Available online at www.sciencedirect.com
ScienceDirect
Pattern Recognition Letters 28 (2007) 2250–2263
Pattern Recognition Letters
www.elsevier.com/locate/patrec

Robust fuzzy relational classifier incorporating the soft class labels

Weiling Cai, Songcan Chen *, Daoqiang Zhang

Department of Computer Science and Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, PR China

Received 29 July 2006; received in revised form 3 June 2007
Available online 6 August 2007

Communicated by W. Pedrycz

Abstract

Fuzzy relational classifier (FRC) is a recently proposed two-step nonlinear classifier. At first, the unsupervised fuzzy *c*-means (FCM) clustering is performed to explore the underlying groups of the given dataset. Then, a fuzzy relation matrix indicating the relationship between the formed groups and the given classes is constructed for subsequent classification. It has been shown that FRC has two advantages: interpretable classification results and avoidance of overtraining. However, FRC not only lacks the robustness which is very important for a classifier, but also fails on the dataset with non-spherical distributions. Moreover, the classification mechanism of FRC is sensitive to the improper class labels of the training samples, thus leading to considerable decline in classification performance. The purpose of this paper is to develop a Robust FRC (RFRC) algorithm aiming at overcoming or mitigating all of the above disadvantages of FRC and maintaining its original advantages. In the proposed RFRC algorithm, we employ our previously proposed robust kernelized FCM (KFCM) to replace FCM to enhance its robustness against outliers and its suitability for the non-spherical data structures. In addition, we incorporate the soft class labels into the classification mechanism to improve its performance, especially for the datasets containing the improper class labels. The experimental results on 2 artificial and 11 real-life benchmark datasets demonstrate that RFRC algorithm can consistently outperform FRC in classification performance.

Keywords: Fuzzy *c*-means clustering (FCM); Fuzzy relations; Fuzzy relational classifier; Kernelized FCM (KFCM); Soft class label; Pattern classification

1. Introduction

The classification is widely used in various engineering problems such as handwriting recognition, medical diagnosis and document retrieval. Its task is to assign a new sample to a class from a given set of classes based on the features of this sample. Generally speaking, the conventional learning classifiers are designed in a supervised manner, that is, the class labels of the training samples are *directly* employed to guide the classification, such as neural networks (Haykin, 1999) and support vector machines (SVM) (Cristianini and Taylor, 2000). Due to this characteristic, this kind of classifiers just emphasizes the determination of the decision functions, but rarely cares about the revelation of the data structure. Here the data structure means the relative locations of the samples in the high dimensional space (Krzanowski, 1988), which is very helpful to the transparence and the interpretability of classification result. Jain et al. (1999) pointed out that the unsupervised clustering analysis which aims to allocate a collection of unlabeled samples into meaningful clusters is appropriate for the exploration of the inherent data structure. However, such unsupervised clustering cannot be directly applied to classification because (1) class labels of the samples are not used in clustering (Kaufman and Rousseeuw, 1990); (2) although the clustering method can be used for classification, each generated cluster may not be assigned a single class label since the samples from different classes may fall into a common data group (cluster). Therefore, unsupervised clustering and supervised classification approaches are more likely to complement

* Corresponding author. Tel.: +86 25 84896481x12106; fax: +86 25 84498069.
E-mail addresses: caiwl@nuaa.edu.cn (W. Cai), s.chen@nuaa.edu.cn (S. Chen).

0167-8655/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2007.07.013

each other by integrating them together (Pedrycz and Vukovich, 2004; Musavi et al., 1992; Setnes and Babuška, 1999; Ramirez et al., 2003).

From the viewpoint of integration, the design of Radial Basis Function neural network (RBFNN) (Musavi et al., 1992) also follows such a line, i.e., first utilizes unsupervised clustering methods such as k-means or fuzzy c-means (FCM) (Bezdek, 1981) to construct its hidden layer and then uses the mean-squared-error (MSE) criterion between the target and actual outputs to optimize the connection weights between the hidden and output layers. In that way, good generalization performance can be achieved in practical applications. However, it emphasizes the classification more than the revelation of the data structure. As a result, in its design, the clustering method is just used as an auxiliary way rather than a method to explore the underlying structure of the data. To indeed fuse the merits of the unsupervised clustering and the supervised classification methods, an algorithm called Fuzzy Relational Classifier (FRC) (Setnes and Babuška, 1999; Ramirez et al., 2003) was recently proposed. Unlike RBFNN, FRC first performs clustering in an unsupervised manner to discover the inherent structure in data, and then uses the resulted cluster memberships of each sample to establish the fuzzy relation matrix $\boldsymbol{R}$ connecting the formed clusters and the given classes. In this way, FRC indeed integrates the strengths of clustering and classification effectively, and as a result it can explore the structure of the given dataset and then interpret the meaning of the classification results. It has been experimentally demonstrated that the FRC works better than the multi-reference minimum distance classifier (MMDC) and the neuro-fuzzy classifier (NEFCLASS) (Setnes and Babuška, 1999). In summary, FRC has several prominent characteristics as follows: (1) with the integration of the relation matrix $\boldsymbol{R}$ and the cluster memberships, the classification result has an intuitive interpretation, which makes FRC's classification prone to transparence; (2) FRC is based on unsupervised learning and thus unlikely to overfit the training samples (Setnes and Babuška, 1999).

However, one disadvantage of FRC is its adoption of fuzzy c-means (FCM) (Bezdek, 1981) to explore the inherent structure of the given dataset. FCM is proved to be non-robustness (Dave and Krishnapuram, 1997), in other words, the clustering centers (prototypes) yielded by FCM deviate from the original real centers. Besides, FCM is unable to group the datasets consisting of the non-spherical clusters, so that the interpretation of the clustering or classification results may be biased. In recent years, many researchers have tried to solve these two problems about the fuzzy clustering methods. To make the results of clustering less sensitive to noise, Wu and Yang (2002) proposed an alternative c-means clustering algorithm by incorporating robust metrics to the objective function. Assuming the clusters have elliptic shapes, the clustering methods (Klawonn and Keller, 1999; Yao et al., 1999) with Mahalanobis or Minkowski metric were designed, but these methods are just fit for the datasets composed of the groups with the same kind of structure. As far as we know, there are few clustering methods which can simultaneously overcome the above two disadvantages of FCM. In our previously proposed kernelized fuzzy c-means algorithm (KFCM) (Chen and Zhang, 2004; Zhang and Chen, 2003, 2004), the robust distance metric induced by the kernel function is utilized to replace the non-robust Euclidean metric in the objective function of FCM. As a result, such KFCM is not only proved to be robust according to the Huber's robust statistics (Huber, 1981), but also able to group the non-spherical clusters in the given dataset. In addition, KFCM retains the same simplicity in computation as FCM. On the other hand, the other disadvantage of FRC is that the employment of hard class labels heavily demotes the classifier's performance, especially for the dataset containing the improper hard labels. Recently, the fuzzy set theory (Zadeh, 1965) has applied to allow the samples to belong to different classes with varying memberships, aiming to compensate for the imprecision of the hard class labels. Therefore, the soft class labels designed in this manner can suppress the influence of the improper hard class labels (Pizzi and Pedrycz, 2000), and more importantly provide more valuable information (Sohn and Daqli, 2001).

In this paper, we develop a Robust FRC algorithm called RFRC which incorporates the previously proposed KFCM and the soft class labels to mitigate or eliminate the disadvantages of FRC while maintaining its advantages. To be widely used, a classifier should be robust, meaning that the designed classifier (its parameter estimation) can resist the effects of outliers and noises. As presented before, FRC lacks the robustness due to the non-robust FCM and error-sensitive hard class labels. Due to the two-step design of FRC, we realize FRC's robustness by the following two steps. In the first training step, we utilize the KFCM with robust objective function to replace FCM to achieve robust clustering. Then, in the second training step of RFRC, we apply the fuzzy set theory to compute the soft class labels which can more precisely describe the class information than the hard ones. Consequently, RFRC incorporating both KFCM and the soft class labels can make the constructed relation matrix $\boldsymbol{R}$ more really reflect the fuzzy relation between the classes and clusters for the subsequent classification, thus significantly decreasing the reject rate (i.e., the reject decisions yielded by FRC) and boosting the accuracy of FRC. It is worth pointing out that Setnes and Babuška (1999) *just* utilized the relation matrix $\boldsymbol{R}$ to classify but not used it to further analyze the structure of given data. In this paper, we can mine more insightful information from this $\boldsymbol{R}$ to help understand the classification results. For example, we can know whether the formed clusters are prone to pure or not, whether the class of the dataset is composed of single-group or multi-groups, whether the formed clusters for each class is reliable, and so on, which makes the RFRC interpretable to some extent. Therefore, RFRC indeed represents a

transparent alternative to conventional black-box techniques like artificial neural networks.

The rest of this paper is organized as follows: In Section 2, Fuzzy Relational Classifier is reviewed. Section 3 describes our Robust Fuzzy Relational Classifier incorporating KFCM and the soft class labels. The experimental results on 2 artificial dataset and the 11 real-life benchmark datasets are presented in Section 4. Finally, the conclusions are given in Section 5.

2. Fuzzy relational classifier

2.1. Training of the classifier

The training of the classifier involves two steps. Firstly, FCM (Bezdek, 1981) algorithm is applied on the training samples to reveal the natural structures in the given dataset. Secondly, a fuzzy relation matrix $\boldsymbol{R}$ is established from the obtained fuzzy partition and the given class labels to uncover the relationship between the clusters and classes. These two steps are illustrated in Fig. 1.

In the first step, FCM algorithm is chosen and its objective function can be formulated as follows:

$$J_{\mathrm{FCM}}(U,V)=\sum_{j=1}^{c}\sum_{i=1}^{N}u_{ji}^{m}\|\boldsymbol{x}_i-\boldsymbol{v}_j\|^2 \tag{1}$$

where N is the total number of the training samples and c is the number of clusters; $X=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_N\}$ and $V=\{\boldsymbol{v}_1,\boldsymbol{v}_2,\ldots,\boldsymbol{v}_c\}$ are the training samples and cluster centers, respectively; and the fuzzy matrix $U=(u_{ji})_{c\times N}$ makes up of the fuzzy memberships of the each training sample $\boldsymbol{x}_i$ to each cluster $\boldsymbol{v}_j$. By definition, each sample $\boldsymbol{x}_i$ satisfies the constraint $\sum_{j=1}^{c}u_{ji}=1$. The parameter m $(1\leqslant m<\infty)$ is a weighting exponent on each fuzzy membership that determines the amount of fuzziness of the resulting classification. In the following experiment, the value of m is set to 2. The outputs of such algorithm allow the training sample to belong to multiple clusters with varying degrees of membership, in this way FCM can get much more information from dataset than the hard clustering methods. Since FCM clustering method performs in an unsupervised manner (class labels are not used), the number of prototype in FCM is independent of the number of classes. Consequently, the resulted fuzzy partition of the training dataset can more closely represent the underlying structures.

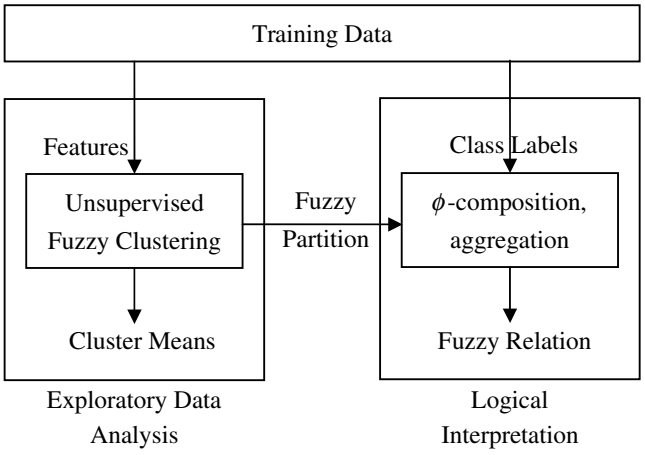

Fig. 1. Training of the fuzzy relational classifier.

In the second step, a fuzzy relation matrix $\boldsymbol{R}$ is established from the obtained cluster membership matrix U and the hard class labels. Such original class label is denoted by one-of-c encoding (Sung-Bae and Kim, 1995)

$$\boldsymbol{y}_i=[y_{1i},y_{2i},\ldots,y_{li},\ldots,y_{Li}]^{\mathrm{T}} \tag{2}$$

where y_{li} is the class membership of the ith sample to the lth class and L is the number of classes. Here $y_{li}\in\{1,0\}$, meaning that the ith sample belongs completely to the lth class or not. For each training sample $\boldsymbol{x}_i$, the partial relation R_i is described by a $c\times L$ matrix

$$\boldsymbol{R}_i=\begin{bmatrix}(r_{11})_i & (r_{12})_i & \cdots & (r_{1L})_i\\ (r_{21})_i & (r_{22})_i & \cdots & (r_{2L})_i\\ \cdots & \cdots & \cdots & \cdots\\ (r_{c1})_i & (r_{c2})_i & \cdots & (r_{cL})_i\end{bmatrix} \tag{3}$$

where $(r_{jl})_i$ is computed by the ϕ-composition operator (Pedrycz, 1994)

$$(r_{jl})_i=\min(1,1-u_{ji}+y_{li}),\quad l=1,2,\ldots,L,\quad j=1,2,\ldots,c \tag{4}$$

All these relations $\boldsymbol{R}_i$s can be aggregated into the $\boldsymbol{R}$ in terms of a fuzzy conjunction operator as follows:

$$\boldsymbol{R}=\bigcap_{i=1}^{N}\boldsymbol{R}_i \tag{5}$$

implemented element-wise by the minimum function

$$r_{jl}=\min_{i=1,2,\ldots,N}[(r_{jl})_i] \tag{6}$$

where r_{jl} represents the fuzzy relationship between the jth cluster and the lth class.

2.2. Classification of test samples

The classification of a test sample $\boldsymbol{x}$ involves three steps. Firstly, the cluster membership degree $\hat{\boldsymbol{u}}_x=[\hat{u}_{1x},\hat{u}_{2x},\ldots,\hat{u}_{jx},\ldots,\hat{u}_{cx}]$ is computed by measuring the distances between the $\boldsymbol{x}$ and the cluster centers

$$\hat{u}_{j\boldsymbol{x}}=\frac{\|\boldsymbol{x}-\boldsymbol{v}_j\|^{-2/(m-1)}}{\sum_{j=1}^{c}\|\boldsymbol{x}-\boldsymbol{v}_j\|^{-2/(m-1)}} \tag{7}$$

where $\hat{u}_{jx}$ represents the cluster membership of the $\boldsymbol{x}$ to the jth cluster. Secondly, using the obtained $\hat{\boldsymbol{u}}_x$ and $\boldsymbol{R}$, the class membership $\hat{\boldsymbol{y}}_x=[\hat{y}_{1x},\hat{y}_{2x},\ldots,\hat{y}_{lx},\ldots,\hat{y}_{Lx}]$ of the $\boldsymbol{x}$ can be computed by fuzzy relational composition

$$\hat{\boldsymbol{y}}_x=\hat{\boldsymbol{u}}_x\circ_T\boldsymbol{R} \tag{8}$$

where $\circ_T$ is the sup-t composition operator (Klir and Yuan, 1995). This approach is implemented in terms of

$$\hat{y}_{lx} = \max_{1\leqslant j\leqslant c}[\max(\hat{u}_{jx} + r_{jl} - 1, 0)], \quad l = 1, 2, \ldots, L \tag{9}$$

Finally, the class membership degree $\hat{\boldsymbol{y}}_x$ is defuzzified using the maximum operator to obtain a crisp decision for classification

$$\hat{\omega}_{\boldsymbol{x}} = \arg\max_{1\leqslant l\leqslant L} \hat{y}_{lx} \tag{10}$$

where $\hat{\omega}_{\boldsymbol{x}}$ is the final class label. Note that if the difference between the first maximal and the second maximal class membership is too small, a *reject decision* for the test sample $\boldsymbol{x}$ should be made. Such a *reject decision* implies that the training dataset contains the logical conflict information or lacks the information in a particular region.

According to the obtained $\hat{\boldsymbol{y}}_x$, we can make following analysis about the test sample $\boldsymbol{x}$. The low value of $\hat{y}_{\max} = \max_{1\leqslant l\leqslant L}(\hat{y}_{l\boldsymbol{x}})$ means that the inconsistent information more likely exists in the training dataset; on the other hand, the overall high values of $\hat{y}_{lx}$ indicate that available evidence from the training dataset is insufficient for classification. Consequently, FRC gives an intuitive interpretation for the classification result, thus making its classification prone to transparence.

3. Robust fuzzy relational classifier

In this section, we propose the Robust FRC algorithm (RFRC) incorporating the KFCM and soft class labels to overcome the previously mentioned disadvantages of FRC, and at the same time retain its advantages. Firstly, we employ our previously proposed robust KFCM to replace FCM to enhance its robustness against outliers and its adaptability for the non-spherical data groups. Secondly, we apply the fuzzy set theory to obtain the soft class labels aiming to improve the classifier's performance.

3.1. Kernelized fuzzy c-means

Recently, a number of powerful kernel-based learning machines, such as Support Vector Machines (SVM) (Cristianini and Taylor, 2000), Kernel Fisher Discriminate (KFD) (Roth and Steinhage, 2000) and Kernel Principal Component Analysis (KPCA) (Scholkopf et al., 1998) have been successfully applied to pattern recognition and data mining. A common theory behind these algorithms is the kernel trick, which aims at converting the nonlinear problem in the original low dimensional input space into a linear one in the rather high dimensional feature space (Cover, 1965).

A *kernel* is a function K that for all x, z from the original input space $\boldsymbol{X}$ satisfies

$$K(\boldsymbol{x}, \boldsymbol{z}) = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{z}) \rangle \tag{11}$$

where $\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{z}) \rangle$ denotes the inner product operation and ϕ is an implicit nonlinear map from the input space $\boldsymbol{X}$ to a rather high dimensional feature space F

$$\phi : \boldsymbol{x} \to \phi(\boldsymbol{x}) \in F \tag{12}$$

Using this mapping ϕ, the kernelized version of FCM can be described as below

$$J_{\mathrm{KFCM}}(U, V) = \sum_{j=1}^{c}\sum_{i=1}^{N} u_{ji}^m \|\phi(\boldsymbol{x}_i) - \phi(\boldsymbol{v}_j)\|^2 \tag{13}$$

Similar to FCM, each sample $\boldsymbol{x}_i$ satisfies the constraint $\sum_{j=1}^{c} u_{ji} = 1$. Through the kernel substitution, we have

$$\begin{aligned}\|\phi(\boldsymbol{x}_i) - \phi(\boldsymbol{v}_j)\|^2 &= (\phi(\boldsymbol{x}_i) - \phi(\boldsymbol{v}_j))^{\mathrm{T}}(\phi(\boldsymbol{x}_i) - \phi(\boldsymbol{v}_j)) \\ &= \phi(\boldsymbol{x}_i)^{\mathrm{T}}\phi(\boldsymbol{x}_i) - \phi(\boldsymbol{v}_j)^{\mathrm{T}}\phi(\boldsymbol{x}_i) \\ &\quad - \phi(\boldsymbol{x}_i)^{\mathrm{T}}\phi(\boldsymbol{v}_j) + \phi(\boldsymbol{v}_j)^{\mathrm{T}}\phi(\boldsymbol{v}_j) \\ &= K(\boldsymbol{x}_i, \boldsymbol{x}_i) + K(\boldsymbol{v}_j, \boldsymbol{v}_j) - 2K(\boldsymbol{x}_i, \boldsymbol{v}_j)\end{aligned} \tag{14}$$

in this way, a new class of non-Euclidean distance measures in original input space (also an Euclidean distance in the feature space) are obtained. Obviously, different kernels will induce different measures for the original space. Particularly, $K(\boldsymbol{x},\boldsymbol{y})$ in this paper is taken as the radial basis function (RBF) kernel

$$K(\boldsymbol{x}, \boldsymbol{y}) = \exp\left(\frac{-\|\boldsymbol{x} - \boldsymbol{y}\|^2}{\sigma^2}\right) \tag{15}$$

where σ is the kernel parameter and significantly affects the clustering result. In order to simplify the selection of this kernel parameter, we define the parameter σ in terms of (Abe, 2005)

$$\sigma^2 = \frac{\max_{1\leqslant i\leqslant N}\|\boldsymbol{x}_i - \bar{\boldsymbol{x}}\|^2}{\lambda} \tag{16}$$

where N is the number of training samples, the parameter λ is a scale factor and $\bar{\boldsymbol{x}} = \sum_{i=1}^{N}\boldsymbol{x}_i/N$. To get an appropriate value of σ, we indirectly determine it by seeking an appropriate scale factor λ in {0.01, 0.05, 0.1, 0.5, 1, 5, 10, 15} according to the trial-and-error approach (Abe, 2005). Though the parameter σ so obtained is unlikely to be optimal, the selection procedure can drastically be simplified.

By the definition of RBF kernel, we obtain $K(\boldsymbol{x}_i, \boldsymbol{x}_i) = 1$ and $K(\boldsymbol{v}_j, \boldsymbol{v}_j) = 1$, and hence the objective function (13) of KFCM can be simplified to

$$J_{\mathrm{KFCM}}(U, V) = 2\sum_{j=1}^{c}\sum_{i=1}^{N} u_{ji}^m(1 - K(\boldsymbol{x}_i, \boldsymbol{v}_j)) \tag{17}$$

To minimize $J_{\mathrm{KFCM}}(U, V)$, u_{ji} and $\boldsymbol{v}_j$ need to be computed according to the following iterative formulas:

$$u_{ji} = \frac{(1 - K(\boldsymbol{x}_i, \boldsymbol{v}_j))^{-1/(m-1)}}{\sum_{j=1}^{c}(1 - K(\boldsymbol{x}_i, \boldsymbol{v}_j))^{-1/(m-1)}} \tag{18}$$

$$\boldsymbol{v}_j = \frac{\sum_{i=1}^{N} u_{ji}^m K(\boldsymbol{x}_i, \boldsymbol{v}_j)\boldsymbol{x}_i}{\sum_{k=1}^{N} u_{ji}^m K(\boldsymbol{x}_i, \boldsymbol{v}_j)} \tag{19}$$

Note that the obtained centers $\{\boldsymbol{v}_j\}$ still lie in the original space rather than in the transformed higher dimensional feature space, so that the computational simplicity is still retained.

According to Huber's robust statistics (Huber, 1981; Wu and Yang, 2002), a robust procedure should possess all of the following properties: (1) it should have a reasonably good accuracy at the assumed model; (2) small deviations from the model assumptions should impair the performance only by a small amount; (3) larger deviations from the model assumptions should not cause a catastrophe. From this point of view, FCM is not a robust estimator (Dave and Krishnapuram, 1997). In contrast, the distance metric $1-K(x_i,v_j)$ induced by RBF kernel is proved to be a robust measure (Hathaway and Bezdek, 2000; Jajuga, 1991), hence KFCM based on RBF kernel is a robust estimator according to M-estimator (Huber, 1981; Leski, 2003). It is still worth pointing out that according to Huber's robust statistics, KFCM based on polynomial and sigmoid kernels are not robust due to the non-robust property of the distance metrics induced by them. Consequently, in this paper, we just choose RBF kernel to guarantee the so-needed robustness of the clustering result and the subsequent classification.

Furthermore, we compare the regions formed by FCM and KFCM induced by RBF kernel in the input space. For FCM, the ith region determined by the center $\boldsymbol{v}_i$ can be described as

$$\begin{aligned}\mathrm{Region}_i^{\mathrm{FCM}} &= \left\{\boldsymbol{x}|\|\boldsymbol{x}-\boldsymbol{v}_i\|^2 < \|\boldsymbol{x}-\boldsymbol{v}_j\|^2,\ i\neq j\right\}\\ &= \left\{\boldsymbol{x}|(\boldsymbol{v}_i-\boldsymbol{v}_j)^{\mathrm{T}}\left(\boldsymbol{x}-\frac{\boldsymbol{v}_i-\boldsymbol{v}_j}{2}\right)>0,\ i\neq j\right\}\end{aligned} \tag{20}$$

All the regions decided by all the centers form a Voronoi tessellation (Bezdek, 1981) in the input space and the boundary between the regions is a hyper plane which can only be induced by hyper-spherical clusters. In our KFCM, the distance function $1-K(\boldsymbol{x},\boldsymbol{v}_i)$, induced by RBF kernel, is in fact a non-Euclidean distance in the original input space. Similar to FCM, the center $\boldsymbol{v}_i$ obtained by KFCM also forms a region with respect to one of the clusters. The $\mathrm{Region}_i^{\mathrm{KFCM}}$ can be formulated as follows:

$$\begin{aligned}\mathrm{Region}_i^{\mathrm{KFCM}} = \Bigg\{\boldsymbol{x}|&\sum_{n=1}^{\infty}\frac{(-1)^n}{n!}\frac{\|\boldsymbol{x}-\boldsymbol{v}_i\|^{2n}}{\sigma^{2n}}\\ &<\sum_{n=1}^{\infty}\frac{(-1)^n}{n!}\frac{\|\boldsymbol{x}-\boldsymbol{v}_i\|^{2n}}{\sigma^{2n}},\ i\neq j\Bigg\}.\end{aligned} \tag{21}$$

Here we have used

$$\begin{aligned}1-K(\boldsymbol{x},\boldsymbol{v}_i) &= 1-\exp\left(\frac{-\|\boldsymbol{x}-\boldsymbol{v}_i\|^2}{\sigma^2}\right)\\ &= 1-\sum_{n=0}^{\infty}\frac{(-\|\boldsymbol{x}-\boldsymbol{v}_i\|^2/\sigma^2)^n}{n!}\\ &= \sum_{n=1}^{\infty}\frac{(-1)^n}{n!}\frac{\|\boldsymbol{x}-\boldsymbol{v}_i\|^{2n}}{\sigma^{2n}}\end{aligned} \tag{22}$$

Obviously, all the $\mathrm{Region}_i^{\mathrm{KFCM}}$ $(i=1,2,\ldots,c)$ form no longer a Voronoi tessellation in the original space but a more complex partition that is difficult to be described explicitly. From the above analysis, it is obtained that the kernel tricks can make the algorithm more likely adapt to non-spherical shape of clusters in data, which also accords with the conclusion obtained by Chen and Zhang (2004), Girolami (2002) and Kim et al. (2004).

Finally, the process of KFCM algorithm is summarized in Fig. 2 where the number c of the clusters is determined by a cluster validity index or the slightly awkward trial-and-error approach. In conclusion, KFCM can effectively remedy the shortcomings of FCM, and still retain its simplicity in computation.

3.2. Soft class label

For a training sample $\boldsymbol{x}_i$, its hard class label just gives the yes-or-no decisions for the given classes. Naturally, it is unable to differentiate the membership degrees to the different classes, and therefore possibly misses some valuable class information. Even worse, the improper class label

KFCM Algorithm

Step1. Randomly select c training samples as the initial cluster centers and set $\varepsilon>0$ to a very small value.

Step2. Update the partition matrix using (18).

Step3. Update the prototypes using (19)

Repeat Steps2–3 until the following termination criterion is satisfied:

$$|\mathbf{V}_{new}-\mathbf{V}_{old}|<\varepsilon$$

Fig. 2. KFCM algorithm.

caused by various reasons most likely demotes the classifier's performance.

To mitigate the above weaknesses of the hard class label, we utilize the soft one to represent the class information in RFRC. The soft label of the $\boldsymbol{x}_i$ can be written as $\boldsymbol{y}_i = [y_{1i}, y_{2i}, \ldots, y_{li}, \ldots, y_{Li}]^{\mathrm{T}}$, where y_{li} takes its value from $[0,1]$ rather than $\{0,1\}$ in the hard label. Cover and Hart (1967) pointed out that half of the class information for one sample is hidden in its neighbors, and hence it is necessary to incorporate the local class information into the design of the soft class label. In practice, one form of the soft class label motivated by the fuzzy k-nearest neighbor method (FkNN) (Keller et al., 1985) is computed as follows:

$$y_{li} = \begin{cases} 0.51 + 0.49(n_{li}/k) & \text{if } l = \text{the same as the label of the } k\text{th pattern} \\ 0.49(n_{li}/k) & \text{if } l \neq \text{the same as the label of the } k\text{th pattern} \end{cases} \tag{23}$$

where y_{li} represents the class membership of the $\boldsymbol{x}_i$ to the class l, k is the number of the $\boldsymbol{x}_i$'s neighbors and n_{li} stands for the number of neighbors of the $\boldsymbol{x}_i$ that belong to the lth class. The constant value 0.51 makes the "dominant class" membership of training sample not to be affected, that is to say, the sample is not moved to a different class. It is worth noting that the soft class label of each sample satisfies the constraint $\sum_{l=1}^{L} y_{li} = 1$. By examining the formula (23), we can give the following intuitive interpretation that if very few neighbors of $\boldsymbol{x}_i$ belong to the same class, the membership degree is kept close to 0.51; on the other hand, if $n_{li} = k$, meaning that all neighbors of $\boldsymbol{x}_i$ are in the same class, then y_{li} returns 1.0.

From the viewpoint of the classifier FRC, the predetermined hard class label has no chance to correct or suppress the improper class label, thus possibly demoting the FRC's performance (Ramirez et al., 2003). In contrast, it is the incorporation of the soft class labels into the RFRC's classification mechanism that enhances the error tolerance of this classifier. In addition, the hard label in FRC fails to fuzzify or refine the class memberships, while the more precise soft class label possibly makes RFRC obtain more real relationship between clusters and classes, as a result, it can effectively boost the FRC's performance.

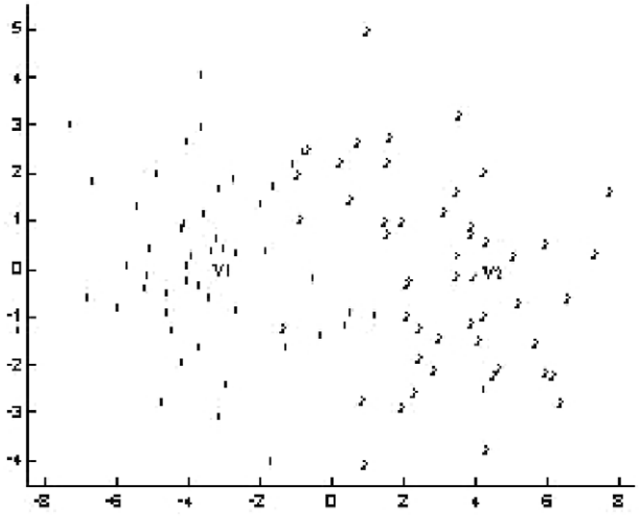

Fig. 3. Synthetic dataset with two Gaussian distributions corrupted by noise.

Here we design a synthetic dataset shown in Fig. 3 to examine the superiority of the soft class label over the hard one. This dataset is randomly generated in terms of two Gaussian distributions with the means $[3.5, 0]$ and $[-3.5, 0]$ respectively and the common variance diag$[2.5, 2.5]$ (diag denotes diagonal matrix). Meanwhile, this dataset is corrupted by the Gaussian noise with mean $[0, 0]$ and the variance diag$[2, 2]$. We use half of this dataset for training and the other half for testing. After the training phase, the two cluster centers are located at $[-3.69, 0.28]$, $[3.35, -0.51]$ in FRC and $[-3.68, 0.27]$, $[3.34, -0.52]$ in RFRC, and the relation matrices $\boldsymbol{R}$s in FRC and RFRC are $\boldsymbol{R}_{\mathrm{FRC}} = \begin{bmatrix} 0.23 & 0.00 \\ 0.00 & 0.06 \end{bmatrix}$ and $\boldsymbol{R}_{\mathrm{RFRC}} = \begin{bmatrix} 0.54 & 0.00 \\ 0.00 & 0.43 \end{bmatrix}$, respectively. Note that the cluster centers in FRC are close to those in RFRC, but their relation matrices are so different from each other. This means that the difference between the performances of FRC and RFRC can attribute to the employment of the different type of class labels. In the test phase, FRC just achieves the classification accuracy of 49.0% and reject rate of 51.0% respectively, while RFRC 87.0% and 0.0% respectively. From this example, we can observe that the soft class label indeed works better than the hard one.

3.3. More insight from relation matrix $\boldsymbol{R}$

In (Setnes and Babuška, 1999), the established $\boldsymbol{R}$ is *just* utilized to classify but not used to further mine more information hidden in it. Intuitively, we can discover more information from this $\boldsymbol{R}$ to help understand further both the structure of given data and the relationship between the structure and their classes. In this subsection, we attempt to compensate their shortcoming by analyzing the distribution characteristics of the elements of the $\boldsymbol{R}$. For such a purpose, we need to introduce the following definition of the so-called *row dominant element* of a matrix.

Definition of the *row dominant element*: For any $c \times L$ matrix A, the *row dominant element* of the ith row is defined as follows: first sort $a_{i1}, a_{i2}, \ldots, a_{ij}, a_{i(j+1)}, \ldots, a_{iL}$ in a non-increasing order to $b_1, b_2, \ldots, b_j, b_{(j+1)}, \ldots, b_L$, i.e., $b_1 \geqslant b_2 \cdots b_j \geqslant b_{(j+1)} \geqslant \cdots \geqslant b_L$; secondly find the smallest j $(1 \leqslant j \leqslant L-1)$ that satisfies $b_j - b_{(j+1)} \geqslant \tau > 0$. If such j exists, then $b_1, b_2, \ldots, b_j$ are termed as *row dominant elements*; otherwise, this row has no dominant elements. In the following experiments, τ is empirically set to 0.4. In what follows, we will apply those *row dominant elements* to mine the structural knowledge of dataset hidden in the

$\boldsymbol{R}$. For a given $c \times L$ relation matrix (where c is the number of the clusters, also the number of the rows, and L that of the classes, also the number of the columns)

$$\boldsymbol{R} = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1L} \\ r_{21} & r_{22} & \cdots & r_{2L} \\ \cdots & \cdots & \cdots & \cdots \\ r_{c1} & r_{c2} & \cdots & r_{cL} \end{bmatrix}$$

its ith *row* elements denote the memberships of the ith cluster to all the classes, while its jth *column* elements denote the belongings of the jth class to all the clusters. For a row corresponding to a cluster, if the multiple dominant elements exist in the row, it indicates that the cluster consists of the samples mainly from those classes corresponding to the row dominant elements; in particular only one dominant element appears in one row, the corresponding cluster consists of the samples mainly from only one class, meaning that the cluster is relatively pure; if there is no dominant element in one row, indicating that the corresponding cluster consists of the samples from multiple classes. For the columns corresponding to the given classes, we still utilize these *row dominant elements* to analyze its properties. If multiple *row dominant elements* appear in one column, we can infer that the corresponding class contains multi-clusters which correspond to the row dominant elements; while the class associating with a single row dominant element only consists of single cluster; further if no row dominant element exists, the samples in the class are scattered or distributed to multiple clusters. It is worth pointing out that the above analysis depends on both the clustering methods and composite operators used. Consequently, the relation matrix $\boldsymbol{R}$ indeed plays an important role not only in classification but also in discovery of the structural knowledge in given dataset.

Now let us give an example below, given $\boldsymbol{R}_{\mathrm{RFRC}} = \begin{bmatrix} \boxed{0.54} & 0.00 \\ 0.00 & \boxed{0.43} \end{bmatrix}$ in the above subsection, in which the difference between the two row elements 0.54 (0.43) and 0.00 (0.00) is larger than 0.4, and thus 0.54 (0.43) marked by the squares is a row dominant element. This indicates that all the samples in the cluster v_1 (v_2) belong to the class 1 (class 2) and the relationship between clusters and classes is reliable mainly due to the other element of absolute *zero*.

3.4. Time complexity of RFRC

RFRC proceeds in two steps. In the first step, performing KFCM costs the time complexity of O(NcI), where I is the number of iterations. Then, in the second step, computing the relation matrix $\boldsymbol{R}$ costs the time complexity of O(NcL). For a given dataset, the values N and L are fixed, therefore, the value of c determines the time complexity of RFRC to great extent. Consequently, an appropriate c value should be chosen to make a tradeoff between the time complexity and classification accuracy of RFRC, which is very consistent with the conclusion drawn in (Alippi et al., 2001).

RFRC Algorithm

Training Phase:

Step1: Apply the KFCM on the training dataset $\{\mathbf{x}_i\}$, and obtain the cluster membership degrees $\{\mathbf{u}_i\}$ and cluster centers $\{\mathbf{v}_j\}$ according to (18) and (19), respectively.

Step2: Compute the soft class labels $\{\mathbf{y}_i\}$ for all the training samples by (23).

Step3: Establish the fuzzy relation matrix $\mathbf{R}$ using $\{\mathbf{u}_i\}$ and $\{\mathbf{y}_i\}$ of all the training samples according to (4-6).

Test Phase:

Step4: For a test sample $\mathbf{x}$, compute the cluster membership degrees $\hat{\mathbf{u}}_x$ by (18).

Step5: Compute the class memberships $\hat{\mathbf{y}}_x$ using $\mathbf{R}$ and $\hat{\mathbf{u}}_x$ according to (8) and (9).

Step6: Examine the values of the $\hat{\mathbf{y}}_x$, if the difference between the first maximal and the second maximal class membership is too small, a *reject decision* for the sample $\mathbf{x}$ should be made; otherwise the class label $\hat{\omega}_{\mathbf{x}}$ is yielded in items of (10).

Fig. 4. RFRC algorithm.

3.5. Summary of the RFRC algorithm

In this subsection, the whole process of the proposed Robust FRC (RFRC) algorithm is summarized in Fig. 4 and again it is worth mentioning that RFRC still retains the following characteristics of FRC:

(1) RFRC utilizes a fuzzy relation matrix $\boldsymbol{R}$ to establish the correspondence between structures and the class labels.
(2) It can effectively deal with the classes that cannot be described by a single structure in the dataset.
(3) It always learns the given training dataset without overtraining.
(4) It represents a transparent alternative to conventional black-box techniques.

4. Experimental results and discussion

In order to evaluate the performances among FRC, RFRC and RBFNN, we carry out the experiments on 2 artificial and 11 real-life datasets, and then compare their performances including the classification accuracies and reject rates.

4.1. Artificial datasets

4.1.1. Artificial dataset1

In this subsection, we testify the robustness of RFRC including the *training robustness* and *classification robustness*. The *training robustness* means that the parameter estimation of a classifier can resist the effects of outliers and noises; the *classification robustness* implies that a classifier can make a reasonable decision for a test outlier. For this propose, the two-dimensional artificial dataset1 composed of two classes is designed. The samples in class 1 are randomly generated from a Gaussian distribution with mean $[0,0]$ and variance diag$[0.5,0.5]$; the samples in class 2 are uniformly generated in a 3×3 rectangle centered at $[3.5,0]$. The training and test datasets follow the same distribution. In order to examine the *training robustness*, two outliers $[-20,100]$ and $[-100,20]$ as the training samples are added to class 1 and one outlier $[100,100]$ is added to class 2. For testing the *classification robustness*, 3 outliers located at $[130,150]$, $[50,-210]$ and $[-140,-30]$ are added to the test dataset.

Firstly, we analyze the important parameters in both FRC and RFRC under the condition of the number of centers is set to 2 to examine their *training robustness*. The corresponding parameters including the estimated cluster centers and relation matrices $\boldsymbol{R}$s are listed in Table 1. It can seen from Table 1 that the clustering centers $[23.58,93.46]$ and $[1.39,0.07]$ obtained by FCM in FRC are heavily deviated from the original centers, meaning that they are very sensitive to outliers and thus naturally fail to reveal the inherent structure of the dataset. Such non-robust clustering result causes the $\boldsymbol{R}_{\mathrm{FRC}}$ lack the dominant elements in each row. According to the analysis in Section 3.3, this $\boldsymbol{R}_{\mathrm{FRC}}$ reveals the unreliable relationship between the clusters and classes. In contrast, the centers $[-0.18,0.19]$ and $[3.32,0.11]$ obtained by KFCM in RFRC are very close to the original centers so that they can still relatively really uncover the structure of this dataset. Based on these unbiased clustering centers, the obtained $\boldsymbol{R}_{\mathrm{RFRC}}$ has the row dominant elements 0.66 and 0.51 which are marked by the squares in Table 1, implying that more real relationship between the formed clusters and classes is revealed by this $\boldsymbol{R}_{\mathrm{RFRC}}$. Hence, it is the robustness of the estimation for the cluster centers obtained by KFCM in RFRC that makes $\boldsymbol{R}_{\mathrm{RFRC}}$ able to relatively really reflect the underlying data structure.

Table 1
Parameters of FRC and RFRC on artificial dataset1 including three training outliers

	FRC	RFRC
Cluster centers	$v_1=[23.58\ \ 93.46]$ $v_2=[1.39\ \ 0.07]$	$v_1=[-0.18\ \ 0.19]$ $v_2=[3.32\ \ 0.11]$
Relation matrix	$\boldsymbol{R}_{\mathrm{FRC}}=\begin{bmatrix}0.23 & 0.16\\ 0.00 & 0.00\end{bmatrix}$	$\boldsymbol{R}_{\mathrm{RFRC}}=\begin{bmatrix}\boxed{0.66} & 0.00\\ 0.00 & \boxed{0.51}\end{bmatrix}$

Secondly, we inspect whether such a robustness of the training can lead to the *classification robustness* (in fact, this is our aim of designing RFRC classifier), in other words, whether RFRC can make the reasonable decision for the test outlier. Before testifying the corresponding classification results on the test outliers, we still need to analyze the experimental results on the normally-generated test samples which are illustrated in Fig. 5a–c. The character 'r' in these figures denotes a reject decision for a test sample and the 'e' denotes a misclassified sample. Besides, the 'o' and '+' are the correctly classified samples in class 1 and 2, respectively. From Fig. 5a, we can see that most (97.5%) of the test samples are not identified by FRC and thus the corresponding reject decisions are resulted, and while in Fig. 5c, about half of the samples are misclassified by RBFNN. In contrast, in Fig. 5b, RFRC correctly classifies all of the test samples due to its training robustness.

Further, we analyze the classification results on three test outliers and the corresponding parameters including the cluster and class memberships are displayed in Table 2. From this table, we can observe that for FRC, the cluster memberships of the outlier $[130,150]$ to the two clusters are 0.27 and 0.73 respectively, such an apparent difference between the two values makes the outlier categorized to the second cluster. For the other two outliers, there are also similar consequences. Actually, the outliers, as isolated points far deviated from the given distribution, should not be categorized to any clusters or classes, hence the cluster memberships now obtained by FCM in FRC are obviously unreasonable and anti-intuitive. The same result can be obtained for RBFNN. In contrast, in RFRC, the cluster

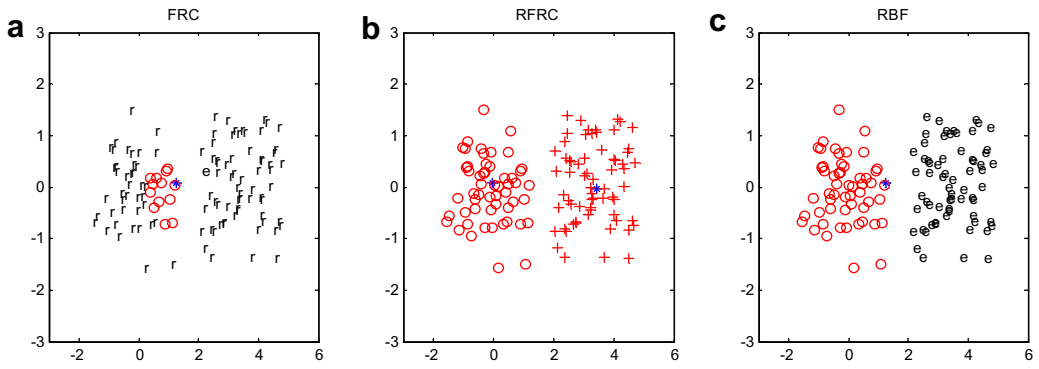

Fig. 5. Classification results on normally-generated test samples of artificial dataset1 by FRC, RFRC and RBFNN, respectively.

Table 2
Cluster and class memberships of three test outliers by FRC, RFRC and RBFNN

Outlier	FRC	RFRC	RBFNN
[130, 150]	$\hat{\boldsymbol{u}} = [0.27\ 0.73]$	$\hat{\boldsymbol{u}} = [0.50\ 0.50]$	$\hat{\boldsymbol{u}} = [0.27\ 0.73]$
	$\hat{\boldsymbol{y}} = [0.00\ 0.00]$	$\hat{\boldsymbol{y}} = [0.09\ 0.00]$	$\hat{\boldsymbol{y}} = [0.13\ 0.33]$
[50, −210]	$\hat{\boldsymbol{u}} = [0.67\ 0.33]$	$\hat{\boldsymbol{u}} = [0.50\ 0.50]$	$\hat{\boldsymbol{u}} = [0.67\ 0.33]$
	$\hat{\boldsymbol{y}} = [0.00\ 0.00]$	$\hat{\boldsymbol{y}} = [0.09\ 0.00]$	$\hat{\boldsymbol{y}} = [0.02\ 0.00]$
[−140, −30]	$\hat{\boldsymbol{u}} = [0.67\ 0.33]$	$\hat{\boldsymbol{u}} = [0.50\ 0.50]$	$\hat{\boldsymbol{u}} = [0.67\ 0.33]$
	$\hat{\boldsymbol{y}} = [0.00\ 0.00]$	$\hat{\boldsymbol{y}} = [0.09\ 0.00]$	$\hat{\boldsymbol{y}} = [0.15\ 0.19]$

Table 3
Comparison between FRC and RFRC on artificial dataset2

	FRC	RFRC
Cluster centers	$v_1 = [1.43\ 0.49\ 0.44]$	$v_1 = [1.25\ 0.45\ 0.02]$
	$v_2 = [0.52\ 0.55\ 0.51]$	$v_2 = [1.15\ 0.49\ 0.93]$
Relation matrix	$\boldsymbol{R}_{\mathrm{FRC}} = \begin{bmatrix} 0.18 & 0.13 \\ 0.14 & 0.18 \end{bmatrix}$	$\boldsymbol{R}_{\mathrm{RFRC}} = \begin{bmatrix} \boxed{0.52} & 0.02 \\ 0.01 & \boxed{0.51} \end{bmatrix}$
Accuracy	19.2%	100%
Reject rate	77.5%	0%

memberships for all the three outliers are all $[0.50, 0.50]$, meaning that the outliers belong equally to the two clusters and thus are impossible to be categorized to any cluster, which reflects exactly the nature of the outliers. Due to its biased cluster memberships and biased parameter $\boldsymbol{R}$, FRC makes reject decisions for almost all the test samples (seen from Fig. 5) including the three outliers. Obviously, such reject decisions attribute to the non-robustness of $\boldsymbol{R}_{\mathrm{FRC}}$ in the training phase, which further leads to the robustness lack of the FRC in the classification. For RBFNN, though rejecting the outliers $[50, -210]$ and $[-140, -30]$ relatively reliably, it still classifies the outlier $[130, 150]$ to a class with the bigger class membership (0.33) as shown in Table 2, as a result, the classification robustness of RBFNN is difficult to be ensured. In contrast, RFRC generates the same class memberships of 0.09 and 0.00 for all the three outliers to the two classes respectively, the margin between these memberships is so small that the outliers will not be categorized to any given classes and thus a reject decision can reasonably be made. In conclusion, RFRC cannot only classify the normally-generated test samples correctly (seen from Fig. 5), but also make the reasonable reject decision for the test outlier, therefore we have reason to believe that it is the training robustness that ensures the classification robustness of RFRC.

4.1.2. Artificial dataset2

This artificial dataset named dataset2 is designed to explore the classification ability for the non-spherical shape of data groups. This is a three-dimensional dataset with two classes. The samples in this dataset are randomly generated on two planes in which one is zero-section (i.e. $z = 0$) plane and the other is one-section (i.e. $z = 1$) planes. The samples in each plane (class) are uniformly generated in a 1×2 rectangle centered at $[1, 0.5]$. The training and test datasets are generated from the same distribution. Table 3 gives the parameters of FRC and RFRC including the cluster centers, relation matrix $\boldsymbol{R}$, accuracy and reject rate when the number c of the centers is set to 2 and Fig. 6a–c illustrate the corresponding classification results.

From the results listed in Table 3, we can observe that: (1) the cluster centers in FRC deviate from their corresponding original centers $[1, 0.5, 0]$ and $[1, 0.5, 1]$, while the generated centers in RFRC are relatively close to the original centers, indicating that KFCM can group the non-Euclidean inherent structures of dataset; (2) as seen from $\boldsymbol{R}_{\mathrm{FRC}}$ in Table 3, due to the absence of the row dominant elements, FRC is difficult to establish the reliable relationship between the clusters and classes, while RFRC can fulfill this through the $\boldsymbol{R}_{\mathrm{RFRC}}$; (3) FRC makes the reject decisions for the test samples of 77.5%, while RFRC classifies all of the test samples correctly due to the unbiased cluster centers and reliable relation matrix. In summary, RFRC is also suitable for the non-spherical clusters compared with FRC.

From Fig. 6a, we can see that most test samples are not identified by FRC and thus corresponding reject decisions are resulted, and in Fig. 6c about half of the samples are misclassified by RBFNN. In contrast, in Fig. 6b all the samples are correctly classified by RFRC.

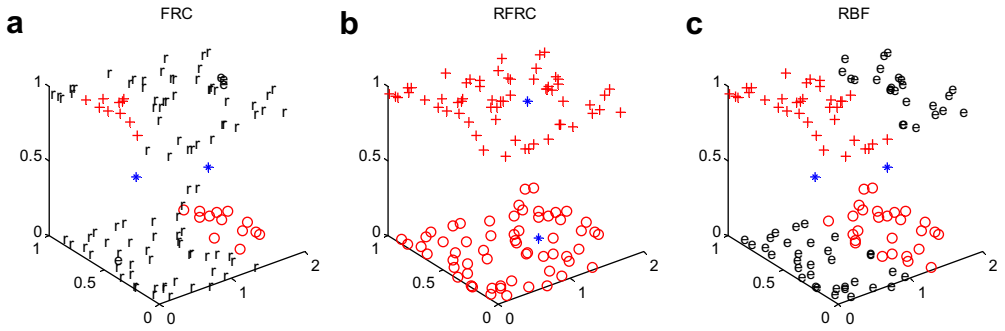

Fig. 6. Classification results of FRC, RFRC and RBFNN on artificial dataset2 respectively.

Finally, we also examine the classification robustness on this dataset as Section 4.1.1, though the dataset assumes a non-spherical distribution different from the dataset1, the similar conclusion can still be drawn and thus omitted here.

4.2. Real-life datasets

4.2.1. Data description

To further investigate the performance of RFRC, we use 11 real-life datasets cited from the UCI Machine Learning Repository (Blake et al., 1998) which is a repository of databases, domain theories and data generators collected by the machine learning community for the empirical analysis of machine learning algorithms.

Table 4 summarizes the characteristics of the employed datasets including the name of the dataset, the number of samples, the number of features and the number of classes. Taking the dataset *Balance* as an example, it has 625 3-class samples where three classes contain 49, 288 and 288 samples, respectively, and each sample is composed of four features: left-weight, left-distance, right-weight and right-distance.

4.2.2. Experiment setup

In the following Section 4.2.3, we first employ the Xie–Beni index (Xie and Beni, 1991) to determine the value c of the cluster centers and then compare the classification results of FRC, RFRC and RBFNN under this setting. However, the value of c determined by such particular cluster validity index may not lead to the optimal performance of the classifier. Therefore, in Section 4.2.4 we instead utilize the trial-and-error approach to seek the optimal value of c in the range from the number of classes up to $c_{\max}$. Here the parameter $c_{\max}$ is set to $\sqrt{N}$ in items of J.C. Bezdek's suggestion (Bezdek, 1998), where N is the number of the training samples.

Table 4
Overview of the characteristics of the datasets used in experiment

Name of dataset	Number of samples	Number of classes	Number of features
Water	116	2	38
Wisconsin Diagnostic Breast Cancer (WDBC)	569	2	30
Ionosphere	351	2	34
Wisconsin Breast Cancer Database (WBCD)	683	2	9
Twonorm	7400	2	20
Thyroid	215	3	5
Iris	150	3	4
Wine	178	3	13
Ecoli	336	8	7
Glass	214	6	9
Balance scale (Balance)	625	3	4

In all of our experiments, each dataset is randomly partitioned into two halves: one half is used for training and the other for testing. This process runs repeatedly and independently for 100 times, and only their averaged results are reported. It is worth mentioning that the features of each sample are normalized to the range between 0 and 1. For comparison fairness, we introduce the reject level $\delta \in [0,1]$ to control the number of reject decisions for a given test dataset. Concretely, we first record the maximal and minimal class memberships (denoted respectively by *y_max* and *y_min*) yielded on the training dataset and then compute the reject threshold θ for the classification of the test dataset

$$\theta = (y\_max - y\_min) \times \delta \tag{24}$$

If the margin between the first maximal and the second maximal class memberships of a test sample is lower than the reject threshold θ, then a reject decision is made. In order to obtain the relatively reasonable reject decision, the reject level δs are all set to the same (relatively small) value of 0.05 in our experiment.

In RFRC, the scale factor λ of RBF kernel is determined by searching in {0.01, 0.05, 0.1, 0.5, 1, 5, 10, 15} with trial-and-error approach and in the same way the k in the soft class labels is also selected in {5, 7, 9, 11, 13, 15}. In addition, the weighting exponent m is set to 2 in both FCM and KFCM.

4.2.3. Comparison among FRC, RFRC and RBFNN

In this section, we compare FRC, RFRC and RBFNN under the condition of the value of c determined by the Xie–Beni index. Table 5 lists the parameters including the scale factor λ, the parameter k and the number c of

Table 5
Performance Comparison among FRC, RFRC and RBFNN on the Benchmark UCI datasets

Dataset	Parameter			FRC		RFRC		RBFNN	
	λ	k	c	Accuracy	Rej rate	Accuracy	Rej rate	Accuracy	Rej rate
Water	1	5	2	91.4 ± 5.4	6.8 ± 5.9	$\mathbf{97.2 \pm 2.1}$	1.1 ± 1.3	96.5 ± 1.9	2.4 ± 1.5
WDBC	5	13	2	49.6 ± 11.4	49.9 ± 12.0	$\mathbf{92.3 \pm 1.3}$	1.9 ± 1.7	91.4 ± 1.2	2.9 ± 0.8
Ionosphere	15	5	2	25.7 ± 4.9	73.2 ± 5.3	$\mathbf{60.4 \pm 15.4}$	6.0 ± 2.7	54.1 ± 4.7	11.3 ± 4.9
WBCD	0.5	5	2	72.7 ± 7.7	26.8 ± 8.1	$\mathbf{96.5 \pm 0.8}$	0.5 ± 0.4	96.2 ± 0.8	1.1 ± 0.4
Twonorm	1	5	2	54.4 ± 18.3	45.5 ± 18.3	96.2 ± 1.1	2.3 ± 1.2	96.1 ± 0.1	2.5 ± 0.1
Thyroid	0.05	7	3	60.6 ± 19.7	38.5 ± 19.9	83.6 ± 8.2	7.0 ± 6.7	83.8 ± 6.8	4.5 ± 3.1
Iris	0.5	9	3	69.4 ± 12.1	28.2 ± 13.0	83.1 ± 11.5	7.7 ± 11.9	$\mathbf{84.2 \pm 6.3}$	6.9 ± 7.4
Wine	5	5	3	85.3 ± 6.3	12.7 ± 7.2	93.0 ± 2.5	2.8 ± 1.7	$\mathbf{94.7 \pm 2.2}$	3.0 ± 1.7
Ecoli	15	9	12	46.9 ± 10.3	48.0 ± 12.4	78.5 ± 4.7	12.1 ± 4.8	$\mathbf{81.0 \pm 2.0}$	5.8 ± 2.5
Glass	0.01	19	8	16.4 ± 5.6	80.3 ± 5.3	32.7 ± 5.3	42.8 ± 8.8	$\mathbf{49.7 \pm 4.7}$	18.0 ± 4.6
Balance	0.1	13	6	35.7 ± 12.3	59.6 ± 13.1	63.1 ± 9.1	19.8 ± 8.1	$\mathbf{80.3 \pm 3.6}$	8.8 ± 1.3

the clusters, and the experimental results of three classifiers including the average accuracy, the corresponding standard deviation, average reject rate and the corresponding standard deviation.

First, we compare the classification results yielded by FRC and RFRC. It can be seen from Table 5 that the accuracies and the reject rates of RFRC are consistently better than those of FRC on all the datasets, and the averaged standard deviations of the accuracies and reject rates obtained by RFRC are respectively lower than those obtained by FRC except for *Ionosphere*, meaning that RFRC is relatively more stable than FRC. Such a performance promotion of RFRC can attribute to both KFCM and the soft class labels, which corporately construct a more real relation matrix $\boldsymbol{R}$ to uncover the underlying relationship between the clusters and classes.

Now let us take the dataset *Balance* as an example to compare the relation matrices $\boldsymbol{R}$s in both FRC and RFRC. When the c is set to 6, the *transpose* of the 6×3 (where 6 and 3 correspond to the numbers of the clusters and classes respectively) relation matrix $\boldsymbol{R}_{\mathrm{FRC}}$ in FRC is shown below

$$\boldsymbol{R}_{\mathrm{FRC}}^{\mathrm{T}} = \begin{bmatrix} 0.58 & 0.36 & 0.51 & 0.56 & 0.34 & 0.25 \\ 0.51 & 0.46 & 0.46 & 0.80 & \boxed{0.81} & 0.25 \\ 0.59 & 0.55 & 0.62 & 0.56 & 0.34 & \boxed{0.80} \end{bmatrix}$$

We can observe that there is no dominant element in the first four row of the $\boldsymbol{R}_{\mathrm{FRC}}$, meaning that the corresponding four clusters (v_3, v_4, v_5 and v_6) are impure, thus FRC does not more likely find the true relation between those clusters and the classes and results in the high reject rate of 59.6% and the low accuracy of 35.7%. Next let us examine RFRC whose $\boldsymbol{R}_{\mathrm{RFRC}}$ is given as follows:

$$\boldsymbol{R}_{\mathrm{RFRC}}^{\mathrm{T}} = \begin{bmatrix} 0.02 & 0.13 & 0.24 & 0.18 & 0.28 & 0.12 \\ \boxed{0.84} & \boxed{0.79} & \boxed{0.76} & 0.18 & 0.32 & 0.16 \\ 0.02 & 0.21 & 0.27 & \boxed{0.84} & \boxed{0.82} & \boxed{0.75} \end{bmatrix}$$

From which we can find that only one dominant element appears in each row, thus the clusters v_1, v_2 and v_3 are prone to pure and the samples falling into these clusters possibly belong to class 2. A similar analysis can also be made for the other clusters. Compared with FRC, the $\boldsymbol{R}_{\mathrm{RFRC}}$ is relatively more real and thus makes RFRC achieve the higher accuracy of 63.1% and lower reject rate of 19.8%.

In addition, for the same setting of c, some heuristic knowledge about this dataset *Balance* can be obtained from the $\boldsymbol{R}_{\mathrm{RFRC}}$: (1) the classes of this dataset are all composed of multi-groups, for example, class 2 is composed of 3 clusters (v_1, v_2 and v_3) under our KFCM; (2) there is no cluster corresponding to class 1 due to the absence of dominant element in the first column under $c = 6$ setting, consequently, such a setting fails to adequately describe the structure of this dataset, leading RFRC to working unwell on this dataset. On the other hand, such a failure also gives us a valuable heuristic to guide the selection of the value c. Such a property of the relation matrix makes the analyses for the cluster structure and classifier prone to be transparent.

Next, let us make a comparison between the performance of RFRC and RBFNN. From the results in Table 5, RFRC produces the better or comparable classification performance on the datasets *Water*, *WDBC*, *Ionosphere*, *WBCD*, *Thyroid* and *Twonorm*, but worse performance on the other 5 datasets. It is worth noting that the comparison here is in fact not quite favorable for our algorithm, because the parameters in RBFNN are the optimized results based on the MSE criterion, in contrast the parameters in RFRC are directly from the operator-based specific construction rather than optimization. Therefore, the relatively inferior classification performance yielded by RFRC is comprehensible. However, we still need to point out that even so, on 6 out of all the 11 datasets, our classifier still achieves better or comparable classification performance compared with RBFNN. Our next work is to optimize the relation matrix $\boldsymbol{R}$ to further promote its performance.

It is worth emphasizing that it is such a direct constructive approach that makes RFRC possess the following advantages: (1) it requires short training time; (2) it always learns given training dataset without overtraining (Setnes and Babuška, 1999); (3) from the obtained relation matrix $\boldsymbol{R}$, it can acquire some insightful information about the structure of given data and the relation between the structure and their classes; (4) according to the yielded class

memberships, the reasonable reject decision for a test sample can be made or not. In conclusion, RFRC is a transparent alternative to conventional black-box learning classifiers such as RBFNN here.

4.2.4. Influence of the number c of the prototypes on the performance of FRC, RFRC and RBFNN

In the above experiment, both RFRC and RBFNN work well on the 5 datasets (*Water*, *WDBC*, *WBCD*, *Twonorm* and *Wine*) and achieve accuracy of 92% above, indicating that the value c determined by the Xie–Beni index is relatively suitable for the five datasets. However, this index works not well enough on the other 6 datasets, for example, the value 6 of c determined by this index is not large enough to fit the dataset *Balance*. Though we also adopt other cluster validity indexes in the literature, none of the indexes have been claimed to be good for all datasets. Up to now, finding an appropriate number c of clusters is still an important and open issue for partitional clustering (Xu, 1996). Due to such a characteristic of the existing validity indexes, in this subsection, we instead utilize the slightly awkward trial-and-error approach to seek the optimal c. Of course, such optimal value of c is gained at the cost of time.

In this subsection, we record the performance of FRC, RFRC and RBFNN changing as a function of the c on the 6 datasets (*Ionosphere*, *Thyroid*, *Iris*, *Ecoli*, *Glass* and *Balance*). No matter how many classes the dataset is composed of, we find that the changing trends of classification performance on these datasets are basically similar. Hence for simplicity of illustration, we just present the value of c's effect on the performance of three algorithms on the datasets *Balance*.

Fig. 7 illustrates the changing curves (left: accuracy, right: reject rate) of FRC, RFRC and RBFNN on the dataset *Balance* with the c value incrementally varying from the number of class to $c_{\max}(\sqrt{N})$, respectively. From this figure, we can see that that the accuracy (reject rate) of FRC increases (decreases) quickly as the c increases. Different from this, the accuracy (reject rate) yielded by RFRC increases (decreases) considerably as c increases from 6 to 12 and then begins stabilizing at about 75.0% (11.0%) when $c \geqslant 12$, from which we can infer that larger c can reveal more refined, even missing, structure of this dataset than smaller c. In order to illustrate our intuition, we record the $\boldsymbol{R}_{\mathrm{RFRC}}$ when the c equals 12

$$\boldsymbol{R}^{\mathrm{T}}_{\mathrm{RFRC}} = \begin{bmatrix} \boxed{0.74} & \boxed{0.77} & 0.06 & 0.36 & 0.09 & 0.19 & 0.09 & 0.45 & 0.16 & 0.13 & 0.07 & 0.15 \\ 0.23 & 0.37 & \boxed{0.92} & \boxed{0.92} & \boxed{0.88} & \boxed{0.86} & 0.09 & 0.45 & 0.16 & 0.01 & 0.07 & 0.27 \\ 0.33 & 0.25 & 0.06 & 0.36 & 0.09 & 0.19 & \boxed{0.91} & \boxed{0.94} & \boxed{0.87} & \boxed{0.83} & \boxed{0.77} & \boxed{0.73} \end{bmatrix}.$$

From the matrix, we can observe that as the c increases, the original missed relation between class 1 and clusters under $c = 6$ setting can now be found, the class 1 corresponds to the two new clusters v_1 and v_2, while the other relations also yield some changes, concretely class 2 and class 3 correspond 4 clusters and 6 clusters respectively. Such relationship between clusters and classes basically accords with the distribution of this dataset. From the above analysis, it can be concluded that the larger c possibly makes the formed clusters more pure so that the subsequently-constructed $\boldsymbol{R}$ more really reflects the underlying structures of this dataset, thus giving rise to better performance. For RBFNN, the changing curves on the *Balance* are relatively smooth when $c \geqslant 5$. That is because the complexity of RBFNN is determined by the number of hidden nodes, i.e., the number c of the cluster centers, and when the value of c is large enough for the given problem, increasing the c cannot always promote the classifier's performance.

From the above analysis, the relatively large c value seems able to achieve good classification performance. However, the larger the c value is, the more space and time is consumed. More importantly, the larger c makes the regions formed by KFCM more complex and thus more possibly RFRC overfit the given dataset. Therefore, the c value should appropriately be determined to make a balance between the performance and complexity of the algorithms.

5. Conclusion

Fuzzy relational classifier (FRC) is recently proposed two-step nonlinear classifier, which is very different from

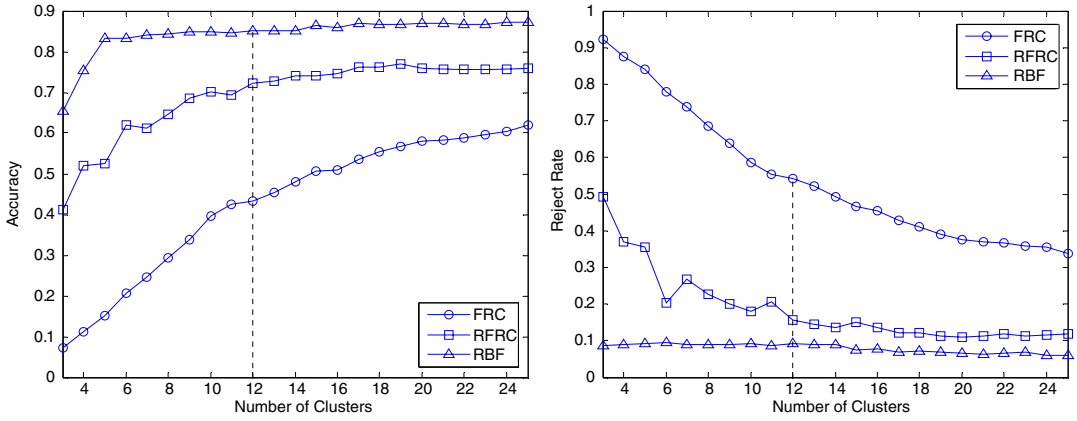

Fig. 7. Performance for varying number c of clusters by FRC, RFRC and RBFNN on *Balance* dataset.

the conventional learning classifier such as neural networks and support vector machines (SVM). In this paper, a Robust FRC algorithm (called RFRC) based on FRC is developed for robust pattern recognition. Concretely, two approaches are adopted to overcome the FRC's disadvantages and maintain its original advantages. Firstly, we utilize our previously proposed robust KFCM to replace FCM to realize the clustering robustness for non-spherical groups in data. Secondly, we employ the soft rather than hard class labels to enhance its error tolerance. Consequently, a significant gain of classification performance is obtained on most of the datasets used here.

The classification results on 2 artificial datasets demonstrate that RFRC not only robustly classifies the dataset including outliers, but also effectively handles the dataset composed of non-spherical groups. On the other hand, the experimental results on 11 real-life datasets can be summarized as follows: (1) RFRC consistently outperform FRC in classification performance on all the datasets; (2) the prototype number c influences the performance of FRC, RFRC and RBFNN and on 6 relatively complex datasets, the increase of the value c can effectively promote the classification performances; (3) the time complexities of FRC, RFRC and RBFNN all depend on the number c, hence, a tradeoff between the performance and complexity of the algorithm should be considered by choosing an appropriate c value and (4) the relation matrix can be used to not only establish effective classifier but also acquire the transparent heuristic information in revealing the structure of given data and the relation between the structure and their classes.

Our further and ongoing works include the adaptive determination for the number c of the prototypes and the kernel parameter in KFCM, the optimization of the relation matrix, the selective usage of the existing various composite operators and their combination.

Acknowledgements

The authors thank the anonymous reviewers for their constructive and valuable comments that greatly improved this paper. We thank Natural Science Foundation of Jiangsu Province under Grant No. BK2006521, National Science Foundation of China under Grant No. 60505004 and Jiangsu "QingLan" Project Foundation for partial supports, respectively.

References

Abe, S., 2005. Training of support vector machines with Mahalanobis kernels. In: Internat. Conf. on Artificial Networks. Lecture Notes in Computer Science, vol. 3697, pp. 571–576.

Alippi, C., Piuri, V., Scotti, F., 2001. Accuracy versus complexity in RBF neural networks. IEEE Instrum. Measur. Mag. 4 (1), 32–36.

Bezdek, J.C., 1981. Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum, New York.

Bezdek, J.C., 1998. Pattern Recognition in Handbook of Fuzzy Computation. IOP Publishing Lte., Boston, NY (Chapter F6).

Blake, C., Keogh, E., Merz, C.J., 1998. UCI repository of machine learning databases. Department of Information and Computer Science, University of California, Irvine, CA, <http://www.ics.uci.edu/~mlearn/MLRepository.html>.

Chen, S.C., Zhang, D.Q., 2004. Robust image segmentation using FCM with spatial constraints based on new kernel-induced distance measure. IEEE Trans. Systems Man Cybernet. B 34 (4), 1907–1916.

Cover, T.M., 1965. Geomeasureal and statistical properties of systems of linear inequalities in pattern recognition. Electron. Comput. 14, 326–334.

Cover, T.M., Hart, P.E., 1967. Nearest neighbor pattern classification. IEEE Trans. Inform. Theory 13, 21–27.

Cristianini, N., Taylor, J.S., 2000. An Introduction to SVMs and Other Kernel-based Learning Methods. Cambridge University Press.

Dave, R.N., Krishnapuram, R., 1997. Robust clustering methods: A unified view. IEEE Trans. Fuzzy Systems 5, 270–293.

Girolami, M., 2002. Mercer kernel-based clustering in input space. IEEE Trans. Neural Networks 13, 780–784.

Hathaway, R.J., Bezdek, J.C., 2000. Generalized fuzzy c-means clustering strategies using L_p norm distance. IEEE Trans. Fuzzy Systems 8, 572–576.

Haykin, S., 1999. Neural Networks: A Comprehensive Foundation, second ed. Prentice-Hall.

Huber, P.J., 1981. Robust Statistics. Wiley, New York.

Jain, A.K., Murty, M.N., Flynn, P.J., 1999. Data Clustering: A Review. ACM Computing Surveys.

Jajuga, K., 1991. L_1 norm based fuzzy clustering. Fuzzy Sets Systems 39 (1), 43–50.

Kaufman, L., Rousseeuw, P.J., 1990. Finding Groups in Data: An Introduction to Cluster Analysis. John Wiley.

Keller, J.M., Gray, M.R., Givens, J.A., 1985. A fuzzy k-nearest neighbor algorithm. IEEE Trans. Systems Man Cybernet. 15, 580–585.

Kim, D.W., Lee, K., Lee, D., 2004. Evaluation of the performance of clustering algorithms in kernel-based input space. Pattern Recognition 38, 607–661.

Klawonn, F., Keller, A., 1999. Fuzzy clustering based on modified distance. In: Proc. Third Internat. Symp. on Intelligent Data Analysis, IDA'99, LNCS 1642, pp. 291–301.

Klir, G.J., Youan, B., 1995. Fuzzy Sets and Fuzzy Logic: Theory and Applications. Prentice-Hall, Englewood Cliffs, NJ.

Krzanowski, W.J., 1988. Principles of Multivariate Analysis: A User's Perspective. Oxford Clarendron Press.

Leski, J., 2003. Towards a robust fuzzy clustering. Fuzzy Sets Systems 137 (2), 215–233.

Musavi, M.T., Ahmed, W., Chan, K.H., Faris, K.B., Hummels, D.M., 1992. On the training of radial basis function classifiers, 5 (4), pp. 595–603.

Pedrycz, W., 1994. In: Kandel, A., Langholz, G. (Eds.), Reasoning by Analogy in Fuzzy Controllers, Fuzzy Control Systems. CRC, Boca Raton, FL, pp. 55–74.

Pedrycz, W., Vukovich, G., 2004. Fuzzy clustering with supervision. Pattern Recognition. 37, 1229–1349.

Pizzi, N.J., Pedrycz, W., 2000. Fuzzy set theoretic adjustment to training set class labels using robust location measures. In: Internat. Joint Conf. on Neural Networks.

Ramirez, L. et al, 2003. Prototypes Stability analysis in the design of fuzzy classifiers to assess the severity of scoliosis. In: IEEE Canadian Conf. on Electrical and Computer Engineering, vol. 3, pp. 1465–1468.

Roth, V., Steinhage, V., 2000. Nonlinear discriminant analysis using kernel functions. Adv. Neural Inform. Process. Systems 12, 568–574.

Scholkopf, B., Smola, A.J., Muller, K.R., 1998. Nonlinear component analysis as a kernel eigenvalue problem. Neural Comput. 10, 1299–1319.

Setnes, M., Babuška, R., 1999. Fuzzy relational classifier trained by fuzzy clustering. IEEE Trans. SMC Part B 29, 619–625.

Sohn, S., Daqli, C.H., 2001. Advantages of using fuzzy class memberships in self-organizing map and support vector machines. In: Internat. Joint Conf. on Neural Networks Proceedings, vol. 3, pp. 1886–1890.

Sung-Bae, C., Kim, J.H., 1995. Multiple network fusion using fuzzy logic. IEEE Trans. Neural Networks 6 (2), 497–501.

Wu, K.L., Yang, M.S., 2002. Alternative *c*-means clustering algorithms. Pattern Recognition 35, 2267–2278.

Xie, X.L., Beni, G.A., 1991. Validity measure for fuzzy clustering. IEEE Trans. Pattern Anal. Machine Intell. 3, 841–846.

Xu, L., 1996. How many clusters? A ying–yang machine based theory for a classical open problem in pattern recognition. In: IEEE Internat. Conf. on Neural Networks, vol. 3, pp. 1546–1550.

Yao, Y.H., Chen, L.H., Chen, Y.Q., 1999. Unsupervised curve-based clustering. Neural Networks 2, 1097–1101.

Zadeh, L.A., 1965. Fuzzy sets. Inf: Control 8, 338–353.

Zhang, D.Q., Chen, S.C., 2003. Clustering incomplete data using kernel-based fuzzy *c*-means algorithm. Neural Processing Lett. 18 (3), 155–162.

Zhang, D.Q., Chen, S.C., 2004. A novel kernelized fuzzy *c*-means algorithm with application in medical image segmentation. Artif. Intell. Med. 32 (1), 37–50.