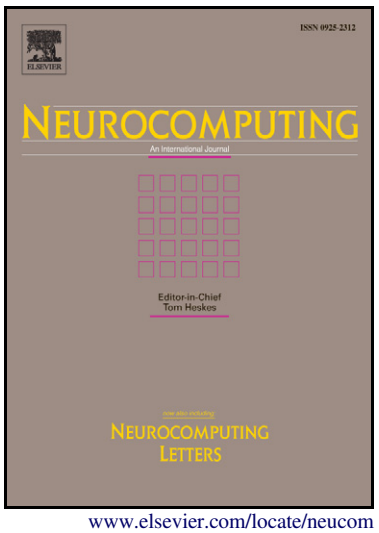

www.elsevier.com/locate/neucom

A weighting k-Modes algorithm for subspace clustering of categorical data

Fuyuan Cao, Jiye Liang, Deyu Li, Xingwang Zhao

PII: S0925-2312(12)00887-9
DOI: http://dx.doi.org/10.1016/j.neucom.2012.11.009
Reference: NEUCOM13040

To appear in: *Neurocomputing*

Received date: 2 October 2011
Revised date: 23 September 2012
Accepted date: 21 November 2012

Cite this article as: Fuyuan Cao, Jiye Liang, Deyu Li, Xingwang Zhao, A weighting k-Modes algorithm for subspace clustering of categorical data, *Neurocomputing,* http://dx.doi.org/10.1016/j.neucom.2012.11.009

A weighting k-Modes algorithm for subspace clustering of categorical data

Fuyuan Cao[a], Jiye Liang[a,*], Deyu Li[a], Xingwang Zhao[a]

[a] *School of Computer and Information Technology, Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, Shanxi University, Taiyuan, 030006, Shanxi, China*

Abstract

Traditional clustering algorithms consider all of the dimensions of an input data set equally. However, in the high dimensional data, a common property is that data points are highly clustered in subspaces, which means classes of objects are categorized in subspaces rather than the entire space. Subspace clustering is an extension of traditional clustering that seeks to find clusters in different subspaces within a data set. In this paper, a weighting k-Modes algorithm is presented for subspace clustering of categorical data and its corresponding time complexity is analyzed as well. In the proposed algorithm, an additional step is added to the k-Modes clustering process to automatically compute the weight of all dimensions in each cluster by using complement entropy. Furthermore, the attribute weight can be used to identify the subsets of important dimensions that categorize different clusters. The effectiveness of the proposed algorithm is demonstrated with real data sets and synthetic data sets.

Keywords:
Subspace clustering, Weight, k-Modes algorithm, Categorical data

1. Introduction

Clustering is a descriptive task that seeks to partition a set of objects into several groups according to the predefined similarity measure [1]. Clustering techniques have been studied extensively in statistics, pattern recognition, machine learning, etc. Detailed surveys on clustering methods can be found in [2], [3].

Subspace clustering or projected clustering [4] is an extension of feature selection [5], [6], [7], that attempts to group objects into clusters on different subsets of dimensions or attributes of a data set. It achieves two purpose, identification of the subsets of dimensions where clusters can be found and discovery

*Corresponding author
Email addresses: cfy@sxu.edu.cn (Fuyuan Cao), ljy@sxu.edu.cn (Jiye Liang), lidy@sxu.edu.cn (Deyu Li), zhaoxw84@163.com (Xingwang Zhao)

of the clusters from different subsets of dimensions. According to the ways with which the subsets of dimensions are identified, we can divide subspace clustering methods into two categories: hard subspace clustering and soft subspace clustering [8]. Hard subspace clustering determines the exact subsets of dimensions for different clusters. Soft subspace clustering determines the subsets of dimensions according to the contributions of the dimensions in discovering the corresponding clusters. The contribution of a dimension is measured by a weight that is assigned to the dimension in the clustering process.

In terms of data type, a data set has numeric(or real-valued) data and categorical(or symbolic) data, and of course the hybrid of the two. In the numeric domain, many subspace clustering algorithms have been presented, such as hard subspace clustering [9], [10], [11], [12], [13], [14], [15], [16] and soft subspace clustering [17], [18], [19], [20], [21], [8]. However, the above-mentioned algorithms working only on numeric data restrict their uses in data mining where categorical data sets are frequently encountered. In the categorical domain, Kim and Ramakrishna [22] proposed a new hierarchical clustering algorithm named PCC based on projected clustering, which avoids the characteristic error propagation through reassigning and deleting bad clusters. Gan [23] designed an iterative algorithm called SUBCAD for clustering high dimensional categorical data set, based on the minimization of an object function for clustering. In addition, various properties of the proposed object function are proved, which are essential to design a fast algorithm to find the subspace associated with each cluster. Gan [24] put forward an algorithm called PARTCAT (Projective Adaptive Resonance Theory for Categorical data clustering) based on a neural network architecture PART for clustering high dimensional categorical data. But, PARTCAT does not outperform SUBCAD. Zaik [25] presented an algorithm called CLICKS based on a search for k-partite maximal cliques, and can guarantee the completeness of the results at a reasonable additional cost without sacrificing scalability by using a novel vertical encoding. However, the above-mentioned algorithms have a common problem that computational cost of these algorithms is very high, and they fall in hard subspace clustering.

To our knowledge, soft subspace clustering algorithms are scarce for categorical data. Chan [26] presented an attributes-weighting algorithm, which is achieved by the development of a new procedure to generate the weight of each attribute from each cluster within the framework of the k-means-types algorithm. The effectiveness of the algorithm was demonstrated with both synthetic and real mixed data sets. However, for categorical data, Chan's clustering algorithm encounters some problems in the weighted computation. In other words, if there is the same attribute values in some dimension in a cluster, then its weight is 1. This means that the rest of attributes will be ignored. In categorical data set, this situation occurs frequently. To overcome these drawbacks, a new method to find the weight of each attribute from each cluster is proposed based on complement entropy for categorical data.

In this paper, we consider that different dimensions provide different clustering characteristics in a cluster. Suppose that there are attribute values of a dimension which occur in a cluster uniformly, that dimension which contains the

maximum uncertainty provides less clustering characteristics. The contribution of a dimension is measured by a weight that is assigned to the dimension in the clustering process. Based on the foregoing, a method to find the weight for each attribute in each cluster is provided by using complement entropy. Furthermore, a weighting k-Modes algorithm (abbreviated as wk-Modes) is presented and its corresponding time complexity is analyzed as well. The effectiveness of the proposed algorithm is demonstrated with real data sets and synthetic data sets.

The rest of the paper is organized as follows. In Section 2, some related works are reviewed. A weighting k-Modes algorithm is presented based on complement entropy in Section 3. In Section 4, we demonstrate the effectiveness of our method with experimental results. Finally, a summary is given to conclude the paper in Section 5.

2. Related work

In general, we assume the set of objects to be clustered is stored in a table, where each row(tuple) represents an object by a series of attributes. Data in the real world are often described by categorical attributes. More formally, a categorical data table can be defined as a quadruple $DT = (U, A, V, f)$, where:

U– a nonempty set of objects, called the universe;

A– a nonempty set of attributes;

V– a union of all attribute domains, i.e., $V = \bigcup_{a \in A} V_a$, where V_a is the domain of attribute a and it is finite and unordered;

$f : U \times A \to V$– a mapping called an information function such that for any $x \in U$ and $a \in A$, $f(x, a) \in V_a$.

Since first published in 1998, the k-Modes algorithm [27] has become an important technique for solving categorical data clustering problems in different domains. The k-Modes algorithm extends the k-Means algorithm by using a simple matching dissimilarity measure for categorical objects, modes instead of means for clusters, and a frequency-based method to update modes in the clustering process to minimize the clustering cost function. These 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. So far, the k-Modes algorithm and its variants, including fuzzy k-Modes algorithm [28], fuzzy k-Modes algorithm with fuzzy centroid [29], and k-prototype algorithm [27], have been used widely in many domains. However, the distance or dissimilarity measure of these algorithms involve all attributes of the data set equally. This is applicable if all or most attributes are important to every cluster. The clustering results become less accurate if a large number of attributes are not important to some clusters. Therefore, variable weighting for clustering has become an important research topic in statistic and data mining [17], [19], [21], [30], [31]. Chan [26] proposed an attribute-weighting algorithm for the mixed data set, which is the direct extension to the k-Means type variable.

However, Chan's algorithm has some deficiency in clustering categorical data. For example, if there are the same attribute values for a dimension in a cluster, and the attribute values of the rest of dimensions are different in the cluster, then the weight for that dimension is 1 and others are 0. The rest of attributes, that is to say, are ignored in the current iteration process. This situation occurs frequently in real data sets, especially for categorical data sets.

Example 1. *A categorical data set is given in Table 1.*

Table 1: A categorical data set

object	a	b	c
x_1	B	F	G
x_2	C	M	P
x_3	B	E	D
x_4	B	M	P
x_5	B	E	Q

This is a categorical data table, where $U = \{x_1, x_2, x_3, x_4, x_5\}$ *and* $A = \{a, b, c\}$. *Suppose that* x_1 *and* x_2 *are chosen as the initial cluster centers. By executing Chan's algorithm, we can obtain the clustering results after the first iteration, i.e.,* $c_1 = \{x_1, x_3, x_5\}$ *and* $c_2 = \{x_2, x_4\}$. *The weight of each attribute in* c_1 *and* c_2 *are shown in Table 2.*

Table 2: The weight of each attribute in c_1 and c_2

cluster	a	b	c
c_1	1	0	0
c_2	0	0.5	0.5

If so, in the next iteration, the distances between x_4 *and the "mode" of two clusters are all zeros, which means that we cannot determine* x_4 *should be assigned to which cluster.*

3. A Weighting *k*-Modes Algorithm

In this section, some basic concepts are reviewed, which are indiscernibility relation [32],[33], complement entropy [34]. The weight of attributes in a cluster is defined based on complement entropy. Furthermore, a weighting k-Modes algorithm is proposed and the corresponding time complexity is analyzed as well.

3.1. Some basic concepts

Definition 1. *Let* $DT = (U, A, V, f)$ *is a categorical data table, for any attribute subset* $P \subseteq A$ *and object subset* $U' \subseteq U$, *a binary relation* $IND(P)$, *called indiscernibility relation, is defined as*

$$IND(P) = \{(x, y) \in U' \times U' | \forall a \in P, f(x, a) = f(y, a)\}.$$

It is obvious that $IND(P)$ is an equivalence relation on U' and $IND(P) = \bigcap_{a\in P} IND(\{a\})$. Given $P \subseteq A$, the relation $IND(P)$ induces a partition of U', denoted by $U'/IND(P) = \{[x]_P | x \in U'\}$, where $[x]_P$ denotes the equivalence class determined by x with respect to P, i.e., $[x]_P = \{y \in U' | (x, y) \in IND(P)\}$. If $(x, y) \in IND(P)$, then x and y are indiscernible by attributes from P.

Entropy is the measurement of information and uncertainty on a random variable. Shannon introduced the concept of entropy in physics to information theory for measure uncertainty of the structure of a system. The bigger entropy value is, the higher out-of-order of a system is. Liang [34] presented a complement entropy for measure uncertainty in a categorical data table.

Definition 2. *Let $DT = (U, A, V, f)$ be a categorical data table, $P \subseteq A$, $U' \subseteq U$. The complement entropy of P is defined as*

$$E(P) = \sum_{X\in U'/IND(P)} \frac{|X|}{|U'|}\frac{|X^c|}{|U'|} = \sum_{X\in U'/IND(P)} \frac{|X|}{|U'|}(1 - \frac{|X|}{|U'|}),$$

where X^c denotes the complement set of X, i.e., $X^c = U' - X$, $\frac{|X|}{|U'|}$ represents the probability of X within the U' and $\frac{|X^c|}{|U'|}$ is the probability of the complement set of X within the U'.

If $U'/IND(P) = \{X | X = \{u\}, u \in U'\}$, then $E(P) = 1 - \frac{1}{|U'|}$.

If $U'/IND(P) = \{X | X = U'\}$, then $E(P) = 0$.

The complement entropy $E(P)$ reflects the uncertainty of object set U' with respect to attributes set P. The bigger the complement entropy value is, the higher the uncertainty is. $E(P) = 0$ means that all objects of U' belong to the same equivalence class with respect to P, that is to say, each object of U' has the same attribute values which provide the maximum certainty. On the contrary, $E(P) = 1 - \frac{1}{|U'|}$ shows that each object of U' forms a equivalence class with respect to P, in other words, the disorder of distributions of attribute values is the most highest.

Based on the complement entropy, the within-cluster entropy is defined as follows [35].

Definition 3. *Let $DT = (U, A, V, f)$ be a categorical data table, which can be separated into k clusters, i.e., $C^k = \{c_1, c_2, \cdots, c_k\}$. For any $c_{k'} \in C^k$, the within-cluster entropy $WE(c_{k'})$ is defined as*

$$WE(c_{k'}) = \sum_{a\in A} \sum_{X\in c_{k'}/IND(\{a\})} \frac{|X|}{|c_{k'}|}(1 - \frac{|X|}{|c_{k'}|}).$$

In fact, the within-cluster entropy reflects the average distance between objects for given attribute set in the same object set. To prove the property, the dissimilarity measure between two objects is given.

Definition 4. *Let $DT = (U, A, V, f)$ be a categorical data table. For any $x, y \in U$, the dissimilarity measure $D_A(x, y)$ is defined as*

$$D_A(x, y) = \sum_{a \in A} d_a(x, y), \tag{1}$$

where $d_a(x, y) = \begin{cases} 0, & f(x, a) = f(y, a), \\ 1, & f(x, a) \neq f(y, a). \end{cases}$

Intuitively, the dissimilarity between two categorical objects is directly proportional to the number of attributes in which they mismatch.

Property 1. $WE(c_{k'}) = \frac{1}{|c_{k'}|^2} \sum_{x \in c_{k'}} \sum_{y \in c_{k'}} D_A(x, y)$.

Proof 1. *For convenience, suppose that $Y_{\{a\}} = c_{k'}/IND(\{a\})$, where $a \in A$. Then,*

$$\begin{aligned} WE(c_{k'}) &= \sum_{a \in A} \sum_{X \in Y_{\{a\}}} \frac{|X|}{|c_{k'}|} \left(1 - \frac{|X|}{|c_{k'}|}\right) \\ &= \sum_{a \in A} \left(1 - \sum_{X \in Y_{\{a\}}} \frac{|X|^2}{|c_{k'}|^2}\right) \\ &= \frac{1}{|c_{k'}|^2} \sum_{a \in A} \left(|c_{k'}|^2 - \sum_{X \in Y_{\{a\}}} |X|^2\right) \\ &= \frac{1}{|c_{k'}|^2} \sum_{a \in A} \sum_{x \in c_{k'}} \sum_{y \in c_{k'}} d_a(x, y) \\ &= \frac{1}{|c_{k'}|^2} \sum_{x \in c_{k'}} \sum_{y \in c_{k'}} \sum_{a \in A} d_a(x, y) \\ &= \frac{1}{|c_{k'}|^2} \sum_{x \in c_{k'}} \sum_{y \in c_{k'}} D_A(x, y) \end{aligned}$$

The above derivation means that the within-cluster entropy can be expressed with the average dissimilarity between objects within a cluster for categorical data.

Example 2. *(Continued from Example 1)*
Suppose that $c_1 = \{x_1, x_3, x_5\}$ and $A = \{a, b, c\}$. By Definition 2, one have that

$$E(\{a\}) = \frac{|\{x_1, x_3, x_5\}|}{|c_1|}\left(1 - \frac{|\{x_1, x_3, x_5\}|}{|c_1|}\right) = 0,$$

$$E(\{b\}) = \frac{|\{x_1, x_3\}|}{|c_1|}\left(1 - \frac{|\{x_1, x_3\}|}{|c_1|}\right) + \frac{|\{x_5\}|}{|c_1|}\left(1 - \frac{|\{x_5\}|}{|c_1|}\right) = \frac{4}{9},$$

and

$$E(\{c\}) = \frac{|\{x_1\}|}{|c_1|}(1 - \frac{|\{x_1\}|}{|c_1|}) + \frac{|\{x_3\}|}{|c_1|}(1 - \frac{|\{x_3\}|}{|c_1|}) + \frac{|\{x_5\}|}{|c_1|}(1 - \frac{|\{x_5\}|}{|c_1|}) = \frac{6}{9}.$$

Obviously, $E(\{a\}) < E(\{b\}) < E(\{c\})$. In other words, $E(\{c\})$ achieves its maximal value, which means that the attribute c provides maximal uncertainty or minimum information in forming cluster c_1 process.

The distances between objects in the cluster c_1 with respect to a, b, c are computed as follows, respectively.

$$\hat{d}_a = d_a(x_1, x_3) + d_a(x_1, x_5) + d_a(x_3, x_5) = 0,$$

$$\hat{d}_b = d_b(x_1, x_3) + d_b(x_1, x_5) + d_b(x_3, x_5) = 2,$$

and

$$\hat{d}_c = d_c(x_1, x_3) + d_c(x_1, x_5) + d_c(x_3, x_5) = 3.$$

According to Definition 3, we have that

$$WE(c_1) = E(\{a\}) + E(\{b\}) + E(\{c\}) = 0 + \frac{4}{9} + \frac{6}{9} = \frac{10}{9}.$$

Obviously, we have that

$$\begin{aligned} WE(c_1) &= \frac{1}{|c_1|^2} \times (D_A(x_1, x_3) + D_A(x_1, x_5) + D_A(x_3, x_5) \\ &+ D_A(x_3, x_1) + D_A(x_5, x_1) + D_A(x_5, x_3)) \\ &= \frac{1}{9}(\hat{d}_a + \hat{d}_b + \hat{d}_c + \hat{d}_a + \hat{d}_b + \hat{d}_c) = \frac{10}{9} \end{aligned}$$

3.2. Weight of Attribute

Instead of identifying exact subspace for clusters, soft subspace clustering assigns a weight to each dimension in clustering process to measure the contribution of the attribute in forming a particular cluster. In the following, based on the complement entropy, the importance of an attribute in forming a cluster is defined as follows.

Definition 5. *Let $DT = (U, A, V, f)$ be a categorical data table. Suppose that a clustering result $C^k = \{c_1, c_2, \cdots, c_k\}$ is given after iteration, where c_i, $1 \leq i \leq k$, is the ith cluster, and k is the number of the clusters. For any $a \in A$, the importance of attribute a in the cluster c_i is defined as*

$$I(c_i, a) = \sum_{X \in c_i/IND(\{a\})} \frac{|X|}{|c_i|} \frac{|X^c|}{|c_i|} = \sum_{X \in c_i/IND(\{a\})} \frac{|X|}{|c_i|}(1 - \frac{|X|}{|c_i|}).$$

The $I(c_i, a)$ has maximum value when the domain values of the attribute a in the cluster c_i have the uniform distribution, which means that the attribute a provides least clustering characteristics for the cluster c_i. The $I(c_i, a)$ reflects the intracluster similarity with respect to a in the cluster c_i. The smaller $I(c_i, a)$ is, the higher intracluster similarity of cluster c_i with respect to a is.

Example 3. *(Continued from Example 1)*

Suppose that a clustering result $c = \{c_1, c_2\}$ *is given after iteration, where* $c_1 = \{x_1, x_3, x_5\}$ *and* $c_2 = \{x_2, x_4\}$.

By Definition 5, the importance of each attribute in c_1 *and* c_2 *is shown in Table 3.*

Table 3: The importance of each attribute in c_1 and c_2

cluster	a	b	c
c_1	0	4/9	6/9
c_2	1/2	0	0

From Table 3, it is clear that the attribute a is the most important for the cluster c_1*. For the cluster* c_2*, the attribute b and c have the same importance. On the basis of the importance of an attribute in forming a cluster, the weight of an attribute is defined as follows.*

Definition 6. *Let* $DT = (U, A, V, f)$ *be a categorical data table. Suppose that a clustering result* $C^k = \{c_1, c_2, \cdots, c_k\}$ *is given after iteration, where* c_i*,* $1 \leq i \leq k$*, is the* i*th cluster, and* k *is the number of the clusters. For any* $a \in A$*, the weight of the attribute* a *in the cluster* c_i *is defined as*

$$\lambda(c_i, a) = \frac{exp(-I(c_i, a))}{\sum_{a' \in A} exp(-I(c_i, a'))}.$$

$\lambda(c_i, a)$ is inversely proportional to $E(\{a\})$. The smaller $E(\{a\})$, the larger $\lambda(c_i, a)$, the more important the corresponding dimension.

Example 4. *(Continued from Example 3)*

By Definition 6, the weights of each attribute in c_1 *and* c_2 *are shown in Table 4.*

Table 4: The weights of each attribute in c_1 and c_2

cluster	a	b	c
c_1	0.5315	0.2729	0.1955
c_2	0.2327	0.3837	0.3837

From Table 4, obviously, the distance between x_4 *and the mode of* c_1 *is greater than that between* x_4 *and the mode of* c_2*. Therefore,* x_4 *should be assigned to the cluster* c_2*.*

From the above analysis, we can find that the new weight is superior to Chan's.

3.3. The wk-Modes Algorithm

Let $DT = (U, A, V, f)$ be a categorical data table. The objective of clustering a set of $n = |U|$ objects into k clusters is to minimize the objective function

$$F(W, Z, \Lambda) = \sum_{l=1}^{k} \sum_{i=1}^{n} \sum_{a \in A} \omega_{li} D_a(z_l, x_i) \lambda(c_l, a) \tag{2}$$

subject to

$$\omega_{li} \in \{0, 1\}, 1 \leq l \leq k, 1 \leq i \leq n, \tag{3}$$

$$\sum_{l=1}^{k} \omega_{li} = 1, 1 \leq i \leq n, \tag{4}$$

$$0 < \sum_{i=1}^{n} \omega_{li} < n, 1 \leq l \leq k, \tag{5}$$

$$\lambda(c_l, a) \geq 0, 1 \leq l \leq k, a \in A, \tag{6}$$

and

$$\sum_{a \in A} \lambda(c_l, a) = 1, 1 \leq l \leq k, \tag{7}$$

where $k(< n)$ is a known number of clusters, $W = [\omega_{li}]$ is a k-by-n $\{0, 1\}$ matrix, $Z = \{z_1, z_2, \cdots, z_k\}$ is the k cluster modes with the attribute set A, $\Lambda = [\lambda(c_l, a)]$ is a k-by-$|A|$ real matrix.

The minimization of $F(W, Z, \Lambda)$ in (2) with the constraints in (3)-(7) forms a class of constrained nonlinear optimization problems whose solutions are unknown. The usual method towards optimization of $F(W, Z, \Lambda)$ in (2) is to use partial optimization for Z, W and Λ. In this method, we first fix Z and Λ and find necessary conditions on W to minimize $F(W, Z, \Lambda)$. Then, we fix W and Λ and minimize $F(W, Z, \Lambda)$ with respect to Z. We then fix W and Z and minimize $F(W, Z, \Lambda)$ with respect to Λ. The process is repeated until no more improvement in the objective function value can be made. The above procedure is formalized in Table 5.

In the wk-modes algorithm, the computations of W and Z are similar to that of the k-modes algorithm in the first iteration, then we use $\lambda(c_i, a)$ to obtain Λ, which is also to minimize the objective function. Therefore, the objective function does not increase in the first iteration procedure. In the rest iterations, as the weight λ and the distance D are in essence the same, they all reflect the similarity of within cluster. Therefore, fixing Λ and Z can minimize the objective function. When fixing Λ and W, we use the method of finding modes in the k-modes algorithm to obtain Z, then we compute Λ. This two steps can also minimize the objective function. Therefore, we consider that the wk-modes algorithm can converge in a finite number of iterations.

The wk-Modes algorithm is scalable to the number of dimensions, the number of objects, and the number of the clusters. This is because the wk-Modes only adds a new step to the k-Modes clustering process to calculate the attribute

Table 5: The pseudo-code of the wk-Modes algorithm

```
1   Input: The number of cluster k and a categorial data table DT
2   Initialize the variable oldmodes as a k × |A|-ary empty array;
3   Randomly choose k distinct objects x_1, x_2, ···, x_k from U
4   and assign [x_1, x_2, ···, x_k] to the k × |A|-ary array variable newmodes
5   and set all initial weights to 1/|A|;
6   while oldmodes <> newmodes do
7       oldmodes = newmodes;
8       for i = 1 to |U|
9           for l = 1 to k
10              calculate the dissimilarity between the ith object and
11              the lth mode and classify the ith object into the cluster
12              whose mode is closest to it;
13          end;
14      end;
15      for l = 1 to k
16          find the mode z_l of each cluster and assign to newmodes;
17          calculate the weight of each dimension of lth cluster;
18      end;
19      if oldmodes == newmodes
20          break;
21      end;
22  end;
23  Output: U = {c_1, c_2, ···, c_k};
```

weight from each cluster. The time complexity of the proposed algorithm can be analyzed as follows. We only consider the three major computational steps:

(1) With respect to a given attribute, the time complexity for computing weight value is $O(|C|^2)$, where $|C| = max\{|c_1|, |c_2|, \cdots, |c_k|\}$.

(2) The computational complexity for assigning the ith object into the lth cluster is $O(|U|mk)$.

(3) The computational complexity for updating all cluster centers is $O(|U|mk)$.

If the clustering process needs t iterations to terminate, the total computational complexity of this algorithm is $O(tm|U|k)$. This shows that the computational complexity increases linearly as the number of dimensions, objects, or clusters increases.

4. Experimental Results

In this Section, we demonstrate the effectiveness of the proposed algorithm on real world data sets and synthetic data sets. In Subsection 4.1, the simulation environment and the data sets used are described. The comparison results of the wk-Modes algorithm and the other algorithms, such as the k-Modes algorithm

and Chan's algorithm, on real world data set are presented in Subsection 4.2. Subsection 4.3 presents the scalability of the proposed algorithm on synthetic data sets.

4.1. Simulation Environment and Data Sets

All of our experiments are conducted on a PC with Intel i3(2.27G) processor, 2GB memory, and Windows XP SP3 professional operating system installed.

To compare the effectiveness of the *wk*-Modes algorithm with the other clustering algorithms, 6 standard data sets are downloaded from the UCI Machine Learning Repository[36]. All these data sets have class labels assigned to the objects. The details of data sets are described as follows:

Soybean Data Set

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: D1, D2, D3, and D4. Except for D4, which has 17 instances, each of the other diseases only have 10 instances. We only selected 21 attributes in this experiment because the other attributes only have one category.

Promoters Data Set

This data set contains 106 objects, each of which is described by 58 categorical attributes. One of attributes indicates that each object belongs to one of two classes, positive and negative. The remaining 57 attributes are the sequence, starting at position -50 (p-50) and ending at position +7 (p7). Each of these attributes is filled with one of a, g, t, c.

Vote Data Set

This is a pure categorical data set with 435 objects described by 16 attributes. Each object belong to one of two classes Republican(168 objects) and Democrats(267 objects). Vote data set contains a mass of objects with missing attribute values. According to the instruction of the data, *a missing attribute value does not mean that it is unknown. It means simply that the value is not "yea" or "nay".* Perhaps, the missing values mean "nonuser" or "unconcern". So missing attribute values are replaced with a special value in the experiments.

Breast-cancer Data Set

Breast-cancer data set consists of 699 data objects described by 9 categorical attributes. The objects with missing attribute values are replaced with a special value in the experiments. It is divided into two known classes, Benign(458 objects) and Malignant(241 objects).

Mushroom Data Set

There are 8140 objects described by 22 categorical attributes in Mushroom data set. Each object belongs to one of two classes, edible(e) and poisonous(p). The 2480 objects with missing attribute values are replaced with a special value in the experiments.

Connect-4 Opening Data Set

This data set contains all legal 8-ply positions in the game of connect-4 in which neither player has won yet, and in which the next move is not forced. This data set has 67557 objects, each of which is described by 42 attributes.

Each object is labeled as one of the three classes: win(44473), loss(16635), draw(6449).

4.2. Clustering Evaluation: Accuracy and Adjusted Rand Index(ARI)

To evaluate the performance of clustering algorithms, we use two popular measures to compare different clustering results in the same data set.

Clustering Accuracy: Let $C = \{C_1, C_2, \ldots, C_k\}$ be a partitions on a data set U with n objects, the clustering accuracy is defined as

$$AC = \frac{\sum_{i=1}^{k} c_l}{|U|},$$

where k is the number of clusters desired, and c_l is the number of objects occurring in both cluster C_i and its corresponding generated cluster label, and $|U| = n$ is the number of objects in the data set.

Adjusted Rand Index(ARI): The adjusted rand index is an external criterion which attempts to measure the similarity between two partitions of objects in the same data set. Let $C = \{C_1, C_2, \ldots, C_k\}$ and $C' = \{C_1^{'}, C_2^{'}, \ldots, C_{k'}^{'}\}$ be two partitions on a data set U with n objects, N_{ij} be the number of objects in cluster C_i in partition C and in cluster $C_j^{'}$ in partition C', i.e, $N_{ij} = |C_i \bigcap C_j'|$, c_i be the number of objects in cluster C_i in partition C, $c_j^{'}$ be the number of objects in cluster C_j' in partition C', the similarity measure between two partitions can be characterized using a contingency matrix as shown in Table 6.

Table 6: The contingency table

clusters	$C_1^{'}$	$C_2^{'}$	$\cdots$	$C_{k'}^{'}$	$\sum$
C_1	N_{11}	N_{12}	$\cdots$	$N_{1k'}$	c_1
C_2	N_{21}	N_{22}	$\cdots$	$N_{2k'}$	c_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
C_k	N_{k1}	N_{k2}	$\cdots$	$N_{kk'}$	c_k
$\sum$	$c_1^{'}$	$c_2^{'}$	$\cdots$	$c_{k'}^{'}$	N

The Adjusted Rand Index(ARI)[37] is defined as follows:

$$ARI(C, C') = \frac{r_0 - r_3}{0.5(r_1 + r_2) - r_3},$$

where $r_0 = \sum_{i=1}^{k}\sum_{j=1}^{k'}\binom{N_{ij}}{2}$, $r_1 = \sum_{i=1}^{k}\binom{c_i}{2}$, $r_2 = \sum_{j=1}^{k'}\binom{c_j'}{2}$, $r_3 = \frac{2r_1r_2}{N(N-1)}$, $\binom{n}{m}$ is the binomial coefficient. If the clustering result is close to the true class distribution, then the value of ARI is high.

In the following, the proposed algorithm is compared with the k-Modes algorithm and Chan's algorithm on accuracy and ARI, respectively. As the k-Modes algorithm is to choose initial cluster centers randomly, different initial

cluster centers will obtain different clustering results on the same data set. Therefore, we carried out 100 runs of the k-Modes algorithm, Chan's algorithm and the wk-Modes on the 6 standard data sets, respectively. In each run, the same initial cluster centers were used for three algorithms. This study sets the weight exponent to 1.1 in Chan's algorithm. The experimental results are shown in Table 7. Each value in the table is the average of 100 times experiments.

Table 7: The clustering accuracy and ARI of different algorithms on real data sets

Data sets	Accuracy			ARI		
	k-Modes	Chan's	wk-Modes	k-Modes	Chan's	wk-Modes
Soybean	0.8660	0.7117	0.8972	0.7293	0.5328	0.8054
Promoters	0.5991	0.7810	0.6099	0.0541	0.3232	0.0675
Vote	0.8620	0.7894	0.8651	0.5231	0.3638	0.5345
Breast-cancer	0.8377	0.7815	0.8471	0.4828	0.2881	0.5130
Mushroom	0.7872	0.6195	0.7905	0.3500	0.0020	0.3586
Connect-4 Opening	0.6583	0.6583	0.6583	0.0016	-0.0026	0.0019

From Table 7, it can be seen that the wk-Modes obviously outperforms the k-Modes algorithm and Chan's algorithm on these real data sets. This result can be explained by the reason that the attribute values in some dimensions are often the same in a cluster for categorical data. For example, on Mushroom data set, the values of attribute "veil-type" are all the same in whole data set.

In the meantime, the subspace of the clusters can also be identified by the weight values after clustering. We extract a few dimensions from each cluster whose weight are greater than $\frac{1}{m}$ after clustering, where m is the number of the attributes. The subspace dimensions associated with each cluster on different data sets are described in Table 8-13, respectively. The subspace information of each cluster in Table 8-13 is derived from the clustering results whose clustering accuracy are the highest among 100 runs of the wk-Modes on the 6 standard data sets, respectively.

Table 8: The subspace dimensions associated with each cluster on the soybean data set ($AC = 1$)

Clusters	Subspace dimensions
c_1	{2 3 8 11 13 14 15 16 17 18 19 20 21}
c_2	{3 4 7 11 13 14 15 16 18 19 20 21}
c_3	{2 3 4 5 11 13 15 16 17 18 19 20 21}
c_4	{2 7 11 12 14 15 17 18 19 20 21}

Table 8-13 show that the subspace dimensions associated with different clusters are different. Although the clustering accuracy of the wk-Modes and the k-Modes algorithm is the same in the Connect-4 Opening data set, the wk-Modes can obtain less dimensions which facilitate to interpret and understand

Table 9: The subspace dimensions associated with each cluster of the promoters data set ($AC = 0.8019$)

Clusters	Subspace dimensions
c_1	{3 4 5 14 15 16 22 24 26 27 28 31 32 34 36 40 42 45 50 57}
c_2	{6 7 11 13 14 15 16 17 18 19 31 33 37 39 40 41 43 49}

Table 10: The subspace dimensions associated with each cluster of the vote data set ($AC = 0.8851$)

Clusters	Subspace dimensions
c_1	{3 4 5 7 8 9 12}
c_2	{4 5 6 8 9 13 14 15}

Table 11: The subspace dimensions associated with each cluster of the Breast-cancer data set ($AC = 0.9413$)

Clusters	Subspace dimensions
c_1	{6 9}
c_2	{2 4 5 6 8 9}

Table 12: The subspace dimensions associated with each cluster of the Mushroom data set ($AC = 0.8887$)

Clusters	Subspace dimensions
c_1	{4 6 7 16 17 18 21}
c_2	{6 7 8 12 13 16 17 18 19}

Table 13: The subspace dimensions associated with each cluster of the Connect-4 Opening data set ($AC = 0.6583$)

Clusters	Subspace dimensions
c_1	{3 4 5 6 10 11 12 15 16 17 18 23 24 26 27 28 29 30 32 33 34 35 36 41 42}
c_2	{5 6 11 12 15 16 17 18 21 22 23 24 28 29 30 33 34 35 36 39 40 41 42}
c_3	{3 4 5 6 9 10 11 12 17 18 22 23 24 27 28 29 30 34 35 36 39 40 41 42}

the clustering result for users.

4.3. Evaluating Scalability

To test the scalability of the *wk*-Modes algorithm, we use a synthetic data generator [38] to generate data sets with different number of objects and attributes. The number of objects varies from 10,000 to 100,000, and the dimensionality is in the range of 10-50. In all synthetic data sets, each dimension

possesses 5 different attribute values. As the different clustering results will be obtained on the same data set when we select different initial cluster centers. Therefore, each value in the Fig.1 and Fig.2 is the average of 10 times experiments.

Fig.1 shows the scalability with data size of three algorithms. This study fixes the dimensionality to 10, and the cluster number to 3, and the data size varies from 10,000 to 100,000. It can be seen that the wk-Modes algorithm is linear with respect to the data size. The execution time of the proposed method is nearly the k-modes algorithm. Therefore, the wk-Modes algorithm can ensure efficient execution when the data size is large.

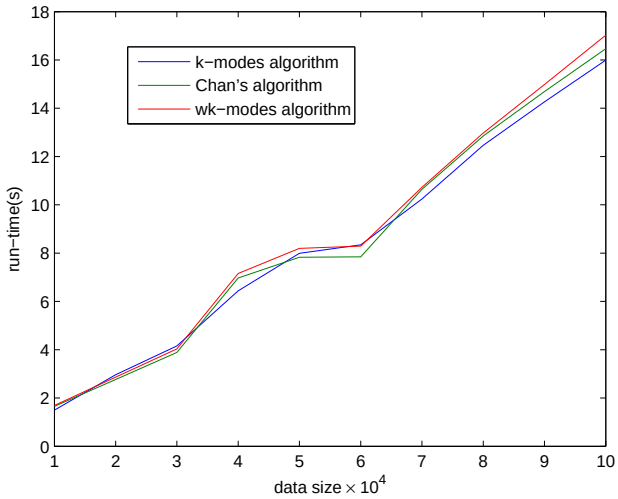

Figure 1: Execution time comparison of three algorithms: scalability with data size

Fig.2 shows the scalability with data dimensionality of three algorithms. We fix the data size to 30,000, and the cluster number to 3, and the number of dimensions varies from 10 to 50. It is clear that the run-time of the wk-Modes algorithm increases linearly as the number of dimensions increases, which is inferior to k-mode algorithm, but it is tolerant in practical use. Furthermore, the run-time of the wk-Modes algorithm is closer to that of Chan's algorithm. The result can be explained by the reason that the weight of each dimension need to be computed on each iteration.

5. Conclusion

Most clustering algorithms do not work efficiently for high dimensional data. Due to the inherent sparsity of objects, it is not feasible to identify interesting clusters in the whole data space. In this paper, we presented the attributes-weighting k-Modes algorithm based on complement entropy for subspace clustering of categorical data. In the proposed algorithm, different dimensions have

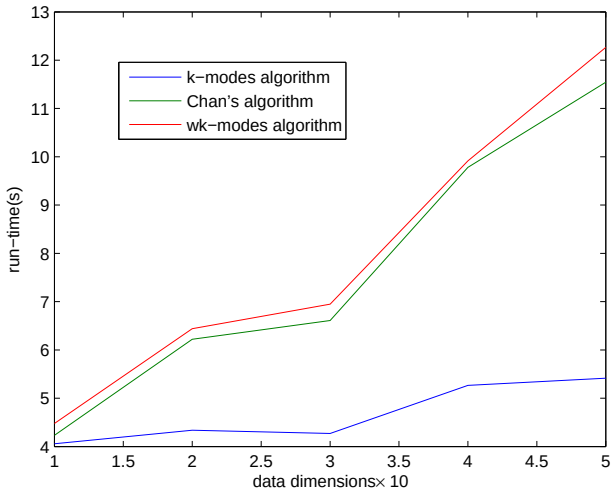

Figure 2: Execution time comparison of three algorithms: scalability with data dimensionality

different weights to a given cluster. According to the analysis of time complexity and scalability test, the wk-Modes algorithm can be used in large scale data sets. Furthermore, we extract a few dimensions from each cluster whose weights are greater than a given threshold after clustering. It follows that the subspace dimensions associated with each cluster can be derived from the clustering results obtained. Experimental results show the effectiveness of the proposed algorithm on real and synthetic data sets.

Acknowledgements

The authors are very grateful to the anonymous reviewers and editor. Their many helpful and constructive comments and suggestions helped us significantly improve this work. This work was supported by the National Natural Science Foundation of China (Nos. 71031006,70971080,60970014), the Special Prophase Project on National Key Basic Research and Development Program of China (973) (No. 2011CB311805), the Natural Science Foundation of Shanxi (Nos. 2010021016-2, 2010011021-1), and China Postdoctoral Science Foundation(No.2012M510046).

[1] J. Han, M. Kamber. Data Mining Concepts and Techniques. San Francisco: Morgan Kaufmann, 2001

[2] R. Xu, D. Wu. Survey of clustering algorithms. IEEE Transactions on Neural Networks, 2005, 16(3): 645-678

[3] P. Berkhin. Survey of clustering data mining techniques. Technical report, Accrue Software, San Jose, CA, 2002

[4] C.C. Aggarwal, J.L. Wolf, P.S. Yu, C. Procopiuc, J.S. Park. Fast algorithms for projected clustering. In Proceedings of the ACM SIGMOD International Conference on Management of Data, 1999, 61-72

[5] A. Blum, P. Langley. Selection of relevant features and examples in machine learning. Artificial Intelligence, 1997, 97: 245-271

[6] H. Liu, H. Motoda. Feature Selection for Knowledge Discovery and Data Mining. Boston: Kluwer Academic Publishers, 1998

[7] J. M. Pena, J. A. Lozano, P. Larranaga, I.Inza. Dimensionality reduction in unsupervised learning of conditional gaussian network. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2001, 23(6): 590-630

[8] L.P. Jing, M.K. Ng, J.Z. Huang. An entropy weighting k-means algorithm for subspace clusteringof high-dimensional sparse data. IEEE Transactions on Knowledge and Data Engineering, 2007, 19(8): 1026-1041

[9] C.C. Hsu, S.H. Lin, W.S. Tai. Apply extended self-organizing map to cluster and classify mixed-type data. Neurocomputing, 2011,74(18):3832-3842

[10] R. Agrawal, J. Gehrke, D. Gunopulos, P. Raghavan. Automatic subspace clusteing of high dimensional data for data mining application. In proceedings of ACM SIGMOD International Conference on Management of Data, 1998, 94-105

[11] C.C Aggarwal, P.S. Yu. Finding generalized projected clusters in high dimensional spaces. In proceedings of ACM SIGMOD International Conference on Management of Data, 2000, 70-81

[12] K. Chakrabarti, S. Mehrotra. Local dimensionality reduction: a new approach to indexing high dimensional space. In proceedings of 26th International Conference on very large data bases, 2000, 89-100

[13] C.M. Procopiuc, M. Jones, P.K. Agarwal, T.M. Murali. A monte carlo algorithm for fast projected clustering. In proceedings of ACM SIGMOD International Conference on Management of Data, 2002, 418-427

[14] K.Y. Yip, D.W. Cheung, M.K. Ng. A practical projected clustering algorithm. IEEE Transactions on Knowldege and Data Engineering, 2004, 16(11): 1397-1397

[15] K.Y. Yip, D.W. Cheung, M.K. Ng. On discovery of extremely low-dimensional clsuters using semi-supervised projected clustering. In proceedings of 21th International Conference on Data Engineering, 2005, 329-340

[16] R. Agrawal, J. Gehrke, D. Gunopulos, P. Raghavan. Automatic subspace clustering of high dimensional data. Data Mining and Knowledge Discovery, 2005, 11: 5-33

[17] W.S. Desarbo, J.D. Carroll, L.A. Clark, P.E. Green. Synthesized clustering: A method for amalgamating clustering bases with differential weighting variables, Psychometrika, 1984, 49: 57-78

[18] G.W. Milligan. A validation study of a variable weighting algorithm for cluster analysis. Journal of Classification, 1989, 6: 53-71

[19] D.S. Modha, W.S. Spangler. Feature weighting in k-Means clustering. Machine Learning, 2003, 52: 217-237

[20] H. Frigui, O. Nasraoui. Unsupervised learning of prototypes and attribute weights. Pattern Recognition, 2004, 37(3): 567- 581

[21] J.Z. Huang, M.K. Ng, H. Rong, Z. Li. Automated variable weighting in k-Means type clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(5): 1-12

[22] M. Kim, R.S. Ramakrishna. Projected clustering for categorical datasets. Pattern Recognization, 2006, 17: 1405-1417

[23] G.J. Gan, J.H. Wu. Subspace clustering for high dimensional categorical data. ACM SIGKDD Explorations Newsletter, 2004, 6(2): 87-94

[24] G.J. Gan, J.H. Wu, Z.J. Yang. PARTCAT: A subspace clustering algorithm for high dimensional categorical data. 2006 International Joint Conference on Nerual Networks, 2006, 16-21

[25] M.J. Zaki, M. Peters, I. Assent, T. Seidl. Clicks: An effective algorithm for mining subspace clusters in categorical datasets. Data & Knowledge Engineering, 2007, 60: 51-70

[26] E.Y. Chan, W.K. Ching, M.K. Ng, J.Z. Huang. An optimization algorithm for clustering using weighted dissimilaity measure. Pattern Recoginzation, 2004, 37(5): 943-952

[27] Z. X. Huang. Extensions to the k-Means algorithm for clustering large data sets with categorical values. Data Mining and Knowledge Discovery, 1998, 2(3): 283-304

[28] J.Z. Huang, M.K. Ng. A fuzzy k-modes algorithm for clustering categorical data. IEEE Transactions on Fuzzy Systems, 1999, 7(4): 446-452

[29] D.W. Kim, K.H. Lee, D. Lee. Fuzzy clustering of categorical data using fuzzy centroids. Pattern Recognition Letters, 2004, 25: 1263-1271

[30] G. De Soete. Optimal variable weighting for ultrametric and additive tree clustering. Quality and Quantity, 1986, 20: 169-180

[31] G. De Soete. OVWTRE: A program for optimal variable weighting for ultrametric and additive tree fitting. Journal of Classification, 1988, 5: 101-104

[32] Z. Pawlak. Rough sets: Theoretical Aspects of Reasoning about Data. Dordrecht: Kluwer Academic Publishers, 1991

[33] F.Y. Cao, J.Y. Liang, L. Bai, X. W. Zhao, C.Y. Dang. A framework for clustering categorical time-evolving data. IEEE Transactions on Fuzzy Systems, 2010,18(5):872-885

[34] J.Y. Liang, K.S. Chin, C.Y. Dang. A new method for measuring uncertainty and fuzziness in rough set theory. International Journal General Systems, 2002, 31(4): 331-342

[35] J.Y Liang, X.W Zhao, D.Y Li, F.Y Cao, C.Y Dang. Determining the number of clusters using information entropy for mixed data. Pattern Recognition, 2012,45(6): 2251-2265

[36] UCI Machine Learning Repository, http://www.ics.uci.edu/mlearn/ MLRepository.html,2009

[37] L. Hubert, P. Arabie. Comparing partitions. Journal of Classification, 1995, 2(1):193-218

[38] Data Generator: Perfect data for an imperfect world, http://www. generatedata.com, 2009

Fuyuan Cao received his M.S. and Ph.D. degrees in Computer Science from Shanxi University in 2004 and 2010, respectively. Now, he is an Associate Professor with the School of Computer and Information Technology in Shanxi University. His research interests include data mining and machine learning.

Jiye Liang is a professor of School of Computer and Information Technology and Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education at Shanxi University. He received his M.S. degree and Ph.D. degree from Xi'an Jiaotong University in 1990 and 2001, respectively. His current research interests include computational intelligence, granular computing, data mining and knowledge discovery. He has published more than 100 journal paper in his research fields.

Deyu Li is a professor of School of Computer and Information Technology of Shanxi University. He received his M.S. degree from Shanxi University in 1998, and his Ph.D.degree from Xi'an Jiaotong University in 2002. His current research interests include rough set theory, granular computing, data mining and knowledge discovery.

Xingwang Zhao is a teaching assistant in the School of Computer and Information Technology in Shanxi University. He received his M.S. degree from Shanxi University in 2011. His research interests are in the areas of data mining and machine learning.

Fuyuan Cao

Jiye Liang

Deyu Li

Xingwang Zhao