Knowledge-Based Systems 24 (2011) 785–795

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

An initialization method to simultaneously find initial cluster centers and the number of clusters for clustering categorical data

Liang Bai [a,b], Jiye Liang [a,*], Chuangyin Dang [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 Shanxi, China

[b] Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong, Hong Kong

ARTICLE INFO

Article history:
Received 19 April 2010
Received in revised form 21 February 2011
Accepted 24 February 2011
Available online 2 March 2011

Keywords:
The *k*-modes-type algorithms
Categorical data
Initial cluster centers
The number of clusters
Density measure

ABSTRACT

The leading partitional clustering technique, *k*-modes, is one of the most computationally efficient clustering methods for categorical data. However, in the *k*-modes-type algorithms, the performance of their clustering depends on initial cluster centers and the number of clusters needs be known or given in advance. This paper proposes a novel initialization method for categorical data which is implemented to the *k*-modes-type algorithms. The proposed method can not only obtain the good initial cluster centers but also provide a criterion to find candidates for the number of clusters. The performance and scalability of the proposed method has been studied on real data sets. The experimental results illustrate that the proposed method is effective and can be applied to large data sets for its linear time complexity with respect to the number of data points.

1. Introduction

Clustering is a process of grouping a set of data points into clusters so that the data points in the same cluster have high similarity but are very dissimilar with data points in other clusters. Various types of clustering algorithms have been proposed in the literature (e.g., [1] and references therein). Clustering algorithms are generally categorized under two different categories, partitional and hierarchical. The *k*-means algorithm [1–4] is a well known partitional clustering algorithm for its efficiency in clustering large data sets. Fuzzy versions of the *k*-means algorithm have been reported by Ruspini [5] and Bezdek [6], where each pattern is allowed to have memberships in all clusters rather than having a distinct membership to one single cluster. Numerous problems in real world applications, such as pattern recognition and computer vision, can be tackled effectively by the fuzzy *k*-means algorithm. However, the use of the *k*-means-type algorithms is only limited to numeric data.

Due to the fact that large categorical data sets exist in many applications, it has been widely recognized that directly clustering the raw categorical data is important. Examples include environmental data analysis [7], market basket data analysis [8], DNA or protein sequence analysis [9], text mining [10], and computer security [11]. Therefore, in 1997, Huang [12,13] extended the *k*-means algorithm to propose the *k*-modes algorithm whose extensions have removed the numeric-only limitation of the *k*-means algorithm and enable the *k*-means clustering process to be used to efficiently cluster large categorical data sets from real world databases. Furthermore, Huang and Ng introduced the fuzzy *k*-modes algorithm [14], a generalized version of the *k*-modes algorithm, which assigns membership degrees to data in different clusters. Since first published, the *k*-modes-type algorithms have become a popular technique for solving problems about clustering categorical data in different application domains [15].

There are two major issues in the application of the *k*-means-type (nonfuzzy or fuzzy) algorithms and the *k*-modes-type algorithms in cluster analysis. The first issue is that these algorithms use alternating minimization methods to solve nonconvex optimization problems in finding cluster solutions [1]. These algorithms require a set of initial cluster centers to start and often end up with different clustering results for different sets of initial cluster centers. Therefore, they are very sensitive to the initial cluster centers. Usually, they begin with an initial set of randomly selected cluster centers. Due to its simplicity, the random initialization method has been widely used. However, these algorithms need to rerun many times with different initializations in an attempt to find a good solution. Moreover, the random initialization method works well only when at least one of the random initializations is close to a good solution. Therefore, how to select initial cluster centers is extremely important as they have a direct impact on the formation of final clusters. For numeric data, many attempts have been reported to solve the initialization problem. For example, several experts

* Corresponding author.
E-mail addresses: sxbailiang@126.com (L. Bai), ljy@sxu.edu.cn (J. Liang), mecdang@cityu.edu.hk (C. Dang).

0950-7051/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.knosys.2011.02.015

[16–19] used genetic algorithms to obtain good initial cluster centers. Arthur and Vassilvitskii [20] proposed and studied a careful seeding for initial cluster centers to improve the performance. Cao et al. [21] presented an selection method based on the neighborhood rough-set model. Likas et al. [22] proposed a global k-means clustering algorithm (GKM) to solve the local minimum problem. However, due to the lack of intuitive geometric properties for categorical data, these techniques for numerical data cannot be applicable to categorical data. For categorical data, Huang [13] suggested to select the first k distinct objects from the data set as the initial k modes or assign the most frequent categories equally to the initial k modes. Although the methods are to make the initial modes diverse, an uniform criteria is not given for selecting k initial modes in [13]. Sun [23] introduced an initialization method which is based on the frame of refining. This method presents a study on applying Bradley's iterative initial-point refinement algorithm [24] to the k-modes clustering, but its time cost is high and the parameters of this method are plenty which need to be asserted in advance. Wu [25] proposed an initialization method based on density and distance, which limits the process in a sub-sample data set and uses a refining framework. But this method needs to randomly select sub-sample, so the sole clustering result can not be guaranteed. Cao [26] redefined the density of data points and proposed the new initialization method which is scalable and capable of dealing with large categorical data sets. However, Wu [25] and Cao [26] proposed that the density of a data point is the total similarity measures between the data point and all data points, which do not consider the local density of a data point. In many cases, the global density of the boundary points among the clusters are often higher than the cluster centers. In summary, currently there are no universally accepted method for obtaining initial cluster centers for categorical data. Hence, it is very necessary to propose a new cluster centers initialization method for categorical data.

The second issue is that the number of clusters k needs to be determined in advance as an input to clustering algorithms. In a real data set, k is usually unknown. In practice, different values of k are tried, and cluster validation techniques are used to measure the clustering results and determine the best value of k, see, for instance, [1]. In [27], Hamerly and Elkan studied statistical methods to learn k in the k-means-type algorithms. In [28], Li et al. presented an agglomerative fuzzy k-means clustering algorithm for numerical data, an extension to the standard fuzzy k-means algorithm by introducing a penalty term to the objective function. The algorithm can determine the number of clusters by analyzing the penalty factor. However, the algorithm needs to randomly select a subset from data set as initial cluster centers, which results in an uncertainty. For categorical data, a bottom-up hierarchical algorithm ACE was proposed in [29], which uses entropy as an index function to capture the candidates for the number of clusters. However, when the data set is very large, the ACE algorithm is not efficient due to its computational burden of $O(n^2 log_2 n)$ with n being the number of data points.

In this paper, we propose a novel initialization method for categorical data to tackle the above two issues in application of the k-modes-type clustering algorithms. In cluster centers initialization, we do not use all data points as the potential exemplars but propose a method to construct a potential exemplars set. We define a new density measure to reflect the cohesiveness of potential exemplars. Based on the density measure and the distance measure, we select the initial cluster centers from the potential exemplars set. In determination of the number of clusters, we propose a criterion to find the candidates for the number of clusters. The proposed initialization method has been used along with the k-modes algorithm and the fuzzy k-modes algorithm, respectively. The time complexity of the proposed method has been analyzed. Comparisons with other initialization methods illustrate the effectiveness of this approach. The major research highlights are as follows:

- Study initialization problems about clustering categorical data. Categorical data is different from continuous or discretized numerical data. Due to the lack of an inherent order on the domains of the categorical attributes, the clustering techniques for numerical data cannot be applicable to categorical data.
- A new initialization method is proposed to simultaneously find the initial cluster centers and the number of clusters for categorical objects. In this paper, we took account of clustering not attributes but objects.
- We applied three widely used external evaluation measures to evaluate the effectiveness of the k-modes type algorithms with the proposed initialization method on several real data sets from UCI. These real data sets included the class information (class labels and the number of classes). The class information was not used to obtain the initial cluster centers and the number of clusters but was only used to evaluate the effectiveness of the proposed initialization method.

The outline of the rest of this paper is as follows. A detailed review of the k-modes algorithm and the fuzzy k-modes algorithm is presented in Section 2. In Section 3, a new cluster centers initialization method is proposed. In Section 4, a criterion to find the candidates for the number of clusters is presented. Section 5 demonstrates the performance and scalability of the new initialization method. Finally, a concluding remark is given in Section 6.

2. The k-modes algorithm and the fuzzy k-modes algorithm

As we know, the structural data are stored in a table, where each row (tuple) represents facts about a data point. A data table is also called an information system in rough set theory [31–35]. Data are prevalently described by categorical attributes [12,36,37]. More formally, a categorical data table is described as a quadruple $IS = (U, A, V, f)$, where:

(1) $U = \{x_1, x_2, \ldots, x_n\}$ is the nonempty set of n data points, called a universe;
(2) $A = \{a_1, a_2, \ldots, a_m\}$ is the nonempty set of m categorical attributes;
(3) V is the union of attribute domains, i.e., $V = \bigcup_{j=1}^{m} V_{a_j}$, where $V_{a_j} = \left\{a_j^{(1)}, a_j^{(2)}, \ldots, a_j^{(n_j)}\right\}$ is the value domain of categorical attribute a_j and is finite and unordered, e.g., for any $1 \leqslant p \leqslant q \leqslant n_j$, either $a_j^{(p)} = a_j^{(q)}$ or $a_j^{(p)} \neq a_j^{(q)}$. Here, n_j is the number of categories of attribute a_j for $1 \leqslant j \leqslant m$;
(4) $f : U \times A \to V$ is an information function such that $f(x_i, a_j) \in V_{a_j}$ for $1 \leqslant i \leqslant n$ and $1 \leqslant j \leqslant m$.

The objective of the k-modes-type algorithms [12–14] is to cluster data points in U into k clusters by minimizing the function

$$F(W, Z) = \sum_{l=1}^{k} \sum_{i=1}^{n} w_{li}^{\alpha} d(z_l, x_i)$$

subject to

$$w_{li} \in [0, 1], \quad 1 \leqslant l \leqslant k, \ 1 \leqslant i \leqslant n,$$

$$\sum_{l=1}^{k} w_{li} = 1, \quad 1 \leqslant i \leqslant n,$$

and

$$0 < \sum_{i=1}^{n} w_{li} < n, \quad 1 \leqslant l \leqslant k,$$

where n is the number of data points in U, $k(\leqslant n)$ is a given number of clusters. The parameter $\alpha \in [1,+\infty)$ is a positive coefficient for controlling the membership of each datum. $\alpha = 1$ gives the k-modes algorithm. $W = [w_{li}]$ is a k-by-n real matrix, w_{li} indicates whether x_i belongs to the lth cluster for the k-modes algorithm, $w_{li} = 1$ if x_i belongs to the lth cluster and 0 otherwise, and for the fuzzy k-modes algorithm, w_{li} is the membership degree of x_i to the lth cluster. $Z = \{z_1, z_2, \ldots, z_k\}$, and z_l is the lth cluster center with the categorical attributes $a_1, a_2, \ldots, a_m$, where m is the number of attributes.

To cluster categorical data, the k-modes-type algorithms apply the simple matching dissimilarity measure to compute the distance between a cluster center and a categorical data point, use modes as cluster centers instead of means for clusters, and update the cluster centers at each iteration to minimize the clustering cost function.

The simple matching dissimilarity measure $d(z_l, x_i)$ between a center z_l and a categorical data point x_i is defined as

$$d(z_l, x_i) = \sum_{j=1}^{m} \delta_{a_j}(z_l, x_i),$$

where

$$\delta_{a_j}(z_l, x_i) \begin{cases} 1, & f(z_l, a_j) \neq f(x_i, a_j), \\ 0, & f(z_l, a_j) = f(x_i, a_j). \end{cases}$$

It is easy to verify that $0 \leqslant d(z_l, x_i) \leqslant m$ and the function d defines a metric space on the set of categorical data points. We note that d is also a kind of generalized Hamming distance.

The lth cluster center z_l, referred to as the lth mode, is updated as follows. Each $f(z_l, a_j)$ for $a_j \in A$ is updated. For the k-modes algorithm ($\alpha = 1$), $f(z_l, a_j)$ satisfies the following criterion:

$$|\{x_i \in U | f(x_i, a_j) = f(z_l, a_j), w_{li} = 1\}| = \max_{q=1}^{n_j} |\{x_i \in U | f(x_i, a_j) = a_j^{(q)}, w_{li} = 1\}|$$

and satisfies the following criterion for the fuzzy k-modes algorithm ($\alpha > 1$):

$$\sum_{f(x_i, a_j) = f(z_l, a_j), x_i \in U} w_{li}^{\alpha} = \max_{q=1}^{n_j} \sum_{f(x_i, a_j) = a_j^{(q)}, x_i \in U} w_{li}^{\alpha}.$$

For the k-modes algorithm ($\alpha = 1$), $W = [w_{li}]$ is updated as

$$w_{li} = \begin{cases} 1, & \text{if } d(z_l, x_i) = \min\limits_{1 \leqslant h \leqslant k} d(z_h, x_i), \\ 0, & \text{otherwise} \end{cases}$$

and for fuzzy k-modes algorithm ($\alpha > 1$), $W = [w_{li}]$ is updated as

$$w_{li} = \begin{cases} 1, & \text{if } x_i = z_l, \\ 0, & \text{if } x_i = z_h, h \neq l, \\ 1 \Big/ \sum\limits_{h=1}^{k} \left[\frac{d(z_l, x_i)}{d(z_h, x_i)}\right]^{1/(\alpha-1)}, & \text{if } x_i \neq z_l \quad \text{and} \quad x_i \neq z_h, 1 \leqslant h \leqslant k. \end{cases}$$

The whole process of the k-modes-type algorithms is described as follows [14]:

Step 1: Choose an initial point set $Z^{(1)} \subseteq R^m$, where $R^m = V_{a_1} \times V_{a_2} \times \cdots \times V_{a_m}$. 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 goto 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 goto Step 2.

This procedure removes the numeric-only limitation of the k-means-type algorithm. Moreover, the fuzzy partition matrix provides more information to help the user to determine the final clustering and to identify the boundary data points. Such information is extremely useful in applications such as data mining in which the uncertain boundary data points are sometimes more interesting than data points which can be clustered with certainty. However, similar to the k-means-type algorithms, the k-modes-type algorithms are sensitive to initial cluster centers and need to give the number of clusters in advance. To solve these problems, a new initialization method for categorical data is proposed in Sections 3 and 4.

3. A new cluster centers initialization method

Currently, most cluster centers initialization methods generally consider all data points as potential exemplars or randomly choose a part of data points from a given data set to be as potential exemplars. The first method has very high computing cost and maybe weaken the representability of exemplars to clusters. The second method maybe result in an uncertainty.

We propose a new method to construct potential exemplars, instead of randomly selecting. Since categorical data can be partitioned into several subsets by each categorical attribute, for each subset, we do not consider all the data points in it as potential exemplars but select a data point to represent it and be as a potential exemplar. These chosen data points make up a set of potential exemplars. Similar to the selection method of cluster representative in the k-modes algorithm, in a given subset, we will select the most frequent attribute value in each attribute domain to compose its representative point. The formal definitions are as follows:

Definition 1. Let $IS = (U, A, V, f)$ be a categorical data table. A subset $U_{a_j^{(q)}} \subseteq U$ is defined as

$$U_{a_j^{(q)}} = \{x_i \in U | f(x_i, a_j) = a_j^{(q)}, 1 \leqslant i \leqslant n\}$$

for $1 \leqslant j \leqslant m$, $1 \leqslant q \leqslant n_j$.

Definition 2. Let $IS = (U, A, V, f)$ be a categorical data table. The representative point $c_{a_j^{(q)}}$ of the subset $U_{a_j^{(q)}}$ is defined as

$$c_{a_j^{(q)}} = \left[f(c_{a_j^{(q)}}, a_1), f(c_{a_j^{(q)}}, a_2), \ldots, f(c_{a_j^{(q)}}, a_m)\right],$$

where

$$|\{x | f(x, a_h) = f(c_{a_j^{(q)}}, a_h), x \in U_{a_j^{(q)}}\}| = \max_{t=1}^{n_h} \left|\left\{x | f(x, a_h) = a_h^{(t)}, x \in U_{a_j^{(q)}}\right\}\right|, \quad 1 \leqslant h \leqslant m$$

for $1 \leqslant j \leqslant m$, $1 \leqslant q \leqslant n_j$.

Property 1. *Let $IS = (U, A, V, f)$ be a categorical data table. The representative point $c_{a_j^{(q)}}$ of the subset $U_{a_j^{(q)}}$ satisfies*

$$\sum_{x \in U_{a_j^{(q)}}} d(c_{a_j^{(q)}}, x) = \min_{v \in R^m} \sum_{x \in U_{a_j^{(q)}}} d(v, x),$$

for $1 \leqslant j \leqslant m$, $1 \leqslant q \leqslant n_j$, where $R^m = V_{a_1} \times V_{a_2} \times \ldots \times V_{a_m}$.

Proof. For a given $U_{a_j^{(q)}}$, we write the sum of the distance between $v \in R^m$ and every object x in $U_{a_j^{(q)}}$ as

$$\sum_{x\in U_{a_j^{(q)}}} d(v,x) = \sum_{x\in U_{a_j^{(q)}}} \sum_{h=1}^{m} \delta_{a_h}(v,x) = \sum_{h=1}^{m} \sum_{x\in U_{a_j^{(q)}}} \delta_{a_h}(v,x)$$
$$= \sum_{h=1}^{m} (|U_{a_j^{(q)}}| - |\{x | f(x,a_h) = f(v,a_h), x \in U_{a_j^{(q)}}\}|).$$

It is clear that $\sum_{x\in U_{a_j^{(q)}}} d(v,x)$ is minimized iff $|\{x|f(x,a_h) = f(v,a_h), x \in U_{a_j^{(q)}}\}|$ is maximal for $1 \leqslant h \leqslant m$. The result follows. □

Intuitively, the smaller the sum of difference between a data point and each data point in a given subset is, the higher representability the data point has in the subset. Property 1 tells us that the selected representative point $c_{a_j^{(q)}}$ minimizes the sum of distance between the representative point and each data point in $U_{a_j^{(q)}}$. Therefore, $c_{a_j^{(q)}}$ has good representability in the subset $U_{a_j^{(q)}}$ and can be seen as a center point of $U_{a_j^{(q)}}$ which is similar to the mean of numerical data. For each subset, we use a center point (mode) of the subset as a potential exemplar to represent the subset. According to Definition 2, we know that the representative point of a subset $U_{a_j^{(q)}}$ is not unique. Therefore, while more than one representative point exists in the subset, we select one from them to be as the representative point at random or select one which is different from the representative points of other subsets.

Definition 3. Let $IS = (U,A,V,f)$ be a categorical data table. The potential exemplars set S is defined as

$$S = \{c_{a_j^{(q)}} | c_{a_j^{(q)}} \neq c_{a_i^{(p)}}, 1 \leqslant p \leqslant n_i, 1 \leqslant q \leqslant n_j, 1 \leqslant i < j \leqslant m\}.$$

According to the definition, we know that $|S| \leqslant |V|$, where $|V| = \sum_{i=1}^{m} n_j$. When the data set is very large, $|V| \ll |U|$.

Let us consider the following example to demonstrate the process of constructing a set of potential exemplars from a given data set.

Example 1. Given a categorical data table $IS = (U,A,V,f)$ shown in Table 1, where $U = \{x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8,x_9,x_{10},x_{11},x_{12},x_{13},x_{14},x_{15}\}$, $A = \{a_1,a_2,a_3,a_4\}$, $V_{a_1} = \{B,C,D\}$, $V_{a_2} = \{E,F,G\}$, $V_{a_3} = \{H,I,J,K\}$ and $V_{a_4} = \{L,M,N\}$. There are three classes with their modes and their five objects.

According to the attribute a_1, U is partitioned into three subsets:

$$U_B = \{x_2,x_3,x_4,x_5\}, \quad U_C = \{x_6,x_7,x_8,x_{10},x_{12}\}, \quad U_D = \{x_1,x_9,x_{11},x_{13},x_{14},x_{15}\};$$

Table 1
An example data set.

Objects\attributes	a_1	a_2	a_3	a_4
x_1	D	E	H	L
x_2	B	E	I	L
x_3	B	G	H	N
x_4	B	E	H	L
x_5	B	E	K	L
The mode z_1 of Class 1	B	E	H	L
x_6	C	F	J	M
x_7	C	F	I	N
x_8	C	G	I	M
x_9	D	E	H	M
x_{10}	C	E	I	M
The mode z_2 of Class 2	C	F	I	M
x_{11}	D	G	I	N
x_{12}	C	G	H	N
x_{13}	D	G	J	N
x_{14}	D	E	J	N
x_{15}	D	G	J	M
The mode z_3 of Class 3	D	G	J	N

According to the attribute a_2, U is partitioned into three subsets:

$$U_E = \{x_1,x_2,x_4,x_5,x_{10},x_{14}\}, \quad U_F = \{x_6,x_7,x_9\},$$
$$U_G = \{x_3,x_8,x_{11},x_{12},x_{13},x_{15}\};$$

According to the attribute a_3, U is partitioned into four subsets:

$$U_H = \{x_1,x_3,x_4,x_9,x_{15}\}, \quad U_I = \{x_2,x_7,x_8,x_{10},x_{11}\},$$
$$U_J = \{x_6,x_{13},x_{14},x_{15}\}, \quad U_K = \{x_5\};$$

According to the attribute a_4, U is partitioned into three subsets:

$$U_L = \{x_1,x_2,x_3,x_5\}, \quad U_M = \{x_6,x_8,x_9,x_{10},x_{15}\},$$
$$U_N = \{x_4,x_7,x_{11},x_{12},x_{13},x_{14}\};$$

By computing, the potential exemplars set S is shown in Table 2.

Next, we will integrate density measure and distance measure to select k exemplars as the initial cluster centers from the potential exemplars set S. That implies a hypothesis: $k \leqslant |S|$ which is consistent with real data sets. In real large data sets, $k \leqslant |S| \ll |U|$. In the selecting process, we use the simple dissimilarity measure (see Section 2) as distance measure to reflect the difference between any two exemplars. How to measure the density of an exemplar? Intuitively, for any two exemplars c_i, $c_j \in S (1 \leqslant i < j \leqslant |S|)$, if c_i is thought to have higher density than c_j, the number of data points around c_i should be more than c_j. Therefore, for any exemplar in S, we will use the number of data points that is the nearest to the exemplar to measure its density. The definition is given as follows:

Definition 4. Let $IS = (U,A,V,f)$ be a categorical data table. The density of a potential exemplar $c \in S$ is defined as

$$Dens(c) = |\{x_i \in U | d(x_i,c) = \min_{c' \in S} d(x_i,c'), 1 \leqslant i \leqslant n\}|.$$

Obviously, we have $0 \leqslant Dens(c) \leqslant n$. In the potential exemplars set, the more $Dens(c)$ is, if can be expressed in a graph, the more data points around the c is, the more possible c is to be as a cluster center. So we select the potential exemplar with the maximum density as the first initial cluster center. For selection of the rest of initial cluster centers, we consider not only the density of the potential exemplars, but also the distance between the potential exemplars. If the distance between the potential exemplars is the only considered factor, it is possible that outliers are taken as cluster centers. Similarly, if only the density of the potential exemplars is taken into account, it is utmost possible that many cluster centers locate in the surrounding of one center. They are unreliable initial points which could lead to bad partitions after the clustering process. To avoid these potential problems, we integrate distance with density together to measure the possibility of each potential exemplar to be an initial cluster center.

Generally, methods to integrate the two factors can be classified into two categories. The one is a linear combination defined as the sum of distance and density with appropriate weighting factor, i.e., $\eta Dens + (1-\eta)d$, where $\eta \in [0,1]$ is a weighting value. The other is a nonlinear combination characterized by the multiplication of

Table 2
The potential exemplars set in Example 1.

Exemplars\attributes	a_1	a_2	a_3	a_4
c_1	B	E	H	L
c_2	C	F	I	M
c_3	D	G	J	N
c_4	C	F	I	M
c_5	D	G	H	N
c_6	C	E	I	M
c_7	B	E	K	L
c_8	D	G	J	M

distance and density, i.e., $Dens^{\eta} \times d^{1-\eta}$. Here, we use the first method to select initial cluster centers. There are three reasons: (1) the second method can become the first method by logarithmic transformation, i.e., $\ln(Dens^{\eta} \times d^{1-\eta}) = \eta \ln Dens + (1-\eta)\ln d$. This means that the two methods have mathematical properties in common. (2) The second method is too sensitive to the changes in the values of the two factors. For example, in the first method, if *Dens* becomes $Dens + \triangle Dens$, $Dens + d$ increases by $\triangle Dens$ (to simplify the analysis, we omit the weight η in the example). However, in the second method, if *Dens* becomes $Dens + \triangle Dens$, $Dens \times d$ increases by $\triangle Dens \times d$. (3) In experimental studies, we also found that the first method is more appropriate to select initial cluster centers than the second method. Therefore, we will define the possibility of a potential exemplar from S to be the $l+1$th cluster center ($0 \leqslant l < |S|$) based on the summation type.

Definition 5. Let $IS = (U, A, V, f)$ be a categorical data table, and $Z_l = \{z_1, z_2, \ldots, z_l\}$ be a set of the first l initial cluster centers obtained, where $0 < l < k$. For any $c \in S$, the possibility of the potential exemplar c to be the $l+1$th cluster center z_{l+1} is defined as

$$Possibility_{l+1}(c) = \begin{cases} \eta \frac{Dens(c)}{n} + (1-\eta)\frac{\min_{i=1}^{l} d(c,z_i)}{m}, & \min_{i=1}^{l} d(c, z_i) \neq 0, \\ 0, & \min_{i=1}^{l} d(c, z_i) = 0, \end{cases}$$

where $\eta \in [0,1]$ is a weight value.

According to Definition 5, we have $0 \leqslant Possibility_{l+1}(c) \leqslant 1$. The weight η is to maintain a balance between the effect of density and that of distance. We set $\eta = 1/2$ which means that the effect of density is seen as important as that of distance on selecting initial cluster centers.

A new cluster centers initialization method is described as follows:

Input: $IS = (U, A, V, f)$ and k, where k is the number of clusters desired.
Output: *Centers*.
Step 1: $Centers = \emptyset$;
Step 2: Construct the potential exemplars set S;
Step 3: For each $c \in S$, calculate the $Dens(c)$ and choose the densest potential exemplar c as the first cluster center. Set $Centers = Centers \bigcup \{c\}$ and $i = 1$;
Step 4: If $i < k$, then let $i = i + 1$ and choose the most probable exemplar c from S as the $i+1$ cluster center, $Centers = Centers \bigcup \{c\}$ where c satisfies

$$Possibility_i(c) = \max_{c' \in S} Possibility_i(c'),$$

and goto Step 4, otherwise goto Step 5;
Step 5: End.

Constructing the potential exemplars set S will take $O(nm|V|)$, where $|V| = \sum_{i=1}^{m} n_j$. Computing the density value for each potential exemplar in S will take $O(nm|V|)$. For selection of the first cluster center, the computation is $O(|V|)$, and for the rest of initial cluster centers, the computation is $O(mk^2|V|)$. Therefore, the computational cost of the proposed method is $O(2nm|V| + |V| + mk^2|V|)$ which is linear with respect to the number of data points. When the number of data points is large, m, k, $|V| \ll n$. Therefore, the method is scalable to large data sets.

Let us continue Example 1 to show the process of selecting initial cluster centers from the constructed set of potential exemplars when the number of clusters is known, i.e., $k = 3$.

Example 2 (*Continued from Example 1).* By Definition 4, the density of each potential exemplar in S is computed as

$$\begin{aligned} Dens(c_1) &= |\{x_1, x_2, x_3, x_4\}| = 4; \\ Dens(c_2) &= |\{x_6, x_7, x_8\}| = 3; \\ Dens(c_3) &= |\{x_{11}, x_{12}, x_{14}\}| = 3; \\ Dens(c_4) &= |\{x_6, x_9\}| = 2; \\ Dens(c_5) &= |\{x_{11}, x_{12}\}| = 2; \\ Dens(c_6) &= |\{x_8, x_{10}\}| = 2; \\ Dens(c_7) &= |\{x_{15}\}| = 1; \\ Dens(c_8) &= |\{x_2, x_5\}| = 2. \end{aligned}$$

We select the densest potential exemplar c_1 as the first cluster center z_1.

Furthermore, we compute the possibility of each potential exemplar $c \in S$ to be the second cluster center z_2 as follows

$$\begin{aligned} Possibility_2(c_1) &= 0; \\ Possibility_2(c_2) &= (3/15 + 4/4)/2 = 0.6000; \\ Possibility_2(c_3) &= (3/15 + 4/4)/2 = 0.6000; \\ Possibility_2(c_4) &= (2/15 + 3/4)/2 \approx 0.4417; \\ Possibility_2(c_5) &= (2/15 + 3/4)/2 \approx 0.4417; \\ Possibility_2(c_6) &= (2/15 + 3/4)/2 \approx 0.4417; \\ Possibility_2(c_7) &= (1/15 + 4/4)/2 \approx 0.5333; \\ Possibility_2(c_8) &= (2/15 + 1/4)/2 \approx 0.1917. \end{aligned}$$

Since the maximum of $Possibility_2(\cdot)$ is not unique, we can select any of c_2 and c_3 as the second cluster center z_2. Here, we select the potential exemplar c_2 of first achieving the maximum.

Finally, we compute the possibility of each potential exemplar $c \in S$ to be the third cluster center z_3 as follows

$$\begin{aligned} Possibility_3(c_1) &= 0; \\ Possibility_3(c_2) &= 0; \\ Possibility_3(c_3) &= (3/15 + 4/4)/2 = 0.6000; \\ Possibility_3(c_4) &= (2/15 + 1/4)/2 \approx 0.1917; \\ Possibility_3(c_5) &= (2/15 + 3/4)/2 \approx 0.4417; \\ Possibility_3(c_6) &= (2/15 + 1/4)/2 \approx 0.1917; \\ Possibility_3(c_7) &= (1/15 + 3/4)/2 \approx 0.4083; \\ Possibility_3(c_8) &= (2/15 + 1/4)/2 \approx 0.1917. \end{aligned}$$

We select the potential exemplar c_3 with the maximum of $Possibility_3(\cdot)$ as the third cluster center z_3. We see that the first three initial cluster centers are consistent with the true modes of these classes.

4. Finding the number of clusters

While choosing initial cluster centers, we choose a potential exemplar which is with high density and far from other initial cluster centers to be as an initial cluster center. Suppose the best clustering structure has k clusters. Intuitively, when we choose the $k+1$th potential exemplar as the $k+1$th cluster center, the potential exemplar will be representative of the same cluster as one of the first k initial cluster centers. This means that $\max_{c \in S} Possibility_{k+1}(c)$ is far smaller than $\max_{c \in S} Possibility_l(c)$, $(1 \leqslant l \leqslant k)$. The values from k to $k+1$ have dramatic change. Moreover, the values from $k+1$ to $k+n (k+n \leqslant |S|)$ should be much less distinctive, because each of these chosen $k+2, k+3, \ldots, k+n$ potential exemplars also will be representative of the same cluster as one of the first k initial cluster centers. This heuristic tells us that the value of $\max_{c \in S} Possibility_l(c)$, $(1 \leqslant l \leqslant |S|)$ could reflect the possibility of the lth cluster existing. The higher the value is, the more probably the lth cluster exist. We will explore the changes of the values from

k to $k+1$ to find the candidates for number of clusters. Next, we will define the possibility function of a cluster existing.

Definition 6. Let $IS = (U, A, V, f)$ be a categorical data table and S be the potential exemplars set. The possibility of the lth cluster existing is defined as

$$P(l) = \begin{cases} \max\limits_{c \in S} Possibility_l(c), & l \neq 1, \\ P(2), & l = 1. \end{cases}$$

Since the number of clusters is usually not less than 2, the function values from $k = 1$ to 2 should not have dramatic change, otherwise it is meaningless. Therefore, we set $P(1) = P(2)$ in the definition of the function P.

Property 2. *Let $IS = (U, A, V, f)$ be a categorical data table. $P(l) \geqslant P(l+1)$, $0 < l < |S|$, where $|S|$ is the number of the potential exemplars set S.*

We use a curve to describe the function P with different k (Fig. 1). When the curve from k to $k+1$ is dramatic and from $k+1$ goes into a plateau, we consider k as a candidate for the number of clusters. This means that $k+1$ should be a knee point on the function P.

In order to easily find the number of clusters, we define a function which reflects difference between the neighboring values on the function P.

Definition 7. Let $IS = (U, A, V, f)$ be a categorical data table. The difference between $P(l)$ and $P(l+1)$ is defined as

$$C(l) = P(l) - P(l+1), \quad 1 \leqslant l < |S|.$$

When $P(k)$ has dramatic difference with $P(k+1)$, $C(k)$ is very large. When the function P from $k+1$ goes into a plateau, $C(k+1), C(k+2), \ldots, C(k+n)$ are very small. Therefore, $C(k)$ is an obvious peak on the function C (Fig. 2).

We need to input a value k' and analyze the function P and C with different k in the range of $1 \leqslant k \leqslant k'$. k' is a estimated number larger than the possible number of clusters in the given data set. In real world, k' is estimated easier than the real number of clusters k. However, when the value of k' cannot be determined, we set $k' = |S|$. In Section 3, we have known that obtaining the first k' initial cluster centers will take $O(2nm|V| + |V| + mk'^2|V|)$. After the initial cluster centers are obtained, the computation complexity of finding the number of clusters is $O(k')(k' \leqslant |V| \ll n)$.

Let us consider the examples in Section 3 again to show the process of determining the number of clusters that is assumed to be unknown.

Example 3 (*Continued from Example 2*). We set $k' = 8$ and compute the possibility of the lth cluster existing for $1 \leqslant l \leqslant 8$ by Definition 6 as follows

$$\begin{aligned} P(1) &= P(2) = \max_{c \in S} Possibility_2(c) = 0.6000; \\ P(3) &= \max_{c \in S} Possibility_3(c) = 0.6000; \\ P(4) &= \max_{c \in S} Possibility_4(c) \approx 0.1917; \\ P(5) &= \max_{c \in S} Possibility_5(c) \approx 0.1917; \\ P(6) &= \max_{c \in S} Possibility_6(c) \approx 0.1917; \\ P(7) &= \max_{c \in S} Possibility_7(c) \approx 0.1917; \\ P(8) &= \max_{c \in S} Possibility_8(c) \approx 0.1583. \end{aligned}$$

Fig. 3 shows $P(4)$ is a knee point on the function P. This indicates that $k = 3$ may be the true number of clusters of the given data set in Table 1.

Furthermore, we compute the difference between $P(l)$ and $P(l+1)$ by Definition 7 for $1 \leqslant l \leqslant 7$ as follows

$$\begin{aligned} C(1) &= P(1) - P(2) = 0; \\ C(2) &= P(2) - P(3) = 0; \\ C(3) &= P(3) - P(4) \approx 0.4083; \\ C(4) &= P(4) - P(5) = 0; \\ C(5) &= P(5) - P(6) = 0; \\ C(6) &= P(6) - P(7) = 0; \\ C(7) &= P(7) - P(8) \approx 0.0333. \end{aligned}$$

Fig. 4 shows $C(3)$ is an obvious peak on the function C. Therefore, we determine that the number of clusters is 3 which is consistent with the true number of clusters of the given data set.

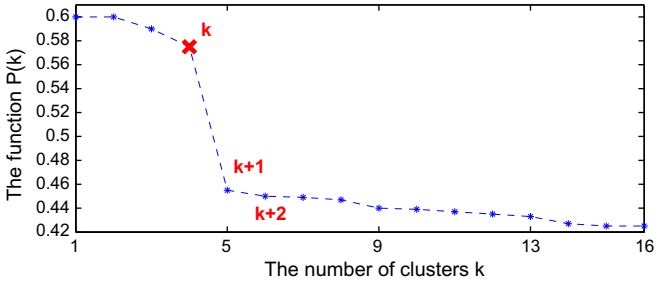

Fig. 1. Sketch of the function $P(k)$.

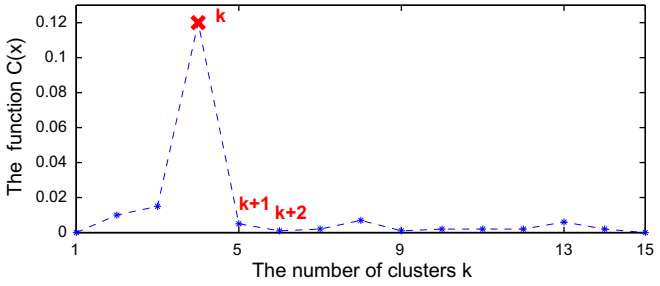

Fig. 2. Sketch of the function $C(k)$.

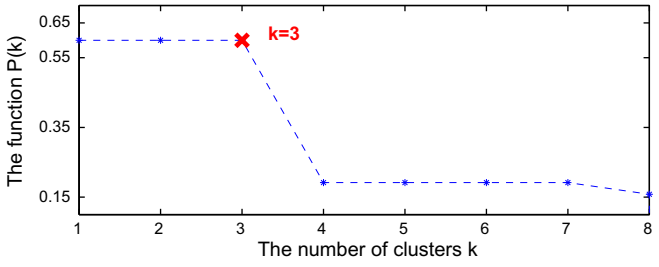

Fig. 3. The value of the function $P(k)$ against the number of clusters k.

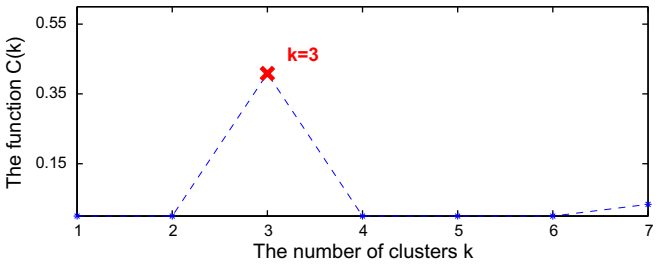

Fig. 4. The value of the function $C(k)$ against the number of clusters k.

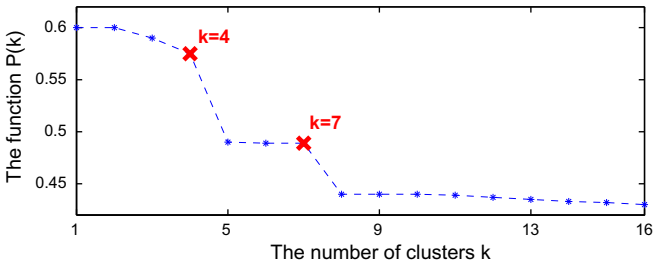

Fig. 5. Sketch of the function $P(k)$ with several knee points.

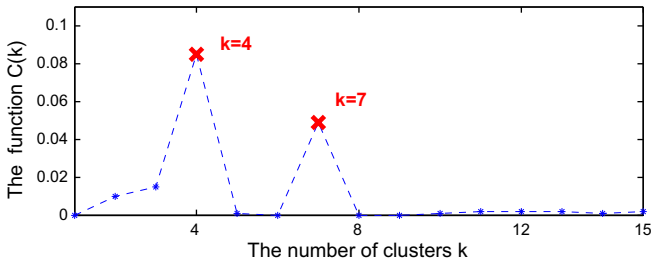

Fig. 6. Sketch of the function $C(k)$ with several peaks.

In the above example, we see only one obvious peak on the function C. However, for real data sets, there may be more than one obvious peak on the function C. In this case, we cannot decide the exact number of clusters. Therefore, we consider all the obvious peaks on the functions C as the candidates for the number of clusters, which fit reality. For example, a data set has three big clusters and four small clusters which are very close to each other. When the function P and C are used to analyze the number of clusters, there may be two knee points, $P(4)$ and $P(7)$ on the function P (showed in Fig. 5) and two obvious peaks, $C(4)$ and $C(7)$ on the function C (showed in Fig. 6). Whether the number of clusters k is 4 or 7 depends on if the four small clusters is viewed as one or four clusters, which implies that it is possible that the number of clusters is not unique. The proposed method is implemented in the framework shown in Fig. 7.

5. Experimental analysis

In this section, in order to evaluate the performance and scalability of the proposed initialization method, several standard data sets are downloaded from the UCI Machine Learning Repository [43].

The performance analysis of the proposed method consists of two parts. The one is to evaluate the effectiveness of the initial cluster centers obtained by the proposed method. Generally speaking, there are two types of clustering validation techniques [1,38–42], which are based on external and internal criteria, respectively. The focus of this paper is on the evaluation of external clustering validation measures. In this part, we first introduce three commonly used external evaluation measures [42] which are used to compare a clustering result with the true class distribution on a given data set. As external criteria, these measures use external information-class labels and the number of clusters. If the cluster result is close to the true class distribution, then the values of these evaluation measures are high. To ensure that the comparisons are in a uniform environmental condition, we set that the number of clusters is equal to the true number of clusters for each of the given data sets. Furthermore, we use these evaluation measures to evaluate and compare the clustering results of k-modes-type algorithms based on different initialization cluster centers methods including random initialization method, Cao's method [26], Wu's method [25] and the proposed method. For the random initialization method, we randomly select 100 initial cluster centers to carry out 100 runs of the k-modes algorithm and the fuzzy k-modes algorithm on each of the given data sets and compute the average values of AC, PR, RE for 100 clustering results. For the fuzzy k-modes algorithm, we specified $\alpha = 1.1$ that is suggested in [14]. Tables 3–10 show the summary results for the four initialization methods on the given data sets. The other is to evaluate the effectiveness of candidates for the number of clusters determined by the proposed method. In this part, we suppose that the number of clusters is unknown in each of the given data sets and use the proposed method in Section 4 to find the candidates for the number of clusters. We compare the found candidates with the true number of clusters. The closer the found candidates are to the true number of clusters, the more effective the proposed method is. Figs. 8–11 show the results of applying the proposed method to find candidates for the number of clusters on each of the given data sets from UCI. In the scalability analysis, we test the proposed method on the connect-4 data set from UCI [43].

5.1. Performance analysis

To evaluate the performance of clustering algorithms, three evaluation measures are introduced in [42]. If data set contains k classes for a given clustering, let a_i denote the number of data points that are correctly assigned to class C_i, let b_i denote the number of data points that are incorrectly assigned to the class C_i, and let c_i denote the number of data points that are incorrectly rejected from the class C_i. The accuracy, precision and recall are defined as follows

$$AC = \frac{\sum_{i=1}^{k} a_i}{n}, \quad PR = \frac{\sum_{i=1}^{k} \left(\frac{a_i}{a_i+b_i}\right)}{k}, \quad RE = \frac{\sum_{i=1}^{k} \left(\frac{a_i}{a_i+c_i}\right)}{k},$$

respectively.

We present comparative results of clustering on soybean data, lung cancer data, zoo data and mushroom data, respectively.

5.1.1. Soybean data

The soybean data set has 47 records, each of which is described by 35 attributes. Each record is labeled as one of the four diseases: Diaporthe Stem Canker, Charcoal Rot, Rhizoctonia Root Rot, and Phytophthora Rot. Except for Phytophthora Rot which has 17 records, all other diseases have 10 records each. We only selected 21 attributes in this experiment, since the other attributes only have one category. The clustering results of the k-modes-type algorithms with different initial cluster centers on the soybean data are summarized in Tables 3 and 4. The candidate for the number of clusters is generated by the proposed method on the soybean data (Fig. 8 clearly indicates that 4 is the only significant k).

5.1.2. Lung cancer data

Lung cancer data set contains 32 instances described by 56 categorical attributes. Data set has three classes. The clustering results of the k-modes-type algorithms with different initial cluster centers on the lung cancer data are summarized in Tables 5 and 6. The candidate for the number of clusters is generated by the proposed method on the lung cancer data (Fig. 9 clearly indicates that 3 is the only significant k).

5.1.3. Zoo data

Zoo data set contains 101 elements described by 17 Boolean-valued attributes. It has seven classes. The clustering results of

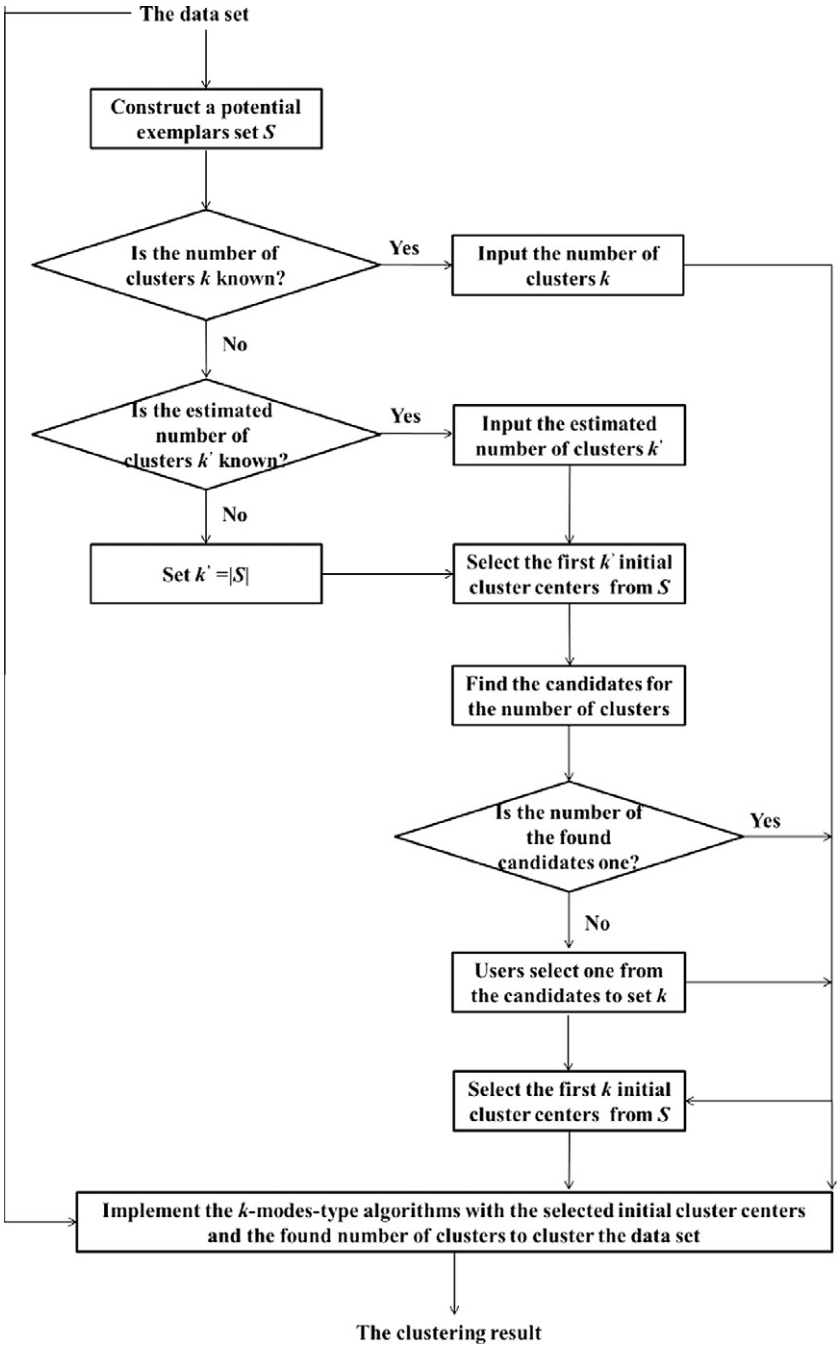

Fig. 7. The flowchart of the overall implementation of the proposed method.

Table 3
The summary clustering results of the *k*-modes algorithm on the soybean data.

	Random	Wu's method	Cao's method	Proposed method
AC	0.8553	1.0000	1.0000	1.0000
PR	0.9020	1.0000	1.0000	1.0000
RE	0.8407	1.0000	1.0000	1.0000

Table 4
The summary clustering results of the fuzzy *k*-modes algorithm on the soybean data.

	Random	Wu's method	Cao's method	Proposed method
AC	0.8336	1.0000	1.0000	1.0000
PR	0.8840	1.0000	1.0000	1.0000
RE	0.8176	1.0000	1.0000	1.0000

the *k*-modes-type algorithms with different initial cluster centers on the zoo data are summarized in Tables 7 and 8. Due to the fact that the zoo data has three big clusters and four small clusters which are very close to each other, we can see that there are more than one obvious peak in Fig. 10. In Fig. 10, although $C(3)$ is the maximum value in the function C, $k = 3$ is not seen as a candidate for the number of clusters. Because $C(4)$ is also a very high value. This indicates that when the change level from $P(3)$ to $P(4)$ is dramatic, the curve of the function P from $k = 4$ does not go into a plateau. Therefore, the candidates for the number of clusters for the zoo data should be $k = 4$ and 7.

Table 5
The summary clustering results of the *k*-modes algorithm on the lung cancer data.

	Random	Wu's method	Cao's method	Proposed method
AC	0.5313	0.5000	0.5000	0.6250
PR	0.5880	0.5584	0.5584	0.7930
RE	0.5374	0.5014	0.5014	0.5744

Table 6
The summary clustering results of the fuzzy *k*-modes algorithm on the lung cancer data.

	Random	Wu's method	Cao's method	Proposed method
AC	0.5497	0.5000	0.5000	0.6250
PR	0.5965	0.4880	0.4880	0.6852
RE	0.5626	0.5630	0.5630	0.5667

Table 7
The summary clustering results of the *k*-modes algorithm on the zoo data.

	Random	Wu's method	Cao's method	Proposed method
AC	0.8324	0.8812	0.8812	0.9505
PR	0.8433	0.8702	0.8702	0.9378
RE	0.6576	0.6714	0.6714	0.8571

Table 8
The summary clustering results of the fuzzy *k*-modes algorithm on the zoo data.

	Random	Wu's method	Cao's method	Proposed method
AC	0.8375	0.8812	0.9208	0.9505
PR	0.8442	0.8717	0.8819	0.9116
RE	0.6471	0.6714	0.7857	0.8571

Table 9
The summary clustering results of the *k*-modes algorithm on the mushroom data.

	Random	Wu's method	Cao's method	Proposed method
AC	0.7176	0.8754	0.8754	0.8892
PR	0.7453	0.9019	0.9019	0.9042
RE	0.7132	0.8709	0.8709	0.8858

Table 10
The summary clustering results of the fuzzy *k*-modes algorithm on the mushroom data.

	Random	Wu's method	Cao's method	Proposed method
AC	0.7001	0.8754	0.8754	0.8892
PR	0.7166	0.9013	0.9013	0.9042
RE	0.6947	0.8709	0.8709	0.8858

5.1.4. Mushroom data

Mushroom data set consists of 8124 data objects and 22 categorical attributes. Each object belongs to one of two classes, edible (4208 objects) and poisonous (3916 objects). The clustering results of the *k*-modes-type algorithms with different initial cluster centers on the mushroom data are summarized in Tables 9 and 10. The candidate for the number of clusters is generated by the proposed method on the mushroom data (Fig. 11 clearly indicates that 2 is the only significant *k*).

According to Tables 3–10, the performance of the *k*-modes-type algorithms based on the proposed cluster centers initialization method is better than other methods for AC, PR and RE. According to Figs. 8, 9 and 11, we see that the found candidates for the numbers of clusters are consistent with the real numbers of clusters on these data sets from UCI.

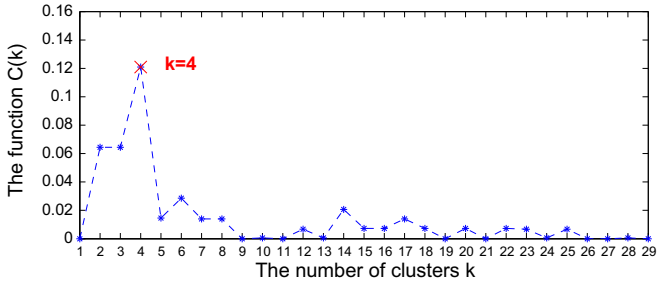

Fig. 8. The candidates for the number of clusters on the soybean data.

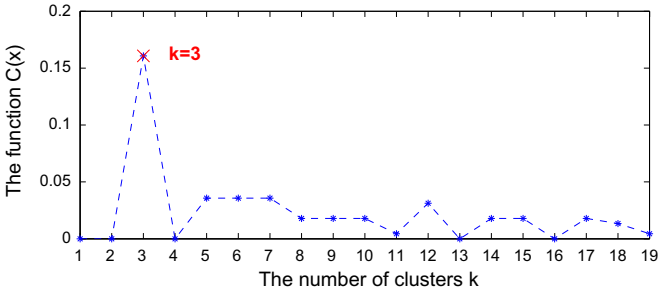

Fig. 9. The candidates for the number of clusters on the lung cancer data.

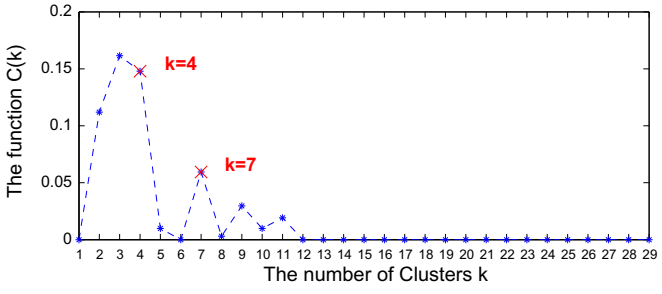

Fig. 10. The candidates for the number of clusters on the zoo data.

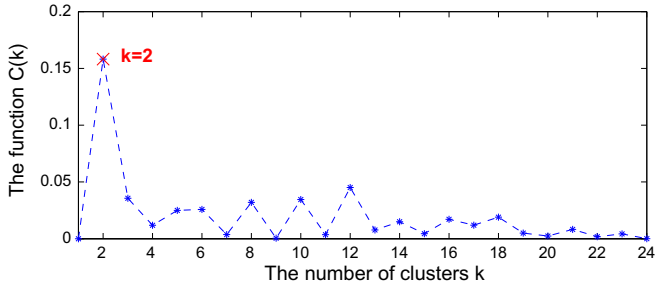

Fig. 11. The candidates for the number of clusters on the mushroom data.

5.2. Scalability analysis

To test the scalability of the new method, we choose the connect-4 data set from UCI. The data set contains 67,557 data points and 42 categorical attributes. It has three class: win (44,473), loss (16,635) and draw (6449). The computational results are performed by using a machine with an Intel Q9400 and 2G RAM. The computational times of the proposed method are plotted with respect to the number of data points, attributes and clusters, while the other corresponding parameters are fixed.

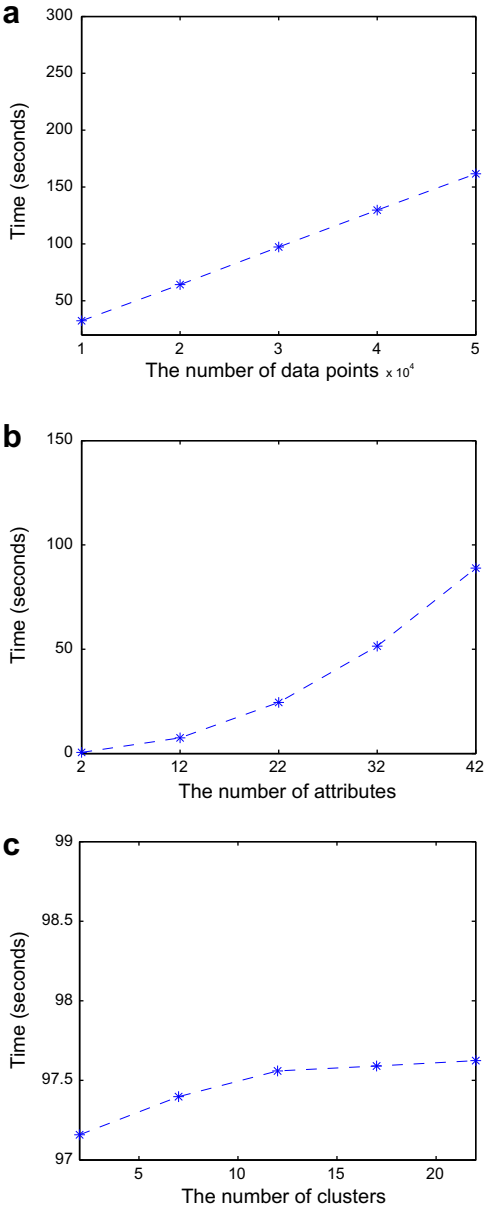

Fig. 12. (a) Computational times for different numbers of data points. (b) Computational times for different numbers of attributes. (c) Computational times for different numbers of clusters.

Fig. 12(a) shows the computational times against the number of data points, while the number of attributes is 42 and the number of clusters is 3. Fig. 12(b) shows the computational times against the number of attributes, while the numbers of clusters is 3 and the number of data points is 30,000. Fig. 12(c) shows the computational times against the number of clusters, while the number of attributes is 42 and the number of data points is 30,000. According to the figures, we see that the proposed method is scalable, i.e., it can efficiently deal with large categorical data.

6. Conclusions

Categorical data are ubiquitous in real-world databases. The development of the *k*-modes-type algorithms was motivated to solve this problem. However, the performance of these algorithms strongly depends on two parameters, an initial set of cluster centers and the number of clusters. When the prior information about setting the two parameters for a data set is not available, it is difficult for users to implement these algorithms to effectively cluster the data set. In this paper, a new initialization method for categorical data clustering has been proposed. The proposed method can simultaneously obtain the good initial cluster centers and the candidates for the number of clusters. Furthermore, the time complexity of the proposed method has been analyzed. We tested the proposed method on real world data sets from UCI Machine Learning Repository. The experimental results have illustrated that the proposed method is effective and efficient for initializing categorical data.

Acknowledgement

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 Hong Kong SAR Government, the National Key Basic Research and Development Program of China (973) (No. 2007CB311002), the Natural Science Foundation of Shanxi (No. 2010021016-2), the Foundation of Doctoral Program Research of Ministry Education of China (No. 20101401110002).

References

[1] A.K. Jain, R.C. Dubes, Algorithms for Clustering Data, Prentice Hall, 1988.
[2] M.R. Anderberg, Cluster Analysis for Applications, Academic, 1973.
[3] G.H. Ball, D.J. Hall, A clustering technique for summarizing multivariate data, Behavioral Science 12 (2) (1967) 153–155.
[4] J.B. MacQueen, Some methods for classification and analysis of multivariate observationsin, in: Proc. of 5th Berkley Symposium on Mathematical Statistics and Probability, University of California Press, California, 1967.
[5] E.R. Ruspini, A new approach to clustering, Information Control 15 (1) (1969) 22–32.
[6] J.C. Bezdek, A convergence theorem for the fuzzy ISODATA clustering algorithms, IEEE Transactions on Pattern Analysis and Machine Intelligence 2 (1) (1980) 1–8.
[7] N. Wrigley, Categorical Data Analysis for Geographers and Environmental Scientists, Longman, London, 1985.
[8] C.C. Aggarwal, C. Magdalena, P.S. Yu, Finding localized associations in market basket data, IEEE Transactions on Knowledge and Data Engineering 14 (1) (2002) 51–62.
[9] A. Baxevanis, F. Ouellette (Eds.), Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins, second ed., Wiley, NY, 2001.
[10] J. Wang, G. Karypis, On efficiently summarizing categorical databases, Knowledge and Information Systems 9 (1) (2006) 19–37.
[11] D. Barbara, S. Jajodia (Eds.), Applications of Data Mining in Computer Security, Kluwer, Dordrecht, 2002.
[12] Z.X. Huang, A fast clustering algorithm to cluster very large categorical data sets in data mining, in: Proceedings of SIGMOD Workshop Research Issues on Data Mining and Knowledge Discovery, 1997, pp. 1–8.
[13] Z.X. Huang, Extensions to the *k*-means algorithm for clustering large data sets with categorical values, Data Mining and Knowledge Discovery 2 (3) (1998) 283–304.
[14] Z.X. Huang, M.K. Ng, A fuzzy *k*-modes algorithm for clustering categorical data, IEEE Transactions on Fuzzy Systems 7 (4) (1999) 446–452.
[15] B. Andreopoulos, A. An, X. Wang, Clustering the internet topology at multiple layers, WSEAS Transactions on Information Science and Applications 10 (2) (2005) 625–1634.
[16] G.P. Babu, M.N. Murty, A near-optimal initial seed value selection for *k*-means algorithm using genetic algorithm, Pattern Recognition Letters 14 (10) (1993) 763–769.
[17] K. Krishna, M.N. Murty, Genetic *k*-means algorithm, IEEE Transactions on Systems Man and Cybernetics-Part B: Cybernetics 29 (3) (1999) 433–439.
[18] M. Laszlo, S. Mukherjee, A genetic algorithm using hyper-quadtrees for low-dimensional *k*-means clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (4) (2006) 533–543.
[19] M. Laszlo, S. Mukherjee, A genetic algorithm that exchanges neighboring centers for *k*-means clustering, Pattern Recognition Letters 28 (16) (2007) 2359–2366.
[20] D. Arthur, S. Vassilvitskii, *K*-means++: the advantages of careful seeding, in: SODA '07: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, 2007, pp. 1027–1035.
[21] F.Y. Cao, J.Y. Liang, G. Jiang, An initialization method for the *k*-means algorithm using neighborhood model, Computers and Mathematics with Application 58 (3) (2009) 474–483.

[22] A. Likas, M. Vlassis, J. Verbeek, The global *k*-means clustering algorithm, Pattern Recognition 35 (2) (2003) 451–461.
[23] Y. Sun, Q.M. Zhu, Z.X. Chen, An iterative initial-points refinement algorithm for categorical data clustering, Pattern Recognition Letters 23 (7) (2002) 875–884.
[24] P.S. Bradley, U.M. Fayyad, Refining initial points for *k*-means clustering, in: J. Sharlik (Ed.), Proc. 15th Internat Conf. on Machine Learning (ICML98), Morgan Kaufmann, San Francisco, CA, 1998, pp. 91–99.
[25] S. Wu, Q.S. Jiang, Z.X. Huang, A new initialization method for categorical data clustering, Lecture Notes in Computer Science 4426 (2007) 972–980.
[26] F.Y. Cao, J.Y. Liang, L. Bai, A new initialization method for categorical data clustering, Expert Systems with Applications 33 (7) (2009) 10223–10228.
[27] G. Hamerly, C. Elkan, Learning the *k* in *k*-means, in: Proc. 17th Ann. Conf. Neural Information Processing Systems (NIPS 03), 2003.
[28] J.J. Li, M.K. Ng, Y.M. Cheng, Z.H. Huang, Agglomerative fuzzy *k*-means clustering algorithm with selection of number of clusters, IEEE Transactions on Knowledge and Data Engineering 20 (11) (2008) 1519–1534.
[29] K.K. Chen, L. Liu, Best *K*: critical clustering structures in categorical datasets, Knowledge and Information Systems 20 (1) (2008) 1–33.
[31] Z. Pawlak, Rough Sets-Theoretical Aspects of Reasoning about Data Dordrecht Boston, Kluwer Academic Publishers, London, 1991.
[32] J.Y. Liang, D.Y. Li, Uncertainty and Knowledge Acquisition in Information Systems, Science Press, Beijing, China, 2005.
[33] J.Y. Liang, J.H. Wang, Y.H. Qian, A new measure of uncertainty based on knowledge granulation for rough sets, Information Sciences 179 (4) (2009) 458–470.
[34] Y.H. Qian, J.Y. Liang, W. Pedrycz, C.Y. Dang, Positive approximation: an accelerator for attribute reduction in rough set theory, Artificial Intelligence 174 (2010) 597–618.
[35] Y.H. Qian, J.Y. Liang, D.Y. Li, Approximation reduction in inconsistent incomplete decision tables, Knowledge-Based Systems 23 (5) (2010) 427–433.
[36] K.C. Gowda, E. Diday, Symbolic clustering using a new dissimilarity measure, Pattern Recognition 24 (6) (1999) 567–578.
[37] F.Y. Cao, J.Y. Liang, L. Bai, A framework for clustering categorical time-evolving data, IEEE Transactions on Fuzzy Systems 18 (5) (2010) 872–882.
[38] M.A. Gluck, J.E. Corter, Information uncertainty and the utility of categories, in: Proceedings of the Seventh Annual Conference of Cognitive Science Society, 1985, pp. 283–287.
[39] L. Hubert, P. Arabie, Comparing partitions, Journal of Classification 2 (1) (1985) 193–218.
[40] K.Y. Huang, Applications of an enhanced cluster validity index method based on the fuzzy C-means and rough set theories to partition and classification, Expert Systems With Applications 37 (2) (2010) 8757–8769.
[41] K.Y. Huang, A hybrid particle swarm optimization approach for clustering and classification of datasets, Knowledge-Based Systems 24 (3) (2011) 420–426.
[42] Y.M. Yang, An evaluation of statistical approaches to text categorization, Journal of Information Retrieval 1 (1–2) (1999) 67–88.
[43] UCI Machine Learning Repository, 2010. Available from: <http://www.ics.uci.edu/mlearn/MLRepository.html>.