Available online at www.sciencedirect.com
ScienceDirect
Fuzzy Sets and Systems 161 (2010) 564 – 577
FUZZY sets and systems
www.elsevier.com/locate/fss

# Adaptive learning of ordinal–numerical mappings through fuzzy clustering for the objects of mixed features

Mahnhoon Lee[a,*], Witold Pedrycz[b,c]

[a] Department of Computing Science, Thompson Rivers University, 900 McGill Rd, Kamloops, BC, Canada V2C 5N3
[b] Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada
[c] Systems Research Institute, Polish Academy of Science, Warsaw, Poland

Received 20 June 2008; received in revised form 26 May 2009; accepted 27 May 2009
Available online 10 June 2009

**Abstract**

Ordinal feature values are totally ordered labels that can be considered as fuzzy sets. The formulation of proper fuzzy sets for ordinal labels is important for the systems that deal with the objects of mixed feature types. When a proper ordinal–numerical mapping for an ordinal feature of interest is given, proper fuzzy sets for the labels of the ordinal feature can be easily formulated. In this paper, we propose an adaptive method to learn proper ordinal–numerical mappings for ordinal features of interest from a given objects of mixed features including the ordinal features. The method starts with uniform ordinal–numerical mappings, and performs two steps iteratively. The first step computes a fuzzy partition over the given object set with the ordinal–numerical mappings. The second step learns new ordinal–numerical mappings from the new fuzzy partition in the way that the new mappings make the similarity between two ordinal labels be similar to the average similarity between the objects having the two labels, respectively. Through the alternate repetition of the two steps, both of the ordinal–numerical mappings and the clustering quality become gradually improved. The validity of the proposed method is strongly supported through the experiments with a modified fuzzy C-means clustering algorithm in which the proposed method is implemented.


*Keywords:* Ordinal–numerical mapping; Fuzzy clustering; Similarity; Fuzzy C-means clustering algorithm

## 1. Introduction

It is natural to consider identification of object types as one of the first steps to learn about the surrounding environments. The identification of object types comes from the comparison of similarities between objects. The similarity between two objects is an aggregation of the similarities between feature values of the two objects. Feature types can be numerical, ordinal or categorical (also called as nominal). The similarity between two numerical values can be measured with the distance between the two values. The similarity between two categorical values of the same type can be measured as 0 or 1 according to the equality of the two values. However, it is not clear how to measure the similarities between ordinal values of the same type. Note that the ordinal values of the same type have a total ordering. An ordinal feature may be treated as categorical, and then the information of total ordering among the ordinal labels

* Corresponding author. Tel.: +1 250 377 6022; fax: +1 250 377 6133.
*E-mail address:* mlee@tru.ca (M. Lee).

0165-0114/$ - see front matter Crown Copyright © 2009 Published by Elsevier B.V. All rights reserved.
doi:10.1016/j.fss.2009.05.011

cannot be used at all. An ordinal feature may be treated as numerical with the uniform ordinal–numerical mapping. This idea utilizes the total ordering among the ordinal labels. However, incorrect dissimilarity measurements can be obtained, because the actual dissimilarities between adjacent labels do not have to be all equal.

The values of an ordinal feature also can be considered as fuzzy sets. The formulation of proper fuzzy sets for an ordinal feature is important for the systems that deal with the objects of mixed feature types. The triangle-shape fuzzy sets can be easily formulated, when a proper *ordinal–numerical mapping* is given, which maps ordinal labels to numerical values in the way of preserving the relative similarities between labels. Therefore, it is very important for fuzzy systems to formulate proper ordinal–numerical mappings for ordinal features.

Meanwhile, following the survey work in [1], there have been a lot of efforts in the modeling of similarities between objects. Modeling the similarities between objects involves assumptions about both the representational structures used to describe the objects and the processes used to assess the similarities across these structures. In a seminal paper [2], the author points out that the dimensional approach, which is independent of a given context, is not always consistent with experimental results in psychology. The author defines a set-theoretic model for the similarity that considers the representation of objects in a given context, but not the objects themselves.

The author in [2] points out that the similarities between objects are dependent on the representation of objects in a given context. Similarly, we assert that the similarities between feature values are also dependent on the representation of feature values in a given context. This is because the similarity between objects is an aggregation of the similarities between feature values that describe the objects.

In this paper, we propose an adaptive method to learn proper ordinal–numerical mappings for ordinal features of interest from a given object set. To the extent of our knowledge, there has not been similar work, even though there have been a lot of work [3–8] for discretization of numerical values, which can be considered as the opposite concept of finding ordinal–numerical mappings.

In this paper, the given object set is understood as a context in which the labels of the ordinal features of interest are used to describe the objects. A fuzzy partition computed over the object set is considered as the structural information stored in the object set. The similarity between two objects is measured from the fuzzy partition, not from their geometrical closeness in the multidimensional feature space. The similarity between two ordinal labels is measured as the average similarity between the objects having the two labels, respectively. Now ordinal–numerical mappings are formulated in the way of preserving the relative similarities between labels. In summary, ordinal–numerical mappings are formulated to be adapted to a fuzzy partition computed over the given object set.

The fuzzy partition can be obtained through fuzzy clustering. However, one-time use of fuzzy clustering is not likely to give a correct fuzzy partition in the object set, when the objects have ordinal feature values. This is because the correct similarities between labels of the ordinal feature of interest are not known yet. In order to solve this problem, the proposed method in this paper uses the similar approach of iterative fuzzy clustering algorithms such as fuzzy C-means (hereinafter denoted by FCM) [9,10], G-K FCM [11], fuzzy C-shells [12] and expectation–maximization [13,14]. The method starts with uniform ordinal–numerical mappings, one for each ordinal feature of interest, and iteratively performs two steps. The first step computes a fuzzy partition over the given object set with the ordinal–numerical mappings. The second step learns new ordinal–numerical mappings from the new fuzzy partition. Through the alternate repetition of the two steps, both of the ordinal–numerical mappings and the clustering quality become gradually improved.

The validity of the proposed method is strongly supported through the experiments with synthetic object sets and real object sets in the UCI machine learning repository [15]. In the experiments, the proposed method is implemented in a fuzzy clustering algorithm (hereinafter denoted by AFCMO), which includes the two refinement steps of the FCM algorithm to obtain membership degrees and prototypes, respectively. (Note that any fuzzy clustering algorithm can be included in the AFCMO algorithm as a step to compute fuzzy partitions.) Throughout the experiments, we show that the AFCMO algorithm finds not only proper ordinal–numerical mappings but also better clusters than the FCM algorithm.

In the next section, we explain in detail the concept of adaptive learning of ordinal–numerical mappings. The formulation of ordinal–numerical mappings from the similarity degrees at the ordinal feature level, which is obtained from a given fuzzy partition, is explained. In Section 3 we give the background information of the general FCM algorithm to deal with the objects of mixed features. The AFCMO algorithm that implements the adaptive learning method is also explained. In Section 4, we explain the experimental results. Conclusion in Section 5 follows.

## 2. Adaptive learning of ordinal–numerical mappings

### 2.1. Model

It is not easy to measure the correct dissimilarity (or similarities) between two objects of mixed features including ordinal. This is because the dissimilarity between two objects is an aggregation, e.g., Euclidean distance, of feature-level dissimilarities, and there has not been a proper way to measure the dissimilarity between two ordinal labels. The next two simple dissimilarity measures have been used to measure the dissimilarities between ordinal labels. For all two labels, $l_p$ and $l_q$, of an ordinal feature having $n$ labels, $l_1, \ldots, l_n$,

$$d_1(l_p, l_q) = \begin{cases} 0 & \text{if } p \neq q \\ 1 & \text{otherwise} \end{cases} \tag{1}$$

$$d_2(l_p, l_q) = |g(l_p) - g(l_q)| \tag{2}$$

where $g$ is the *uniform ordinal–numerical mapping* for the ordinal feature, which is defined in Definition 1. The first dissimilarity measure in (1) deals with an ordinal feature as categorical, and the information of total ordering among the ordinal labels is not used at all. The second dissimilarity measure in (2) handles an ordinal feature as numerical with the uniform ordinal–numerical mapping for the ordinal feature. This idea utilizes the information of total ordering among the ordinal labels. However, this can lead to incorrect dissimilarity measurements because the actual dissimilarities between adjacent labels do not have to be all equal.

**Definition 1.** Let $l_1, \ldots, l_n$ be the labels of an ordinal feature, satisfying $l_1 < \cdots < l_n$.

An *ordinal–numerical mapping* is a function that maps the labels to the unit interval of real numbers.

$$g : \{l_1, \ldots, l_n\} \to [0, 1] \ni g(l_1) < \cdots < g(l_n) \tag{3}$$

The *uniform ordinal–numerical mapping* is an ordinal–numerical mapping such that

$$\forall r = 1..n, \quad g(l_r) = \frac{r}{n} - \frac{1}{2n} \tag{4}$$

The proposed method in this paper to learn proper ordinal–numerical mappings from a given object set uses an adaptive approach: the method starts with uniform ordinal–numerical mappings. The mappings are improved by being adapted to the fuzzy partition computed over the given object set with the previous ordinal–numerical mappings. Fig. 1 shows an illustration of the method. (Note that the fuzzy partition can be obtained through fuzzy clustering steps of any fuzzy clustering algorithm.) However, one-time use of fuzzy clustering steps is not likely to give the correct fuzzy partition in

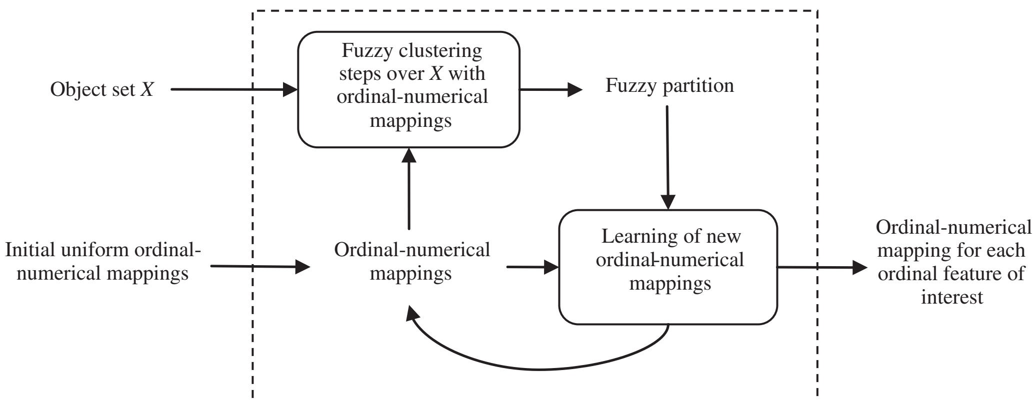


Fig. 1. An illustration of the adaptive method learning ordinal–numerical mappings from an object set. Fuzzy partition and ordinal–numerical mappings are gradually improved.

the object set, because the ordinal–numerical mappings probably do not reflect yet the correct dissimilarities between ordinal labels. The fuzzy partition also becomes gradually improved by using the new ordinal–numerical mappings, as illustrated in Fig. 1.

### 2.2. Formulation of ordinal–numerical mappings

As illustrated in Fig. 1, new ordinal–numerical mappings are repeatedly recomputed over the new fuzzy partition, so that the new ordinal–numerical mappings become gradually improved. The computation of new ordinal–numerical mappings is based on the similarities between adjacent ordinal labels. The similarity between two ordinal labels is defined as the average similarity between the objects having the two labels, respectively. The similarity between objects is obtained from the fuzzy partition.

These processes are explained in the bottom-up manner along the following definitions, starting with the definitions of similarity and fuzzy partition. In the following definitions and explanations, $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ is a given set of $N$ objects having $M$ features and $\mathbf{U} = [u_{ij}]_{c\times N}$ is a partition matrix over $X$ with the number of clusters $c$.

**Definition 2.** A *similarity* over a domain $D$ is a function to the unit interval of real numbers:

$$sim : D \times D \to [0, 1] \ni \forall x, y \in D, \quad sim(x, x) = 1; \quad sim(x, y) = sim(y, x); \quad sim(x, y) \le 1 \tag{5}$$

$\mathbf{U}$ is a *fuzzy c-partition* over $X$ when the followings are satisfied:

$$\begin{aligned} &\forall\, i = 1..c;\ j = 1..N, \quad 0 < u_{ij} < 1 \\ &\forall\, j = 1..N, \quad \sum_{i=1}^{c} u_{ij} = 1 \end{aligned} \tag{6}$$

Then $\mathbf{u}_i = (u_{i1}, \ldots, u_{iN})$ is a fuzzy set having $u_{ij}$ as the membership value of $\mathbf{x}_j$ on the crisp domain $X$. Every object $\mathbf{x}_t$ also can be expressed as a fuzzy set, called *fuzzy object* with respect to a fuzzy partition $\mathbf{U}$, on the domain of fuzzy sets, $\{\mathbf{u}_1, \ldots, \mathbf{u}_c\}$, with the memberships:

$$\mu_{\mathbf{x}_t}(\mathbf{u}_i) = \frac{u_{it}}{\sum_{j=1}^{N} u_{ij}} \tag{7}$$

The authors in [16,17] define a *fuzzy conditional probability relation* and use it in fuzzy information systems. A fuzzy conditional probability relation is a special case of *weak fuzzy similarity relation*. A weak fuzzy similarity relation is a generalization of *fuzzy similarity relation* in [18]. By applying the concept of fuzzy conditional probability relation to the definition in (7), the *conditional similarity* between two objects with respect to a fuzzy partition is obtained as follows. The conditional similarities are not symmetric in general. Hence we define the *context-based similarity between two objects* with respect to a fuzzy partition as the average of two conditional similarities between the two objects, so that the context-based similarities between two objects become symmetric. We finally define the *context-based similarity between two ordinal labels*.

**Definition 3.** Let the $r$th feature be ordinal. Let $L = \{l_1, \ldots, l_n\}$ be the set of all labels of the ordinal feature, satisfying $l_1 < l_2 < \cdots < l_n$, and let $g_r()$ be an ordinal–numerical mapping for the ordinal feature. For each $l_s$, let $X_s \subseteq X$ be the subset of the objects containing $l_s$, and assume $X_s \neq \emptyset$ without loss of generality.

The *conditional similarity* between two objects $\mathbf{x}_s$ and $\mathbf{x}_t$ with respect to $\mathbf{U}$ is defined by

$$f(\mathbf{U}, \mathbf{x}_s, \mathbf{x}_t) = \frac{\sum_{i=1}^{c} \min\{\mu_{\mathbf{x}_s}(\mathbf{u}_i), \mu_{\mathbf{x}_t}(\mathbf{u}_i)\}}{\sum_{i=1}^{c} \mu_{\mathbf{x}_t}(\mathbf{u}_i)} \tag{8}$$

The above definition is used in [16,17] to define a fuzzy conditional probability relation. Note that $\mu_{\mathbf{x}_s}(\mathbf{u}_i)$ is the normalized membership degree of $\mathbf{x}_s$ in the $i$th cluster, and the next equation is explained in [19]:

$$\sum_{i=1}^{c} \min\{\mu_{\mathbf{x}_s}(\mathbf{u}_i), \mu_{\mathbf{x}_t}(\mathbf{u}_i)\} = 1 - \frac{\sum_{i=1}^{c} |\mu_{\mathbf{x}_s}(\mathbf{u}_i) - \mu_{\mathbf{x}_t}(\mathbf{u}_i)|}{2} \tag{9}$$

The *context-based similarity between two objects* $\mathbf{x}_s$ and $\mathbf{x}_t$ with respect to $\mathbf{U}$ is defined by

$$h(\mathbf{U}, \mathbf{x}_s, \mathbf{x}_t) = \frac{f(\mathbf{U}, \mathbf{x}_s, \mathbf{x}_t) + f(\mathbf{U}, \mathbf{x}_t, \mathbf{x}_s)}{2} \tag{10}$$

From the formulas in (9) and (10), we can see that the context-based similarity explains the intrinsic fact that two similar objects must belong to the same clusters with similar membership degrees.

The *context-based similarities between two ordinal labels* with respect to $\mathbf{U}$ is defined as

$$cbsim(\mathbf{U}, l_s, l_t) = \begin{cases} 1 & \text{if } s = t \\ \dfrac{\sum_{\mathbf{x}_j \in X_s; \mathbf{x}_k \in X_t} h(\mathbf{U}, \mathbf{x}_j, \mathbf{x}_k)}{|X_s||X_t|} & \text{otherwise} \end{cases} \tag{11}$$

It can easily be shown that the above definition satisfies the three conditions in (5).

Now we explain how a new ordinal–numerical mapping can be formulated from the context-based similarities between ordinal labels, given a fuzzy partition and an ordinal–numerical mapping. Similarly let the $r$th feature be ordinal, $L = \{l_1, \ldots, l_n\}$ be the set of all labels of the ordinal feature, satisfying $l_1 < l_2 < \cdots < l_n$, and $g_r()$ be an ordinal–numerical mapping. Let us consider three consecutive labels, $l_{p-1}, l_p$ and $l_{p+1}$. When $1 - cbsim(\mathbf{U}, l_{p-1}, l_p) < 1 - cbsim(\mathbf{U}, l_p, l_{p+1})$, we can generally say $l_p$ is closer to $l_{p-1}$ than to $l_{p+1}$, for all $p = 2, \ldots, n-1$, in the given object set $X$. The numerical values for the two end labels are fixed first, and then the numerical values for the other labels are adjusted according to the relative similarities between adjacent labels, as follows:

$$\begin{aligned} & g_r^{(\text{new})} : L \to [0, 1] \ni \\ & \qquad g_r^{(\text{new})}(l_1) = \frac{1}{n+1} \\ & \qquad g_r^{(\text{new})}(l_s) = \frac{1}{n+1} + \left(1 - \frac{2}{n+1}\right) \frac{\sum_{i=2}^{s} (1 - cbsim(\mathbf{U}, l_{i-1}, l_i))}{\sum_{i=2}^{n} (1 - cbsim(\mathbf{U}, l_{i-1}, l_i))}, \quad \forall s = 2, \ldots, n-1 \\ & \qquad g_r^{(\text{new})}(l_n) = \frac{n}{n+1} \end{aligned} \tag{12}$$

## 3. A modified FCM clustering algorithm to learn ordinal–numerical mappings

As explained in the previous section, fuzzy clustering steps of any clustering algorithm can be used in the adaptive learning model illustrated in Fig. 1. Among many fuzzy clustering algorithms, the FCM algorithm [9,10] and its variances have been used successfully in many areas. We also use the clustering steps in the FCM algorithm for the experiments in Section 4.

The original FCM algorithm [9,10] partitions the objects of only numerical feature types. The original FCM algorithm can be generalized for the object set, in which objects have not only numerical but also ordinal and categorical features. This generalization can be done by replacing the Euclidean distance measure by different distance measures for different types of feature. Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ be an object set of $N$ objects of $M$ features, and let $\mathbf{x}_k = (x_{k1}, \ldots, x_{kM})$. The features can be numerical, ordinal or categorical. The general FCM algorithm is defined by the minimization of the next objective function

$$P = \sum_{i=1}^{c} \sum_{j=1}^{N} u_{ij}^m \sum_{k=1}^{M} d_k(x_{jk}, v_{ik})^2 \tag{13}$$

where $c$ is the number of clusters; $\mathbf{U} = [u_{ij}]_{c\times M}$ is a fuzzy partition over $X$; $m$ is the fuzzification coefficient; $\mathbf{v}_i = (v_{i1}, \dots, v_{iM})$ is the prototype of the $i$th cluster; and $d_k(x_{jk}, v_{ik})$ is the feature-level distance (or dissimilarity) measure between the values of the $k$th feature.

The prototype of each cluster can be obtained by fixing the membership and finding the prototype for each feature. Let $\mathbf{v}_s = (v_{s1}, \dots, v_{sM})$ be the prototype of the $s$th cluster. When the $r$th feature is numerical, $v_{sr}$ and the feature-level distance measure can be obtained as follows:

$$v_{sr} = \frac{\sum_{j=1}^{N} u_{sj}^m x_{jr}}{\sum_{j=1}^{N} u_{sj}^m} \tag{14}$$

$$d_r(x_{tr}, v_{sr})^2 = (x_{tr} - v_{sr})^2 \tag{15}$$

When the $r$th feature is ordinal, $v_{sr}$ can be obtained as follows, with an ordinal–numerical mapping $g_r()$ for the feature. $v_{sr}$ is now treated as numerical, not ordinal. The prototype and the feature-level distance measure are similarly defined as

$$v_{sr} = \frac{\sum_{j=1}^{N} u_{sj}^m g_r(x_{jr})}{\sum_{j=1}^{N} u_{sj}^m} \tag{16}$$

$$d_r(x_{tr}, v_{sr})^2 = (g_r(x_{tr}) - v_{sr})^2 \tag{17}$$

(Note another approach to handle ordinal feature for prototypes is to treat ordinal feature as categorical. But this approach is not used in this paper.)

There have been several approaches for the categorical prototype. Let us suppose the $r$th feature is categorical. In [20–26], a *medoid*, which is a most central value, is used as an element of a prototype. The prototype and the feature-level distance measure are defined as

$$v_{sr} = x_{tr} \ni t = \max_j \arg u_{sj} \tag{18}$$

$$d_r(x_{jr}, v_{sr})^2 = \begin{cases} 0, & x_{jr} \neq v_{sr} \\ 1, & x_{jr} = v_{sr} \end{cases} \tag{19}$$

Another approaches in [22,27] use a *mode* as an element of a prototype, which is a frequency distribution of labels. (Note that the convergence of the FCM algorithm with modes is proved in [28], and [22] explains the hard clustering version.)

$$\begin{aligned} v_{sr} &= (v_{sr1}, \dots, v_{srn_r}) \\ v_{srp} &= \frac{\sum_{j=1}^{N} u_{sj}^m \delta(l_{rp}, x_{jr})}{\sum_{j=1}^{N} u_{sj}^m} \\ \delta(l_{rp}, x_{jr}) &= \begin{cases} 0, & l_{rp} \neq x_{jr} \\ 1, & l_{rp} = x_{jr} \end{cases} \end{aligned} \tag{20}$$

where $l_{r1}, \dots, l_{rn_r}$ are the labels of the $r$th feature. The feature-level distance measure is defined as

$$d_r(x_{tr}, v_{sr}) = 1 - v_{srp} \quad \text{where } x_{tr} = l_{rp} \tag{21}$$

The membership in the general FCM algorithm is obtained in the same way of the original FCM algorithm, by fixing the prototypes and solving for the membership, with the distance measures defined in (15), (17) and (21) for numerical, ordinal and categorical features, respectively:

$$u_{st} = \frac{1}{\sum_{i=1}^{c} \left( \frac{\sum_{k=1}^{M} d_k(x_{tk}, v_{sk})^2}{\sum_{k=1}^{M} d_k(x_{tk}, v_{ik})^2} \right)^{1/(m-1)}} \tag{22}$$

In the general FCM clustering algorithm, the formulas for prototypes in (14), (16), and (20), and the formula for membership in (22) are computed repeatedly in a loop until some termination criteria are met. In the remainder of this paper, the general FCM clustering algorithm is used for the experiments, and the FCM clustering algorithm or FCM means the general FCM clustering algorithm.

The refining two steps are included in the fuzzy clustering algorithm, called AFCMO, to learn ordinal–numerical mappings. The uniform ordinal–numerical mappings are initially given. Fuzzy partition is obtained, and the fuzzy partition is used to obtain new ordinal–numerical mappings using the formula in (12). The new mappings would reflect the quality of clusters better than the previous mappings. The algorithm produces a new fuzzy partition with the new ordinal–numerical mappings, which is used to compute new ordinal–numerical mappings again. This cooperation continues until some criteria are satisfied.

**Algorithm.** Adaptive FCM for ordinal features (AFCMO).
Initialization:
Initialize all the uniform ordinal–numerical mappings for all the ordinal features of interest;
Initialize the memberships $u_{ij},\ \forall i = 1..c;\ \forall j = 1..N$;
Set $\varepsilon = 0.0001$;
Repeat:
(Step 1) Compute the prototypes $\mathbf{v}_i,\ \forall i = 1..c$, using (14), (16), and (20);
(Step 2) Compute the memberships $u_{ij},\ \forall i = 1..c;\ \forall j = 1..N$, using (22);
(Step 3) If $\max_{i,j}\{|old\_u_{ij} - new\_u_{ij}|\} \le 0.1$, then for each ordinal feature of interest, compute the new ordinal–numerical mapping, using (12);
while $\varepsilon \le \max_{i,j}\{|old\_u_{ij} - new\_u_{ij}|\}$;

## 4. Experiments with the adaptive FCM clustering algorithm for ordinal features (AFCMO)

The validity of the adaptive model to learn ordinal–numerical mappings, which was explained in Section 2, is strongly supported through three phases of extensive experiments. As explained in the previous section, the model is implemented in the AFCMO algorithm.

First we show that the AFCMO algorithm converges. If the adaptive model is wrong, the new ordinal–numerical mappings in Step 3 will be incorrect. This will make the new prototypes in Step 1 and the new memberships in Step 2 become much different from the old ones, and hence the AFCMO algorithm will not converge.

Second we show that the clustering results obtained by the AFCMO algorithm are generally better than the clustering results obtained by the FCM algorithm that uses uniform ordinal–numerical mappings. The two clustering algorithms are run in the situation that the algorithms extract the correct structural information stored in given object sets, i.e., the correct number of clusters is used for clustering. We use multiple fuzzy cluster validity indices to decide the correct number of clusters.

Third we use the $\chi^2$ statistics. The $\chi^2$ statistics have been used in discretization of numerical values in many proposals [3–8], in which the $\chi^2$ statistics determine if the relative class frequencies of adjacent intervals are distinctly different. If the $\chi^2$ value for two adjacent intervals is small, then the two intervals are generally merged in the proposals [3–8]. In this paper, we use the $\chi^2$ statistics as a tool to compare the ordinal–numerical mappings for each ordinal feature, by treating the mapped-numerical values of ordinals as numerical intervals, and fuzzy clusters as classes. We expect the $\chi^2$ statistics with the clustering results obtained by the AFCMO algorithm is better than the $\chi^2$ statistics with the clustering results obtained by the FCM algorithm.

### 4.1. Performance measures used in the experiments

Seven performance measures were used in the experiments to compare the clustering results between the AFCMO algorithm and the FCM algorithm. First, *number of iterations* (*NI*) is used to see if the AFCMO algorithm converges. Second, five fuzzy cluster validity indices are used to show the AFCMO algorithm computes better fuzzy clusters than the FCM algorithm. The five indices are *MPC* [12], *CPI* [29], *XB* [30], *TANG* [31] and *COC* [32], and it was reported in [32] that these indices show better performance especially for the objects of mixed features. Third, $CHI^2$, which is defined in the following, is used to compare the $\chi^2$ statistics.

For the definitions of the above performance measures, we let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ be an object set of $N$ objects of $M$ features, $\mathbf{x}_j = \{x_{j1}, \ldots, x_{jM}\}$ be the $j$th object, $c$ be the number of clusters, $\mathbf{U} = [u_{ij}]_{c\times M}$ be a fuzzy partition over $X$, $m$ be the fuzzification coefficient, and $\mathbf{v}_i = (v_{i1}, \ldots, v_{iM})$ be the prototype of the $i$th cluster.

*MPC* in [12] finds $c$ maximizing

$$MPC(c) = 1 - \frac{c}{c-1}\left(1 - \frac{1}{N}\sum_{i=1}^{c}\sum_{j=1}^{N} u_{ij}^2\right) \tag{23}$$

*CPI* in [29] finds $c$ maximizing

$$CPI(c) = \frac{1}{c}\sum_{i=1}^{c}\frac{\sum_{j\neq k} u_{ij}u_{ik}\, prox(\mathbf{U}, \mathbf{x}_j, \mathbf{x}_k)}{\sum_{j\neq k} u_{ij}u_{sk}} - \frac{1}{c(c-1)}\sum_{i_1=1;i_2=1;i_1\neq i_2}^{c}\frac{\sum_{j\neq k} u_{i_1 j}u_{i_2 k}\, prox(\mathbf{U}, \mathbf{x}_j, \mathbf{x}_k)}{\sum_{j\neq k} u_{i_1 j}u_{i_2 k}} \tag{24}$$

where $prox(\mathbf{U}, \mathbf{x}_j, \mathbf{x}_k)$ is the proximity (i.e., similarity) between two objects with respect to $\mathbf{U}$. In the experiments, $h(\mathbf{U}, \mathbf{x}_j, \mathbf{x}_k)$ in (10) is used as $prox(\mathbf{U}, \mathbf{x}_j, \mathbf{x}_k)$.

*XB* in [30] finds $c$ minimizing

$$XB(c) = \frac{\sum_{i=1}^{c}\sum_{j=1}^{N} u_{ij}^m d(\mathbf{x}_j, \mathbf{v}_i)^2}{N \min_{i\neq k} d(\mathbf{v}_i, \mathbf{v}_k)^2} \tag{25}$$

In this paper, the dissimilarity between two prototypes at the categorical feature level is defined as the average difference between frequencies.

*TANG* in [31] finds $c$ minimizing

$$TANG(c) = \frac{\sum_{j=1}^{N}\sum_{i=1}^{c} u_{ij}^2 d(\mathbf{x}_j, \mathbf{v}_i)^2 + \dfrac{1}{c(c-1)}\sum_{k\neq i}^{c} d(\mathbf{v}_k, \mathbf{v}_i)^2}{\dfrac{1}{c} + \min_{k\neq i} d(\mathbf{v}_k, \mathbf{v}_i)^2} \tag{26}$$

*COC* in [32] finds $c$ maximizing

$$COC(c) = \frac{\dfrac{1}{N}\sum_{j=1}^{N} \max_{1\leq i\leq c}\{u_{ij}\} - \dfrac{1}{c}}{1 - \dfrac{1}{c}} \tag{27}$$

The formula for computing the $\chi^2(k)$ value for the two adjacent $k$th and $(k+1)$th intervals is defined in [3–8] as follows:

$$\chi^2(k) = \sum_{j=k}^{k+1}\sum_{i=1}^{c}\frac{(A_{ij} - E_{ij})^2}{E_{ij}} \tag{28}$$

where $c$ is the number of classes; $A_{ij}$ the number of examples in $j$th interval and $i$th class; $R_j$ the number of examples in $j$th interval $= \sum_{i=1}^{c} A_{ij}$; $C_i$ the number of examples in $i$th class $= \sum_{j=k}^{k+1} A_{ij}$; $P$ the total number of examples $= \sum_{i=1}^{c} C_i$; $E_{ij}$ the expected frequency of $A_{ij} = R_j \times C_i / P$.

(Note that according to [8], in accurate discretization, two adjacent intervals should not have similar relative class frequencies, i.e., small $\chi^2$ values.) Let the $r$th feature be ordinal, and let $a_1, \ldots, a_n$ be all the labels of the ordinal feature. The above definition of $\chi^2(k)$ value for the $k$th and $(k+1)$th intervals, which now represent the mapped-numerical values of $a_k$ and $a_{k+1}$ in our experiments, can be redefined with a fuzzy partition matrix, by replacing $c$ and $A_{ij}$, as follows:

$$c = \text{number of clusters}$$

$$A_{ij} = \sum_{k=1;x_{kr}=a_j}^{N} u_{ik}$$

Table 1
The abbreviations used in the experiments.

| | Abbr. | Description | |
|---|---|---|---|
| Objects | NO | Number of objects | |
| | NNF | Number of numerical features | |
| | NC | Number of classes | |
| | NFC | Number of found clusters by using fuzzy cluster validity indices | |
| | NOF | Number of ordinal features | |
| | NOFV | Number of ordinal feature values | |
| | NCF | Number of categorical features | |
| Clustering algorithm | FCM | The general fuzzy C-means algorithm | |
| | AFCMO | The adaptive FCM algorithm for ordinal | |
| Clustering performance measure | *MPC* | Modified partition coefficient | The larger, the better |
| | *CPI* | Proximity based index | The larger, the better |
| | *XB* | Xie and Ben's index | The smaller, the better |
| | *TANG* | Tang's index | The smaller, the better |
| | *COC* | Crispness of clusters | The larger, the better |
| | $CHI^2$ | $\chi^2$ index | The larger, the better |
| | *NI* | Number of iterations | The smaller, the better |

Table 2
Three synthetic object sets, in which objects have only numerical feature values in [0,1].

| Synthetic object sets | NO | NNF | NC |
|---|---|---|---|
| *Syn*-I | 500 | 2 | 2 |
| *Syn*-II | 500 | 2 | 5 |
| *Syn*-III | 500 | 10 | 6 |

The objects were generated with Gaussian distributions. As shown in Fig. 2, *Syn*-I has two well-separated clusters, *Syn*-II has three well-separated clusters and two overlapped clusters, and *Syn*-III has six separated clusters.

Now we define $CHI^2$ that is used to compare the $\chi^2$ statistics for the FCM algorithm and the AFCMO algorithm:

$$CHI^2 = \frac{\sum_{r=1}^{M} B_r}{\text{the number of ordinal features}} \quad \text{where } B_r = \begin{cases} \underset{k}{average}\, \chi^2(k) & \text{if the } r\text{th feature is ordinal} \\ 0 & \text{otherwise} \end{cases} \tag{29}$$

We expect a larger $CHI^2$ value for the fuzzy partition matrix obtained with more proper ordinal–numerical mappings.

### 4.2. Experimental results

In the experiments, some synthetic object sets were used for the purpose of explanation and visualization, and some real object sets obtained from the UCI machine learning repository [15] were also used for the comparison of clustering performance between the AFCMO algorithm and the FCM algorithm (Table 1). All the numerical features were mapped to the unit interval. Tables 2–4 show the information about the object sets used in the experiments, including the explanation how the synthetic object sets were obtained. Figs. 2 and 3 show the visualization of the synthetic object sets.

The AFCMO algorithm and the FCM algorithm were run 100 times to get average values for each experimental case. When the two algorithms were run for the experiments, the proper number of clusters was used for each object set, which was obtained by using the five fuzzy cluster validity indices, *MPC*, *CPI*, *XB*, *TANG* and *COC*, with the FCM algorithm. Different fuzzification coefficients are used for different object sets to obtain distinctive fuzzy cluster validity index values, especially for the objects of mixed features. $\varepsilon = 0.0001$ was used for the termination criterion.

Table 3
Five synthetic object sets, in which objects have ordinal feature values.

| Synthetic object sets | NO | NNF | NOF (NOFV) | NC | NFC |
|---|---|---|---|---|---|
| *Syn*-I.1 | 500 | 1 | 1 (5) | 2 | 2 |
| *Syn*-I.2 | 500 | 1 | 1 (10) | 2 | 2 |
| *Syn*-II.1 | 500 | 1 | 1 (5) | 5 | 4 |
| *Syn*-II.2 | 500 | 1 | 1 (10) | 5 | 4 |
| *Syn*-III.2 | 500 | 7 | 3 (10) | 6 | 6 |

The object sets were obtained from the object sets in Table 2, by discretizing one numerical feature each in *Syn*-I and *Syn*-II, and by discretizing three numerical features in *Syn*-III. The discretized values are used as ordinal labels.

Table 4
The real object sets obtained from the UCI machine learning repository [15].

| Real object sets | NO | NNF | NOF (NOFV) | NCF | NC | NFC |
|---|---|---|---|---|---|---|
| *CarEvaluation* | 1728 | 0 | 4 (4,4,3,3) | 2 | 4 | 3 |
| *ContraceptiveMethodChoice* (*CMC*) | 1473 | 2 | 4 (4,4,4,4) | 3 | 3 | 5 |
| *HayesRoth* | 132 | 0 | 2 (4,4,) | 2 | 3 | 10 |
| *StatlogHeart* | 170 | 6 | 1 (3) | 5 | 2 | 2 |
| *BreastCancerWisconsin* (*BCW*) | 699 | 6 | 3 (10,10,10) | | 2 | 2 |
| *CreditApproval* | 653 | 4 | 2 (5,5) | 9 | 2 | 3 |
| *Ecoli* | 336 | 5 | 2 (5,5) | | 8 | 6 |
| *Iris* | 150 | 3 | 1 (10) | 0 | 3 | 2 |

The numerical features were mapped to [0,1]. There are not many object sets that use ordinal features. Some numerical features are converted to ordinal features for experiments by using discretization, for the object sets, *BreastCancerWisconsin*, *CreditApproval*, *Ecoli*, and *Iris*.

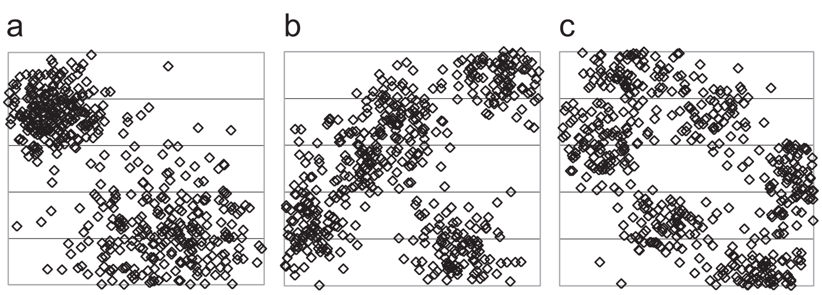


Fig. 2. Visualization of the synthetic object sets in Table 2. (c) shows the objects with only the values of the first two features, not all of the 10 features.

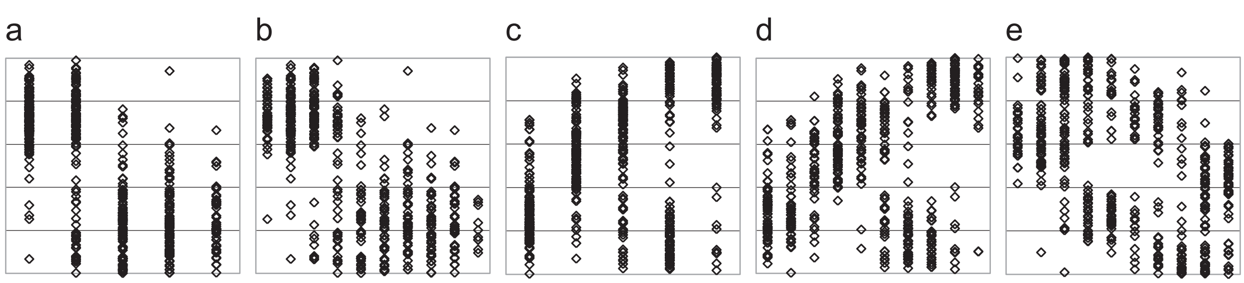


Fig. 3. Visualization of the object sets in Table 3, with uniform ordinal–numerical mappings on the horizontal axis. (e) shows only the values of the first two features, not all of the 10 features.

Table 5
Performance comparison between the AFCMO algorithm and the FCM algorithm, with the synthetic object sets in Table 3.

| Synthetic object sets | NFC | Algorithm | *MPC* | *CPI* | *XB* | *TANG* | *COC* | $CHI^2$ | *NI* |
|---|---|---|---|---|---|---|---|---|---|
| *Syn*-I.1 | 2 | FCM | 0.739 (3.64e − 07) | 0.585 (7.07e − 07) | 0.0958 (1.86e − 07) | 17.402 (3.20e − 05) | 0.838 (4.83e − 07) | 26.95 (5.19e − 05) | 10.17 (1.29e + 00) |
| | | AFCMO | 0.86 (3.60e − 07) | 0.752 (5.54e − 07) | 0.0897 (1.33e − 07) | 22.389 (5.95e − 05) | 0.907 (3.82e − 07) | 39.76 (1.44e − 04) | 13.92 (7.96e − 01) |
| *Syn*-I.2 | 2 | FCM | 0.746 (3.02e − 07) | 0.59 (6.08e − 07) | 0.0919 (1.32e − 07) | 16.706 (2.34e − 05) | 0.842 (4.04e − 07) | 3.82 (1.64e − 05) | 10.05 (1.04e + 00) |
| | | AFCMO | 0.854 (6.86e − 08) | 0.743 (1.10e − 07) | 0.0924 (5.27e − 08) | 22.974 (1.90e − 05) | 0.903 (5.83e − 08) | 6.45 (4.17e − 08) | 20.65 (1.35e + 00) |
| *Syn*-II.1 | 4 | FCM | 0.632 (2.37e − 06) | 0.457 (1.53e − 06) | 0.115 (1.29e − 06) | 17.848 (1.21e − 03) | 0.76 (6.57e − 07) | 61.78 (1.60e − 03) | 26.26 (3.35e + 00) |
| | | AFCMO | 0.664 (3.29e − 07) | 0.482 (2.61e − 07) | 0.11 (1.58e − 02) | 18.453 (7.11e − 06) | 0.777 (3.57e − 07) | 67.727 (2.30e − 03) | 25.19 (3.38e + 00) |
| *Syn*-II.2 | 4 | FCM | 0.645 (3.06e − 07) | 0.47 (5.02e − 08) | 0.116 (1.14e − 02) | 17.237 (7.01e − 04) | 0.77 (2.13e − 07) | 11.022 (9.77e − 05) | 22.46 (2.22e + 00) |
| | | AFCMO | 0.663 (3.61e − 08) | 0.488 (2.92e − 08) | 0.107 (1.78e − 02) | 18.76 (6.88e − 06) | 0.782 (3.62e − 08) | 11.745 (2.47e − 04) | 23.22 (2.62e + 00) |
| *Syn*-III.2 | 6 | FCM | 0.595 (3.05e − 07) | 0.432 (3.16e − 07) | 0.383 (7.30e − 07) | 28.237 (9.24e − 05) | 0.766 (1.93e − 07) | 10.81 (3.64e − 06) | 23.44 (2.98e + 00) |
| | | AFCMO | 0.598 (1.32e − 08) | 0.436 (1.29e − 08) | 0.384 (5.84e − 08) | 28.376 (5.59e − 06) | 0.768 (8.51e − 09) | 10.897 (2.37e − 06) | 23.4 (2.68e + 00) |

The fuzzification coefficient of $m = 2.0$ was used. The values in parenthesis are standard deviations.

First, the convergence of the AFCMO algorithm is obviously shown in Tables 5 and 6 with the *NI* values. *NI* values for the AFCMO algorithm are a bit larger than *NI* values for the FCM algorithm, in most object sets. This is because the computation of membership degrees in the AFCMO algorithm is a bit affected with the new ordinal–numerical mappings obtained by the adaptive learning method.

Second, for the quality of clusters computed by the AFCMO algorithm, five fuzzy clustering indices were used as we explained in the previous subsection. Tables 5 and 6 show the better results in most cases were obtained by the AFCMO algorithm.

Third, for the test of $\chi^2$ statistics, Tables 5 and 6 also show larger $CHI^2$ values for the AFCMO algorithm than the FCM algorithm. This means, when we treat the mapped-numerical values of ordinal labels as intervals, the $\chi^2$ values with the final ordinal–numerical mappings obtained by the AFCMO algorithm are better than the $\chi^2$ values with the initial uniform ordinal–numerical mappings used in the FCM algorithm.

For some object sets, the AFCMO algorithm shows much better fuzzy cluster validity index values. For example, for *Syn*-I.1, *Syn*-I.2, *Syn*-II.1 and *Syn*-II.2 in Table 5, the index values with the AFCMO algorithm are much better than the index values with the FCM algorithm. For *Syn*-III.2, the index values with the AFCMO algorithm are similar to the index values with the FCM algorithm. This explains that the degree how much the proposed adaptive model reflects the information stored in a given object set can be various with different object sets. Table 7 and Fig. 4 show the ordinal–numerical mappings obtained by the AFCMO algorithm reflect very well the clusters that are well separated in *Syn*-I.1, *Syn*-I.2, *Syn*-II.1 and *Syn*-II.2, but not in *Syn*-III.2.

We performed another experiment. *Syn*-I.1 and *Syn*-I.2 were obtained from the same synthetic object set *Syn*-I by discretizing into different numbers of ordinal labels, 5 and 10, respectively. Similarly, *Syn*-II.1 and *Syn*-II.2 were

Table 6
Performance comparison between the AFCMO algorithm and the FCM algorithm, with the real object sets in Table 4.

| Real object sets | NFC | Algorithm | *MPC* | *CPI* | *XB* | *TANG* | *COC* | *CHI*$^2$ | *NI* |
|---|---|---|---|---|---|---|---|---|---|
| *CarEvaluation* | 3 | FCM (with $m = 1.2$) | 0.826 (1.65e − 05) | 0.743 (2.27e − 05) | 1.612 (8.23e − 05) | 2.00e + 3 (6.66e − 02) | 0.906 (9.08e − 06) | 3.09e − 13 (2.75e − 13) | 22.99 (2.51e + 00) |
| | | AFCMO (with $m = 1.2$) | 0.826 (1.48e − 05) | 0.743 (2.04e − 05) | 1.612 (6.53e + 00) | 2.00e + 3 (5.97e − 02) | 0.909 (8.15e − 06) | 2.85e − 13 (3.08e − 13) | 22.59 (2.54e + 00) |
| *CMC* | 5 | FCM (with $m = 1.3$) | 0.943 (2.92e − 02) | 0.834 (4.50e − 02) | 0.285 (7.82e − 02) | 294.347 (9.13e + 01) | 0.965 (2.05e − 02) | 24.849 (1.04e − 01) | 18.46 (4.43e + 00) |
| | | AFCMO (with $m = 1.3$) | 0.95 (1.58e − 02) | 0.847 (3.05e − 02) | 0.27 (4.09e − 02) | 278.442 (4.68e + 01) | 0.97 (1.11e − 02) | 24.838 (1.15e − 01) | 19.47 (3.31e + 00) |
| *HayesRoth* | 10 | FCM (with $m = 1.3$) | 0.93 (4.17e − 03) | 0.843 (1.42e − 02) | 0.292 (4.29e − 02) | 42.106 (1.02e + 00) | 0.942 (4.63e − 03) | 13.01 (2.96e − 01) | 47.1 (1.31e + 01) |
| | | AFCMO (with $m = 1.3$) | 0.931 (3.07e − 03) | 0.848 (1.04e − 02) | 0.298 (4.50e − 02) | 41.917 (9.94e − 01) | 0.944 (3.24e − 03) | 13.068 (2.44e − 01) | 36.12 (8.08e + 00) |
| *StatlogHeart* | 2 | FCM (with $m = 1.3$) | 0.213 (2.35e − 05) | 0.072 (1.24e − 05) | 4.67 (2.63e − 04) | 508.081 (1.71e − 02) | 0.411 (2.39e − 05) | 4.02 (4.94e − 04) | 30.72 (4.56e + 00) |
| | | AFCMO (with $m = 1.3$) | 0.213 (2.57e − 05) | 0.0722 (1.35e − 05) | 4.665 (4.15e − 04) | 507.688 (3.16e − 02) | 0.411 (2.61e − 05) | 4.07 (4.83e − 04) | 34.3 (5.08e + 00) |
| *BCW* | 2 | FCM (with $m = 2.0$) | 0.677 (4.85e − 07) | 0.47 (1.97e − 07) | 0.338 (1.98e − 06) | 65.441 (6.49e − 04) | 0.787 (1.03e − 07) | 2.28 (1.40e − 05) | 11.55 (1.06e + 00) |
| | | AFCMO (with $m = 2.0$) | 0.698 (1.65e − 06) | 0.505 (1.41e − 06) | 0.319 (1.80e − 06) | 61.14 (6.18e − 04) | 0.805 (3.03e − 08) | 2.71 (4.20e − 05) | 14.88 (1.25e + 00) |
| *CreditApproval* | 3 | FCM (with $m = 1.2$) | 0.309 (1.88e − 05) | 0.18 (1.22e − 05) | 5.793 (2.77e − 01) | 1723 (1.23e − 01) | 0.5 (1.77e − 05) | 9.82e − 09 (6.29e − 09) | 58.35 (6.96e + 00) |
| | | AFCMO (with $m = 1.2$) | 0.306 (2.09e − 05) | 0.178 (1.35e − 05) | 5.849 (2.80e − 01) | 1727 (2.26e − 01) | 0.498 (1.98e − 05) | 1.16e − 08 (6.73e − 09) | 58.85 (6.99e + 00) |
| *Ecoli* | 6 | FCM (with $m = 2.0$) | 0.25 (2.56e − 06) | 0.137 (1.54e − 06) | 0.726 (3.06e − 01) | 54.499 (3.68e − 04) | 0.424 (3.85e − 06) | 14.57 (1.26e − 04) | 41.58 (5.86e + 00) |
| | | AFCMO (with $m = 2.0$) | 0.27 (4.49e − 06) | 0.142 (2.63e − 06) | 0.654 (3.11e − 01) | 55.411 (6.12e − 04) | 0.443 (1.18e − 05) | 17.003 (3.74e − 04) | 45.38 (1.05e + 01) |
| *Iris* | 2 | FCM (with $m = 2.0$) | 0.713 (6.98e − 06) | 0.517 (8.87e − 06) | 0.149 (1.14e − 06) | 8.266 (6.99e − 05) | 0.82 (5.73e − 06) | 1.16 (2.07e − 05) | 9.31 (1.14e + 00) |
| | | AFCMO (with $m = 2.0$) | 0.735 (1.34e − 05) | 0.545 (1.60e − 05) | 0.161 (3.08e + 00) | 10.063 (7.01e − 05) | 0.832 (1.51e − 05) | 1.939 (9.87e − 06) | 18.79 (1.80e + 00) |

The different fuzzification coefficients $m$-values were used to obtain distinctive index values. The values in parenthesis are standard deviations.

obtained from the same synthetic object set *Syn*-II by discretizing into different numbers of ordinal labels, 5 and 10, respectively. In this way, as shown in Fig. 3, *Syn*-I.1 and *Syn*-I.2 have the very similar structural information, and *Syn*-II.1 and *Syn*-II.2 also have the very similar structural information. If the adaptive learning model is valid, the AFCMO algorithm that uses the model must show the very similar results. Fig. 4 shows the clusters in the four object sets, which were computed by the algorithm with the correct number of clusters. (a) and (b) show that *Syn*-I.1 and *Syn*-I-2

Table 7
The final ordinal–numerical mappings obtained by using the AFCMO algorithm.

| Labels | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| *Syn*-I.1 | 0.167 | 0.248 | 0.756 | 0.801 | 0.833 | | | | | |
| *Syn*-I.2 | 0.091 | 0.111 | 0.153 | 0.311 | 0.752 | 0.799 | 0.837 | 0.869 | 0.893 | 0.909 |
| *Syn*-II.1 | 0.167 | 0.347 | 0.445 | 0.656 | 0.833 | | | | | |
| *Syn*-II.2 | 0.091 | 0.137 | 0.250 | 0.337 | 0.389 | 0.484 | 0.623 | 0.729 | 0.855 | 0.909 |
| *Syn*-III.2 | 0.091 | 0.148 | 0.241 | 0.347 | 0.436 | 0.535 | 0.629 | 0.729 | 0.834 | 0.909 |

The AFCMO algorithm was run with the numbers of clusters $c = 2, 2, 4, 4, 6$, respectively. The last row shows the numerical values equivalent to ordinal labels of the first ordinal feature.

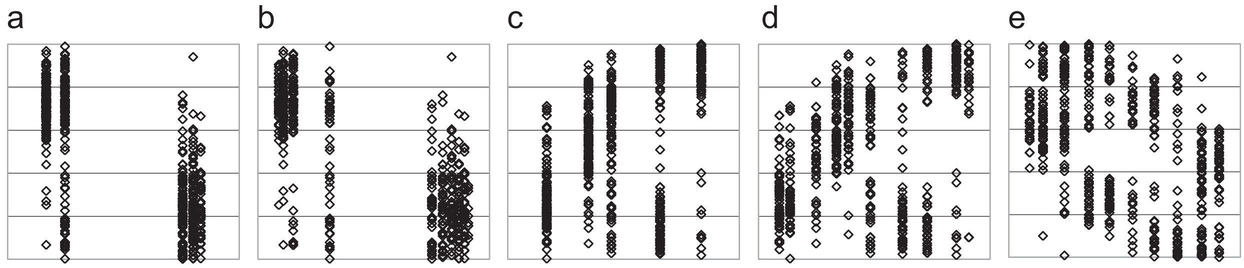


Fig. 4. The synthetic object sets with the ordinal–numerical mappings in Table 7, on the horizontal axis. (e) shows the values of the first two features, not all the 10 features.

have the very similar clusters, and (c) and (d) show that *Syn*-II.1 and *Syn*-II.2 also have the very similar clusters, as we expected.

## 5. Conclusion

We presented an adaptive method to learn proper ordinal–numerical mappings for ordinal features of interest from an object set, in which the ordinal feature labels are used to describe the objects. In the method fuzzy partitioning steps are included to compute a fuzzy partition over a given object set, and new ordinal–numerical mappings are formulated in the way to be adapted to the new fuzzy partition. These two processes are repeatedly performed so that the ordinal–numerical mappings as well as the fuzzy partition become gradually improved. The ordinal–numerical mapping is ready to be used to formulate triangular-shape fuzzy sets for the ordinal labels.

The proposed adaptive method was implemented in the adaptive fuzzy C-means clustering algorithm. The extensive experiments with the clustering algorithm showed that the method is valid and can be used to formulate proper ordinal–numerical mappings. The method also can be used as a tool to improve fuzzy clustering within any type of partitional iterative fuzzy clustering algorithm, for the objects of mixed features including ordinal.

The proposed adaptive method is dependent on fuzzy partitions. Different fuzzy partitions may be obtained with different fuzzy clustering algorithms for a given object set. Hence the method may find different ordinal–numerical mappings for a given object set, when different fuzzy clustering algorithms are used.

## References

[1] D.J. Navarro, M.D. Lee, Common and distinctive features in stimulus similarity: a modified version of the contrast model, Psychonomic Bulletin & Review 11 (2004) 961–974.
[2] A. Tversky, Features of similarity, Psychological Review 84 (1977) 327–352.
[3] H. Liu, R. Setiono, Chi2: feature selection and discretization of numeric attributes, in: IEEE 7th Internat. Conf. on Tools with Artificial Intelligence, 1995.
[4] M. Boulle, Khiops: a statistical discretization method of continuous attributes, Machine Learning 55 (2004) 53–69.
[5] F.E.H. Tay, L. Shen, A modified Chi2 algorithm to discretization, IEEE Transactions on Knowledge and Data Engineering 14 (2002) 666–670.

[6] C.T. Su, J.H. Hsu, An extended Chi2 algorithm for discretization of real value attributes, IEEE Transactions on Knowledge and Data Engineering 17 (2005) 437–441.
[7] S. Kotsiantis, D. Kanellopoulos, Discretization techniques: a recent survey, GESTS International Transactions on Computer Science and Engineering 32 (2006) 47–58.
[8] R. Kerber, ChiMerge: discretization of numeric attributes, in: AAAI-92, 1992, pp. 123–128.
[9] J.C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters, Journal of Cybernetics 3 (1973) 32–57.
[10] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1981.
[11] D.E. Gustafson, W.C. Kessel, Fuzzy clustering with a fuzzy covariance matrix, in: IEEE Conf. on Decision and Control Including the 17th Symp. on Adaptive processes, San Diego, CA, USA, 1979, pp. 761–766.
[12] R.N. Dave, Validating fuzzy partitions obtained through C-shells clustering, Pattern Recognition Letters 17 (1996) 613–623.
[13] A.P. Dempster, N.M. Laird, R.D. B., Maximum-likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society B (1977) 1–38.
[14] S. Zhong, J. Ghosh, A unified framework for model-based clustering, Journal of Machine Learning Research 4 (2003) 1001–1037.
[15] The UCI Machine Learning Repository, Center for Machine Learning and Intelligent Systems, UCI, 2008.
[16] R. Intan, M. Mukaidono, Degree of Similarity in Fuzzy Partition, in: AFSS Internat. Conf. on Fuzzy Systems, Calcutta, India, 2002, pp. 99–107.
[17] R. Intan, M. Mukaidono, Fuzzy conditional probability relations and their applications in fuzzy information systems, Knowledge and Information Systems 6 (2004) 345–365.
[18] L.A. Zadeh, Similarity relations and fuzzy orderings, Information Science 3 (1970) 177–200.
[19] M. Lee, Mapping of ordinal feature values to numerical values through fuzzy clustering, in: IEEE Internat. Conf. on Fuzzy Systems, Hong Kong, China, 2008, pp. 732–737.
[20] Z. Huang, Extensions to the k-means algorithm for clustering large data sets with categorical values, Data Mining and Knowledge Discovery 2 (1998) 283–304.
[21] Z. He, S. Deng, X. Xu, Improving K-modes algorithm considering frequencies of attribute values in mode, in: Computational Intelligence and Security, 2005, pp. 157–162.
[22] O.M. San, V.-N. Huynh, Y. Nakamori, An alternative extension of the k-means algorithm for clustering categorical data, International Journal of Applied Mathematics and Computer Science 14 (2004) 241–247.
[23] M.K. Ng, M.J. Li, J.Z. Huang, Z. He, On the impact of dissimilarity measure in k-modes clustering algorithm, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (2007) 503–507.
[24] Z. Huang, M.K. Ng, A fuzzy k-modes algorithm for clustering categorical data, IEEE Transactions on Fuzzy Systems 7 (1999) 446–452.
[25] S.C. Chu, J.F. Roddick, J.S. Pan, A comparative study and extensions to k-medoids algorithms, in: Internat. Conf. on Optimization: Techniques and Applications, Hong Kong, China, 2001, pp. 1708–1717.
[26] R. Krishnapuram, A. Joshi, L. Yi, A fuzzy relative of the k-medoids algorithm with application to web document and snippet clustering, in: IEEE Internat. Fuzzy Systems Conf., Seoul, Korea, 1999, pp. 1281–1286.
[27] D.W. Kim, K.H. Lee, D. Lee, Fuzzy clustering of categorical data using fuzzy centroids, Pattern Recognition Letters 25 (2004) 1263–1271.
[28] M. Lee, Fuzzy P-mode prototypes: a generalization of frequency-based cluster prototypes for clustering categorical objects, in: IEEE Computational Intelligence and Data Mining, Nashville, Tennessee, USA, 2009.
[29] M. Lee, Fuzzy cluster validity index based on object proximities defined over fuzzy partition Matrices, in: IEEE Internat. Conf. on Fuzzy Systems, Hong Kong, China, 2008, pp. 336–340.
[30] X.L. Xie, G.A. Beni, Validity measures for fuzzy clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 3 (1991) 841–846.
[31] Y. Tang, F. Sun, Z. Sun, Improved validation index for fuzzy clustering, in: American Control Conf., 2005, pp. 1120–1125.
[32] M. Lee, On fuzzy clustering validity indices for the objects of mixed features, Thompson Rivers University CS-TR-2008-3, December 2008.