The Impact of Cluster Representatives on the Convergence of the K-Modes Type Clustering

Liang Bai, Jiye Liang, Chuangyin Dang, *Senior Member*, *IEEE*, and Fuyuan Cao

Abstract—As a leading partitional clustering technique, k-modes is one of the most computationally efficient clustering methods for categorical data. In the k-modes, a cluster is represented by a "mode," which is composed of the attribute value that occurs most frequently in each attribute domain of the cluster, whereas, in real applications, using only one attribute value in each attribute to represent a cluster may not be adequate as it could in turn affect the accuracy of data analysis. To get rid of this deficiency, several modified clustering algorithms were developed by assigning appropriate weights to several attribute values in each attribute. Although these modified algorithms are quite effective, their convergence proofs are lacking. In this paper, we analyze their convergence property and prove that they cannot guarantee to converge under their optimization frameworks unless they degrade to the original k-modes type algorithms. Furthermore, we propose two different modified algorithms with weighted cluster prototypes to overcome the shortcomings of these existing algorithms. We rigorously derive updating formulas for the proposed algorithms and prove the convergence of the proposed algorithms. The experimental studies show that the proposed algorithms are effective and efficient for large categorical datasets.

Index Terms—Clustering, K-modes type clustering algorithms, categorical data, weighted cluster prototype, convergence

1 Introduction

Clustering is an unsupervised classification technique that aims at grouping a set of unlabeled objects into meaningful clusters so that the objects in the same cluster have high similarity but are very dissimilar to objects in other clusters. Many types of clustering techniques have been studied in the literature (e.g., [1] and references therein), which has extensive applications in various domains. Recently, increasing attention has been paid to clustering categorical data, where records are made up of nonnumerical data, since this task is of great practical relevance in several fields ranging from statistics to psychology [2], [3], [4], [5], [6].

Several algorithms for categorical data have been reported [7], [8], [9], [10], [11], 12], [13], [14], [15], [16], [17]. Among them, the k-modes type (nonfuzzy or fuzzy) clustering algorithms [16], [17], [19] are very popular techniques for solving categorical data clustering problems in different application domains, which have removed the numeric-only limitation of the k-means type algorithms [18] and enable the k-means clustering process to effectively cluster large categorical datasets from real-world databases.

In the k-modes, the prototype of a cluster is composed of the attribute value that occurs most frequently in each attribute domain of the cluster. Although this cluster representative is simple, using only one attribute value in each attribute domain to represent a cluster is questionable, as it often ignores the representability of other attribute values whose frequencies in the cluster may be close to the largest one. To get rid of this deficiency, several modified algorithms were developed in [20], [21], [22], [23], [24], [25], [26], where a prototype in a cluster is a list of several categorical values in the attribute with their frequencies in the cluster as the weights. The higher the weight of a categorical value in the cluster is, the more representability the categorical value has in the cluster. Although these modified algorithms are quite effective in enhancing the performance of the original k-modes type algorithms, the convergence proofs of these algorithms are lacking. However, in real applications, the main concerns for an iterative algorithm are whether it "stops" (successive iterations stabilize at an apparent fixed point of the process up to some margin of error) and, even more importantly, when it does stop, is the terminal iterate an (at least local) optimal solution of its objective function? Therefore, we need to address the following two problems:

1. Can these modified algorithms converge to the local optimal solutions of their objective functions in a finite number of iterations?
2. When the convergence of these modified algorithms cannot be guaranteed, how do we design the k-modes type algorithms with frequency-based prototypes which can guarantee the convergence?

On the basis of the above motivations, the major contributions in this paper are as follows: We first analyze

- *L. Bai is with the School of Computer and Information Technology, Shanxi University, Taiyuan 030006, Shanxi, China, and the Department of Systems Engineering and Engineering Management, City University of Hong Kong. E-mail: sxbailiang@126.com.*
- *J. Liang and F. Cao are with the School of Computer and Information Technology, Shanxi University, Taiyuan 030006, Shanxi, China. E-mail: {ljy, cfy}@sxu.edu.cn.*
- *C. Dang is with the Department of Systems Engineering and Engineering Management, City University of Hong Kong, Kowloon, Hong Kong. E-mail: mecdang@cityu.edu.hk.*

Manuscript received 1 Apr. 2011; revised 22 May 2012; accepted 22 Sept. 2012; published online 12 Oct. 2012.
Recommended for acceptance by M. Meila.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2011-04-0194.
Digital Object Identifier no. 10.1109/TPAMI.2012.228.

0162-8828/13/$31.00 © 2013 IEEE Published by the IEEE Computer Society

the convergence of the existing modified k-modes type algorithms [20], [21], [22], [23], [24], [25] and prove that the iterative sequences generated by these algorithms can converge to the local minimal solutions under their optimization frameworks only if they degrade to the original k-modes type algorithms. Furthermore, we propose two new k-modes type algorithms with frequency-based cluster prototypes, called MKM_NOF and MKM_NDM, respectively, which overcome, in different ways, the shortcomings of the existing modified algorithms as follows:

1. In the MKM_NOF algorithm, while keeping the formats of the dissimilarity measures in these algorithms, we modify their objective functions by adding the weight entropy term.
2. In the MKM_NDM algorithm, while keeping the formats of the objective functions in these algorithms, we modify the dissimilarity measures by adding an uncertainty measure.

These approaches can simultaneously minimize the within cluster dispersions and use the frequency of each categorical value in a cluster to reflect the representability of the categorical value in the cluster. We rigorously derive updating formulas of the MKM_NOF and MKM_NDM algorithms, respectively. It is proven that the clustering process with these updating formulas converges under the optimization framework. Finally, the experimental studies on several real datasets from UCI show that the proposed algorithms are effective and suitable for large categorical datasets thanks to its linear time complexity with respect to the number of data objects, attributes, or clusters.

The outline of this paper is as follows: In Section 2, we review the k-modes type algorithms. In Section 3, we introduce several modified k-modes type algorithms with frequency-based prototypes and analyze the reasons of the non convergence of these algorithms. In Section 4, we present a new objective function and the MKM_NOF algorithm. In Section 5, we propose a new dissimilarity measure and the MKM_NDM algorithm. In Section 6, we analyze the convergence of the two proposed algorithms. In Section 7, the experimental analysis is given to illustrate the convergence, effectiveness, and efficiency of the proposed algorithms. Finally, a concluding remark is given in Section 8.

2 The K-Modes Type Algorithms

Let $U = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be a set of n objects, $A = \{a_1, a_2, \ldots, a_m\}$ be a set of m attributes, and D_{a_j} be the domain of attribute a_j for $1 \le j \le m$. Here, we only consider two general data types, numeric and categorical, and assume other types used in database systems can be mapped to one of these two types. A numeric domain consists of real numbers. A domain D_{a_j} is defined as categorical if it is finite and unordered, i.e., $D_{a_j} = \{a_j^{(1)}, a_j^{(2)}, \ldots, a_j^{(n_j)}\}$, where n_j is the number of categories of attribute a_j for $1 \le j \le m$. For any $1 \le p \le q \le n_j$, either $a_j^{(p)} = a_j^{(q)}$ or $a_j^{(p)} \neq a_j^{(q)}$. For $1 \le i \le n$, object $\mathbf{x}_i \in U$ is represented as $[x_{i1}, x_{i2}, \ldots, x_{im}]$, where $x_{ij} \in D_{a_j}$, for $1 \le j \le m$. If each attribute in A is categorical, U is called a categorical dataset.

The k-modes type algorithms use the k-means type paradigm to cluster categorical datasets. The objective of clustering a set of n categorical objects into k clusters is to find W and Z that minimize [17]

$$F(W, Z) = \sum_{l=1}^{k} \sum_{i=1}^{n} w_{li}^{\alpha} d(\mathbf{z}_l, \mathbf{x}_i), \tag{1}$$

subject to

$$\begin{cases} w_{li} \in [0, 1], 1 \le l \le k, 1 \le i \le n, \\ \sum_{l=1}^{k} w_{li} = 1, 1 \le i \le n, \\ 0 < \sum_{i=1}^{n} w_{li} < n, 1 \le l \le k, \end{cases} \tag{2}$$

where

- n is the number of objects in U, $k(\le n)$ is a known number of clusters;
- $\alpha \in [1, +\infty)$ is the fuzzy index; $\alpha = 1$ gives the k-modes algorithm;
- $W = [w_{li}]$ is a k-by-n real matrix, w_{li} indicates whether $\mathbf{x}_i$ belongs to the lth cluster for the k-modes algorithm, $w_{li} = 1$ if $\mathbf{x}_i$ belongs to the lth cluster and 0 otherwise, and for the fuzzy k-modes algorithm, w_{li} is the membership degree of $\mathbf{x}_i$ to the lth cluster;
- $Z = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_k\} \subseteq R$, where $R = D_{a_1} \times D_{a_2} \times \cdots \times D_{a_m}$ and $\mathbf{z}_l = [z_{l1}, z_{l2}, \ldots, z_{lm}]$ is the lth cluster prototype with categorical attributes $a_1, a_2, \ldots, a_m$;
- $d(\mathbf{z}_l, \mathbf{x}_i)$ is the simple matching dissimilarity measure between object $\mathbf{x}_i$ and the prototype $\mathbf{z}_l$ of the lth cluster which is defined as

$$d(\mathbf{z}_l, \mathbf{x}_i) = \sum_{j=1}^{m} \delta(z_{lj}, x_{ij}), \tag{3}$$

where

$$\delta(z_{lj}, x_{ij}) = \begin{cases} 1, & z_{lj} \neq x_{ij}, \\ 0, & z_{lj} = x_{ij}. \end{cases} \tag{4}$$

Minimization of F in (1) with the constraints in (2) forms a class of constrained nonlinear optimization problems whose solutions are unknown. The usual method toward optimization of F in (1) is to use partial optimization for Z and W. In this method, we first fix Z and find necessary conditions on W to minimize F. Then, we fix W and minimize F with respect to Z. The above optimization problem can be solved by iteratively solving the following two minimization problems:

1. **Problem** P_1. Fix $Z = \hat{Z}$, solve the reduced problem $F(W, \hat{Z})$.
2. **Problem** P_2. Fix $W = \hat{W}$, solve the reduced problem $F(\hat{W}, Z)$.

For the k-modes algorithm ($\alpha = 1$), Problem P_1 is solved by

$$\hat{w}_{li} = \begin{cases} 1, & if \ d(\hat{\mathbf{z}}_l, \mathbf{x}_i) \le d(\hat{\mathbf{z}}_h, \mathbf{x}_i), 1 \le h \le k, \\ 0, & otherwise, \end{cases} \tag{5}$$

for $1 \le i \le n$, $1 \le l \le k$. For the fuzzy k-modes algorithm ($\alpha > 1$), Problem P_1 is solved by

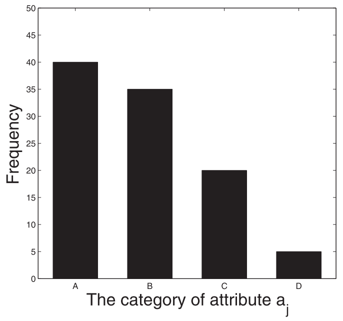


Fig. 1. An example of an attribute distribution in the cluster, where each bar corresponds to each categorical value.

$$w_{li} = \begin{cases} 1, & if\ \ d(\hat{\mathbf{z}}_l, \mathbf{x}_i) = 0, \\ 0, & if\ \ d(\hat{\mathbf{z}}_h, \mathbf{x}_i) = 0, h \neq l, \\ 1 \Big/ \sum_{h=1}^{k} \left[\dfrac{d(\hat{\mathbf{z}}_l, \mathbf{x}_i)}{d(\hat{\mathbf{z}}_h, \mathbf{x}_i)} \right]^{1/(\alpha-1)}, & if\ \ d(\hat{\mathbf{z}}_h, \mathbf{x}_i) \neq 0, 1 \leq h \leq k, \end{cases} \tag{6}$$

for $1 \leq i \leq n$, $1 \leq l \leq k$.

Problem P_2 is solved by

$$z_{lj} = a_j^{(r)} \in D_{a_j}, \tag{7}$$

where

$$\sum_{x_{ij}=a_j^{(r)}, \mathbf{x}_i \in U} w_{li}^{\alpha} = \max_{q=1}^{n_j} \sum_{x_{ij}=a_j^{(q)}, \mathbf{x}_i \in U} w_{li}^{\alpha}, \tag{8}$$

for $1 \leq j \leq m$. Here, $D_{a_j} = \{a_j^{(1)}, a_j^{(2)}, \ldots, a_j^{(n_j)}\}$, n_j is the number of categories of attribute a_j for $1 \leq j \leq m$.

This process is formalized in the k-modes type algorithms as follows [17]:

Step 1. Choose an initial point set $Z^{(1)} \subseteq R$. Determine $W^{(1)}$ such that $F(W, Z^{(1)})$ is minimized. Set $t = 1$.

Step 2. Determine $Z^{(t+1)}$ such that $F(W^{(t)}, Z^{(t+1)})$ is minimized. If $F(W^{(t)}, Z^{(t+1)}) = F(W^{(t)}, Z^{(t)})$, then stop; otherwise, go to Step 3.

Step 3. Determine $W^{(t+1)}$ such that $F(W^{(t+1)}, Z^{(t+1)})$ is minimized. If $F(W^{(t+1)}, Z^{(t+1)}) = F(W^{(t)}, Z^{(t+1)})$, then stop; otherwise, set $t = t + 1$ and go to Step 2.

We remark that Z is determined based on the frequencies of attribute values in the cluster. The most frequent attribute value in each attribute domain in a cluster is selected to represent the cluster, which minimizes the within-cluster dissimilarity. However, this approach often ignores the representability of other attribute values whose frequencies in the cluster may be close to the largest one.

Let us consider the following example to demonstrate the problem: We suppose that there is a categorical attribute a_j which has four categorical values: "A," "B," "C," and "D," and a cluster c_l which contains 40 "A," 35 "B," 20 "C," and 5 "D" in attribute a_j. Fig. 1 shows the categorical attribute distribution in cluster c_l. Although "A" is the most frequent categorical value in cluster c_l, the frequency of "B" is close to "A" in cluster c_l. When we select "A" from the attribute domain to represent cluster c_l, other 60 percent categorical values will be ignored.

3 The Convergence Property of Several Modified K-Modes Type Algorithms

To get rid of this deficiency, several modified algorithms were developed in [20], [21], [22], [23], [24], [25] by assigning appropriate weights to several attribute values in each attribute. San et al. [20] introduced frequency-based cluster prototypes to represent clusters, which are applied to the k-modes clustering algorithm. A prototype in a cluster is a list of all the categorical values in the attribute with their frequencies in the cluster as the weights. The higher the frequency of a categorical value in the cluster is, the more representability the categorical value has in the cluster. Kim et al. [21] presented a fuzzy k-modes algorithm with frequency-based prototypes. He et al. [22] and Ng et al. [23], [24] used the relative attribute frequencies in a cluster as weights to reflect the representability of cluster mode in the cluster and applied them to measure the similarity between objects and cluster prototypes. This modification can help the k-modes clustering process to recognize a cluster with weak intrasimilarity. Lee and Pedrycz in [25] introduced a generalization of the k-modes type clustering algorithms with fuzzy p-mode prototypes. The above modified algorithms can be seen as the special cases of the generalized k-modes type algorithm.

In the generalized algorithm, a generalization, called fuzzy p-mode prototype, of frequency-based prototypes is defined. A cluster prototype at a categorical attribute is expressed as a list of p categories that have larger frequencies than others in the cluster.

The definition of the lth cluster prototype $\mathbf{z}'_l = [z'_{l1}, z'_{l2}, \ldots, z'_{lm}]$ is formalized as

$$z'_{lj} = \left\{ \left(a_j^{(q)}, f_{ljq}\right) | a_j^{(q)} \in D_{a_j}^{(p_{lj})}, 1 \leq q \leq n_j \right\}, \tag{9}$$

where $D_{a_j}^{(p_{lj})} \subseteq D_{a_j}$ is a set of p_{lj} $(1 \leq p_{lj} \leq n_j)$ categorical values of a_j that have larger frequencies than others in the lth cluster for $1 \leq j \leq m$.

When given the cluster prototypes $Z' = \{\mathbf{z}'_1, \mathbf{z}'_2, \ldots, \mathbf{z}'_k\}$, the dissimilarity measure $d'(\mathbf{z}'_l, \mathbf{x}_i)$ is defined as follows:

$$d'(\mathbf{z}'_l, \mathbf{x}_i) = \sum_{j=1}^{m} \delta'(z'_{lj}, x_{ij}), \tag{10}$$

where

$$\delta'(z'_{lj}, x_{ij}) = \begin{cases} 1 - f_{ljq}, & if\ x_{ij} \in D_{a_j}^{(p_{lj})}, \\ 1, & otherwise. \end{cases} \tag{11}$$

Here, f_{ljq} is the relative frequency of the categorical value $a_j^{(q)}$ in the lth cluster, i.e.,

$$f_{ljq} = \frac{|c_{ljq}|}{|c_l|}, \tag{12}$$

where $|c_{ljq}| = \sum_{i=1, x_{ij}=a_j^{(q)}}^{n} w_{li}^{\alpha}$ and $|c_l| = \sum_{i=1}^{n} w_{li}^{\alpha}$.

Based on the p-mode prototypes instead of the modes and the dissimilarity measure d' instead of the simple matching dissimilarity measure d, Lee and Pedrycz presented a generalization of the k-modes type algorithms. More precisely, they use the iterative method to minimize

$$F'(W, Z') = \sum_{l=1}^{k} \sum_{i=1}^{n} w_{li}^{\alpha} d'(\mathbf{z}'_l, \mathbf{x}_i), \tag{13}$$

subject to the same conditions as those in (2).

When $p_{lj} = 1$ for each attribute a_j, $1 \le j \le m$, Z' is equal to Z of the original k-modes type algorithms and δ' becomes

$$\delta'(z'_{lj}, x_{ij}) = \begin{cases} 1 - f_{ljq}, & if\ x_{ij} = z_{lj}, \\ 1, & otherwise. \end{cases} \tag{14}$$

Then, the generalized k-modes type algorithm becomes He et al. and Ng et al.'s algorithms [22], [23], [24].

When $p_{lj} = n_j$ for each attribute a_j, $1 \le j \le m$, a prototype in a cluster is a list of all the categories in the attribute, with their frequencies in the cluster as the weights, i.e.,

$$z'_{lj} = \{ (a_j^{(q)}, f_{ljq}) | a_j^{(q)} \in D_{a_j}, 1 \le q \le n_j \}, \tag{15}$$

and

$$\begin{aligned} \delta'(z'_{lj}, x_{ij}) &= \sum_{q=1}^{n_j} f_{ljq} \delta(a_j^{(q)}, x_{ij}) \\ &= \sum_{q=1, x_{ij} \neq a_j^{(q)}}^{n_j} f_{ljq} = 1 - f_{ljr}, \end{aligned} \tag{16}$$

where $x_{ij} = a_j^{(r)}$ and $1 \le r \neq q \le n_j$, for $1 \le j \le m$, $1 \le l \le k$, $1 \le i \le n$. In this case, the generalized k-modes type algorithm becomes San et al. and Kim et al.'s algorithms [20], [21].

To analyze the convergence of these modified algorithms, we rewrite the objective function (13) as

$$F_f(W, V) = \sum_{l=1}^{k} \sum_{i=1}^{n} w_{li}^{\alpha} d_f(\mathbf{v}_l, \mathbf{x}_i), \tag{17}$$

subject to (2) and

$$\begin{cases} v_{ljq} \in [0, 1], 1 \le l \le k, 1 \le j \le m, 1 \le q \le n_j, \\ 0 < \sum_{q=1}^{n_j} v_{ljq} \le 1, 1 \le l \le k, 1 \le j \le m, \\ v_{ljq} = 0,\ if\ a_j^{(q)} \in D_{a_j} - D_{a_j}^{(p_{lj})}, 1 \le q \le n_j, \\ f_{ljq} \ge \max_{a_j^{(s)} \in D_{a_j} - D_{a_j}^{(p_{lj})}} f_{ljs},\ if\ a_j^{(q)} \in D_{a_j}^{(p_{lj})}, 1 \le q, s \le n_j, \end{cases} \tag{18}$$

where

- p_{lj} is the number of elements in $D_{a_j}^{(p_{lj})}$ and $1 \le p_{lj} \le n_j$ for $1 \le j \le m$.
- $V = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k]'$ and $\mathbf{v}_l = [v_{l11}, v_{l12}, \ldots, v_{l1n_1}, v_{l21}, v_{l22}, \ldots, v_{l2n_2}, \ldots, v_{lm1}, v_{lm2}, \ldots, v_{lmn_m}]'$ is a list of weights of all categorical values which is used to summarize and characterize the lth cluster. The larger v_{ljq} is, the more representability the categorical value $a_j^{(q)}$ has in the lth cluster. Here, $\mathbf{v}_l$ is seen as the lth cluster prototype.
- $d_f(\mathbf{v}_l, \mathbf{x}_i)$ is a dissimilarity measure between object $\mathbf{x}_i$ and the prototype $\mathbf{v}_l$ of the lth cluster which is defined as

$$d_f(\mathbf{v}_l, \mathbf{x}_i) = \sum_{j=1}^{m} \psi_{a_j}(\mathbf{v}_l, \mathbf{x}_i), \tag{19}$$

where

$$\psi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) = 1 - v_{ljr},\ if\ x_{ij} = a_j^{(r)}, 1 \le r \le n_j. \tag{20}$$

Similarly to solving (1), the optimization problem needs to be solved by iteratively solving the following two minimization problems:

1. **Problem** P_1**.** Fix $V = \hat{V}$, solve the reduced problem $F_f(W, \hat{V})$.
2. **Problem** P_2**.** Fix $W = \hat{W}$, solve the reduced problem $F_f(\hat{W}, V)$.

When $\alpha = 1$, Problem P_1 is solved in [20] by

$$\hat{w}_{li} = \begin{cases} 1, & if\ \ d_f(\hat{\mathbf{v}}_l, \mathbf{x}_i) \le d_f(\hat{\mathbf{v}}_h, \mathbf{x}_i), 1 \le h \le k, \\ 0, & otherwise, \end{cases} \tag{21}$$

for $1 \le i \le n$, $1 \le l \le k$. When $\alpha > 1$, Problem P_1 is solved in [21] by

$$w_{li} = \frac{1}{\sum_{h=1}^{k} \left[\frac{d_f(\hat{\mathbf{v}}_l, \mathbf{x}_i)}{d_f(\hat{\mathbf{v}}_h, \mathbf{x}_i)} \right]^{1/(\alpha-1)}}, \tag{22}$$

for $1 \le i \le n$, $1 \le l \le k$.

Problem P_2 is solved in [20], [21] by

$$v_{ljq} = \begin{cases} f_{ljq}, & if\ a_j^{(q)} \in D_{a_j}^{(p_{lj})}, \\ 0, & otherwise, \end{cases} \tag{23}$$

for $1 \le 1 \le k, 1 \le j \le m, 1 \le q \le n_j$.

Theorem 1. *Let* $W = \hat{W}$ *be fixed.* $F_f(\hat{W}, V)$ *is minimized iff*

$$v_{ljr} = \begin{cases} 1, & f_{ljr} = \max_{q=1}^{n_j} f_{ljq}, \\ 0, & otherwise, \end{cases} \tag{24}$$

for $1 \le l \le k$, $1 \le j \le m$, $1 \le r \le n_j$.

Proof. Let

$$\vartheta_{l,j} = \sum_{i=1}^{n} w_{li}^{\alpha} \psi_{a_j}(\mathbf{v}_l, \mathbf{x}_i),$$

for $1 \le l \le k$ and $1 \le j \le m$. Then,

$$\begin{aligned} \sum_{l=1}^{k} \sum_{i=1}^{n} w_{li}^{\alpha} d_f(\mathbf{v}_l, \mathbf{x}_i) &= \sum_{l=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{m} w_{li}^{\alpha} \psi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) \\ &= \sum_{l=1}^{k} \sum_{j=1}^{m} \sum_{i=1}^{n} w_{li}^{\alpha} \psi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) = \sum_{l=1}^{k} \sum_{j=1}^{m} \vartheta_{l,j}. \end{aligned}$$

For $1 \le l \le k$ and $1 \le j \le m$, each $\vartheta_{l,j}$ is nonnegative and independent. Thus, minimizing the objective function is equivalent to minimizing each $\vartheta_{l,j}$. Note that

$$\begin{aligned} \vartheta_{l,j} &= \sum_{i=1}^{n} w_{li}^{\alpha} \psi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) = \sum_{q=1}^{n_j} \sum_{i=1, x_{ij} = a_j^{(q)}}^{n} w_{li}^{\alpha} \psi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) \\ &= \sum_{q=1}^{n_j} \sum_{i=1, x_{ij} = a_j^{(q)}}^{n} w_{li}^{\alpha} (1 - v_{ljq}) \\ &= |c_l| - \sum_{q=1}^{n_j} \sum_{i=1, x_{ij} = a_j^{(q)}}^{n} w_{li}^{\alpha} v_{ljq} = |c_l| - |c_l| \sum_{q=1}^{n_j} v_{ljq} f_{ljq}. \end{aligned}$$

When W is given, $|c_l|$ is fixed. It is clear that $\vartheta_{l,j}$ is minimized iff $\sum_{q=1}^{n_j} v_{ljq} f_{ljq}$ is maximal for $1 \leq t \leq n_j$.

Because of

$$0 < \sum_{q=1}^{n_j} v_{ljq} \leq 1 \text{ and } v_{ljq} = 0,$$
$$if\ a_j^{(q)} \in D_{a_j} \setminus D_{a_j}^{(p_{lj})}, 1 \leq q \leq n_j,$$

we know that

$$\sum_{q=1}^{n_j} v_{ljq} f_{ljq} = \sum_{a_j^{(s)} \in D_{a_j}^{(p_{lj})}} v_{ljs} f_{ljs} \leq \max_{q=1}^{n_j} f_{ljq}.$$

Therefore,

$$v_{ljr} = \begin{cases} 1, & f_{ljr} = \max_{q=1}^{n_j} f_{ljq}, \\ 0, & otherwise, \end{cases}$$

maximizes $\sum_{q=1}^{n_j} v_{ljq} f_{ljq}$ for $1 \leq l \leq k$, $1 \leq j \leq m$, $1 \leq r \leq n_j$. The result follows. □

According to Theorem 1, we can see that while solving Problem 2 for each attribute, only one categorical value with the relatively maximum frequency has the representability in the cluster. This means that Theorem 1 is equivalent to the updating formula for cluster prototypes in the original k-modes type algorithms. While Theorem 1 is used to compute V, the distance function d_f also becomes the simple matching dissimilarity measure, i.e.,

$$\psi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) = \begin{cases} 1, & x_{ij} = a_j^{(q)}\ and\ v_{ljq} = 0, \\ 0, & x_{ij} = a_j^{(q)}\ and\ v_{ljq} = 1. \end{cases} \tag{25}$$

The analysis tells us that the cluster process can converge to a local minimal solution under the optimization framework only if the modified algorithms are degenerate to the original k-modes type algorithms.

To overcome the deficiencies of these existing modified algorithms, in the next sections we will propose two new modified k-modes type clustering algorithms, called MKM_NOF and MKM_NDM, respectively. They will apply different techniques to simultaneously guarantee the convergence of the clustering process and implement the representation of a cluster by using several categorical values in each attribute with appropriate weights.

4 The MKM_NOF Algorithm

To avoid the problem of identifying clusters by a single categorical value from each attribute, a weight entropy term is added to the objective function (17). This term is inspired by the principle of maximum entropy, which provides an unbiased probability assignment for ill-defined problems on the basis of the given information. The principle was first expounded by Jaynes [27] in 1957 and currently has been applied to fuzzy clustering and subspace clustering [28], [29]. Here, we will use the weight entropy term to help us simultaneously minimize the within-cluster dispersion and stimulate more categorical values from each attribute to contribute to the identification of clusters.

The new objective function and optimization problem can be written as follows:

$$F_e(W,V) = \sum_{l=1}^{k}\sum_{i=1}^{n}\left[w_{li}^{\alpha} d_f(\mathbf{v}_l, \mathbf{x}_i) + \gamma \sum_{j=1}^{m}\sum_{q=1}^{n_j} v_{ljq} \log v_{ljq}\right], \tag{26}$$

subject to the same conditions as in those in (2) and

$$\begin{cases} v_{ljq} \in [0,1], 1 \leq l \leq k, 1 \leq j \leq m, 1 \leq q \leq n_j, \\ \sum_{q=1}^{n_j} v_{ljq} = 1, 1 \leq l \leq k, 1 \leq j \leq m. \end{cases} \tag{27}$$

In the objective function, the first term is the sum of the within-cluster dispersions that we want to minimize and the second term is the negative weight entropy that we want to maximize. Due to the second term, a cluster will be represented by several categorical values with nonzero weights in an attribute instead of one, which makes a significant difference between the proposed approach and the existing ones. For any attribute a_j $(1 \leq j \leq m)$, when v_{ljq^*} is close to one for some q^* and v_{ljq} is close to zero for all $q \neq q^*$, the value of negative entropy $-\sum_{q=1}^{n_j} v_{ljq} \log v_{ljq}$ is close to zero. In this case, the lth cluster will certainly be represented by the single q^*th categorical value of a_j, and the corresponding entropy value is small. However, when some of v_{ljq} are about the same and greater than zero and the others are close to zero, the negative entropy will become more positive, i.e., much larger than zero. In this situation, the lth cluster will be represented by several categorical values of a_j. Therefore, with the weight entropy term, the clustering process attempts to simultaneously minimize the within-cluster dispersions and maximize the negative weight entropy, which can stimulate more categorical values to contribute to the description of clusters. In the minimization process of (26), the value of parameter γ determines which term will play a more important role. The larger the value of γ is, the more the second term contributes in the optimization process and the "smoother" or fuzzier the resulting V are. However, the value of γ should not be too large. The reason is that when γ is very large for each cluster, v_{ljq} is close to $1/n_j$, which makes the descriptions of all the clusters become identical.

Similarly to solving (17), we minimize (26) by iteratively solving Problems 1 and 2. When V is fixed, W is updated by (21) and (22). Now, the key issue is to rigorously derive the updating formula of V for solving Problem 2 when W is fixed. Theorem 2 below presents the updating formula of V.

Theorem 2. *Let $W = \hat{W}$ be fixed. $F_e(\hat{W}, V)$ is minimized iff*

$$v_{ljr} = \frac{\exp\left(\frac{|c_{ljr}|}{\gamma}\right)}{\sum_{q=1}^{n_j} \exp\left(\frac{|c_{ljq}|}{\gamma}\right)}, \tag{28}$$

for $1 \leq l \leq k$, $1 \leq j \leq m$, $1 \leq r \leq n_j$.

Proof. Let

$$\kappa_{l,j} = \sum_{i=1}^{n} w_{li}^{\alpha} \psi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) + \gamma \sum_{q=1}^{n_j} v_{ljq} \log v_{ljq},$$

for $1 \leq l \leq k$ and $1 \leq j \leq m$. Then,

$$\begin{aligned}
&\sum_{l=1}^{k}\sum_{i=1}^{n}\left[w_{li}^{\alpha}d_f(\mathbf{v}_l,\mathbf{x}_i)+\gamma\sum_{j=1}^{m}\sum_{q=1}^{n_j}v_{ljq}\log v_{ljq}\right]\\
&=\sum_{l=1}^{k}\sum_{i=1}^{n}\sum_{j=1}^{m}\left[w_{li}^{\alpha}\psi_{a_j}(\mathbf{v}_l,\mathbf{x}_i)+\gamma\sum_{q=1}^{n_j}v_{ljq}\log v_{ljq}\right]\\
&=\sum_{l=1}^{k}\sum_{j=1}^{m}\sum_{i=1}^{n}\left[w_{li}^{\alpha}\psi_{a_j}(\mathbf{v}_l,\mathbf{x}_i)+\gamma\sum_{q=1}^{n_j}v_{ljq}\log v_{ljq}\right]\\
&=\sum_{l=1}^{k}\sum_{j=1}^{m}\kappa_{l,j}.
\end{aligned}$$

For $1 \leq l \leq k$ and $1 \leq j \leq m$, each $\kappa_{l,j}$ is nonnegative and independent. Thus, minimizing the objective function is equivalent to minimizing each $\kappa_{l,j}$. Note that

$$\begin{aligned}
\kappa_{l,j}&=\sum_{i=1}^{n}w_{li}^{\alpha}\psi_{a_j}(\mathbf{v}_l,\mathbf{x}_i)+\gamma\sum_{q=1}^{n_j}v_{ljq}\log v_{ljq}\\
&=\sum_{q=1}^{n_j}\sum_{i=1,x_{ij}=a_j^{(q)}}^{n}w_{li}^{\alpha}\psi_{a_j}(\mathbf{v}_l,\mathbf{x}_i)+\gamma\sum_{q=1}^{n_j}v_{ljq}\log v_{ljq}\\
&=\sum_{q=1}^{n_j}\sum_{i=1,x_{ij}=a_j^{(q)}}^{n}w_{li}^{\alpha}(1-v_{ljq})+\gamma\sum_{q=1}^{n_j}v_{ljq}\log v_{ljq}\\
&=|c_l|-\sum_{q=1}^{n_j}\sum_{i=1,x_{ij}=a_j^{(q)}}^{n}w_{li}^{\alpha}v_{ljq}+\gamma\sum_{q=1}^{n_j}v_{ljq}\log v_{ljq}\\
&=|c_l|-\sum_{q=1}^{n_j}|c_{ljq}|v_{ljq}+\gamma\sum_{q=1}^{n_j}v_{ljq}\log v_{ljq},
\end{aligned}$$

where $|c_l|$ and $|c_{ljq}|$ $(1 \leq q \leq n_j)$ are constants for fixed W. This means that minimizing $\kappa_{l,j}$ is equivalent to minimizing

$$-\sum_{q=1}^{n_j}|c_{ljq}|v_{ljq}+\gamma\sum_{q=1}^{n_j}v_{ljq}\log v_{ljq}. \tag{29}$$

Since $\kappa_{l,j}$ is a strictly convex function, the well-known K-K-T necessary optimization condition is also sufficient. Therefore, $\hat{\mathbf{v}}_{lj}$ is an optimal solution if and only if there exists $\hat{\lambda}$ together with $\hat{\mathbf{v}}_{lj}$ satisfying the following system of equations:

$$\begin{aligned}
&\nabla_{\mathbf{v}_{lj}}\tilde{\kappa}_{l,j}(\mathbf{v}_{lj},\lambda)=0,\\
&\sum_{q=1}^{n_j}v_{ljq}=1,
\end{aligned}\tag{30}$$

where $\mathbf{v}_{lj}=\{v_{lj1},v_{lj2},\dots,v_{ljn_j}\}$ and

$$\begin{aligned}
\tilde{\kappa}_{l,j}(\mathbf{v}_{lj},\lambda)=&-\sum_{q=1}^{n_j}|c_{ljq}|v_{ljq}+\gamma\sum_{q=1}^{n_j}v_{ljq}\log v_{ljq}\\
&+\lambda\left(\sum_{q=1}^{n_j}v_{ljq}-1\right).
\end{aligned}\tag{31}$$

We have

$$\frac{\partial\tilde{\kappa}_{l,j}(\mathbf{v}_{lj},\lambda)}{\partial v_{ljr}}=-|c_{ljr}|+\gamma(1+\log v_{ljr})+\lambda, 1 \leq q \leq n_j. \tag{32}$$

From (30) and (32), we obtain the optimal solution

$$\hat{v}_{ljr}=\frac{\exp\left(\frac{|c_{ljr}|}{\gamma}\right)}{\sum_{q=1}^{n_j}\exp\left(\frac{|c_{ljq}|}{\gamma}\right)}.$$

This completes the proof. □

Due to $|c_{ljr}|=f_{ljr}|c_l|$, v_{ljr} is proportional to f_{ljr}. Therefore, the larger f_{ljr}, the larger v_{ljr}, the more representability the categorical value $a_j^{(q)}$ has in the lth cluster.

Let us consider the example in Section 2 again. Without loss of generality, assume $\gamma = 10$. According to (28), we can compute the representability of the categorical values "A," "B," "C," and "D" in the cluster c_l as follows: $v_{lj1} = 0.5643$, $v_{lj2} = 0.3423$, $v_{lj3} = 0.0764$, and $v_{lj4} = 0.0170$. We see that the proposed representation method can sufficiently reflect the representability of all the categorical values in the cluster compared to the original k-modes type algorithms. And the larger the frequency of a categorical value in c_l is, the higher its representability of c_l is.

Based on Theorem 2, an algorithm is proposed to minimize (26), which is as follows:

Algorithm-MKM_NOF

```
Input: The number of clusters $k$ and the parameters $\alpha$ and $\gamma$. Randomly choose a set of $k$ objects $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\} \subset U$ to initialize $V^{(1)}$, i.e., set $v_{ljq} = 1$ if $x_{lj} = a_j^{(q)}$, otherwise, $v_{ljq} = 0$, for $1 \leq l \leq k$, $1 \leq j \leq m$ and $1 \leq q \leq n_j$.
REPEAT
    Update the partition matrix $W$ by (21) or (22);
    Update the weights of cluster prototypes $V$ by Theorem 2;
UNTIL the value of the objective function $F_e$ does not change.
```

If the clustering process needs t iterations to converge, the total computational complexity of the MKM_NOF algorithm is $O(mnkt)$ which is as much as the original k-modes type algorithms ($O(mnkt)$). This shows that the computational complexity increases linearly as either the number of objects, attributes or clusters increases. As for the storage, we need $O(mn + nk + 2\sum_{j=1}^{m} n_jk)$ space to hold the set of n objects, the partition matrix W, the cluster prototypes V, and the frequencies of all the categorical values in each cluster. The storage space is very close to that of the original k-modes type algorithms. ($O(mn + nk + mk + \sum_{j=1}^{m} n_jk)$).

5 The MKM_NDM Algorithm

In this section, we introduce a new dissimilarity measure into the objective function (17). More precisely, we will minimize

$$F_n(W,V)=\sum_{l=1}^{k}\sum_{i=1}^{n}w_{li}^{\alpha}d_n(\mathbf{v}_l,\mathbf{x}_i), \tag{33}$$

subject to the same conditions as those in (2) and (27), where the dissimilarity measure $d_n(\mathbf{v}_l, \mathbf{x}_i)$ is defined as follows:

$$d_n(\mathbf{v}_l, \mathbf{x}_i) = \sum_{j=1}^{m} \phi_{a_j}(\mathbf{v}_l, \mathbf{x}_i), \tag{34}$$

with

$$\phi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) = (1 - v_{ljr})^2 + \sum_{q=1, q\neq r}^{n_j} v_{ljq}^2, \; if \; x_{lj} = a_j^{(r)}, 1 \leq r \leq n_j. \tag{35}$$

According to (35), $\phi_{a_j}(\mathbf{v}_l, \mathbf{x}_i)$ depends on two factors, i.e., v_{ljr} and $\sum_{q=1,q\neq r}^{n_j} v_{ljq}^2$. The first factor v_{ljr} is the representability of $a_j^{(r)}$ in the lth cluster. The larger v_{ljr} is, the more representability $a_j^{(r)}$ has in the lth cluster, the smaller the dissimilarity between $\mathbf{v}_l$ and $\mathbf{x}_i$ in the attribute a_j. When the representability of $a_j^{(r)}$ is one, $\phi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) = 0$ and thus the corresponding function value is the same as that in the simple matching dissimilarity measure d in the original k-modes type algorithms. The second factor $\sum_{q=1,q\neq r}^{n_j} v_{ljq}^2$ is an uncertainty measure on the representability of other categories of a_j in the lth cluster. Since $\sum_{q=1,q\neq r}^{n_j} v_{ljq}^2$ is a strictly convex function, the K-K-T necessary optimality condition is also sufficient. Thus, $\mathbf{v}'_{lj} = \{v_{ljq} | 1 \leq q \leq n_j, q \neq r\}$ is an optimal solution of $\min \sum_{q=1,q\neq r}^{n_j} v_{ljq}^2$ subject to $\sum_{q=1}^{n_j} v_{ljq} - 1 + v_{ljr} = 0$ if and only if there is some $\hat{\lambda}$ together with $\hat{\mathbf{v}}'_{lj}$ satisfying the following system of equations:

$$\begin{aligned} &\nabla_{v'_{lj}} \tilde{\varphi}(\mathbf{v}'_{lj}, \lambda) = 0, \\ &1 - \sum_{q=1}^{n_j} v_{ljq} = v_{ljr}, \end{aligned} \tag{36}$$

where

$$\tilde{\varphi}(\mathbf{v}'_{lj}, \lambda) = \sum_{q=1,q\neq r}^{n_j} v_{ljq}^2 + \lambda\left(\sum_{q=1}^{n_j} v_{ljq} - 1 + v_{ljr}\right). \tag{37}$$

Note that

$$\frac{\partial \tilde{\varphi}(\mathbf{v}'_{lj}, \lambda)}{\partial v_{ljq}} = 2v_{ljq} + \lambda, 1 \leq q \leq n_j, q \neq r. \tag{38}$$

From (36) and (38), we obtain that

$$\hat{v}_{ljq} = \frac{1 - v_{ljr}}{n_j - 1}, 1 \leq q \leq n_j, q \neq r. \tag{39}$$

The above analysis shows that, when v_{ljq}, $1 \leq q \leq n_j$, $q \neq r$, are equal, $\sum_{q=1,q\neq r}^{n_j} v_{ljq}^2$ achieves its minimum value given by

$$\frac{(1 - v_{ljr})^2}{n_j - 1}.$$

We also know that

$$\sum_{q=1,q\neq r}^{n_j} v_{ljq}^2 \leq \left(\sum_{q=1,q\neq r}^{n_j} v_{ljq}\right)^2 = (1 - v_{ljr})^2.$$

Hence, if only one of v_{ljq}, $1 \leq q \leq n_j$, is nonzero, $\sum_{q=1,q\neq r}^{n_j} v_{ljq}^2$ achieves its maximum value given by

$$(1 - v_{ljr})^2.$$

The value of $\sum_{q=1,q\neq r}^{n_j} v_{ljq}^2$ reflects an uncertainty degree on the representability of categorical values $a_j^{(q)}$ ($1 \leq q \leq n_j$, $q \leq r$) in the lth cluster. The larger $\sum_{q=1,q\neq r}^{n_j} v_{ljq}^2$ is, the smaller the uncertainty degree is, the larger the dissimilarity between $\mathbf{v}_l$ and $\mathbf{x}_i$ in the attribute a_j is.

Property 1 (Maximum). *The maximum value of $\phi_{a_j}(\mathbf{v}_l, \mathbf{x}_i)$ is 2. This value is achieved only if there exists some $q \leq n_j$ such that $v_{ljq} = 1$ and $x_{ij} \neq a_j^{(q)}$.*

Property 2 (Minimum). *The minimum value of $\phi_{a_j}(\mathbf{v}_l, \mathbf{x}_i)$ is 0. This value is achieved only if $x_{ij} = a_j^{(q)}$ for some q and $v_{ljq} = 1$.*

Property 3. *$\phi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) = 2\psi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) + \sum_{q=1}^{n_j} v_{ljq}^2 - 1$, where $\psi_{a_j}(\mathbf{v}_l, \mathbf{x}_i)$ can be found in Section 3.*

Similarly to the way for solving (17), we will minimize (33) by iteratively solving Problems 1 and 2.

When $\alpha = 1$, Problem P_1 is solved by

$$\hat{w}_{li} = \begin{cases} 1, & if \;\; d_n(\hat{\mathbf{v}}_l, \mathbf{x}_i) \leq d_n(\hat{\mathbf{v}}_h, \mathbf{x}_i), 1 \leq h \leq k, \\ 0, & otherwise, \end{cases} \tag{40}$$

for $1 \leq i \leq n, 1 \leq l \leq k$. When $\alpha > 1$, Problem P_1 is solved by

$$w_{li} = \frac{1}{\sum_{h=1}^{k} \left[\frac{d_n(\hat{\mathbf{v}}_l, \mathbf{x}_i)}{d_n(\hat{\mathbf{v}}_h, \mathbf{x}_i)}\right]^{1/(\alpha - 1)}}, \tag{41}$$

for $1 \leq i \leq n$, $1 \leq l \leq k$.

Theorem 3 below rigorously shows the updating formula of V to solve Problem 2 when W is fixed.

Theorem 3. *Let $W = \hat{W}$ be fixed. $F_n(\hat{W}, V)$ is minimized iff*

$$v_{ljr} = f_{ljr}, \tag{42}$$

for $1 \leq l \leq k$, $1 \leq j \leq m$, $1 \leq r \leq n_j$.

Proof. Let

$$\theta_{l,j} = \sum_{i=1}^{n} w_{li}^{\alpha} \phi_{a_j}(\mathbf{v}_l, \mathbf{x}_i),$$

for $1 \leq l \leq k$ and $1 \leq j \leq m$. Then,

$$\begin{aligned} \sum_{l=1}^{k}\sum_{i=1}^{n} w_{li}^{\alpha} d_n(\mathbf{v}_l, \mathbf{x}_i) &= \sum_{l=1}^{k}\sum_{i=1}^{n}\sum_{j=1}^{m} w_{li}^{\alpha} \phi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) \\ &= \sum_{l=1}^{k}\sum_{j=1}^{m}\sum_{i=1}^{n} w_{li}^{\alpha} \phi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) = \sum_{l=1}^{k}\sum_{j=1}^{m} \theta_{l,j}. \end{aligned}$$

For $1 \leq l \leq k$ and $1 \leq j \leq m$, each $\theta_{l,j}$ is nonnegative and independent. Thus, minimizing the objective function is equivalent to minimizing each $\theta_{l,j}$. Note that

$$\begin{aligned} \theta_{l,j} &= \sum_{i=1}^{n} w_{li}^{\alpha} \phi_{a_j}(\mathbf{v}_l, \mathbf{x}_i) \\ &= \sum_{q=1}^{n_j} \sum_{i=1, x_{ij}=a_j^{(q)}}^{n} w_{li}^{\alpha} \theta_{a_j}(\mathbf{v}_l, \mathbf{x}_i) \\ &= \sum_{q=1}^{n_j} \left[|c_{ljq}|(1 - v_{ljq})^2 + (|c_l| - |c_{ljq}|) v_{ljq}^2\right] \\ &= \sum_{q=1}^{n_j} \left(|c_l| v_{ljq}^2 - 2|c_{ljq}| v_{ljq}\right) + |c_l|, \end{aligned}$$

where $|c_l|$ and $|c_{ljq}|$ $(1 \leq q \leq n_j)$ are constants for fixed W. This means that minimizing $\theta_{l,j}$ is equivalent to minimizing

$$\sum_{q=1}^{n_j} \left(|c_l| v_{ljq}^2 - 2|c_{ljq}| v_{ljq}\right). \tag{43}$$

Since $\theta_{l,j}$ is a strictly convex function, the well-known K-K-T necessary optimization condition is also sufficient. Therefore, $\hat{\mathbf{v}}_{lj}$ is an optimal solution if and only if there exists $\hat{\lambda}$ together with $\hat{\mathbf{v}}_{lj}$ satisfying the following system of equations:

$$\begin{aligned} &\nabla_{\mathbf{v}_{lj}} \tilde{\theta}_{l,j}(\mathbf{v}_{lj}, \lambda) = 0, \\ &\sum_{q=1}^{n_j} v_{ljq} = 1, \end{aligned} \tag{44}$$

where $\mathbf{v}_{lj} = \{v_{lj1}, v_{lj2}, \ldots, v_{ljn_j}\}$ and

$$\tilde{\theta}_{l,j}(\mathbf{v}_{lj}, \lambda) = \sum_{q=1}^{n_j} \left(|c_l| v_{ljq}^2 - 2|c_{ljq}| v_{ljq}\right) + \lambda\left(\sum_{q=1}^{n_j} v_{ljq} - 1\right). \tag{45}$$

We have

$$\frac{\partial \tilde{\theta}_{l,j}(\mathbf{v}_{lj}, \lambda)}{\partial v_{ljr}} = 2|c_l| - 2|c_{ljr}| v_{ljr} + \lambda, 1 \leq q \leq n_j. \tag{46}$$

From (44) and (46), we obtain the optimal solution

$$\hat{v}_{ljr} = \frac{|c_{ljr}|}{|c_l|}.$$

This completes the proof. □

Here, the relative frequency of each categorical value in a cluster is used to reflect its representability in the cluster. It is obvious that the larger the frequency of a categorical value in a cluster is, the larger its representability in the cluster is. Let us consider the example in Section 2 again. According to (42), we can compute the representability of the categorical values "A," "B," "C," and "D" in cluster c_l as follows: $v_{lj1} = 0.4$, $v_{lj2} = 0.35$, $v_{lj3} = 0.2$, and $v_{lj4} = 0.05$.

Based on Theorem 3, an algorithm is proposed to minimize (33), which is as follows:

```
Algorithm-MKM_NDM
Input: The number of clusters $k$ and the parameter $\alpha$;
Randomly choose a set of $k$ objects $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k\} \subset U$ to initialize $V^{(1)}$, i.e., set $v_{ljq} = 1$ if $x_{lj} = a_j^{(q)}$, otherwise, $v_{ljq} = 0$, for $1 \leq l \leq k$, $1 \leq j \leq m$ and $1 \leq q \leq n_j$.
REPEAT
    Update the partition matrix $W$ by (40) or (41);
    Update the weights of cluster prototypes $V$ by Theorem 3;
UNTIL the value of the objective function $F_n$ does not change.
```

The total time and space complexities of the MKM_NDM algorithm are as much as the MKM_NOF algorithm.

6 Convergence Analysis

When $\alpha = 1$, the convergence of the MKM_NOF and MKM_NDM algorithms can be obtained as in Theorems 4 and 5 below.

Theorem 4. *When $\alpha = 1$, the MKM_NOF algorithm converges to a local minimal solution in a finite number of iterations.*

Proof. We first note that there are only a finite number of possible partitions W. We then show that each possible partition W appears at most once by the algorithm. Assume that $W^{(t_1)} = W^{(t_2)}$, where $t_1 \neq t_2$. We note that, given $W^{(t)}$, we can compute the minimizer $V^{(t)}$ according to Theorem 2. For $W^{(t_1)}$ and $W^{(t_2)}$, we have the minimizers $V^{(t_1)}$ and $V^{(t_2)}$, respectively. It is clear that $V^{(t_1)} = V^{(t_2)}$ since $W^{(t_1)} = W^{(t_2)}$. Therefore, we obtain

$$F_e(W^{(t_1)}, V^{(t_1)}) = F_e(W^{(t_2)}, V^{(t_1)}) = F_e(W^{(t_2)}, V^{(t_2)}).$$

However, the sequence $F_e(\cdot, \cdot)$ generated by the MKM_NOF algorithm is strictly decreasing. Hence, the result follows. □

Theorem 5. *When $\alpha = 1$, the MKM_NDM algorithm converges to a local minimal solution in a finite number of iterations.*

Proof. Similar to Theorem 4. □

Next, we will analyze the convergence of the MKM_NOF and MKM_NDM algorithms when $\alpha > 1$. For convenience, we define:

- $$M_{hw} = \left\{W \in \mathbb{R}^{kn}: w_{li} \in \{0,1\} \; and \; \sum_{l=1}^{k} w_{li} = 1, \forall l, i\right\}.$$
- $M_{fw} = \{W \in \mathbb{R}^{kn} : w_{li} \; satisfies \; (2), \forall l, i\}$.
- $$H_{hv} = \left\{V \in \mathbb{R}^{ks} : v_{ljq} \in \{0,1\} \; and \; \sum_{q=1}^{n_j} v_{ljq} = 1, \forall l, j, q\right\},$$
  where $s = \sum_{j=1}^{m} n_j$.
- $H_{fv} = \{V \in \mathbb{R}^{ks} : v_{ljq} \; satisfies \; (27), \forall l, j, q\}$.
- $G_{e1} : H_{fv} \rightarrow M_{fw}$, $G_{e1}(V) = W = [w_{li}]$, where the entries of W are calculated via (22).
- $G_{e2} : M_{fw} \rightarrow H_{fv}$, $G_{e2}(W) = V = [v_{ljq}]$, where the entries of V are calculated via Theorem 2.
- $J_e : (M_{fw} \times H_{fv}) \rightarrow (M_{fw} \times H_{fv})$, $J_e = G_{e2} \circ G_{e1}$.
- $G_{n1} : H_{fv} \rightarrow M_{fw}$, $G_{n1}(V) = W = [w_{li}]$, where the entries of W are calculated via (41).
- $G_{n2} : M_{fw} \rightarrow H_{fv}$, $G_{n2}(W) = V = [v_{ljq}]$, where the entries of V are calculated via Theorem 3.
- $J_n : (M_{fw} \times H_{fv}) \rightarrow (M_{fw} \times H_{fv})$, $J_n = G_{n2} \circ G_{n1}$.

Similar to the approach by which Bezdek analyzed the convergence of the fuzzy k-means algorithm [30], [31], our strategy will be to apply Zangwill's theorem [32] to discuss the convergence of the MKM_NOF and MKM_NDM algorithms ($\alpha > 1$).

Theorem 6 [32]. *Let $f: D_f \subset \mathbb{R}^m \rightarrow \mathbb{R}$: $S = \{x^* \in D_f : f(x^*) < f(y) \forall y \in B^0(x^*, r)\}$, where $B^0(x^*, r) = \{y \in \mathbb{R}^m: \|x^* - y\| < r, \| \cdot \| \text{ any norm on } \mathbb{R}^m\}$, $A : D_f \rightarrow D_f$ be an iterative algorithm, $x_{k+1} = A(x_k)$, and g be attached to* *sequences of iterates generated by A to monitor the progress of A in seeking a solution $x^* \in S$. If the following conditions hold, g is a descent function for $\{A, S\}$, A is continuous on $D_f \backslash S$, and the iterate sequences $\{A(x_k): k = 1, 2, \ldots; x_1 \in D_f\} \subset K$ are contained in a compact set $K \subseteq D_f$ for arbitrary $x_1 \in D_f$, then for each iterative sequence $\{x_k\}$ generated by A, we have either $\{x_k\}$ terminates at a solution $x^* \in S$ or $\exists$ a subsequence $\{x_{k_j}\} \subseteq \{x_k\}$ so that $\{x_{k_j}\} \to x^* \in S$.*

Theorem 6 and its generalizations can be used to obtain convergence proofs for almost all of the classical iterative optimization algorithms, e.g., steepest descent, Newton's method, etc., by using this approach as an alternative to more conventional arguments.

According to Theorems 2 and 3, we know the sequences $F_e(\cdot,\cdot)$ and $F_n(\cdot,\cdot)$ generated by the MKM_NOF and MKM_NDM algorithms, respectively, are strictly decreasing. This indicates the MKM_NOF and MKM_NDM algorithms satisfy the first requirement of Theorem 6.

The second requirement of Theorem 6 is that algorithms J_e and J_n be continuous on the domains of $F_e \setminus S$ and $F_n \setminus S$, respectively. J_e and J_n are in fact continuous on all of $M_{fw} \times H_{fv}$, as we show in the following.

Theorem 7. *J_e is continuous on $(M_{fw} \times H_{fv})$.*

Proof. Since $J_e = G_{e2} \circ G_{e1}$, and the composition of the continuous functions is again continuous, it suffices to show that G_{e1} and G_{e2} are each continuous. To see that G_{e1} is continuous in the (kn) variables $\{w_{li}\}$, note that G_{e1} is a vector field, with the resolution by (ks) scalar field where $s = \sum_{j=1}^{m} n_j$, say

$$G_{e1} = [G_{e1}^{(111)}, G_{e1}^{(112)}, \ldots, G_{e1}^{(ljr)}, \ldots, G_{e1}^{(kmn_m)}]: \ \mathbb{R}^{kn} \to \mathbb{R}^{ks},$$

where $G_{e1}^{(ljr)}: \mathbb{R}^{kn} \to \mathbb{R}$ is defined via Theorem 2 as

$$G_{e1}^{(ljr)}(W) = \frac{\exp\left(\frac{|c_{ljr}|}{\gamma}\right)}{\sum_{q=1}^{n_j} \exp\left(\frac{|c_{ljq}|}{\gamma}\right)} = v_{ljr}, \forall l, j, r.$$

Now, $\{w_{li} \to w_{li}^{\alpha}\}$ is continuous, $\{|c_{ljr}| \to \exp(|c_{ljr}|/\gamma)\}$ is continuous, and the sum of continuous functions is continuous; thus, $G_1^{(ljr)}(W)$ is the quotient of two continuous scalar fields for all $1 \leq l \leq k$, $1 \leq j \leq m$, $1 \leq r \leq n_j$. In view of constraint (6), the denominator of $G_{e1}^{(ljr)}(W)$ never vanishes, so $G_{e1}^{(ljr)}(W)$ is also continuous $\forall l, j, r$. Therefore, G_{e1} is continuous on their entire domains. Next, we show that G_{e2} is a continuous function of the (ks) variables $\{v_{ljq}\}$. G_{e2} is a vector field with the resolution by (kn) scalar fields:

$$G_{e2} = \left[G_{e2}^{(11)}, G_{e2}^{(12)}, \ldots, G_{e2}^{(li)}, \ldots, G_{e2}^{(kn)}\right]: \ \mathbb{R}^{ks} \to \mathbb{R},^{kn}$$

where $G_{e2}^{(li)}: \mathbb{R}^{ks} \to \mathbb{R}$ is defined via (18) as

$$G_{e2}^{(li)}(V) = 1 \Big/ \sum_{h=1}^{k} \left[\frac{d_f(\mathbf{v}_l, \mathbf{x}_i)}{d_f(\mathbf{v}_h, \mathbf{x}_i)}\right]^{1/(\alpha-1)}.$$

According to (19), we know that $\{\mathbf{v}_l \to d_f(\mathbf{v}_l, \mathbf{x}_i)\}$ is continuous. Since the sum of continuous functions is continuous, $G_{e2}^{(li)}(V)$ is the quotient of two continuous scalar fields for all $1 \leq l \leq k$,$1 \leq i \leq n$. In view of our general hypothesis that $d_f(\mathbf{v}_l, \mathbf{x}_i) > 0 \, \forall l, i$, $G_{e2}^{(li)}$ is continuous for all l, i. Therefore, G_{e2} is continuous on their entire domains. Thus, $J = G_{e2} \circ G_{e1}$ is continuous on $(M_{fw} \times H_{fv})$. □

Theorem 8. *J_n is continuous on $(M_{fw} \times H_{fv})$.*

Proof. Similar to Theorem 7. □

The final condition needed for Theorem 6 is compactness of $(M_{fw} \times H_{fv})$, which contains all of the possible iterate sequences generated by J_e and J_n.

Theorem 9. *$M_{fw} \times H_{fv}$ is a compact set.*

Proof. Since H_{fv} is the k-fold Cartesian product of the convex hull of H_{hv} and H_{hv} is a finite set, H_{fv} is closed and bound in $\mathbb{R}^{ks}$. Therefore, H_{fv} is compact. Similarly, since M_{fw} is the k-fold Cartesian product of the convex hull of M_{hw} and M_{hw} is a finite set, M_{fw} is closed and bound in $\mathbb{R}^{kn}$. Therefore, M_{fw} is compact. Thus, $M_{fw} \times H_{fv}$ is compact. □

We now assemble the hypotheses and results of the above theorems into a formal statement for convergence of the MKM_NOF and MKM_NDM algorithms.

Theorem 10. *The MKM_NOF algorithm $(\alpha > 1)$ either terminates at a point (W^*, V^*) in the solution set Ω or a subsequence exists convergent to a point in Ω where*

$$\begin{aligned}\Omega = \{(W^*, V^*) \in M_{fw} \times H_{fv} | F_e(W^*, V^*) \\ \leq F_e(W, V^*) \ and \ F_e(W^*, V^*) \\ \leq F_e(W^*, V) \ for \ all \ V \in H_{fv}\}.\end{aligned}$$

Theorem 11. *The MKM_NDM algorithm $(\alpha > 1)$ either terminates at a point (W^*, V^*) in the solution set Ω or a subsequence exists convergent to a point in Ω where*

$$\begin{aligned}\Omega = \{(W^*, V^*) \in M_{fw} \times H_{fv} | F_n(W^*, V^*) \\ \leq F_n(W, V^*) \ and \ F_n(W^*, V^*) \\ \leq F_n(W^*, V) \ for \ all \ V \in H_{fv}\}.\end{aligned}$$

7 Experimental Results

The main aim of this section is to illustrate the convergence results and evaluate the clustering performance and efficiency of the MKM_NOF and MKM_NDM algorithms. We used five datasets obtained from the UCI Machine Learning Repository [33] to test the proposed algorithms. These datasets are shown in Table 1.

TABLE 1
The Five Datasets from UCI

Data set	Objects	Attributes	Clusters
Soybean	47	35	4
Heart disease	303	8	2
Breast cancer	699	9	2
Mushroom	8124	22	2
Connect-4	67557	45	3

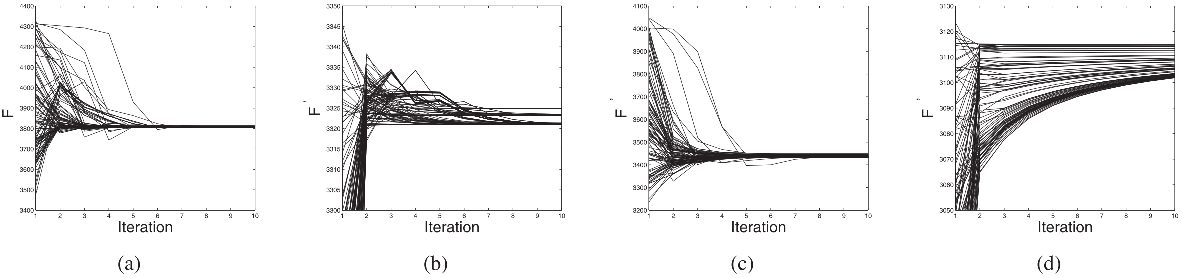


Fig. 2. (a) The objective function values F' against the iterations with different initial guesses when $p_{lj} = 1$ for $1 \leq l \leq k$, $1 \leq j \leq m$, and $\alpha = 1$. (b) The objective function values F' against the iterations with different initial guesses when $p_{lj} = 1$ for $1 \leq l \leq k$, $1 \leq j \leq m$, and $\alpha = 1.5$. (c) The objective function values F' against the iterations with different initial guesses when $p_{lj} = n_j$ for $1 \leq l \leq k$, $1 \leq j \leq m$, and $\alpha = 1$. (d) The objective function values F' against the iterations with different initial guesses when $p_{lj} = n_j$ for $1 \leq l \leq k$, $1 \leq j \leq m$, and $\alpha = 1.5$.

7.1 Convergence Results

For the existing modified k-modes [20], [21], [22], [23], [24], MKM_NOF and MKM_NDM algorithms, we tested the convergence of their hard and fuzzy clustering processes, i.e., $\alpha = 1$ and $\alpha = 1.5$, respectively. In the testing procedure, we carried out 100 runs of these algorithms on the breast cancer dataset, respectively. In each run, different initial cluster prototypes were used in these algorithms. The convergence behaviors are shown in Figs. 2 and 3. In each subfigure, we show the 100 curves, where each curve refers to the objective function values with the iterations of an algorithm in each run.

Fig. 2 shows the convergence behaviors of the objective function F' with random initializations and different parameters. When the parameters $p_{lj} = 1$ for $1 \leq l \leq k$, $1 \leq j \leq m$, and $\alpha = 1$, the objective function F' represents the algorithm proposed by He and Ng et al. [22], [23]. When the parameters $p_{lj} = 1$ for $1 \leq l \leq k$, $1 \leq j \leq m$, and $\alpha > 1$, the objective function F' represents the algorithm proposed by Ng et al. [24]. When the parameters $p_{lj} = n_j$ for $1 \leq l \leq k$, $1 \leq j \leq m$, and $\alpha = 1$, the objective function F' represents the algorithm proposed by San et al. [20]. When the parameters $p_{lj} = n_j$ for $1 \leq l \leq k$, $1 \leq j \leq m$, and $\alpha > 1$, the objective function F' represents the algorithm proposed by Kim et al. [21]. According to Fig. 2, we see that some of the sequences of the objective function values generated by these algorithms are not decreasing in iterative processes. This indicates that they cannot guarantee to obtain the local minimum solutions of their objective functions in the clustering processes.

Fig. 3 illustrates the convergence behaviors of the MKM_NOF and MKM_NDM algorithms on the breast cancer dataset. It is clear from Fig. 3 that the objective function values are decreasing in each curve. We also see in these subfigures that the MKM_NOF and MKM_NDM algorithms stop after a finite number of iterations, i.e., the objective function values do not decrease any more. This is exactly the results we showed in Section 6. Therefore, the MKM_NOF and MKM_NDM algorithms can be used safely.

7.2 Performance Results

To evaluate the performance of clustering algorithms, we considered three widely used evaluation methods.

The category utility (CU) function: The category utility function [35] is an internal criterion which attempts to maximize both the probability that two data objects in the same cluster obtain the same attribute values and the probability that data points from different clusters have different attributes. CU is defined as follows:

$$CU = \sum_{l=1}^{k} \frac{|c_l|}{n} \sum_{j=1}^{m} \sum_{q=1}^{n_j} \Big[P(a_j^{(q)}|c_l)^2 - P(a_j^{(q)})^2 \Big],$$

where

$$P(a_j^{(q)}|c_l) = \frac{|\{\mathbf{x}_i | x_{ij} = a_j^{(q)}, \mathbf{x}_i \in c_l\}|}{|c_l|}, P\big(a_j^{(q)}\big)$$
$$= \frac{|\{\mathbf{x}_i | x_{ij} = a_j^{(q)}, \mathbf{x}_i \in U\}|}{n},$$

and c_l is a set of objects in the lth cluster.

The adjusted rand index: The adjusted rand index is an external criterion which attempts to measure the similarity between two partitions of objects in the same dataset. Given a set U of n data objects and two groupings (e.g.,

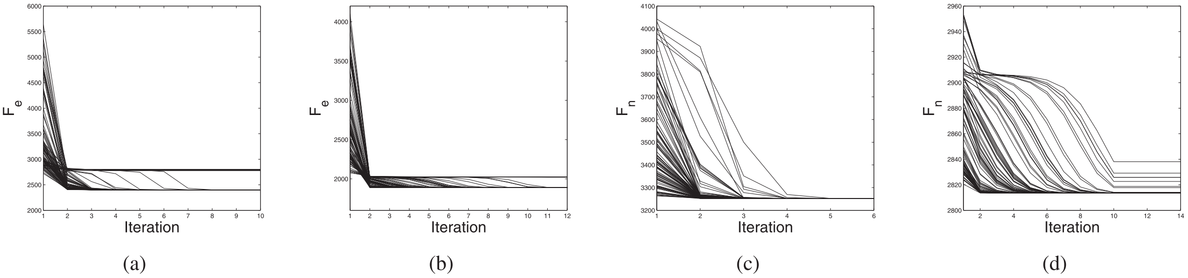


Fig. 3. (a) The objective function values F_e against the iterations with different initial guesses when $\alpha = 1$. (b) The objective function values F_e against the iterations with different initial guesses when $\alpha = 1.5$. (c) The objective function values F_n against the iterations with different initial guesses when $\alpha = 1$. (d) The objective function values F_n against the iterations with different initial guesses when $\alpha = 1.5$.

TABLE 2
Notation for the Contingency Table for Comparing Two Partitions

$C\backslash P$	p_1	p_2	$\cdots$	$p_{k'}$	$Sums$
c_1	n_{11}	n_{12}	$\cdots$	$n_{1k'}$	b_1
c_2	n_{21}	n_{22}	$\cdots$	$n_{2k'}$	b_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
c_k	n_{k1}	n_{k2}	$\cdots$	$n_{kk'}$	b_k
$Sums$	d_1	d_2	$\cdots$	$d_{k'}$	

clusterings) of these objects, namely, $C = \{c_1, c_2, \ldots, c_k\}$ and $P = \{p_1, p_2, \ldots, p_{k'}\}$, the overlappings between C and P can be summarized in a contingency table where n_{ij} denotes the number of common objects of groups c_i and p_{lj}: $n_{ij} = |c_i \cap p_{lj}|$. The adjusted rand index is defined as $AdjustedIndex = \frac{Index-ExpectedIndex}{MaxIndex-ExpectedIndex}$, more specifically,

$$ARI = \frac{\sum_{ij}\binom{n_{ij}}{2} - [\sum_i \binom{b_i}{2}\sum_j \binom{d_j}{2}]/\binom{n}{2}}{\frac{1}{2}[\sum_i \binom{b_i}{2} + \sum_j \binom{d_j}{2}] - [\sum_i \binom{b_i}{2}\sum_j \binom{d_j}{2}]/\binom{n}{2}},$$

where n_{ij}, b_i, d_j are values from the contingency table (Table 2). Since these given datasets contain the clustering label on each data object, we will evaluate the clustering results by using ARI to compare them with the original clustering labels. If the clustering result is close to the true class distribution, then the value of ARI is high.

The set matching technique: This category of methods is based on measuring the shared set cardinality between two clusterings. Similar to the adjusted rand index, the set matching technique is also an external criterion in which external information-class labels need be used. It computes the best matches between clusters (in terms of shared points) from each of the two clusterings and returns a value equal to the total number of points shared between pairs of matched clusters. The simplest form of the set matching technique is called the set matching accuracy (AC) [37], which is defined as

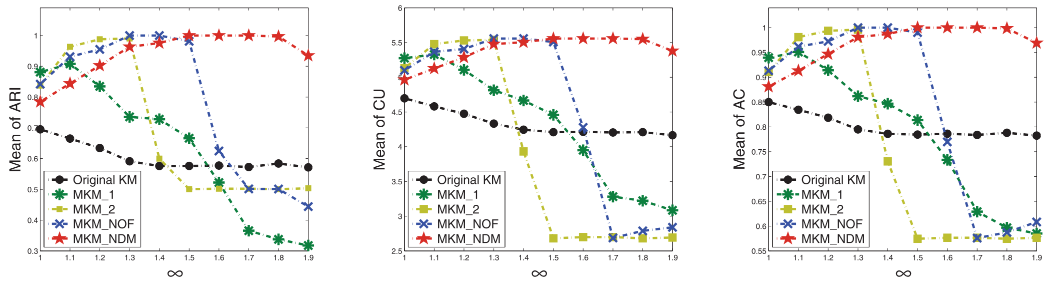
$$AC = \sum_{i=1}^{k} \frac{\max_{j=1}^{k'} n_{ij}}{n},$$

where n_{ij}, k, k' are values from Table 2. If the clustering result is close to the true class distribution, then the value of AC is high.

Based on the above evaluation measures, we compared the proposed algorithms with the existing k-modes type algorithms [16], [19], [20], [21], [22], [23], [24], [25] on four real datasets: the soybean data, the heart disease data, the breast cancer data, and the mushroom data, respectively. To ensure that the comparisons were in a uniform environmental condition, we first set the number of clusters is equal to the "true" number of classes for each of the given datasets. Next, due to the fact that the performance of these algorithms depends on initial cluster centers, we randomly selected 100 initial cluster prototypes for each of the given datasets. Furthermore, we selected 10 values of α that were from 1 to 1.9 with step length of 0.1. For each value of α, we carried out 100 runs of each algorithm on each dataset and computed the average values of its 100 clustering results for ARI, CU, and AC. When $\alpha > 1$, these algorithms produced a fuzzy partition matrix W. We obtained the cluster memberships from W as follows. The object $\mathbf{x}_i$ was assigned to the lth cluster if $w_{li} = \max_{1 \leq h \leq k} w_{hi}$. If the maximum was not unique, then $\mathbf{x}_i$ was assigned to the cluster of first achieving the maximum. Since the convergence of these existing k-modes type algorithms cannot be guaranteed, we set the maximum number of their iterations in each run as 30. For the MKM_NOF algorithm, we set $\gamma = 0.03 * n$ in the experimental analysis (we tried several values of γ and found that the value of γ/n in the interval $[0.01, 0.05]$ can provide the better clustering results on most real datasets), where n is the number of objects.

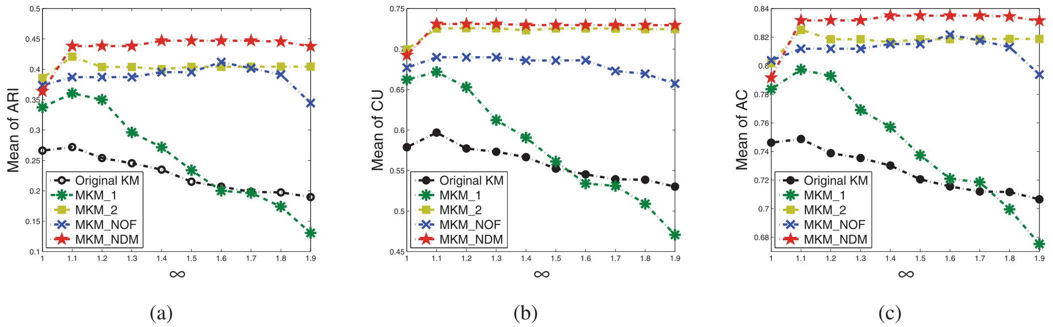
Figs. 4, 5, 6, and 7 show the comparison results of the these algorithms with different α values. In these figures, "Original KM," "MKM_1," and "MKM_1" stand for the original k-modes type algorithm, the generalized k-modes type algorithm with $p_{lj} = 1$ for $1 \leq l \leq k$ and $1 \leq j \leq m$ (nbsp;equivalent to He et al. and Ng et al.'s algorithms) and the generalized k-modes type algorithm with $p_{lj} = n_j$ for $1 \leq l \leq k$ and $1 \leq j \leq m$ (equivalent to San et al. and Kim et al.'s algorithms), respectively.

According to Figs. 4, 5, 6, and 7, we see that MKM_NOF and MKM_NDM algorithms can effectively enhance the performance of the original k-modes type algorithms and are superior to the MKM_1 algorithm for ARI, CU, and AC. Moreover, we also see that the performance of the MKM_NDM algorithm is slightly better than the MKM_2 algorithm. Compared to the MKM_NDM algorithm, the clustering results of the MKM_NOF algorithm are sensitive to the change of α values. This indicates that the MKM_NDM algorithm has much better robustness than the MKM_NOF algorithm. We found that the performance of the MKM_NOF



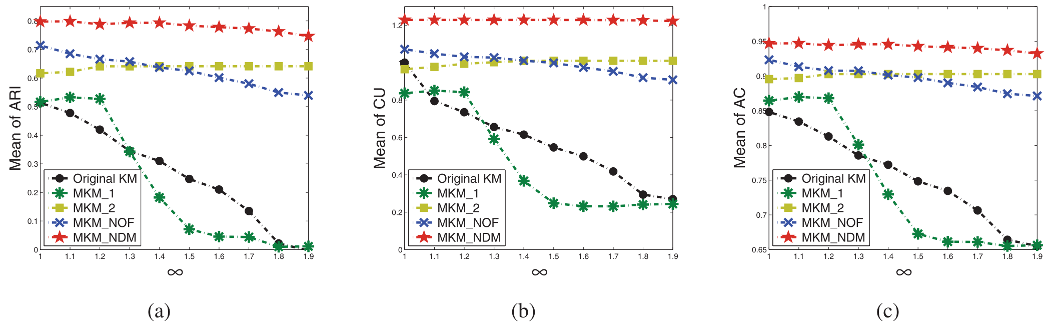
(a) (b) (c)

Fig. 4. (a) Means of ARI with respect to different values of α on the soybean data. (b) Means of CU with respect to different values of α on the soybean data. (c) Means of AC with respect to different values of α on the soybean data.



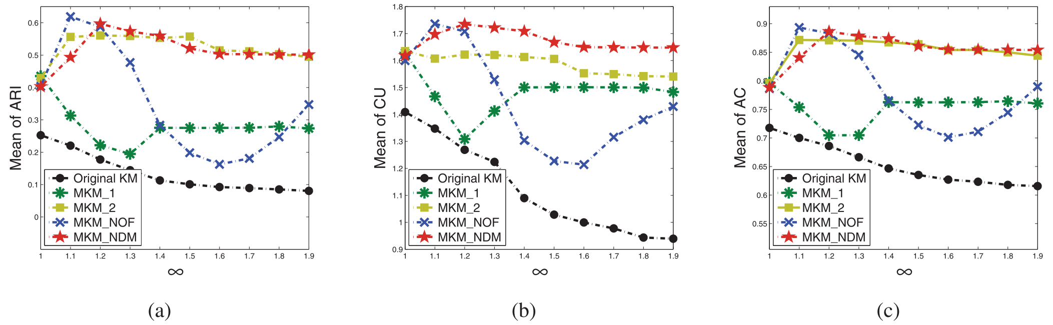
(a) (b) (c)

Fig. 5. (a) Means of ARI with respect to different values of α on the heart disease data. (b) Means of CU with respect to different values of α on the heart disease data. (c) Means of AC with respect to different values of α on the heart disease data.



(a) (b) (c)

Fig. 6. (a) Means of ARI with respect to different values of α on the breast cancer data. (b) Means of CU with respect to different values of α on the breast cancer data. (c) Means of AC with respect to different values of α on the breast cancer data.



(a) (b) (c)

Fig. 7. (a) Means of ARI with respect to different values of α on the mushroom data. (b) Means of CU with respect to different values of α on the mushroom data. (c) Means of AC with respect to different values of α on the mushroom data.

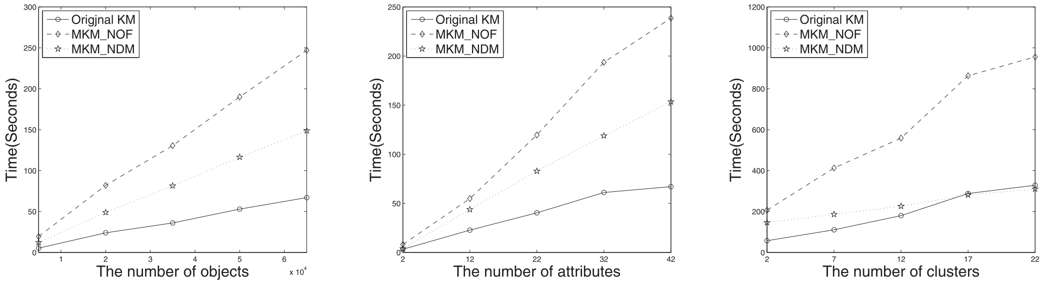
algorithm with the α value in the interval $[1, 1.5)$ is close to that of the MKM_NDM and MKM_2 algorithms.

Therefore, the above experimental results tell us that the proposed algorithms can not only guarantee to be convergent but can also obtain the better clustering results.

7.3 Scalability Results

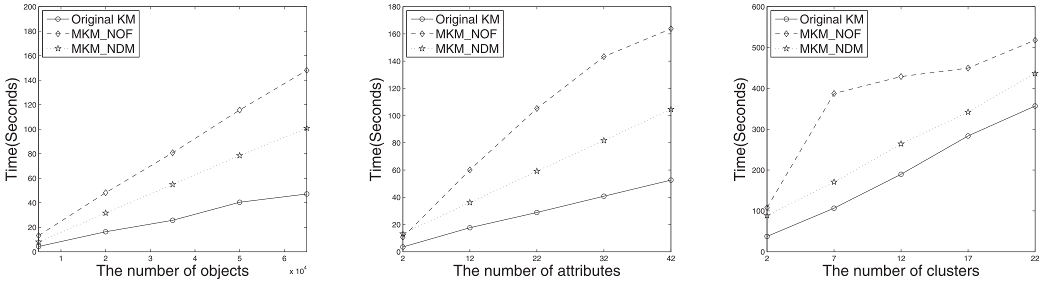
In the scalability analysis, we tested the original k-modes type algorithm, the MKM_NOF algorithm, and the MKM_NDM algorithm on the connect-4 dataset from UCI [33]. The computational results were performed by using a machine with an Intel Q9400 and 2 G RAM. The computational times of algorithms were plotted with respect to the number of objects, attributes, and clusters, while the other corresponding parameters were fixed. All of the experiments were repeated five times and the average computational times were depicted. For each of the three algorithms, we tested the computational times of the hard and fuzzy clustering processes, i.e., $a = 1$ and $a = 1.5$, respectively.

Figs. 8a and 9a show the computational times against the number of objects while the number of attributes is 42 and the number of clusters is 3. Figs. 8b and 9b show the computational times against the number of attributes while the number of clusters is 3 and the number of objects is 680,000. Figs. 8c and 9c show the computational times against the number of clusters while the number of attributes is 42 and the number of objects is 680,000. According to the figures, we see that all three algorithms are scalable, i.e., the computational times increase linearly with respect to either the number of objects, attributes, or clusters. The MKM_NOF and MKM_NDM algorithms require more computational times than the original k-mode type algorithms. It is an expected outcome since the proposed algorithms require some additional arithmetic operations for the weight calculation of cluster prototypes and iterations for searching a local minimal solution compared to the original k-mode type algorithms. However, according to the tests, the MKM_NOF and MKM_NDM algorithms are still scalable, i.e., they can cluster categorical objects efficiently. In addition, we also



(a) (b) (c)

Fig. 8. The runtimes of three algorithms on the connect-4 data with $\alpha = 1$. (a) Computational times for different numbers of objects. (b) Computational times for different numbers of attributes. (c) Computational times for different numbers of clusters.



(a) (b) (c)

Fig. 9. The runtimes of three algorithms on the connect-4 data with $\alpha = 1.5$. (a) Computational times for different numbers of objects. (b) Computational times for different numbers of attributes. (c) Computational times for different numbers of clusters.

see that the computational times of the MKM_NOF algorithm are more than the MKM_NDM algorithm because the MKM_NOF algorithm requires more iterations than the MKM_NDM algorithm in clustering process.

8 Conclusion

In this paper, we have analyzed the convergence of several modified k-modes algorithms using the frequency-based cluster prototypes. It is proven that these modified algorithms cannot converge to the local minimum solutions of their objective functions unless they degrade to the original k-modes type algorithms. To remedy this shortcoming, we have proposed two new modified algorithms, called MKM_NOF and MKM_NDM, respectively, which apply different techniques to represent a cluster by weighted cluster prototypes. We rigorously derive the updating formulas of the two algorithms and prove their convergence under their optimization frameworks. Experimental results have shown that the MKM_NOF and MKM_NDM algorithms are efficient and effective in clustering categorical datasets.

Acknowledgments

The authors are very grateful to the editors and reviewers for their valuable comments and suggestions. This work was supported by the National Natural Science Foundation of China (Nos. 71031006, 70971080), GRF: CityU 112809 of the Hong Kong SAR Government, the National Key Basic Research and Development Program of China (973) (No. 2013CB329404), the Special Prophase Project on National Key Basic Research and Development Program of China (973) (No. 2011CB311805).

References

[1] A.K. Jain and R.C. Dubes, *Algorithms for Clustering Data.* Prentice Hall, 1988.

[2] N. Wrigley, *Categorical Data Analysis for Geographers and Environmental Scientists.* Longman, 1985.

[3] C.C. Aggarwal, C. Magdalena, and P.S. Yu, "Finding Localized Associations in Market Basket Data," *IEEE Trans. Knowledge and Data Eng.* vol. 14, no. 1, pp. 51-62, Jan./Feb. 2002.

[4] A. Baxevanis, *Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins,* second ed. Wiley, 2001.

[5] D. Barbara et al., *Applications of Data Mining in Computer Security.* Kluwer, 2002.

[6] K.C. Gowda and E. Diday, "Symbolic Clustering Using a New Dissimilarity Measure," *Pattern Recognition,* vol. 24, no. 6, pp. 567-578, 1991.

[7] D.H. Fisher, "Knowledge Acquisition via Incremental Conceptual Clustering," *Machine Learning,* vol. 2, no. 2, pp. 139-172, 1987.

[8] S. Guha, R. Rastogi, and S. Kyuseok, "ROCK: A Robust Clustering Algorithm for Categorical Attributes," *Proc. 15th Int'l Conf. Data Eng.,* vol. 23-26, pp. 512-521, 1999.

[9] V. Ganti, J.E. Gekhre, and R. Ramakrishnan, "CACTUS-Clustering Categorical Data Using Summaries," *Proc. Fifth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining,* pp. 73-83, 1999.

[10] D. Barbara, Y. Li, and J. Couto, "Coolcat: An Entropy-Based Algorithm for Categorical Clustering," *Proc. 11th Int'l Conf. Information and Knowledge Management,* pp. 582-589, 2002.

[11] P. Andritsos, P. Tsaparas, R. Miller, and K. Sevcik, "LIMBO: Scalable Clustering of Categorical Data," *Proc. Ninth Int'l Conf. Extending Database Technology,* pp. 123-146, 2004.

[12] E. Cesario, G. Manco, and R. Ortale, "Top-Down Parameter-Free Clustering of High-Dimensional Categorical Data," *IEEE Trans. Knowledge and Data Eng.,* vol. 19, no. 12, pp. 1607-1624, Dec. 2007.

[13] F.Y. Cao et al., "A Framework for Clustering Categorical Time-Evolving Data," *IEEE Trans. Fuzzy Systems,* vol. 18, no. 5, pp. 872-885, Oct. 2010.

[14] T.K. Xiong, S.R. Wang, A. Mayers, and E. Monga, "A New MCA-Based Divisive Hierarchical Algorithm for Clustering Categorical Data," *Proc. Ninth IEEE Int'l Conf. Data Mining,* pp. 1058-1063, 2009.

[15] T.K. Xiong, S.R. Wang, A. Mayers, and E. Monga, "DHCC: Divisive Hierarchical Clustering of Categorical Data," *Data Mining and Knowledge Discovery,* vol. 24, no. 1, pp. 103-135, 2012.

[16] Z.X. Huang, "A Fast Clustering Algorithm to Cluster Very Large Categorical Data Sets in Data Mining," *Proc. SIGMOD Workshop Research Issues on Data Mining and Knowledge Discovery,* pp. 1-8, 1997.

[17] Z.X. Huang, "Extensions to the *k*-Means Algorithm for Clustering Large Data Sets with Categorical Values," *Data Mining and Knowledge Discovery,* vol. 2, no. 3, pp. 283-304, 1998.

[18] J.B. MacQueen, "Some Methods for Classification and Analysis of Multivariate Observations," *Proc. Fifth Berkeley Symp. Math. Statistics and Probability,* vol. 1, pp. 281-297, 1967.

[19] Z.X. Huang and M.K. Ng, "A Fuzzy *k*-Modes Algorithm for Clustering Categorical Data," *IEEE Trans. Fuzzy Systems,* vol. 7, no. 4, pp. 446-452, Aug. 1999.

[20] O. San, V. Huynh, and Y. Nakamori, "An Alternative Extension of the *k*-Means Algorithm for Clustering Categorical Data," *Int'l J. Applied Math. and Computer Science,* vol. 14, no. 2, pp. 241-247, 2004.

[21] D.W. Kim, K.Y. Lee, D.K. Lee, and K.H. Lee, "A *k*-Populations Algorithm for Clustering Categorical Data," *Pattern Recognition,* vol. 38, no. 3, pp. 1131-1134, 2005.

[22] Z. He, S. Deng, and X. Xu, "Improving *k*-Modes Algorithm Considering Frequencies of Attribute Values in Mode," *Proc. Int'l Conf. Computational Intelligence and Security,* pp. 157-162, 2005.

[23] M.K. Ng, M.J. Li, Z.X. Huang, and Z.Y. He, "On the Impact of Dissimilarity Measure in *k*-Modes Clustering Algorithm," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 3, pp. 503-507, Mar. 2007.

[24] M.K. Ng and L.P. Jing, "A New Fuzzy K-Modes Clustering Algorithm for Categorical Data," *Int'l J. Granular Computing Rough Sets and Intelligent Systems,* vol. 1, no. 1, pp. 105-119, 2009.

[25] M. Lee and W. Pedrycz, "The Fuzzy C-Means Algorithm with Fuzzy P-Mode Prototypes for Clustering Objects Having Mixed Features," *Fuzzy Sets and Systems,* vol. 160, no. 24, pp. 3590-3600, 2009.

[26] H.L. Chen, K.T. Chuang, and M.S. Chen, "On Data Labeling for Clustering Categorical Data," *IEEE Trans. Knowledge and Data Eng.,* vol. 20, no. 11, pp. 1458-1472, Nov. 2008.

[27] E.T. Jaynes, "Information Theory and Statistical Mechanics," *Physical Rev. Series II,* vol. 106, no. 4, pp. 620-630, 1957.

[28] S. Miyamoto and M. Mukaidono, "Fuzzy C-Means as a Regularization and Maximum Entropy Approach," *Proc. Seventh Int'l Fuzzy Systems Assoc. World Congress,* vol. 2, pp. 86-92, 1997.

[29] L.P. Jing, M.K. Ng, and Z.X. Huang, "An Entropy Weighting *k*-Means Algorithm for Subspace Clustering of High-Dimensional Sparse Data," *IEEE Trans. Knowledge and Data Eng.,* vol. 19, no. 8, pp. 1026-1041, Aug. 2007.

[30] J.C. Bezdek, "A Convergence Theorem for the Fuzzy ISODATA Clustering Algorithms," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 2, no. 1, pp. 1-8, Jan. 1980.

[31] J.C. Bezdek, R.J. Hathaway, M.J. Sabin, and W.T. Tucker, "Convergence Theory for Fuzzy C-Means: Counterexamples and Repairs," *IEEE Trans. Systems, Man, and Cybernetics,* vol. 17, no. 5, pp. 873-877, Sept./Oct. 1987.

[32] W. Zangwill, *Nonlinear Programming: A Unified Approach,* chapter 4. Prentice-Hall, 1969.

[33] UCI Machine Learning Repository, http://www.ics.uci.edu/mlearn/MLRepository.html, 2010.

[34] M. Zait and H. Messatfa, "A Comparative Study of Clustering Methods," *Future Generation Computer Systems,* vol. 13, pp. 149-159, 1997.

[35] M.A. Gluck and J.E. Corter, "Information Uncertainty and the Utility of Categories," *Proc. Seventh Ann. Conf. Cognitive Science Soc.,* pp. 283-287, 1985.

[36] L. Hubert and P. Arabie, "Comparing Partitions," *J. Classification,* vol. 2, no. 1, pp. 193-218, 1985.

[37] Y.M. Yang, "An Evaluation of Statistical Approaches to Text Categorization," *J. Information Retrieval,* vol. 1, nos. 1/2, pp. 67-88, 1999.

Liang Bai received the MS degree in computer science from Shanxi University in 2009, and is working toward the PhD degree from the School of Computer and Information Technology at Shanxi University, China. His research interests are in the areas of data mining and machine learning.

Jiye Liang received the MS and PhD degrees from Xi'an Jiaotong University, Xi'an, China, in 1990 and 2001, respectively. He is currently a professor with the School of Computer and Information Technology and the Key Laboratory of Computational Intelligence and Chinese Information Processing of the Ministry of Education, Shanxi University, Taiyuan, China. He has authored or coauthored more than 150 journal papers in his research fields. His current research interests include computational intelligence, granular computing, data mining, and machine learning.

Chuangyin Dang received the MS degree in applied mathematics from Xidian University, Xian, China, in 1986, and the PhD degree in operations research/economics from the University of Tilburg, The Netherlands, in 1991. He is currently an associate professor with the Department of Systems Engineering and Engineering Management, City University of Hong Kong, Kowloon, Hong Kong. His research interests include computational intelligence and optimization theory and technology. He is a senior member of the IEEE.

Fuyuan Cao received the MS and PhD degrees in computer science in 2004 and 2009, respectively, from Shanxi University, Taiyuan, China, where he is currently an associate professor with the School of Computer and Information Technology. His research interests include data mining and machine learning.