3)	0.6969(6)	0.6890(7)	0.7787(2)	0.7692(4)	0.8476(1)	0.7692(4)	0.7718(2)	0.6945(6)	0.6882(7)	0.7703(3)	0.7664(5)
Lymph	0.1894(1)	0.1767(2)	0.1571(5)	0.1460(6)	0.0554(7)	0.1613(4)	0.1709(3)	0.2248(1)	0.1787(4)	0.1952(2)	0.1647(6)	0.1210(7)	0.1897(3)	0.1741(5)
Soybean	0.7627(1)	0.6545(3)	0.6945(2)	0.6086(7)	0.6443(4)	0.6345(6)	0.6399(5)	0.7475(1)	0.6148(3)	0.6477(2)	0.5664(6)	0.5779(5)	0.1264(7)	0.6052(4)
PTumor	0.3983(1)	0.3843(3)	0.3866(2)	0.3743(6)	0.3614(7)	0.3751(5)	0.3778(4)	0.3867(1)	0.3729(3)	0.3756(2)	0.3610(5)	0.3437(6)	0.0787(7)	0.3656(4)
CVotes	0.4657(1)	0.4489(5)	0.4521(2)	0.4508(3)	0.0034(7)	0.4488(6)	0.4506(4)	0.4770(2)	0.4666(4)	0.4820(1)	0.4473(5)	0.4675(3)	0.4467(6)	0.4402(7)
BCancer	0.7203(1)	0.6306(4)	0.6446(2)	0.3623(6)	0.0001(7)	0.6322(3)	0.5949(5)	0.7009(1)	0.4969(4)	0.6868(2)	0.3822(7)	0.4564(6)	0.5443(3)	0.4937(5)
Mush	0.4915(1)	0.3657(3)	0.3265(5)	0.2142(6)	0.0001(7)	0.3706(2)	0.3492(4)	0.3156(1)	0.1810(5)	0.2705(2)	0.2606(3)	0.0025(7)	0.2246(4)	0.1778(6)
Balance	0.0296(2)	0.0261(4)	0.0268(3)	0.0228(6)	0.0348(1)	0.0213(7)	0.0253(5)	0.0422(1)	0.0242(3)	0.0216(7)	0.0231(5)	0.0225(6)	0.0265(2)	0.0237(4)
Dema	0.7614(1)	0.6473(3)	0.6767(2)	0.5969(6)	0.5380(7)	0.6121(5)	0.6354(4)	0.7889(1)	0.6693(5)	0.7147(2)	0.6109(7)	0.6902(3)	0.6851(4)	0.6588(6)
AvgR	**1.11**	3.56	2.89	5.78	6	4.44	4.22	**1.11**	3.89	2.44	5.56	5.56	4.33	5.11
	Sub-space(i)							Sub-space(ii)						
Datasets	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC
Zoo	0.8485(1)	0.7812(3)	0.7916(2)	0.6906(7)	0.7007(6)	0.7658(5)	0.7765(4)	0.8477(1)	0.7769(3)	0.7888(2)	0.6909(6)	0.6898(7)	0.7469(5)	0.7749(4)
Lymph	0.2020(1)	0.1707(2)	0.1524(6)	0.1604(5)	0.0577(7)	0.1665(4)	0.1682(3)	0.2141(1)	0.1686(4)	0.1826(2)	0.1584(6)	0.1047(7)	0.1775(3)	0.1640(5)
Soybean	0.7605(1)	0.6520(3)	0.6957(2)	0.6078(7)	0.6330(5)	0.6295(6)	0.6347(4)	0.7492(1)	0.6211(3)	0.6516(2)	0.5646(6)	0.5690(5)	0.1732(7)	0.6078(4)
PTumor	0.3913(1)	0.3734(3)	0.3759(2)	0.3618(4)	0.3439(7)	0.3457(6)	0.3614(5)	0.3813(1)	0.3654(3)	0.3689(2)	0.3548(5)	0.3328(6)	0.0704(7)	0.3559(4)
CVotes	0.4754(1)	0.4489(5)	0.4480(6)	0.4512(2)	0.0034(7)	0.4495(4)	0.4503(3)	0.4732(1)	0.4663(3)	0.4691(2)	0.4551(4)	0.4360(7)	0.4526(5)	0.4465(6)
BCancer	0.7174(1)	0.6025(3)	0.4471(5)	0.4134(6)	0.0001(7)	0.6342(2)	0.5553(4)	0.7051(1)	0.5075(4)	0.6553(2)	0.4175(7)	0.4729(5)	0.5942(3)	0.4721(6)
Mush	0.4766(1)	0.3496(4)	0.3207(5)	0.2533(6)	0.0001(7)	0.3781(2)	0.3529(3)	0.3082(1)	0.1922(5)	0.2712(2)	0.2304(3)	0.0036(7)	0.2234(4)	0.1851(6)
Balance	0.0453(1)	0.0325(5)	0.0348(3)	0.0277(7)	0.0345(4)	0.0369(2)	0.0318(6)	0.0454(1)	0.0347(3)	0.0388(2)	0.0303(5)	0.0293(6)	0.0284(7)	0.0342(4)
Dema	0.7418(1)	0.6351(3)	0.6645(2)	0.5891(6)	0.5250(7)	0.6098(5)	0.6241(4)	0.7821(1)	0.6634(4)	0.7185(2)	0.6110(7)	0.6475(6)	0.6732(3)	0.6513(5)
AvgR	**1**	3.44	3.67	5.56	6.33	4	4	**1**	3.56	2	5.44	6.22	4.89	4.89

Table 11
The average rank R of the seven compared algorithms.

Algorithm	KALM	KBM	KRM	CSPA	HGPA	MCLA	SEC
R	**1.19**	3.67	3.13	5.69	6.24	3.97	4.17

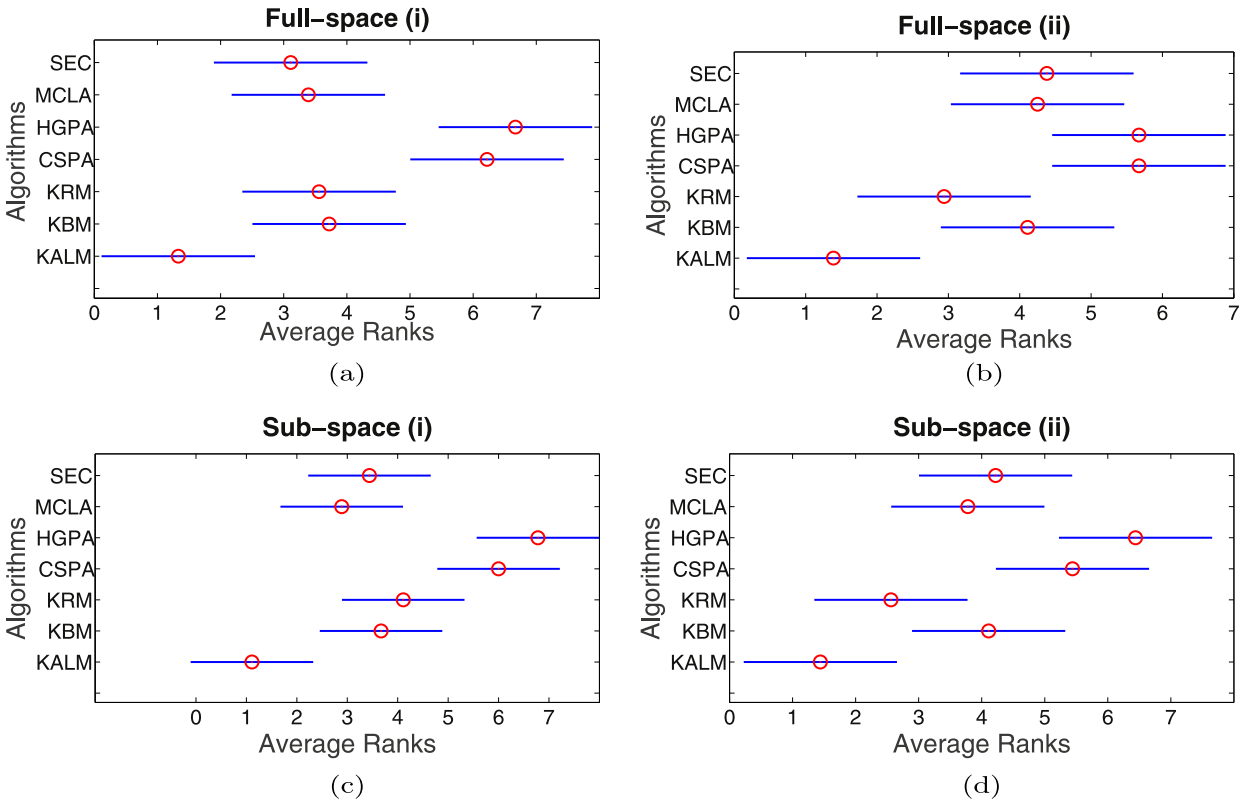

Fig. 2. The Bonferroni-Dunn test of the KALM algorithm in terms of AC.

higher value getting rank 2, ..., as shown in the parentheses in Tables 8–10. AvgR represents the average rank of these algorithms on nine data sets. From Tables 8–10, we can see that AvgR of the KALM ranks the first on all evaluation indexes. That is to say, the KALM outperforms other compared algorithms in general.

In detail, when the number of clusters is equal to K (K is the number of real clusters), we can see that the KALM algorithm is better than the other ensemble clustering algorithms on eight data sets at least for all evaluation indexes. When the number of clusters is random, there are seven data sets at least for all evaluation indexes. Particularly, for the index NMI, the KALM algorithm is better than the other ensemble clustering algorithms on all data sets under the condition of Sub-space method. For Full-space method, there are also eight data sets.

To give a comprehensive comparison [39], we use the Friedman test and Bonferroni-Dunn test [40] to analyze the differences of the seven compared algorithms. According to AvgR we can get the average rank R of these algorithms for all cases. Suppose that r_i^j represents the rank of the jth algorithm on the ith case, the average rank of the jth algorithm $R_j = \frac{1}{B}\sum_{i=1}^{B} r_i^j$, where B represent the number of cases. The Friedman test compares the average ranks of algorithms. As there are $A = 7$ algorithms and $B = 12$ cases (i.e., 4 generating ensemble member methods and 3 external criterions), the average ranks of the seven algorithms can be computed shown in Table 11. According to the average ranks R, we still know the KALM is better than the other six algorithms.

Under the null-hypothesis, all algorithms are equivalent and so their ranks should be equal (i.e., $R_j = 4$ for four algorithms). The Friedman test aims to check whether the measured average ranks are significantly different from the mean rank $R_j = 4$ expected under the null-hypothesis:

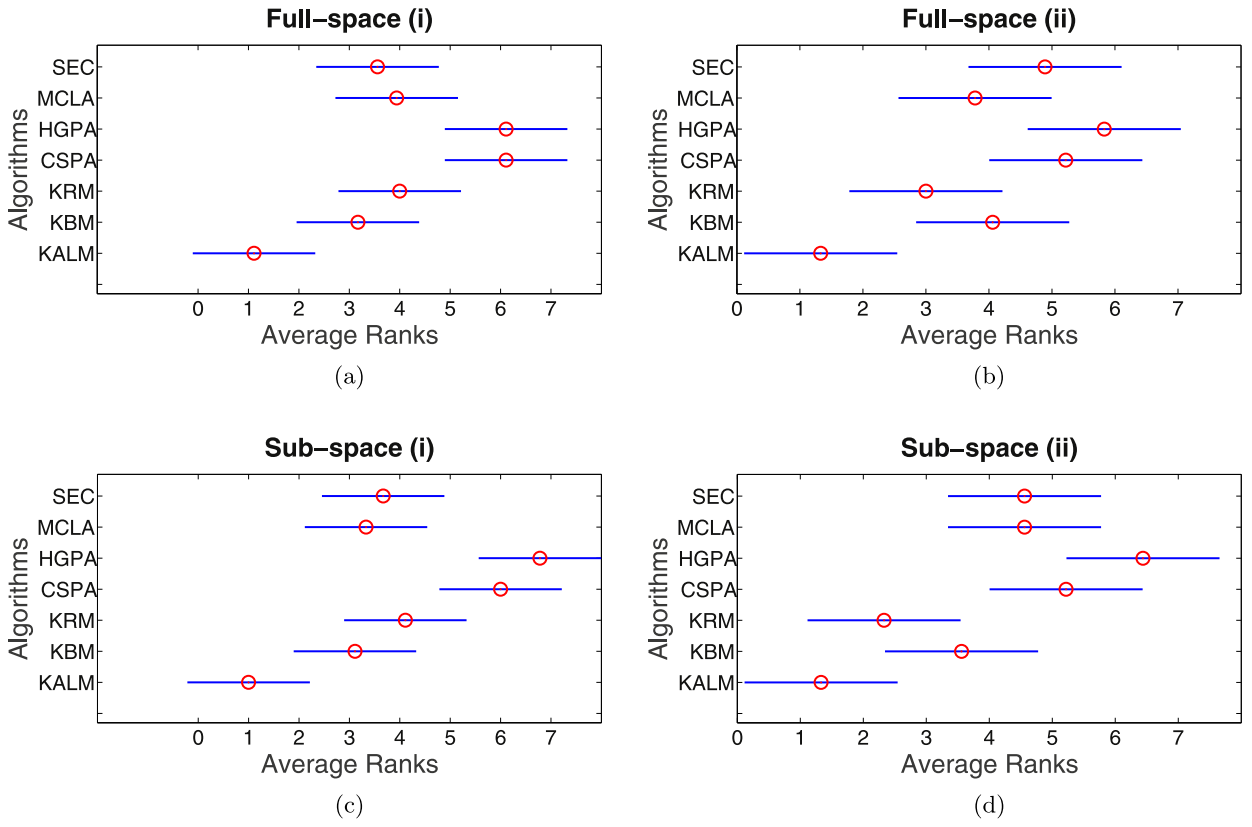

Fig. 3. The Bonferroni-Dunn test of the KALM algorithm in terms of ARI.

$$\begin{aligned}\chi_F^2 &= \frac{12B}{A(A+1)}\left[\sum_{j=1}^{A} R_j^2 - \frac{A(A+1)^2}{4}\right] \\ &= \frac{12\cdot 12}{7\cdot(7+1)}\left[1.19^2 + 3.67^2 + 3.13^2 + 5.69^2 + 6.24^2 + 3.97^2 + 4.17^2 - \frac{7\cdot(7+1)^2}{4}\right] \\ &= 44.09.\end{aligned}$$

With the seven algorithms, the Friedman statistic is distributed according to the χ_F^2 distribution with $A-1=6$ degrees of freedom. The critical value for $\alpha = 0.1$ is $10.64 < 44.09$, so we reject the null-hypothesis and we think the compared seven algorithms have differences.

Then, we use the Bonferroni-Dunn test to reveal the differences. The critical value is 2.394 when we use $\alpha = 0.1$ according to [40]. So the critical difference for nine data sets can be computed as $CD = q_\alpha\sqrt{\frac{A(A+1)}{6N}} = 2.394\cdot\sqrt{\frac{7\cdot(7+1)}{6\cdot 9}} = 2.43$. N is the number of data sets. By the critical difference we can identify the algorithms for all cases. If the difference of the average rank AvgR is larger than the critical difference CD for two algorithms, we think they are significantly different. If the difference of AvgR is larger than the half of CD and is smaller than CD for two algorithms, we think they are comparable. If the difference of AvgR is close to 0 for two algorithms, we think they almost have not difference. Figs. 2–4 show the Bonferroni-Dunn test of the seven algorithms in terms of the three evaluation indexes. The circles represent the average ranks AvgR of algorithms and the length of the bar is the critical difference CD.

From Figs. 2–4, we can find that the KALM algorithm can be differentiated with the other six algorithms. In terms of AC, we can see from Fig. 2(a) that the KALM algorithm is significantly different from CSPA, HGPA while these two algorithms almost have not difference. Meanwhile, the KALM algorithm is comparable with KBM, KRM, MCLA, SEC while there aren't almost difference among these four algorithms. In Fig. 2(b)–(d), the KALM algorithm is significantly different from five algorithms at least. Therefore, we can think the KALM algorithm outperforms other compared algorithms. In the same way, the difference among compared algorithms can be analyzed in terms of ARI and NMI. In addition, the KALM algorithm ranks first on all cases according to Figs. 2–4. Above all, the KALM algorithm outperforms other compared algorithms because it has higher ranks and has some differences compared with those algorithms.

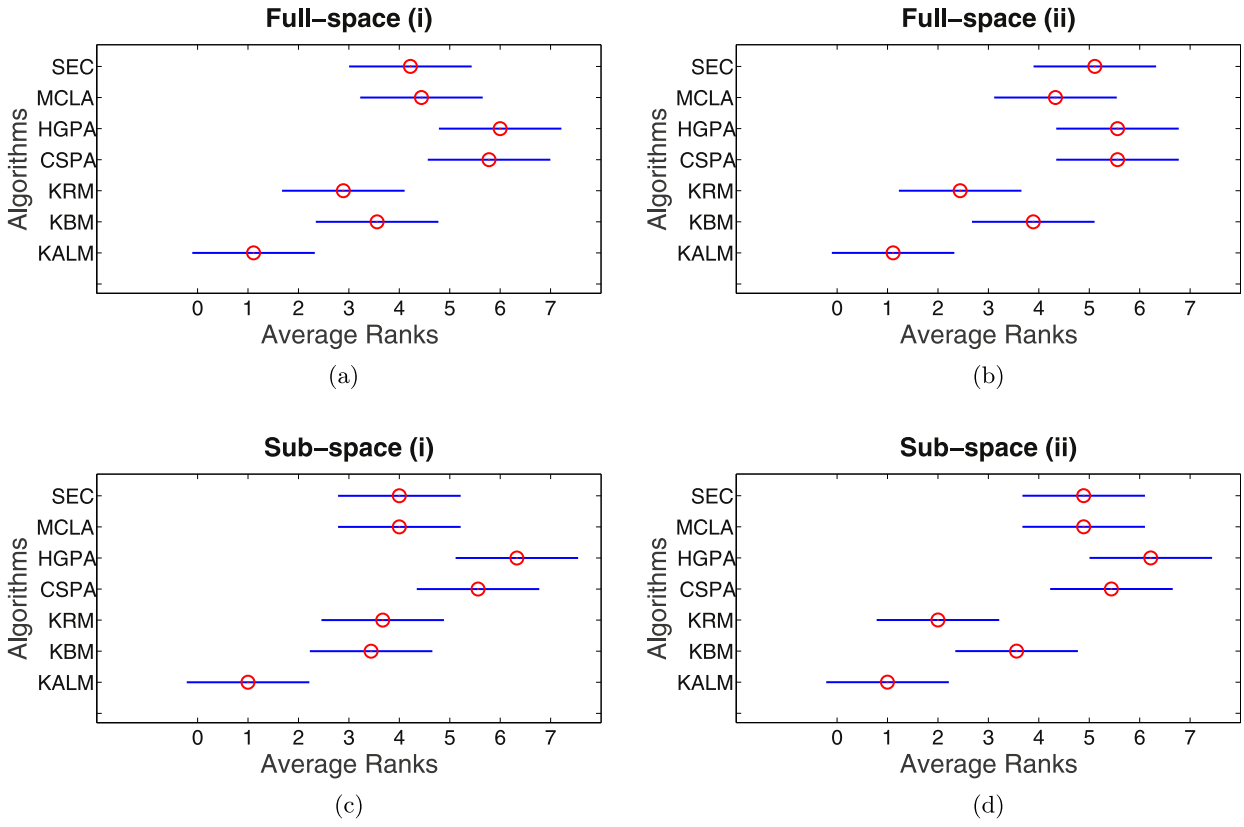

Fig. 4. The Bonferroni-Dunn test of the KALM algorithm in terms of NMI.

6. Conclusions and future work

In this paper, we proposed a new ensemble clustering framework for categorical data, in which original data information and label information of ensemble members are combined to construct the information matrix. For the new framework, different methods can be proposed to combine original data information and label information of ensemble members, different methods can be used to generate the set of base clusterings, and the information matrix can be clustered by different algorithms. The new framework can be instantiated as many ensemble clustering algorithms according to different requirements. Under this new framework, we proposed a new algorithm to construct the ALM matrix, in which the current-member-based similarities and the all-members-based similarities are defined. A better final partition is obtained by the ALM matrix, because it considers the distribution of attribute content in each ensemble member and the relationships among ensemble members based on the distribution. We carried out some experiments on real data sets and the results have shown that the benefits of the ALM matrix by comparing some ensemble clustering algorithms for categorical data. For other types of data, the new framework also can work by using corresponding object-to-cluster similarity measures to consider original data information. This contents will be studied in our future work.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (under grants 61976128, 61773247), the Innovation Project of excellent postgraduates in Shanxi Province (under grant 2019BY002), the Key Research and Development Projects of Shanxi Province (under grant 201803D31022), the Fund Program for the Scientific Activities of Selected Returned Overseas Professionals in Shanxi Province (under grant 2016-001), the Research Project Supported by Shanxi Scholarship Council of China (under grant 2016-003) and the 1331 Engineering Project of Shanxi Province, China.

References

[1] J. Han, M. Kamber, J. Pei, Data Mining Concept and Techniques, 2011.
[2] J. Macqueen, Some methods for classification and analysis of multivariate observations, in: Proceedings of Berkeley Symposium on Mathematical Statistics and Probability, 1967, pp. 281–297.
[3] M. Ester, H.P. Kriegel, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: International Conference on Knowledge Discovery and Data Mining, 1996, pp. 226–231.

[4] A.K. Jain, Data clustering: 50 years beyond k-means, Pattern Recognit. Lett. 31 (2010) 651–666.
[5] A.K. Jain, M.N. Murty, P.J. Flynn, Data clustering:a review, ACM Comput. Surv. 31 (3) (1999) 264–323.
[6] R. Xu, D. Wunsch II, Survey of clustering algorithms, IEEE Trans. Neural Netw. 16 (3) (2005) 645–678.
[7] Z. Huang, Extensions to the k-means algorithm for clustering large data sets with categorical values, Data Min. Knowl. Discov. 2 (3) (1998) 283–304.
[8] Z. Huang, M.K. Ng, A fuzzy k-modes algorithm for clustering categorical data, IEEE Trans. Fuzzy Syst. 7 (4) (1999) 446–452.
[9] M.K. Ng, M.J. Li, Z. Huang, Z. He, On the impact of dissimilarity measure in k-modes clustering algorithm, IEEE Trans. Pattern Anal. Mach. Intell. 29 (3) (2007) 503–507.
[10] L. Bai, J. Liang, C. Dang, F. Cao, The impact of cluster representatives on the convergence of the k-modes type clustering, IEEE Trans. Pattern Anal. Mach. Intell. 35 (6) (2013) 1509–1522.
[11] F. Cao, J. Liang, D. Li, X. Zhao, A weighting k-modes algorithm for subspace clustering of categorical data, Neurocomputing 108 (5) (2013) 23–30.
[12] L. Chen, S. Wang, K. Wang, J. Zhu, Soft subspace clustering of categorical data with probabilistic distance, Pattern Recognit. 51 (2016) 322–332.
[13] F. Cao, Z. Huang, J. Liang, X. Zhao, Y. Meng, K. Feng, Y. Qian, An algorithm for clustering categorical data with set-valued features, IEEE Trans. Neural Netw. Learn. Syst. 29 (10) (2018) 4593–4606.
[14] S. Guha, R. Rastogi, K. Shim, ROCK: A robust clustering algorithm for categorical attributes, Inf. Syst. 25 (5) (2000) 345–366.
[15] D. Barbara, J. Couto, Y. Li, COOLCAT: an entropy-based algorithm for categorical clustering, in: Proceedings of the 11th international conference on Information and knowledge management, 2002, pp. 582–589.
[16] X. Zhao, F. Cao, J. Liang, A sequential ensemble clusterings generation algorithm for mixed data, Appl. Math. Comput. 335 (2018) 264–277.
[17] J. Ghosh, A. Acharya, Cluster ensembles, Wiley Interdiscip. Rev. Data Min.Knowl. Discov. 1 (4) (2011) 305–315.
[18] H.G. Ayad, M.S. Kamel, On voting-based consensus of cluster ensembles, Pattern Recognit. 43 (5) (2010) 1943–1953.
[19] D. Huang, J. Lai, C.D. Wang, Ensemble clustering using factor graph, Pattern Recognit. 50 (2016) 131–142.
[20] N. Iam-On, T. Boongoen, S. Garrett, C. Price, A link-based cluster ensemble approach for categorical data clustering, IEEE Trans. Knowl. Data Eng. 24 (3) (2012) 413–425.
[21] M. Al-Razgan, C. Domeniconi, D. Barbara, Random subspace ensembles for clustering categorical data, in: Supervised and Unsupervised Ensemble Methods and their Applications, 2008, pp. 31–48.
[22] N. Iam-On, T. Boongoen, S. Garrett, Refining pairwise similarity matrix for cluster ensemble problem with cluster relations, in: Proceedings of International Conference on Discovery Science, 2008, pp. 222–233.
[23] G. Jeh, J. Widom, SimRank: a measure of structural-context similarity, in: Proceedings of the 8th Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, 2002, pp. 538–543.
[24] Y. Lu, Y. Wan, Pha: a fast potential-based hierarchical agglomerative clustering method, Pattern Recognit. 46 (5) (2013) 1227–1239.
[25] R.L. Cilibrasi, P.M.B. Vitnyi, A fast quartet tree heuristic for hierarchical clustering, Pattern Recognit. 44 (3) (2011) 662–677.
[26] Z. He, X. Xu, S. Deng, A cluster ensemble method for clustering categorical data, Inf. Fusion 6 (2) (2005) 143–151.
[27] G. Karypis, V. Kumar, Multilevelk-way partitioning scheme for irregular graphs, J. Parallel Distrib. Comput. 48 (2) (1998) 96–129.
[28] A. Ng, M. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm, in: Advances in Neural Information Processing Systems, 14, 2001, pp. 849–856.
[29] L. Jing, K. Tian, Z. Huang, Stratified feature sampling method for ensemble clustering of high dimensional data, Pattern Recognit. 48 (11) (2015) 3688–3702.
[30] Z. Yu, L. Li, Y. Gao, J. You, J. Liu, H.S. Wong, G. Han, Hybrid clustering solution selection strategy, Pattern Recognit. 47 (10) (2014) 3362–3375.
[31] H.L. Chen, K.T. Chuang, M.S. Chen, Labeling unclustered categorical data into clusters based on the important attribute values, in: IEEE International Conference on Data Mining, 2006, p. 8.
[32] F. Cao, L. Yu, J.Z. Huang, J. Liang, *k*-mw-modes: an algorithm for clustering categorical matrix-object data, Appl. Soft Comput. 57 (2017) 605–614.
[33] K. Bache, M. Lichman, UCI machine learning repository, 2014, http://archive.ics.uci.edu/ml.
[34] J. Liang, L. Bai, C. Dang, F. Cao, The *k*-means-type algorithms versus imbalanced data distributions, IEEE Trans. Fuzzy Syst. 20 (4) (2012) 728–745.
[35] A. Strehl, J. Ghosh, Cluster ensembles—a knowledge reuse framework for combining multiple partitions, J. Mach. Learn. Res. 3 (2003) 583–617.
[36] A. Strehl, J. Ghosh, Cluster ensembles: a knowledge reuse framework for combining partitionings, J. Mach. Learn. Res. 3 (2002) 583–617.
[37] H. Liu, T. Liu, J. Wu, D. Tao, F. Yun, Spectral ensemble clustering, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015, pp. 715–724.
[38] Z. Yu, Graph-based consensus clustering for class discovery from gene expression data, Bioinformatics 23 (21) (2007) 2888–2896.
[39] X. Zhao, J. Liang, C. Dang, Clustering ensemble selection for categorical data based on internal validity indices, Pattern Recognit. 69 (2017) 150–168.
[40] J. Ar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (1) (2006) 1–30.