Proceedings of the 2007 IEEE Symposium on
Computational Intelligence and Data Mining (CIDM 2007)

Fuzzy c-Means Classifier for Relational Data

Hidetomo Ichihashi, Katsuhiro Honda, Yasuhiro Kuramoto, and Fumiaki Matsuura
Graduate School of Engineering, Osaka Prefecture University
1-1 Gakuen-cho, Naka-ku, Sakai, Osaka 599-8531 Japan

***Abstract*— This paper proposes a relational version of the fuzzy c-means (FCM) classifier in which relational data instead of object data are used. The classifier based on the relational clustering is called "relational classifier". The classifier is useful when a feature space has an extremely high dimensionality that exceeds the number of objects and many of the feature values are missing, or when only relational data are available instead of the object data. The relational data is represented by a matrix in terms of distances (dissimilarity) between object data, and is not concerned with the relational database. The clustering algorithm used in the classifier includes, as a special case, the relational dual of FCM proposed by Hathaway, Davenport and Bezdek and can be seen as a simultaneous application of multidimensional scaling and clustering.**

The computational intensity of the classifier is comparable to Gaussian mixture classifier (GMC). The proposed classifier outperforms well established relational classifier known as k-nearest neighbor (k-NN) on several benchmark datasets from the UCI ML repository.

I. Introduction

The unsupervised partitioning of data is often called clustering, which forms a significant area of research effort. The fuzzy c-means (FCM) algorithms [1], [2] are widely used, effective tools for the problem of clustering n objects into groups of similar individuals when the data is available as object data, consisting of a set of feature vectors ($\boldsymbol{x}$).

In the FCM clustering, an entropy method that uses an additional entropy term or quadratic term for fuzzification was proposed [3]. Gaussian mixture models or normal mixtures [4] with the expectation maximizing algorithm [5], [6] is parameterized and derived from the FCM clustering with regularization by K-L information.

From the above consideration, a generalized FCM clustering is proposed and applied to a post-supervised classifier design and is called IRLS-FCM classifier (FCMC) [7], [8], [9]. The classifier with deterministic initialization clearly outperforms many well established classifiers [10]. Although the classification performace of the FCMC is high, its computation is not feasible when a feature space has an extremely high dimensionality (m) that exceeds the number of objects (n). The size of covariance matrices is $m \times m$, and its eigenvalue decomposition and inverse operation become difficult due to the restrictions of memory and computation time. This paper tackles the problem.

In the post-supervised design, clustering is implemented by using the data from one class at a time (i.e., on a per class basis). When working with the data class by class, the prototypes that are found for each labeled class already have the assigned physical labels. The unsupervised clustering plays a major role in the post-supervised classifier.

The fundamental distinction between types of clustering algorithms resides in types of data available. A second form of data that may appear is relational data. For example, text (character sequences) and web page sequences are non-numerical pattern sequences that can be represented numerically using (pairwise) relation matrices.

The fuzzy clustering algorithm for relational data was initiated by Ruspini [11] followed by Roubens [12], Windham [13], Hathaway, Davenport and Bezdek [14], and Kaufman and Rousseeuw [15], [16]. Davé and Sen [17] extended the relational FCM approach to handle data sets containing noise and outliers. Clustering method for relational data can also be found by relational alternating cluster estimation (RACE) [18]. Fuzzy c-medoids (FCMdd) [19] is based on selecting c representative objects (medoids) from the data set. An object datum that has the maximum membership in cluster i is chosen and specified as the cluster center $\boldsymbol{v}_i$.

The relational clustering used in the classifier of this paper is of the type of [14] but takes covariance structure of clusters into account. In the relational version of FCM algorithm, the n objects are implicitly described in terms of relational data, which consists of a set of n^2 measurements of relations between each of the pairs of objects. The relation can be represented by a matrix $R = (r_{ij})$, which comes about in either of two ways. First, when $\boldsymbol{x}$ is given as a numerical vector (object data), then distance between data vector pairs or some appropriate two place function like Pearson's correlation coefficient, can produce the data matrix R. This kind of data is used in the Internet related systems often referred to as recommender or collaborative filtering [20].

When a data set includes large number of objects and the number exceeds the data dimensionality, some method as local principal component analysis might be effective [21]. But, if the data dimension of $\boldsymbol{x}$ is extremely large (Case 1) and/or $\boldsymbol{x}$ includes missing values (Case 2), one may not be able to explicitly partition objects. Hence, we need to implicitly partition the set of objects by instead operating on R using relational clustering methods. The second way that R is obtained is directly from a human expert, or measuring device that supplies estimates of r_{ij} by observing or measuring the relation between ith and jth object pairs (Case 3).

We extend the generalized FCM clustering to those based on relational dissimilarity measures, which partitions the data set into flexible elliptic shapes of clusters. The relational

FCM clustering (RFCM) in [14] uses only the Euclidean distance so is a special case of our proposed method.

For reducing dimensionality of data space in each cluster, a way to control the number of parameters in the mixture of probabilistic principal component analysis (MPCA) [22] or the character recognition [23] is adopted. Whereas MPCA is a simultaneous approach to PCA and clustering, Our approach can be seen as a simultaneous approach to the multidimensional scaling (MDS) and FCM clustering.

The FCM clustering is applied to a classifier design in a similar manner to the Gaussian mixture model, which is applied to the classifier called Gaussian mixture classifier (GMC). The IRLS-FCM classifier is a modified GMC and the basic difference resides in the membership function and the matrix used. The relational FCM classifier (RFCMC) uses the matrices in terms of inner products instead of covariance matrices. When the data dimension is larger than the number of objects, the matrix of inner products is more convenient than the covariance matrix. In this sense, the computational intensity of the classifier is comparable to GMC.

The basic performance is compared with the well established relational classifier known as k-nearest neighbor (k-NN) by using relational data computed from the object data of the UCI ML repository (http://www.ics.uci.edu/~ mlearn/) [29]. The proposed classifier outperforms k-NN classifier on several benchmark data sets.

The paper is organized as follows. Section II gives a brief description of the generalized FCM clustering and the classifier design based on IRLS. The proposed relational classifier will be described in Section III. Section IV provides the results of numerical experiments. Section V concludes the paper.

II. FCM Classifier Based on IRLS

A. A Generalization of FCM Clustering

The clustering is used as an unsupervised phase of the classifier design. FCM clustering partitions data set by introducing memberships to fuzzy clusters. Let m dimensional vector $\boldsymbol{v}_i$ denote prototype parameter (i.e., cluster centroid). u_{ik} denotes the membership of k-th object datum to i-th cluster.

The objective function of the standard method is:

$$J_{\rm fcm} = \sum_{i=1}^{c}\sum_{k=1}^{n}(u_{ik})^{\lambda} d_{ik}^2, \quad (\lambda > 1). \tag{1}$$

d_{ik}^2 denotes the squared distance between $\boldsymbol{x}_k$ and $\boldsymbol{v}_i$, so the standard FCM objective function is the weighted sum of squared distances. Taking the objective function for the entropy-based method and the quadratic-term-based method [3] into account, we can generalize the standard objective function a little further as:

$$J_{\rm gfc} = \sum_{i=1}^{c}\sum_{k=1}^{n}(u_{ki})^{\lambda} d_{ik}^2 + \eta \sum_{i=1}^{c}\sum_{k=1}^{n}(u_{ik})^{\lambda}. \tag{2}$$

where $\eta > 0, \lambda > 1$. From the necessary condition of optimality, we have

$$u_{ik} = \left[\sum_{j=1}^{c}\left(\frac{\eta + d_{ik}^2}{\eta + d_{ik}^2}\right)^{\frac{1}{\lambda-1}}\right]^{-1}. \tag{3}$$

$$\boldsymbol{v}_i = \frac{\sum_{k=1}^{n}(u_{ik})^{\lambda}\boldsymbol{x}_k}{\sum_{k=1}^{n}(u_{ik})^{\lambda}}. \tag{4}$$

Gustafson and Kessel's modified FCM [24] can handle covariance structure and is derived from an FCM objective function with fuzzifier λ, though, we need to specify the values of determinant $|A_i|$ for all i. Otherwise, we need some modifications.

B. IRLS FCM Clustering

In order to deal with covariance structure within the scope of fuzzy c-means clustering, we proposed a simplified derivation of the algorithm, which is based on the IRLS technique. Runkler and Bezdek's fuzzy clustering scheme called alternating cluster estimation (ACE) [25] is this kind of simplification.

Now we consider to deploy a technique from the robust M-estimation [26], [27]. The M-estimators try to reduce the effect of outliers by replacing the squared residuals with ρ-function, which is chosen to be less increasing than square. Instead of solving directly this problem, we can implement it as the IRLS. While the IRLS approach does not guarantee the convergence to a global minimum, experimental results have shown reasonable convergence points. If one is concerned about local minima, the algorithm can be run multiple times with different initial conditions.

Let the objective function of the IRLS-FCM be

$$J_{\rm ifc} = \sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}\left(d_{ik}^2 + \log|A_i|\right), \tag{5}$$

where c denotes the number of clusters.

$$d_{ik}^2 = (\boldsymbol{x}_k - \boldsymbol{v}_i)^{\top} A_i^{-1}(\boldsymbol{x}_k - \boldsymbol{v}_i) \tag{6}$$

is squared Mahalanobis distance from data vector $\boldsymbol{x}_k$ to i-th cluster centroid, where $\top$ denotes transpose. A_i is a covariance matrix of data samples of the i-th cluster, which is derived from (5) as:

$$A_i = \frac{\sum_{k=1}^{n} u_{ik}(\boldsymbol{x}_k - \boldsymbol{v}_i)(\boldsymbol{x}_k - \boldsymbol{v}_i)^{\top}}{\sum_{k=1}^{n} u_{ik}}, \tag{7}$$

and $\boldsymbol{v}_i$ is derived as:

$$\boldsymbol{v}_i = \frac{\sum_{k=1}^{n} u_{ik}\boldsymbol{x}_k}{\sum_{k=1}^{n} u_{ik}}. \tag{8}$$

To facilitate competitive movements of cluster centroids, we need to define the membership function to be normalized as:

$$u_{ik} = \frac{u_{ik}^*}{\sum_{l=1}^{c} u_{lk}^*}. \tag{9}$$

We confine our discussion to the membership function

$$u_{ik}^{*} = \frac{\pi_i |A_i|^{-1/\gamma}}{(\eta + d_{ik}^2/0.1)^{1/\lambda},} \quad (10)$$

then, u_{ik} is written as:

$$u_{ik} = \pi_i |A_i|^{-1/\gamma} \left[\sum_{j=1}^{c} \left(\frac{\eta + d_{ik}^2/0.1}{\eta + d_{ik}^2/0.1} \right)^{\frac{1}{\lambda}} \pi_j |A_j|^{-1/\gamma} \right]^{-1}. \quad (11)$$

u^* is a modified and parameterized multivariational version of Cauchy's weight function in M-estimator or of the probability density function (PDF) of Cauchy distribution.

$$\pi_i = \frac{\sum_{k=1}^{n} u_{ik}}{\sum_{j=1}^{c} \sum_{k=1}^{n} u_{jk}} = \frac{1}{n} \sum_{k=1}^{n} u_{ik}. \quad (12)$$

C. FCM Classifier

After completing the clustering for each class, the classification is performed by computing class memberships of unseen test data. Let α_q denote the mixing proportion of class q, i.e., the $a\ priori$ probability of class q. The class membership of k-th data $\boldsymbol{x}_k$ to class q is computed as:

$$u_{qjk}^{*} = \pi_{qj} |A_{qj}|^{-1/\gamma} / (\eta + d_{qjk}^2/0.1)^{1/\lambda}, \quad (13)$$

$$\tilde{u}_{qk} = \alpha_q \sum_{j=1}^{c} u_{qjk}^{*} / \sum_{s=1}^{Q} \alpha_s \sum_{j=1}^{c} u_{sjk}^{*}, \quad (14)$$

where c denotes the number of clusters of each class. The denominator in (14) can be disregarded when applied solely for classification.

III. Relational FCM Classifier Based on IRLS

A. Relational FCM Clustering with Mahalanobis Distances

In the relational clustering, clusters are formed using the matrix $R = (d_{ij}^R)$ of relational data corresponding to pairwise distances between objects. Although explicit values of $\boldsymbol{x}'s$ are not known, if they are known, since

$$\begin{aligned} \|\boldsymbol{x}_j - \boldsymbol{x}_i\|^2 &= (\boldsymbol{x}_j - \boldsymbol{x}_i)^\top (\boldsymbol{x}_j - \boldsymbol{x}_i) \\ &= \boldsymbol{x}_j^\top \boldsymbol{x}_j - 2\boldsymbol{x}_i^\top \boldsymbol{x}_j + \boldsymbol{x}_i^\top \boldsymbol{x}_i, \end{aligned} \quad (15)$$

R can be written by a matrix form as

$$\begin{aligned} R = \mathbf{1}_n \mathbf{1}_n^\top \mathrm{diag}(XX^\top) - 2XX^\top \\ + \mathrm{diag}(XX^\top) \mathbf{1}_n \mathbf{1}_n^\top. \end{aligned} \quad (16)$$

$X = (\boldsymbol{x}_1, ..., \boldsymbol{x}_n)^\top$ and $XX^\top$ is a matrix in terms of inner product $(\boldsymbol{x}_k^\top \boldsymbol{x}_l)_{n \times n}$, where $(\ \ )_{n \times n}$ denotes a matrix of $n \times n$ dimension. $\mathbf{1}_n$ denotes the vector of dimension $n \times 1$ with all entries equal to 1. $\mathrm{diag}(XX^\top)$ denotes a diagonal matrix whose diagonal entries are composed of the diagonal elements of $XX^\top$. Let

$$Q_n = I_n - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^\top, \quad (17)$$

then $Q_n X$ is a matrix of centered $\boldsymbol{x}$. A matrix in terms of inner product of centered $\boldsymbol{x}$ is written as

$$\begin{aligned} X_0 X_0^\top &= Q_n X X^\top Q_n \\ &= -\frac{1}{2} Q_n R Q_n \end{aligned} \quad (18)$$

When the matrix in terms of inner product is given instead of R, this centering process can be omitted.

If exact values of the object data are known, fuzzy covariance matrix for the ith cluster is written in the matrix form as:

$$\begin{aligned} A_i &= \frac{1}{n\pi_i} X_i^\top M_i X_i \\ &= ((n\pi_i)^{-\frac{1}{2}} M_i^{\frac{1}{2}} X_i)^\top ((n\pi_i)^{-\frac{1}{2}} M_i^{\frac{1}{2}} X_i), \end{aligned} \quad (19)$$

where $X_i = (\boldsymbol{x}_1 - \boldsymbol{v}_i, ..., \boldsymbol{x}_n - \boldsymbol{v}_i)^\top$ and M_i is a diagonal matrix whose diagonal elements equal to $(u_1, ..., u_n)$, i.e., $M_i = \mathrm{diag}(u_1, ..., u_n)$. $\boldsymbol{v}_i$ is a cluster centroid. The dimension of $\boldsymbol{x}$ and $\boldsymbol{v}$ is m. $\boldsymbol{v}_i$ is a centroid for the ith cluster given by (8).

Eigenvalue decomposition of A_i is written as

$$A_i = W_i \Delta_i^2 W_i^\top, \quad (20)$$

where $W_i = (\boldsymbol{w}_{i1}, ..., \boldsymbol{w}_{ir})$ is an $m \times r$ matrix and $\boldsymbol{w}_{i1}, ..., \boldsymbol{w}_{ir}$ are eigenvectors associated with positive eigenvalues $(\delta_1^2, ..., \delta_r^2)$ of A_i. The vectors are normalized as $\boldsymbol{w}_{il}^\top \boldsymbol{w}_{il} = 1$. $\Delta^2 = \mathrm{diag}(\delta_1^2, ..., \delta_r^2)$ is a diagonal matrix of the eigenvalues. By the singular value decomposition

$$(n\pi_i)^{-\frac{1}{2}} M_i^{\frac{1}{2}} X_i = F_i \Delta_i W_i^\top, \quad (21)$$

we have

$$X_i W_i \Delta_i^{-1} = (n\pi_i)^{\frac{1}{2}} M_i^{-\frac{1}{2}} F_i, \quad (22)$$

where F_i is an $n \times r$ matrix.

The Mahalanobis distance between $\boldsymbol{x}_k$ and cluster centroid $\boldsymbol{v}_i$ can be written by using (22) as:

$$\begin{aligned} d_{ik}^2 &= (\boldsymbol{x}_k - \boldsymbol{v}_i)^\top A_i^{-1} (\boldsymbol{x}_k - \boldsymbol{v}_i) \\ &= (\boldsymbol{x}_k - \boldsymbol{v}_i)^\top W_i \Delta_i^{-2} W_i^\top (\boldsymbol{x}_k - \boldsymbol{v}_i) \\ &= ((\boldsymbol{x}_k - \boldsymbol{v}_i)^\top W_i \Delta_i^{-1}) \\ &\quad \times ((\boldsymbol{x}_k - \boldsymbol{v}_i)^\top W_i \Delta_i^{-1})^\top \\ &= n\pi_i u_{ik}^{-1} \boldsymbol{f}_{ik}^\top \boldsymbol{f}_{ik}, \end{aligned} \quad (23)$$

where $F_i = (\boldsymbol{f}_{i1}, ..., \boldsymbol{f}_{in})^\top$.

Since the explicit values of $\boldsymbol{x}$ are not known, for obtaining the values of F_i and Δ_i, let us define an $n \times n$ matrix K_i as:

$$K_i = (n\pi_i)^{-1} M_i^{\frac{1}{2}} X_i X_i^\top M_i^{\frac{1}{2}}, \quad (24)$$

where

$$X_i = (I_n - \mathbf{1}_n \overline{\boldsymbol{u}_i}^\top) X_0, \quad (25)$$

and I_n is a unit matrix of dimension n.

$$\overline{\boldsymbol{u}_i} = (u_{i1} / \sum_{k=1}^{n} u_{ik}, ..., u_{in} / \sum_{k=1}^{n} u_{ik})^\top, \quad (26)$$

and $X_i X_i^\top$ can be written as

$$\begin{aligned} X_i X_i^\top &= X_0 X_0^\top - X_0 X_0^\top \overline{\boldsymbol{u}_i} \mathbf{1}_n^\top \\ &\quad -\mathbf{1}_n \overline{\boldsymbol{u}_i}^\top X_0 X_0^\top \\ &\quad +(\mathbf{1}_n \overline{\boldsymbol{u}_i}^\top) X_0 X_0^\top (\overline{\boldsymbol{u}_i} \mathbf{1}_n^\top). \end{aligned} \tag{27}$$

F_i and Δ_i are obtained from the eigenvalue decomposition of K_i, since K_i can be rewritten by using (21) as:

$$\begin{aligned} K_i &= ((n\pi_i)^{-\frac{1}{2}} M_i^{\frac{1}{2}} X_i)((n\pi_i)^{-\frac{1}{2}} M_i^{\frac{1}{2}} X_i)^\top \\ &= (F_i \Delta_i W_i^\top)(W_i \Delta_i F_i^\top) \\ &= F_i \Delta_i^2 F_i^\top. \end{aligned} \tag{28}$$

The remaining value that we need for updating $\boldsymbol{u}_i$ is $|A_i|$. If $\mathrm{rank}(A_i) = m$,

$$|A_i| = |W_i||\Delta_i^2||W_i^\top| = |\Delta_i^2| = \prod_{l=1}^{r} \delta_{il}^2. \tag{29}$$

This is not always the case for all clusters and some modifications of the nonsingular covariance matrices are needed. Moreover, because this relational clustering approach can find up to n nonzero eigenvalues of K_i, reduction of the number of decision variables (F_i) is significant. Unlike the global nonlinear approaches, Gaussian mixture models or normal mixture [4] is to model nonlinear structure with a collection, or mixture, of local linear sub-models of PCA. When estimating covariance structures in high dimensions, while not over-constraining the model flexibility, Tipping and Bishop proposed a way to control the number of parameters in the mixture of probabilistic principal component analysis (MPCA) [22]. Honda *et al.* discussed the similarity between MPCA and FCV type fuzzy clustering with regularization by K-L information [28]. A common practice to estimate the unknown dimensionality r is to use the number of positive eigenvalues of A_i. And, we make this parameter an adjustable one to reduce the decision variables (F_i). Let A_i' denotes an approximation of A_i in (19) and (20) for $p < r$ as

$$A_i' = W_i^p((\Delta_i^p)^2 - \sigma_i^2 I_p) W_i^{p\top} + W_i(\sigma_i^2 I_r) W_i^\top, \tag{30}$$

where I_p is a unit matrix of dimension p. Note that $(A_i')^{-1}$ can be computed easily from (30). Δ_i^p is a diagonal matrix whose diagonal elements are p largest eigenvalues of A_i, and W_i^p is an $m \times p$ matrix consisted of corresponding p eigenvectors.

$$\begin{aligned} \sigma_i^2 &= \frac{1}{r-p} \sum_{l=p+1}^{r} \delta_{il}^2 \\ &= \frac{1}{r-p} (\mathrm{trace}(K_i) - \sum_{l=1}^{p} \delta_{il}^2). \end{aligned} \tag{31}$$

The squared distance between $\boldsymbol{x}_k$ and cluster centroid $\boldsymbol{v}_i$ can be approximated as

$$\begin{aligned} d_{ik}^2 &= (\boldsymbol{x}_k - \boldsymbol{v}_i)^\top A_i^{'-1} (\boldsymbol{x}_k - \boldsymbol{v}_i), \\ &= \sum_{l=1}^{p} \frac{1}{\delta_{il}^2} (\boldsymbol{x}_k - \boldsymbol{v}_i)^\top \boldsymbol{w}_{il} \times \\ &\quad (\boldsymbol{x}_k - \boldsymbol{v}_i)^\top \boldsymbol{w}_{il} \\ &+ \frac{1}{\sigma_i^2} \sum_{l=p+1}^{r} (\boldsymbol{x}_k - \boldsymbol{v}_i)^\top \boldsymbol{w}_{il} \times \\ &\quad (\boldsymbol{x}_k - \boldsymbol{v}_i)^\top \boldsymbol{w}_{il} \\ &= n\pi_i u_{ik}^{-1} \left(\sum_{l=1}^{p} f_{ikl}^2 + \frac{1}{\sigma_i^2} \sum_{l=p+1}^{r} f_{ikl}^2 \delta_{il}^2 \right) \\ &= n\pi_i u_{ik}^{-1} \boldsymbol{f}_{ik}^{'\top} \boldsymbol{f}_{ik}' \\ &+ \sigma_i^{-2} (\boldsymbol{x}_k - \boldsymbol{v}_i)^\top (\boldsymbol{x}_k - \boldsymbol{v}_i) \\ &- \sigma_i^{-2} n\pi_i u_{ik}^{-1} \boldsymbol{f}_{ik}^{'\top} (\Delta_i^p)^2 \boldsymbol{f}_{ik}'), \end{aligned} \tag{32}$$

where $\boldsymbol{f}_{ik}' = (f_{ik1}, ..., f_{ikp})^\top$, and $(\boldsymbol{x}_k - \boldsymbol{v}_i)^\top (\boldsymbol{x}_k - \boldsymbol{v}_i)$ is a diagonal element of $X_i X_i^\top$.

We approximate $|A_i|$ by the product of largest p eigenvalues of K_i and $\sigma_i^{2(r-p)}$ where r is the number of positive eigenvalues.

$$|A_i'| \simeq (\prod_{l=1}^{p} \delta_{il}^2) \sigma_i^{2(r-p)}. \tag{33}$$

The membership to cluster u_{ik} and the ratio π_i are given by (10), (11) and (12).

The algorithm is the repetition of these update for all clusters, i.e., $i = 1, ..., c$ and may be described as

Relational-IRLS-FCM clustering algorithm

Step 1: Initialize $\boldsymbol{u}_i$ with random numbers.
Step 2: Calculate π_i for all i by (12).
Step 3: Calculate K_i using (27) and its eigenvalue decomposition using (28).
Step 4: Calculate u_{ik} by (11).
Step 5: If iteration exceeds the predetermined number then terminate, else go to *Step 2*.

B. Relational Classifier

For classification, we need to calculate distances between new data points and cluster centers. From (21)

$$(n\pi_i)^{-\frac{1}{2}} X_i^\top M_i^{\frac{1}{2}} F_i \Delta_i^{-2} = W_i \Delta_i^{-1}, \tag{34}$$

and

$$(n\pi_i)^{-\frac{1}{2}} X_i X_i^\top M_i^{\frac{1}{2}} F_i \Delta_i^{-2} = X_i W_i \Delta_i^{-1}. \tag{35}$$

Hence, if n new unseen data are given, by replacing left-most X_i in both sides of (35) with X_i^{NEW}, we can calculate the right side of (22) by

$$\begin{aligned} X_i^{NEW} W_i \Delta_i^{-1} &= (n\pi_i)^{-\frac{1}{2}} X_i^{NEW} X_i^\top M_i^{\frac{1}{2}} F_i \Delta_i^{-2} \\ &= (n\pi_i)^{-\frac{1}{2}} (X_0^{NEW} X_0^\top \\ &+ X_i X_i^\top - X_0 X_0) M_i^{\frac{1}{2}} F_i \Delta_i^{-2}, \end{aligned} \tag{36}$$

where X_i^{NEW} denotes the matrix of new data centered to the i-th cluster centroid and X_0^{NEW} denotes the matrix of new data centered to the global mean. We used the fact that from (22)

$$\mathbf{1}_n^\top M_i^{\frac{1}{2}} F_i = \mathbf{0}^\top, \tag{37}$$

and

$$\begin{aligned} & (\boldsymbol{x}_i^{NEW})^\top X_i^\top - (\boldsymbol{x}_0^{NEW})^\top X_0^\top + \boldsymbol{x}_i^\top X_i^\top - \boldsymbol{x}_0^\top X_0^\top \\ = \; & (\boldsymbol{x}_i^{NEW} - \boldsymbol{x}_i)^\top X_i^\top - (\boldsymbol{x}_0^{NEW} - \boldsymbol{x}_0)^\top X_0^\top \\ = \; & (\boldsymbol{x}_i^{NEW} - \boldsymbol{x}_i)^\top (X_i - X_0)^\top \\ = \; & (\boldsymbol{x}_i^{NEW} - \boldsymbol{x}_i)^\top (\boldsymbol{v}_i - \boldsymbol{v}_0)\mathbf{1}_n^\top, \end{aligned} \tag{38}$$

where $\boldsymbol{x}_0^{NEW} (= \boldsymbol{x}_{k'} - \boldsymbol{v}_0)$ denotes an unseen new data vector centered to the global mean $\boldsymbol{v}_0$ and $\boldsymbol{x}_i^{NEW} (= \boldsymbol{x}_{k'} - \boldsymbol{v}_i)$ is the vector centered to i-th cluster centroid $\boldsymbol{v}_i$. Note that $\boldsymbol{v}_0$ and $\boldsymbol{v}_i$ are not known explicitly.

Usually, the number of new test data is not equal to n, but we see from (36) that for a single new datum $\boldsymbol{x}_{k'}$

$$\begin{aligned} & (\boldsymbol{x}_{k'} - \boldsymbol{v}_i)^\top W_i \Delta_i^{-1} = (n\pi_i)^{-\frac{1}{2}} ((\boldsymbol{x}_{k'} - \boldsymbol{v}_0)^\top X_0^\top \\ & + (\boldsymbol{x}_l - \boldsymbol{v}_i)^\top X_i^\top - (\boldsymbol{x}_l - \boldsymbol{v}_0)^\top X_0^\top) M_i^{\frac{1}{2}} F_i \Delta_i^{-2}, \end{aligned} \tag{39}$$

for any $l \in \{1, ..., n\}$.

For computing $d_{ik'}^2$ by (32), we need to calculate Euclidean distance between the new unseen data and cluster centroid ($||\boldsymbol{x}_{k'} - \boldsymbol{v}_i||^2$). Since

$$\boldsymbol{x}_i^{NEW} = \boldsymbol{x}_0^{NEW} - X_0^\top \overline{\boldsymbol{u}_i}, \tag{40}$$

the squared Euclidean distance between $\boldsymbol{x}_0^{NEW}$ and $\boldsymbol{v}_i$ is

$$\begin{aligned} ||\boldsymbol{x}_i^{NEW}||^2 &= (\boldsymbol{x}_0^{NEW})^\top \boldsymbol{x}_0^{NEW} - 2\overline{\boldsymbol{u}_i}^\top X_0 \boldsymbol{x}_0^{NEW} \\ &+ \overline{\boldsymbol{u}_i}^\top X_0 X_0^\top \overline{\boldsymbol{u}_i}, \end{aligned} \tag{41}$$

where $(\boldsymbol{x}_0^{NEW})^\top \boldsymbol{x}_0^{NEW}$ and $X_0 \boldsymbol{x}_0^{NEW}$ can be obtained from the entire relational data including the new unseen data in a similar way to (17)-(18).

It should be noted that if object data are available, even though the dimensinality is high, after clustering W_i can be computed from F_i, Δ_i and object data matrix. Therefore, the classification for new unseen data can be more simple and less time-consuming.

C. Relational Duals of FCM classifier

The relational clustering/classifier stated above implicitly handles covariance structure of each cluster and the Mahalanobis distances are used. If we omit the covariance structure and replace the matrix A_i with a unit matrix, then the algorithm reduces to a relational dual of FCM by Hathaway, Davenport and Bezdek [14].

When p=0, d_{ik}^2 in (32) is

$$d_{ik}^2 = \frac{1}{\sigma_i^2} (\boldsymbol{x}_k - \boldsymbol{v}_i)^\top (\boldsymbol{x}_k - \boldsymbol{v}_i), \tag{42}$$

where $(\boldsymbol{x}_k - \boldsymbol{v}_i)^\top (\boldsymbol{x}_k - \boldsymbol{v}_i)$ is a diagonal element of $X_i X_i^\top$. Duality of FCM in [14] states that this Euclidian squared distance between $\boldsymbol{x}_k$ and $\boldsymbol{v}_i$ can be written as

$$d_{ik}^2 = (R\hat{\boldsymbol{u}}_i)_k - \frac{1}{2} \hat{\boldsymbol{u}}_i{}^\top R \hat{\boldsymbol{u}}_i, \tag{43}$$

where

$$\hat{\boldsymbol{u}}_i = \left((u_{i1})^m / \sum_{k=1}^n (u_{ik})^m, ..., \; (u_{in})^m / \sum_{k=1}^n (u_{ik})^m \right)^\top. \tag{44}$$

In the case of RFCMC, if p=0, d_{ik}^2 in (32), i.e., a diagonal element of $X_i X_i^\top$ in (27), becomes equal to

$$d_{ik}^2 = (R\overline{\boldsymbol{u}_i})_k - \frac{1}{2} \overline{\boldsymbol{u}_i}^\top R \overline{\boldsymbol{u}_i}, \tag{45}$$

except that a fixed constant σ_i^2 is multiplied. This is straight forward if we write $X_i X_i^\top$ in terms of $R, \overline{\boldsymbol{u}_i}$ and $P_n = \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^\top$. Since

$$P_n \overline{\boldsymbol{u}_i} \mathbf{1}_n^\top = \mathbf{1}_n \overline{\boldsymbol{u}_i}^\top P_n = P_n, \tag{46}$$

and R is symmetric, we have

$$\begin{aligned} \mathrm{diag}(R P_n \overline{\boldsymbol{u}_i} \mathbf{1}_n^\top) &= \mathrm{diag}(R P_n) \\ &= \mathrm{diag}(P_n R) \\ &= \mathrm{diag}(\mathbf{1}_n \overline{\boldsymbol{u}_i}^\top P_n R), \end{aligned} \tag{47}$$

where diag(A) denotes the vector whose entries are composed of the diagonal elements of a square matrix A.

$$\begin{aligned} P_n R \overline{\boldsymbol{u}_i} \mathbf{1}_n^\top &= \mathbf{1}_n \overline{\boldsymbol{u}_i}^\top P_n R \overline{\boldsymbol{u}_i} \mathbf{1}_n^\top \\ &= \mathbf{1}_n \overline{\boldsymbol{u}_i}^\top R P_n \overline{\boldsymbol{u}_i} \mathbf{1}_n^\top \\ &= \mathbf{1}_n \overline{\boldsymbol{u}_i}^\top R P_n. \end{aligned} \tag{48}$$

Since the diagonal elements of R are all zero, thus we have

$$\begin{aligned} \mathrm{diag}(X_i X_i^\top) &= \mathrm{diag}(\frac{1}{2} R \overline{\boldsymbol{u}_i} \mathbf{1}_n^\top) + \mathrm{diag}(\frac{1}{2} \mathbf{1}_n \overline{\boldsymbol{u}_i}^\top R) \\ &- \mathrm{diag}(\frac{1}{2} \mathbf{1}_n \overline{\boldsymbol{u}_i}^\top R \overline{\boldsymbol{u}_i} \mathbf{1}_n^\top) \\ &= R\overline{\boldsymbol{u}_i} - \frac{1}{2} \mathbf{1}_n \overline{\boldsymbol{u}_i}^\top R \overline{\boldsymbol{u}_i}. \end{aligned} \tag{49}$$

The element of $\mathrm{diag}(X_i X_i^\top)$ is d_{ik}^2 with $p = 0$ and corresponds to the d_{ik}^2 in (43) where (44) is used instead of (26).

The squared Euclidean distance between $\boldsymbol{x}_0^{NEW}$ and $\boldsymbol{v}_i$ is obtained by (41).

IV. Numerical Experiments

We compare the basic performance of the proposed classifier with k-NN by preparing artificial dissimilarity data computed from object data. This approach enables us to compare the performance with well-known classifiers for object data, although the dimensionality of the benchmark data is not so high. We used 8 sets of object data, i.e., Iris, Wisconsin breast cancer, Ionosphere, Glass, Liver disorder, Pima Indian diabetes, Sonar and Wine as shown in Table I. These dat sets were used for comparisons among several prototype-based methods in [30]. All these data sets consist of object data and are available from the UCI ML repository (http://www.ics.uci.edu/˜ mlearn/) [29]. Incomplete samples in the breast cancer data set were eliminated from the training and test sets. All categorical attributes were encoded with multivalue (integer) variables. All attribute values were

TABLE I

DATA SETS USED IN THE EXPERIMENTS

	features	objects	classes
Iris	4	150	3
Breast	9	683	2
Ionosphere	33	351	2
Glass	9	214	6
Liver	6	345	2
Pima	8	768	2
Sonar	60	208	2
Wine	13	178	3

normalized to zero mean and unit variance. Iris and Wine are the sets with three classes. Iris-Vc is a binary classification problem for discriminating between the iris versicolor and the other two iris subspecies. Though Iris is the set with three classes, it is known that Iris setosa is clearly separated from the other two subspecies and Iris versicolor is between Iris setosa and Iris verginica in the feature space [4]. If the problem is defined as binary one we can easily find that two clusters are necessary for the setosa-verginica class. To show this case, we use the binary problem of Iris data. Wine-3 is also a binary problem.

Table II shows the results by RFCMC and k-NN. The nearest neighbor classifier easily overfits to the training data. Accordingly, instead of 1-nearest neighbor, generally k nearest neighboring data objects are considered in k-NN classifier. Then, the class label of unseen objects is established by majority vote. For the parameter (k) of k-NN, we tested all integer values from 1 to 50 by 10-CV with a default partition and the optimum values are shown in Table II. The k-NN classifier itself is a relational classifier, since the relational data consisted of Euclidian distances between objects are used.

The relational data are computed from object data. The boldface letters in Table II indicate that the classification error rate is smaller than or equal to that of the other approache according to the two sample t test with significance level $p = 0.05$ in the comparison between RFCMC and k-NN. RFCMC overwhelmingly surpassed k-NN. The classification performances of other four well-known classifiers are reported in [10]. Although they are not relational classifiers and evaluated by 10-CV with a default partition, generally speaking, RFCMC results are better than those of the classifiers. It should be noted that the performance of FCMC on the object data is theoretically equivalent to those of RFCMC on the relational data which are obtained from the object data.

Table III shows the parameter values used for RFCMC. The parameter optimization with golden section search method for FCMC in [10] may be a good way, but is not used in our numerical experiment. The parameters of RFCMC are optimized by trial and error using 10-CV with a default partition and evaluated by 10 separate runs of 10-CV with random partitions. The results of FCMC on object data and optimized by trial and error are reported in [8].

On the four data sets, RFCMC with plural clusters for each class performs well. For the several data sets, the performance is not so sensitive to the values of λ and η, so the values are fixed to 1 for the data sets. Depending on the data sets, γ and p assume various values. $p = 0$ represents that Euclidean distance, instead of Mahalanobis distance, is used.

TABLE II

CLASSIFICATION ERROR RATES (%) $\pm$ STANDARD DEVIATION ON RELATIONAL DATA COMPUTED FROM BENCHMARK OBJECT DATA. THE RESULTS OF 10 SEPARATE RUNS OF 10-CV WITH RANDOM PARTITIONS.

	RFCMC	k-NN
Iris	**2.20 $\pm$ 0.31**	5.73 $\pm$ 0.61 k=21
Iris-Vc	**2.33 $\pm$ 0.45**	5.73 $\pm$ 0.61 k=21
Breast	**2.93 $\pm$ 0.10**	3.03 $\pm$ 0.10 k=11
Ionosphere	**4.20 $\pm$ 0.31**	13.63 $\pm$ 0.79 k=1
Glass	31.48 $\pm$ 1.14	**29.00 $\pm$ 0.96** k=2
Liver	**31.26 $\pm$ 0.87**	34.97 $\pm$ 1.36 k=23
Pima	**23.87 $\pm$ 0.39**	24.50 $\pm$ 0.72 k=21
Sonar	**11.65 $\pm$ 0.98**	15.30 $\pm$ 1.36 k=3
Wine	**0.59 $\pm$ 0.37**	2.35$\pm$ 0.64 k=25
Wine-3	**0.00 $\pm$ 0.00**	0.94$\pm$ 0.39 k=15

TABLE III

PARAMETER VALUES USED FOR THE RELATIONAL FCM CLASSIFIER

	c	λ	γ	η	p	
Iris	1	1.2	8	1	4	
Iris-Vc	**2**	0.6	3	1	4	
Breast	1	1	20	1	1	
Ionosphere	1	0.55	25.4	50	4	
Glass	1	1	22	13	4	α=1
Liver	1	1	7	1	6	
Pima	**2**	0.5	15	1	5	
Sonar	20	0.2	–	1	0	
Wine	1	1	30	1	13	α=1
Wine-3	**2**	1	30	1	13	α=1

For Sonar data, to use 20 clusters and Euclidean distance was the best choice among several trials by 10-CV. For Breast cancer data, Euclidean distance, (i.e., p=0, c=3), was used and the classification decision was made by the maximum of cluster membership values in [8]. The result for this case by RFCMC was 2.79 ± 0.07, which is slightly better than the result in Table II in which Mahalanobis distances are used ($p = 1$) and the decision was made by the sum of memberships.

In the table, $\alpha = 1$ represents that class mixing proportions are not take into account in classifying new data.

For the data whose classification error rate is lager than 20%, there is not significant difference between RFCMC and k-NN. Usually the classifier with such a high classification error rate is not enough for practical use, and RFCMC works for the data with small classification error rates such as Iris, Ionosphere and Wine as shown in Table II.

V. CONCLUSION

We have proposed a classifier based on relational data and FCM with Mahalanobis distances. Experimental comparisons by using benchmark object data revealed that the performance of RFCMC for relational data is equivalent to that of FCMC for object data.

RFCMC uses the matrices in terms of inner products instead of covariance matrices. When the data dimension is larger than the number of objects, the matrix of inner products is more convenient than the covariance matrix.

When object data are available, even if the dimensionality is high, the classification for new unseen data can be done in a similar manner to FCMC, so is simple and less time-consuming. The derivation of the algorithm seems to be complicated, though the derived algorithm and its computational intensity are similar to those of the classifier based on Gaussian mixture models or normal mixture. The only difference from GMC is that the proposed classifier needs parameter optimization when designing the classifier.

Application and evaluation on large dimensional data such as those used in recommender or collaborative filtering and bioinformatics are our imminent tasks. The β-spread transform in non-Euclidean relational fuzzy clustering [31] to convert a non-Euclidean matrix into an Euclidean matrix may also be a good way when the observation is non-Euclidean. These challenges are left for our future study.

REFERENCES

[1] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, 1981.

[2] F. Höppner, F. Klawonn, R. Kruse, T. Runkler : *Fuzzy Cluster Analysis, (Methods for Classification, Data Analysis and Image Recognition)*, John Wiley & Sons, 1999.

[3] S. Miyamoto, D. Suizu, O. Takata, "Methods of fuzzy c-means and possibilistic clustering using a quadratic term, *Scientiae Mathematicae Japonicae* vol.60, no.2, pp.217-233, 2004.

[4] R. O. Duda and P. E. Hart: *Pattern Classification and Scene Analysis*, Wiley, New York, 1973.

[5] A. P. Dempster, N. M. Laird and D. B. Rubin: Maximum likelihood from incomplete data via the EM algorithm, *Journal of Royal Statistical Society*, vol.B-39, pp.1-38, 1977.

[6] G. J. McLachlan and T. Krishnan: *The EM algorithm and extensions*, John Wiley & Sons, 1997.

[7] H. Ichihashi and K. Honda, "Fuzzy c-means classifier for incomplete data sets with outliers and missing values," *Proc. of the International Conference on Computational Intelligence for Modelling, Control and Automation*, Vienna, November, pp.457-564, 2005.

[8] H. Ichihashi, K. Honda, T. Hattori, "Regularized discriminant in the setting of fuzzy c-means classifier," *Proc. of the IEEE World Congress on Computational Intelligence*, Vancouver, Canada, pp.4266-4271, 2006.

[9] H. Ichihashi, K. Honda, F. Matsuura, "ROC analysis of FCM classifier with Cauchy weight," *Proc. of the 3rd International Conference on Soft Computing and Intelligent Systems*, Tokyo, Japan, pp.1912-1917, 2006.

[10] H. Ichihashi, K. Honda, A. Notsu and T. Yagi, "Fuzzy c-Means classifier with deterministic initialization and missing value imputation," *Proc. of the 2007 IEEE Symposium on Foundations of Computational Intelligence*, Hawaii, April, 2007 (to appear).

[11] E. H. Ruspini, "Numerical methods for fuzzy clustering," *Information Sciences*, vol.2, pp. 319-350, 1970.

[12] M. Roubens, "Pattern classification problems and fuzzy sets," *Fuzzy Sets and Systems*, vol. 1, pp. 239-253, 1978.

[13] M. P. Windham, "Numerical classification of proximity data with assignment measures," *Journal of Classification*, vol.2, pp.157-172, 1985.

[14] R. J. Hathaway, J. W. Davenport and J. C. Bezdek, "Relational duals of the c-means clustering algorithms," *Pattern Recognition*, vol.22, No.2, pp.205-212, 1985.

[15] J. Kaufman and P. J. Rousseeuw, "Clustering by means of medoids," *Statistical Data Analysis Based on the L_1 Norm*, Y. Dodge, Ed. Amsterdam, The Netherlands: North Holland/Elsevier, pp. 405-416, 1987.

[16] J. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, Brussels, Belgium: Wiley, 1990.

[17] R. N. Davé and S. Sen, "Robust fuzzy clustering of relational data," *IEEE Transactions on Fuzzy Systems*, vol.10, No.6, pp.713-727, Dec 2002.

[18] T. A. Runkler and J. C. Bezdek, "Web mining with relational clustering," *Int. J. Approx. Reasoning*, vol. 32, no. 2-3, pp.217-236, 2003.

[19] R. Krishnapuram, A. Joshi, O. Nasraoui and L. Yi, "Low-complexity fuzzy relational clustering algorithms for web mining," *IEEE Trans. Fuzzy Systems*, vol.9, no.4, pp 596-607, 2001.

[20] U. Shardanand and P. Maes, "Social information filtering, Algorithms for automating 'word of mouth'," *Proc. of ACM CHI'95 Conference on Human Factors in Computing Systems*, Denver, CO, 1995.

[21] K.Honda and H. Ichihashi, "Linear fuzzy clustering techniques with missing values and their application to local principal component analysis, *IEEE Transactions on Fuzzy Systems*, vol.12, no.2, pp.183-193, April, 2004.

[22] M.E. Tipping and C.M. Bishop, "Mixtures of probabilistic principal component analysers, "*Neural Computation*, vol.11, pp.443-482, 1999.

[23] F. Sun, S.Omachi, and H. Aso, "Precise selection of candidates for hand written character recognition," *IEICE Trans. Information and Systems*, vol.E79-D, no.3, pp.510-515, 1996.

[24] D. E. Gustafson and W. C. Kessel, "Fuzzy clustering with a fuzzy covariance matrix, "*Proc. IEEE CDC*, vol.2, pp.761-766, 1979.

[25] T. A. Runkler and J. C. Bezdek, "Alternating cluster estimation: a new tool for clustering and function approximation," *IEEE Trans. Fuzzy Syst.*, vol. 7, no. 4, pp. 377-393, 1999.

[26] P. W. Holland and R. E. Welsch, "Robust regression using iteratively reweighted least-squares, "*Communications in Statistics*, vol. A6, no. 9, pp. 813-827, 1977.

[27] P. J. Huber. *Robust Statistics*. New York:Wiley, first edition, 1981.

[28] K. Honda and H. Ichihashi, "Regularized linear fuzzy clustering and probabilistic PCA mixture models," *IEEE Trans. Fuzzy Syst.*, vol.13, no.4, pp.508-516, 2005.

[29] C.L. Blake and C.J. Merz. UCI repository of machine learning databases, 1998.

[30] C.J. Veenman and M.J.T. Reinders, "The nearest sub-class classifier: a compromise between the nearest mean and nearest neighbor classifier," *IEEE Transactions on PAMI*, vol.27, no.9, pp.1417-1429, 2005.

[31] R. J. Hathaway and J. C. Bezdek, !HNERF c-Means: Non-Euclidean relational fuzzy clustering, !I*Pattern Recognition*, vol. 27, no. 3, pp. 429-437, 1994.