Pattern Recognition 44 (2011) 1750–1760

Contents lists available at ScienceDirect

Pattern Recognition

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

Fuzzy *C*-means based clustering for linearly and nonlinearly separable data

Du-Ming Tsai *, Chung-Chan Lin

Department of Industrial Engineering & Management, Yuan-Ze University, 135 Yuan-Tung Road, Nei-Li, Tao-Yuan, Taiwan, ROC

ARTICLE INFO

Article history:
Received 21 June 2010
Received in revised form
18 January 2011
Accepted 9 February 2011
Available online 16 February 2011

Keywords:
Clustering
Fuzzy *C*-means
Kernel fuzzy *C*-means
Distance metric

ABSTRACT

In this paper we present a new distance metric that incorporates the distance variation in a cluster to regularize the distance between a data point and the cluster centroid. It is then applied to the conventional fuzzy *C*-means (FCM) clustering in data space and the kernel fuzzy *C*-means (KFCM) clustering in a high-dimensional feature space. Experiments on two-dimensional artificial data sets, real data sets from public data libraries and color image segmentation have shown that the proposed FCM and KFCM with the new distance metric generally have better performance on non-spherically distributed data with uneven density for linear and nonlinear separation.

1. Introduction

Clustering is an unsupervised learning method to partition a collection of multivariate data points into meaningful groups, where all members within a group represent similar characteristics and data points between different groups are dissimilar to each other. It has been an important technique for pattern recognition, image processing and data mining. It has also been applied successfully in many fields such as marketing that finds groups of customers with similar purchasing behaviors, biology that groups unknown plants/animals into species, and medical image processing that divides an image into a few meaningful regions for diagnosis.

The similarity criterion for distinguishing the difference between data points is generally measured by distance. Two data points belong to the same group if they are close to each other. They are evidently from different groups if the distance between them is distinctly large. The success of a clustering algorithm is highly affected by the data structure including the cluster shape, cluster density and linear/nonlinear separability. The fuzzy *C*-means (FCM) algorithm [1] is one of the most popular techniques used for clustering. The effectiveness of the clustering method relies on the distance measure. The conventional FCM method uses the Euclidean distance as the similarity criterion that measures the distance between each data point $\boldsymbol{x}_i$ and a cluster centroid $\boldsymbol{v}_c$, i.e. $||\boldsymbol{x}_i-\boldsymbol{v}_c||^2$, with a weight w_{ic} which is inversely proportional to the distance. The Euclidean squared-norm distance makes FCM only suitable for clustering hyperspherically distributed data groups. In order to improve the performance of the conventional FCM, Wu and Yang [2] replaced the Euclidean norm with a normalized distance function $1-\exp(-\beta\|\boldsymbol{x}_i-\boldsymbol{v}_c\|^2)$, where β is a positive constant. Zhang and Chen [3,4] used $1-K(\boldsymbol{x}_i, \boldsymbol{v}_c)$ as the distance measure, where $K(\boldsymbol{x}_i,\boldsymbol{v}_c)$ is a kernel function. The normalized distance function proposed by Wu and Yang is only a special case of Zhang and Chen's method when a Gaussian function is used as the kernel. In a multi-dimensional space, some data features could be more critical than others. A feature-weighted distance [5,6] was proposed to improve the performance of FCM. The distance measure is given by $\sum_k \alpha_k[\boldsymbol{x}_i-\boldsymbol{v}_c]_k^2$, where $[\boldsymbol{x}_i-\boldsymbol{v}_c]_k$ is the difference of the kth feature between $\boldsymbol{x}_i$ and $\boldsymbol{v}_c$, and α_k is the assigned feature weight.

The conventional FCM only works for linearly separable data points. Girolami [7] proposed a kernel-based FCM by mapping the data in the observation space to a higher dimensional feature space so that nonlinear separation of clusters can be achieved. It uses the radial basis function (RBF) kernel to implicitly define the mapping function from data space to feature space. For a RBF kernel, the kernel-based FCM can be interpreted as replacing the Euclidean metric in the FCM algorithm by a probability metric. The choice of the RBF kernel bandwidth remains an open question in the paper. All the bandwidth values for the test data sets in the experiments were empirically determined. For the test samples with known class labels, the best bandwidth can be surely determined by an exhaustive search. To apply the kernel FCM to real data sets where the true class labels are not known, the best bandwidth value cannot be determined by trial-and-error or any search process since the true recognition rate is unknown. This problem is common in all unsupervised learning methods.

* Corresponding author. Fax: +886 3 463 8907.
E-mail addresses: iedmtsai@saturn.yzu.edu.tw, s929501@mail.yzu.edu.tw, s968902@mail.yzu.edu.tw (D.-M. Tsai).

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.02.009

Kim et al. [8] evaluated the performance of four kernel-based clustering methods including kernel *K*-means, kernel FCM, kernel average linkage algorithm and kernel mountain algorithm. The RBF function was used as the kernel in all four kernel clustering algorithms. The results in their experiments indicated each kernel clustering algorithm outperforms its conventional counterpart. The choice of the RBF bandwidth value that derived the superiority over the conventional methods for each individual test data set was not addressed in their paper.

Ng et al. [9] presented a spectral clustering method to improve the clustering results in the original data space. It consists of mapping the original space into a compact feature space by means of eigenvector decomposition, followed by *K*-means clustering in the new feature space to obtain better clustering results. The dominant eigenvectors are extracted from an affinity matrix constructed with elements $\exp(-\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2/2\sigma^2)$ for sample pairs $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. The parameter value σ must be searched in a wide range with a pre-determined resolution, and the one that gives the tightest (smallest distortion) clusters in the new feature space is chosen. For each possible σ value, the whole clustering procedure including the calculation of eigenvectors from the large affinity matrix and then the *K*-means clustering must be repeated once. It is thus computationally very expensive. Zelnik-Manor and Perona [10] studied the parameter selections in spectral clustering. The bandwidth parameter is adaptively given by $\sigma_i\cdot\sigma_j=d(\boldsymbol{x}_i, \boldsymbol{x}_{K,i})\ d(\boldsymbol{x}_j, \boldsymbol{x}_{K,j})$ for sample pairs $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, where $\boldsymbol{x}_{K,i}$ is the *K*th neighbor of the individual data point $\boldsymbol{x}_i$, and $d(\cdot,\cdot)$ is some distance measure. The value of neighboring *K* has to be determined empirically, and the parameter setting process is computationally intensive. Von Luxburg [11] also recommended similar parameter setting as a rule of thumb for the choice of the bandwidth value in spectral clustering. Fred and Jain [12] proposed an evidence accumulation clustering (EAC) method for combining various existing clustering algorithms and/or the same clustering algorithm with various parameter values to obtain a partition that is better than individual clustering algorithms. The evidence accumulation technique maps the clustering ensemble into a new similarity measure between patterns by accumulating pairwise patterns with a voting mechanism. It can be expected that the application of the EAC method can lead to even better partitions of complex data sets if more powerful clustering algorithms are used in the combination.

The currently existing fuzzy *C*-means clustering methods both in the observed data space and in the mapped feature space basically consider only the Euclidean distance between each data point and every cluster centroid. It describes only hyperspherical clusters in data space or in feature space. Furthermore, the density of data points in a cluster could be distinctly different from other clusters in a data set. The conventional metric evaluates only the distance between two individual data points. It ignores the global distance variation for all data points in a cluster.

In this study, we add the distance variation of each individual data group to regularize the distance between a data point and the cluster centroid. The new distance metric is then applied to both the conventional FCM and the kernel FCM. The proposed distance metric can be better applied to non-hyperspherically shaped data with uneven densities for linear separation (in the observed data space) and nonlinear separation (in the mapped feature space). For the proposed distance metric in the kernel FCM, we also introduce a simple bandwidth selection rule for the RBF kernel when kernel FCM is used for clustering of unlabeled real data. Two-dimensional artificial data sets are first used to evaluate the robustness of the proposed clustering methods on cluster shape, cluster density and linear/nonlinear separability. Real data sets from public data libraries are then used to evaluate the clustering results. Finally, the application of the conventional FCM and KFCM and the proposed clustering methods to color image segmentation is also given.

This paper is organized as follows: Section 2 describes four versions of the fuzzy *C*-means clustering methods: conventional FCM, the proposed distance metric for FCM, the kernel FCM (KFCM), and the proposed distance metric for KFCM. The selection rule of bandwidth value of a given data set is also discussed in this section. Section 3 presents the experimental results on 2D artificial data sets, real data sets from public databanks and color image segmentation. The paper is concluded in Section 4.

2. Fuzzy *C*-means based clustering

In this section, we present four clustering methods based on fuzzy *C*-means. The conventional FCM and its formulation are first described so that the implication of the remaining three clustering methods can be correspondingly presented. Let $\boldsymbol{X}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_N\}$ be a collection of unlabeled data points, and $\boldsymbol{x}_i\in\Re^d$. The goal of clustering is to partition the data set $\boldsymbol{X}$ into *C* intrinsic groups.

2.1. Fuzzy C-means (FCM) clustering

FCM partitions the data set $\boldsymbol{X}$ into *C* clusters by minimizing the errors in terms of the weighted distance of each data point $\boldsymbol{x}_i$ to all centroids of the *C* clusters. That is,

$$\text{Min } J_{\text{FCM}}=\sum_{c=1}^{C}\sum_{i=1}^{N} w_{ic}^{p}\|\boldsymbol{x}_i-\boldsymbol{v}_c\|^2$$

s.t.

$$\sum_{c=1}^{C} w_{ic}=1,\quad i=1,2,\ldots,N$$

where p is the exponent.

By using the Lagrange multipliers, we can solve for the weight w_{ic}. The weight w_{ic} and the centroid $\boldsymbol{v}_c$ can be updated by the expectation–maximization (E–M) algorithm:

E-step:

$$w_{ic}=1/\sum_{j=1}^{C}\left(\frac{d_{ic}^2}{d_{ij}^2}\right)^{1/(p-1)}\quad \text{for } i=1,2,\ldots,N \text{ and } c=1,2,\ldots,C$$

where

$$d_{ic}^2=\|\boldsymbol{x}_i-\boldsymbol{v}_c\|^2$$

M-step:

$$\boldsymbol{v}_c=\frac{\sum_{j=1}^{N} w_{jc}^{p}\cdot\boldsymbol{x}_j}{\sum_{j=1}^{N} w_{jc}^{p}}\quad \text{for } c=1,2,\ldots,C$$

The E–M algorithm recursively proceeds until a convergence condition is satisfied.

2.2. New distance metric for FCM

The conventional FCM only takes into account the Euclidean distances between individual data points and centroids. It ignores the distance variation of the data points in the same cluster. It thus may degrade the performance of FCM for data points with uneven densities or non-hyperspherical shapes in individual clusters.

In order to improve the effectiveness of FCM, a new metric that takes the distance variation in each cluster as the regularization of the Euclidean distance is proposed in this paper. The new distance

metric is defined as

$$\hat{d}_{ic}^2 = \frac{\|\boldsymbol{x}_i - \boldsymbol{v}_c\|^2}{\sigma_c} \tag{1}$$

where σ_c is the weighted mean distance in cluster c, and is given by

$$\sigma_c = \left\{ \frac{\sum_{j=1}^{N} w_{jc}^p \cdot \|\boldsymbol{x}_j - \boldsymbol{v}_c\|^2}{\sum_{j=1}^{N} w_{jc}^p} \right\}^{1/2} \tag{2}$$

Different from the Mahalanobis distance, the new distance measure normalizes the distance based on the spread of data points from the centroid in a cluster. It is not normalized with respect to the covariance between features. The new fuzzy C-means algorithm, named FCM-σ, searches for C clusters by minimizing the objective:

$$\text{Min} J_{\text{FCM-}\sigma} = \sum_{c=1}^{C} \sum_{i=1}^{N} w_{ic}^p \cdot \frac{\|\boldsymbol{x}_i - \boldsymbol{v}_c\|^2}{\sigma_c}$$

s.t.

$$\sum_{c=1}^{C} w_{ic} = 1, \quad i = 1,2,\ldots,N$$

The E–M algorithm is also iteratively carried out to solve for the weights w_{ic} and the centroids $\boldsymbol{v}_c$:

E-step:

$$w_{ic} = 1 \Big/ \sum_{j=1}^{C} \left(\frac{\hat{d}_{ic}^2}{\hat{d}_{ij}^2} \right)^{1/(p-1)} \quad \text{for } i = 1, 2,\ldots, N \text{ and } c = 1,2,\ldots,C$$

where

$$\hat{d}_{ic}^2 = \frac{\|\boldsymbol{x}_i - \boldsymbol{v}_c\|^2}{\sigma_c}$$

M-step:

$$\boldsymbol{v}_c = \frac{\sum_{j=1}^{N} w_{jc}^p \cdot \boldsymbol{x}_j}{\sum_{j=1}^{N} w_{jc}^p}$$

and

$$\sigma_c = \left\{ \frac{\sum_{j=1}^{N} w_{jc}^p \cdot \|\boldsymbol{x}_j - \boldsymbol{v}_c\|^2}{\sum_{j=1}^{N} w_{jc}^p} \right\}^{1/2}$$

2.3. Kernel fuzzy C-means clustering

The conventional FCM and the proposed FCM-σ can only deal with linearly separable data points in the observation space. The observed data set $\boldsymbol{X}$ can be transformed into a higher dimensional feature space by applying a nonlinear mapping function to achieve nonlinear separation. The mapping function Φ need not be explicitly specified. Rather, the inner product of $\Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}_j)$ in the mapped feature space can be calculated with a kernel function $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ if the kernel meets the Mercer's theorem [13].

The distance between $\boldsymbol{x}_i$ and the centroid $\boldsymbol{v}_c$ in the higher dimensional feature space with a mapping function Φ is given by

$$\Phi_{d_{ic}^2} = \|\Phi(\boldsymbol{x}_i) - \Phi_{\boldsymbol{v}_c}\|^2 \tag{3}$$

The centroid of cluster c in the mapped feature space is calculated by

$$\Phi_{\boldsymbol{v}_c} = \frac{\sum_{j=1}^{N} w_{jc}^p \cdot \Phi(\boldsymbol{x}_j)}{\sum_{j=1}^{N} w_{jc}^p} \tag{4}$$

The distance $\Phi_{d_{ic}^2}$ in feature space is, therefore, obtained by [7,14]

$$\begin{aligned} \Phi_{d_{ic}^2} = \|\Phi(\boldsymbol{x}_i) - \Phi_{\boldsymbol{v}_c}\|^2 &= \left[\Phi(\boldsymbol{x}_i) - \frac{\sum_{j=1}^{N} w_{jc}^p \cdot \Phi(\boldsymbol{x}_j)}{\sum_{j=1}^{N} w_{jc}^p} \right] \\ &\times \left[\Phi(\boldsymbol{x}_i) - \frac{\sum_{j=1}^{N} w_{jc}^p \cdot \Phi(\boldsymbol{x}_j)}{\sum_{j=1}^{N} w_{jc}^p} \right]^{\mathrm{T}} = \Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}_i) \\ &- 2 \times \Phi(\boldsymbol{x}_i) \times \frac{\sum_{j=1}^{N} w_{jc}^p \cdot \Phi(\boldsymbol{x}_j)}{\sum_{j=1}^{N} w_{jc}^p} + \frac{\sum_{j=1}^{N} w_{jc}^p \cdot \Phi(\boldsymbol{x}_j)}{\sum_{j=1}^{N} w_{jc}^p} \\ &\times \frac{\sum_{j=1}^{N} w_{jc}^p \cdot \Phi(\boldsymbol{x}_j)}{\sum_{j=1}^{N} w_{jc}^p} = K_{ii} - 2 \times \frac{\sum_{j=1}^{N} w_{jc}^p \cdot K_{ij}}{\sum_{j=1}^{N} w_{jc}^p} \\ &+ \frac{\sum_{m=1}^{N} \sum_{n=1}^{N} w_{mc}^p \cdot w_{nc}^p \cdot K_{mn}}{\sum_{m=1}^{N} \sum_{n=1}^{N} w_{mc}^p \cdot w_{nc}^p} \end{aligned} \tag{5}$$

where $K_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is a kernel function. A radial basis function (RBF) kernel

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 / h) \tag{6}$$

is popularly implemented for the implicit mapping. The parameter h is the bandwidth that needs to be adjusted for individual data sets.

The kernel fuzzy C-means (KFCM) algorithm searches for C clusters, which minimizes the distance objective with a mapping function Φ in feature space:

$$\text{Min} J_{\text{KFCM}} = \sum_{c=1}^{C} \sum_{i=1}^{N} w_{ic}^p \cdot \|\Phi(\boldsymbol{x}_i) - \Phi_{\boldsymbol{v}_c}\|^2$$

s.t.

$$\sum_{c=1}^{C} w_{ic} = 1, \quad i = 1,2,\ldots,N$$

By using the Lagrange multipliers to convert the constrained objective above as an unconstrained optimization model and setting the derivatives to zero with respect to w_{ic}, we can obtain the weight in an equivalent form of FCM, i.e.

$$w_{ic} = 1 \Big/ \sum_{j=1}^{C} \left(\frac{\Phi_{d_{ic}^2}}{\Phi_{d_{ij}^2}} \right)^{1/(p-1)} \tag{7}$$

Note that $\Phi_{d_{ic}^2}$ can be calculated from Eq. (5) using the kernel induction. The E–M algorithm can also be recursively applied to update the weights:

E-step:

$$w_{ic} = 1 \Big/ \sum_{j=1}^{C} \left(\frac{\Phi_{d_{ic}^2}}{\Phi_{d_{ij}^2}} \right)^{1/(p-1)}$$

M-step:

$$\Phi_{d_{ic}^2} = K_{ii} - 2 \times \frac{\sum_{j=1}^{N} w_{jc}^p \cdot K_{ij}}{\sum_{j=1}^{N} w_{jc}^p} + \frac{\sum_{m=1}^{N} \sum_{n=1}^{N} w_{mc}^p \cdot w_{nc}^p \cdot K_{mn}}{\sum_{m=1}^{N} \sum_{n=1}^{N} w_{mc}^p \cdot w_{nc}^p}$$

In the M-step, the cluster centroid in the mapped feature space cannot be explicitly calculated. Instead, the distance between a data point and the cluster centroid is calculated using the kernel function.

2.4. New distance metric for KFCM

Similar to the FCM-σ algorithm proposed in Section 2.2, we can also revise the distance metric by introducing the distance variation as the regularization in the mapped feature space.

The new metric in feature space is defined as

$$\hat{\Phi}_{d_{ic}^2} = \frac{\|\Phi(\boldsymbol{x}_i) - \Phi_{\boldsymbol{v}_c}\|^2}{\Phi_{\sigma_c}} = \frac{\Phi_{d_{ic}^2}}{\Phi_{\sigma_c}} \tag{8}$$

where Φ_{σ_c} is the weighted mean distance of cluster c in the mapped feature space, which is equivalent to σ_c in the observed data space. It is calculated by

$$\Phi_{\sigma_c} = \left\{ \frac{\sum_{i=1}^{N} w_{ic}^p \cdot \Phi_{d_{ic}^2}}{\sum_{i=1}^{N} w_{ic}^p} \right\}^{1/2} \tag{9}$$

where $\Phi_{d_{ic}^2} = \|\Phi(\boldsymbol{x}_i) - \Phi_{\boldsymbol{v}_c}\|^2$, and is obtained from Eq. (5) using the kernel computation.

The new kernel fuzzy C-means algorithm, named KFCM-σ, searches for C clusters by minimizing the objective:

$$\text{Min} J_{\text{KFCM-}\sigma} = \sum_{c=1}^{C} \sum_{i=1}^{N} w_{ic}^p \cdot \frac{\|\Phi(\boldsymbol{x}_i) - \Phi_{\boldsymbol{v}_c}\|^2}{\Phi_{\sigma_c}}$$

s.t.

$$\sum_{c=1}^{C} w_{ic} = 1, \quad i = 1,2,\ldots,N$$

The E–M algorithm for solving the weights w_{ic} in feature space is thus given as follows:

E-step:

$$w_{ic} = 1 \Big/ \sum_{j=1}^{C} \left(\frac{\hat{\Phi}_{d_{ic}^2}}{\hat{\Phi}_{d_{ij}^2}} \right)^{1/(p-1)} = 1 \Big/ \sum_{j=1}^{C} \left(\frac{\Phi_{d_{ic}^2}/\Phi_{\sigma_c}}{\Phi_{d_{ij}^2}/\Phi_{\sigma_j}} \right)^{1/(p-1)}$$

M-step:

$$\Phi_{d_{ic}^2} = K_{ii} - 2 \times \frac{\sum_{j=1}^{N} w_{jc}^p \cdot K_{ij}}{\sum_{j=1}^{N} w_{jc}^p} + \frac{\sum_{m=1}^{N} \sum_{n=1}^{N} w_{mc}^p \cdot w_{nc}^p \cdot K_{mn}}{\sum_{m=1}^{N} \sum_{n=1}^{N} w_{mc}^p \cdot w_{nc}^p}$$

$$\Phi_{\sigma_c} = \left\{ \frac{\sum_{i=1}^{N} w_{ic}^p \cdot \Phi_{d_{ic}^2}}{\sum_{i=1}^{N} w_{ic}^p} \right\}^{1/2}$$

The KFCM-σ algorithm allows the clustering of non-hyperspherically shaped data with uneven density in the mapped feature space and achieves nonlinear separation for the data in the observation space.

2.5. Bandwidth setting

The kernel-based learning provides powerful nonlinear separability of classes if the parameter value of a given kernel is carefully selected. For a nonlinear support vector machine (SVM) classification with an RBF kernel, the best bandwidth value can be chosen such that the recognition rate of a collection of labeled training data is maximized. The choice of an appropriate bandwidth value for a kernel-based clustering algorithm could be very troublesome since all the data points are unlabeled and their true classes are unknown. To evaluate the performance of the KFCM and the proposed KFCM-σ with an RBF kernel, we propose a fast bandwidth selection rule based on the distance variance of all data points in the collection. The proposed bandwidth setting rule is defined as follows.

Given the collection $\boldsymbol{X}=\{\boldsymbol{x}_1, \boldsymbol{x}_2,\ldots, \boldsymbol{x}_N\}$, the data center of $\boldsymbol{X}$ is given by

$$\boldsymbol{v} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{x}_i$$

Let $d_i = ||\boldsymbol{x}_i - \boldsymbol{v}||$ be the distance from data point $\boldsymbol{x}_i$ to the data center $\boldsymbol{v}$. The mean distance of d_i is then calculated by

$$\overline{d} = \frac{1}{N} \sum_{i=1}^{N} d_i$$

The bandwidth h is set to the variance of d_i, i.e.

$$h = \frac{1}{N-1} \sum_{i=1}^{N} (d_i - \overline{d})^2 \tag{10}$$

The performance of the proposed clustering methods FCM-σ and KFCM-σ with the RFB parameter given by the bandwidth selection rule will be evaluated in the experiment section.

3. Experimental results

This section evaluates the effectiveness of the four clustering methods, FCM, FCM-σ, KFCM and KFCM-σ. Two-dimensional artificial data sets are first used to analyze the tolerance of the four clustering methods under varying data shape, density and nonlinearity. Real test data sets with known data labels obtained from public databanks are then used to examine the clustering performance. Finally, the four clustering methods are applied to color image segmentation to visualize the clustering results. For each test data set, the same initial weight values of w_{ic} and the same termination criterion are applied to all four clustering methods. The bandwidth values of h for the kernel-based clustering methods are chosen according to Eq. (10). In the experiments, the exponent p is set to a fixed value of 2 for all test data sets but the one in Fig. 5. The complex data set in Fig. 5 uses 1.5 for the exponent p.

3.1. Simulated data

Artificial data are created in a two-dimensional plane so that the true data points in each cluster and clustering results can be visually observed and verified. The effect of changes in the distance between two clusters is first evaluated. A circular cluster and an elliptic cluster are simulated, as seen in Fig. 1. The distance between these two clusters is defined as

$$R_{distance} = \frac{D}{r_A + \lambda_B}$$

where D is the distance between the centroids of the two clusters. r_A is the radius of the circular cluster and λ_B is the semiminor-axis of the elliptic cluster. Both r_A and λ_B are fixed (with $r_A=14.37$ and $\lambda_B=9.97$), and D varies as the clusters move toward each other. When the distance ratio $R_{distance}$ is less than 1, the two clusters overlap. A clustering algorithm should work effectively with an $R_{distance}$ value as small as possible. The clustering results in Fig. 1 reveal that the conventional FCM only works well when the two clusters are distinctly apart from each other with an $R_{distance}=1.99$. The simple FCM-σ algorithm can significantly improve the clustering results in terms of the $R_{distance}$ measure. When $R_{distance}=1.74$, no misassigned data points are present. The two kernel-based clustering can further improve the distance tolerance, and KFCM-σ gives a smallest $R_{distance}$ of 1.25.

In order to evaluate the effect of changes in cluster density, two circular clusters (A and B) of various densities are simulated. Cluster A has 50 data points within a fixed radius of 12. Cluster B has the same number of data points with varying radius larger than 12. The density ratio of the two clusters is defined as

$$R_{density} = \frac{N_B/r_B}{N_A/r_A} = \frac{r_A}{r_B}$$

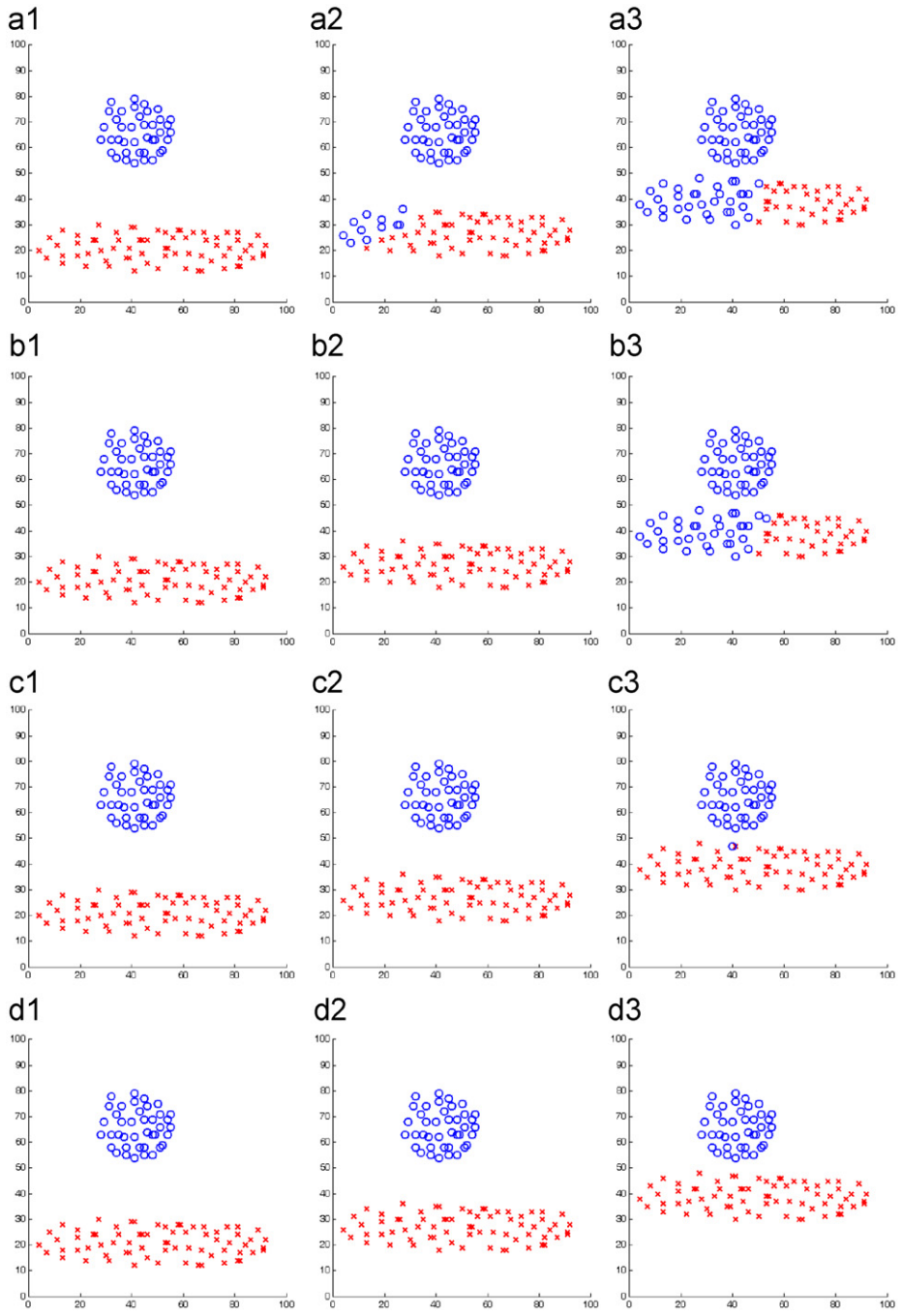

Fig. 1. Effect of changes in cluster distance: (a1)–(a3) clustering results from FCM for $R_{distance}=1.99$, 1.74 and 1.25, respectively; (b1)–(b3) respective clustering results from FCM-σ; (c1)–(c3) respective clustering results from KFCM; (d1)–(d3) respective clustering results from KFCM-σ. (a1) $D=48$ ($R_{distance}=1.99$) FCM (a2) $D=42$ ($R_{distance}=1.74$) FCM (a3) $D=30$ ($R_{distance}=1.25$) FCM (b1) $D=48$ ($R_{distance}=1.99$) FCM-σ (b2) $D=42$ ($R_{distance}=1.74$) FCM-σ (b3) $D=30$ ($R_{distance}=1.25$) FCM-σ (c1) $D=48$ ($R_{distance}=1.99$) KFCM (c3) $D=42$ ($R_{distance}=1.74$) KFCM (c3) $D=30$ ($R_{distance}=1.25$) KFCM (d1) $D=48$ ($R_{distance}=1.99$) KFCM-σ (d2) $D=42$ ($R_{distance}=1.74$) KFCM-σ (d3) $D=30$ ($R_{distance}=1.25$) KFCM-σ.

where N_A and N_B are the numbers of data points in clusters A and B ($N_A=N_B=50$ in the experiment); r_A=a fixed radius of 12 for cluster A, and r_B the variable radius of circular cluster B. A small density ratio $R_{density}$ indicates a sparse distribution.

A good clustering algorithm should be tolerant to a small density ratio. Fig. 2 displays the clustering results of the four clustering algorithms for the simulated data sets with decreasing $R_{density}$ values from 1.00 to 0.10. When both circular clusters have similar data densities and radii, the conventional FCM can partition the data points correctly. The proposed simple FCM-σ algorithm can significantly reduce the misassigned members even if the data points are sparsely distributed. Both kernel-based clustering algorithms further improve the clustering results. The proposed KFCM-σ gives the best partition. There is no misassigned data point even when the density ratio $R_{density}$ is as small as 0.11.

To further evaluate the effect of non-circular data clusters, a circular cluster of fixed radius and an elliptic cluster with varying semimajor-axis are simulated. The roundness measure of the non-circular cluster is defined as

$$R_{roundness}=\frac{\lambda_2}{\lambda_1}$$

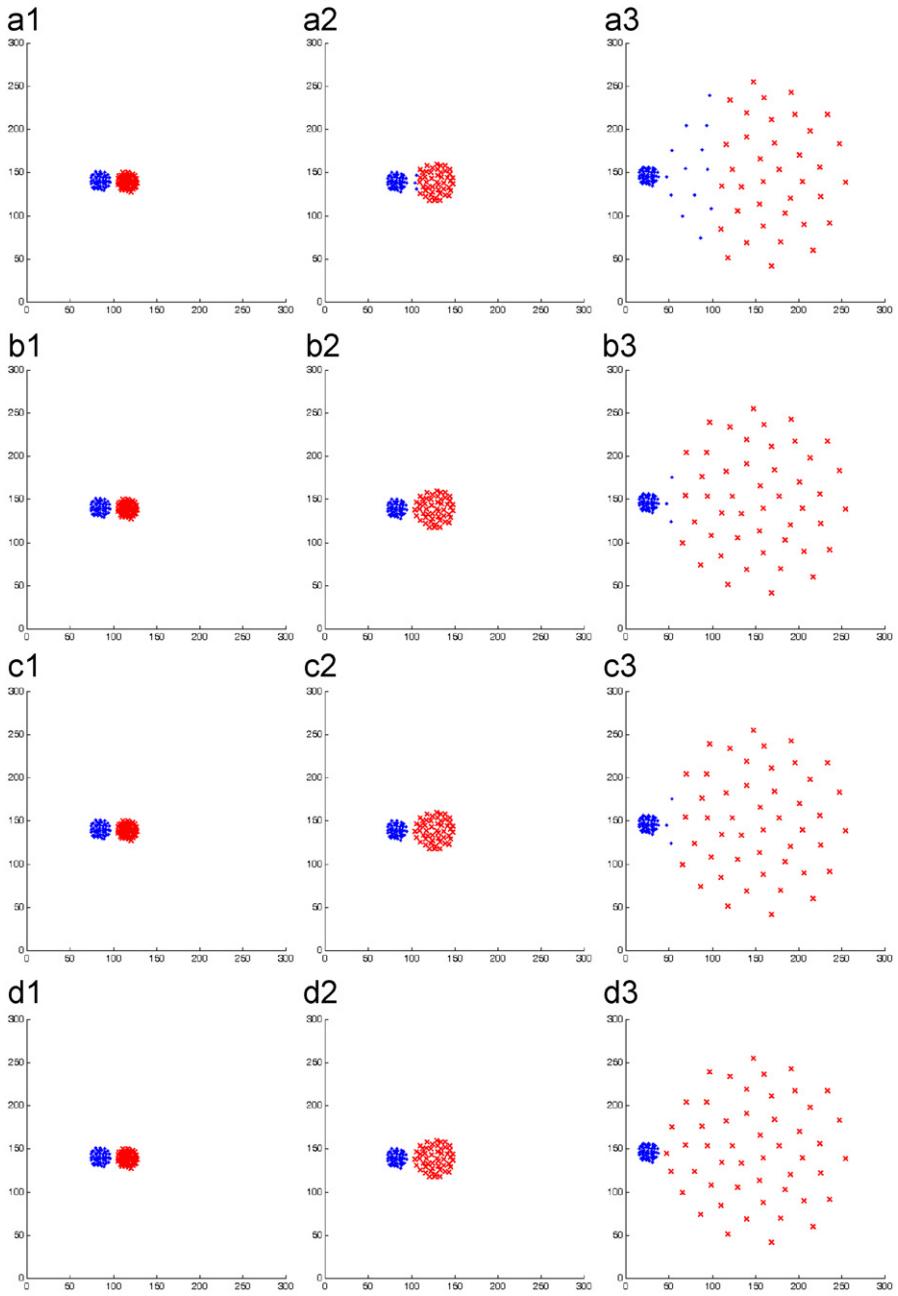

Fig. 2. Effect of changes in cluster density: clustering results of data sets with density ratio $R_{density}=1.00$, 0.48 and 0.11 from (a1)–(a3) FCM; (b1)–(b3) FCM-σ; (c1)–(c3) KFCM; (d1)–(d3) KFCM-σ. (a1)$r_B=12$ ($R_{density}=1.00$) FCM (a2)$r_B=25$ ($R_{density}=0.48$) FCM (a3)$r_B=113$ ($R_{density}=0.11$) FCM (b1) $r_B=12$ ($R_{density}=1.00$) FCM-σ (b2) $r_B=25$ ($R_{density}=0.48$) FCM-σ (b3) $r_B=113$ ($R_{density}=0.11$) FCM-σ (c1) $r_B=12$ ($R_{density}=1.00$) KFCM (c2) $r_B=25$ ($R_{density}=0.48$) KFCM (c3) $r_B=113$ ($R_{density}=0.11$) KFCM (d1) $r_B=12$ ($R_{density}=1.00$) KFCM-σ (d2) $r_B=25$ ($R_{density}=0.48$) KFCM-σ (d3) $r_B=113$ ($R_{density}=0.11$) KFCM-σ.

where λ_1 and λ_2 are the semimajor- and semiminor-axis of the elliptic cluster. When the cluster is circularly distributed, a unity of $R_{roundness}$ is obtained. A good clustering algorithm should be highly tolerant to a small roundness ratio.

Fig. 3 shows the clustering results of the four clustering algorithms, where the conventional FCM generates one misassigned member when the $R_{roundness}$ ratio is 1/3. The proposed simple FCM-σ can generate a reliable partition for $R_{roundness}$ ratio down to 1/4. The KFCM algorithm results in two misassigned members for $R_{roundness}=1/8$, and the proposed KFCM-σ can correctly partition all data points with $R_{roundness}$ as small as 1/8.

Finally, Fig. 4 shows three linearly non-separable data sets used for evaluating nonlinear separability of the four clustering methods. The data set in the first column of Fig. 4 consists of 2 clusters, an inner core and an outer ring. The data set in the second column consists of a disk and a half-ring that forms a sun–moon shape. The data set in the third column shows the interlacing of two half-rings. It clearly shows that both the FCM and FCM-σ in the observed data space separate each data set into two regions by a linear discriminant function. The KFCM and KFCM-σ with the proposed bandwidth setting can correctly partition the core-ring and sun–moon data sets by mapping the observed data to the higher dimensional feature space. However, both kernel-based clustering methods fail to group the data set with the interlacing of two half-rings, no matter what the bandwidth value is used. A polynomial or a sigmoid kernel also fails to cluster such a data set. It is apparent from the clustering results in Fig. 4 that the kernel-based clustering methods can only

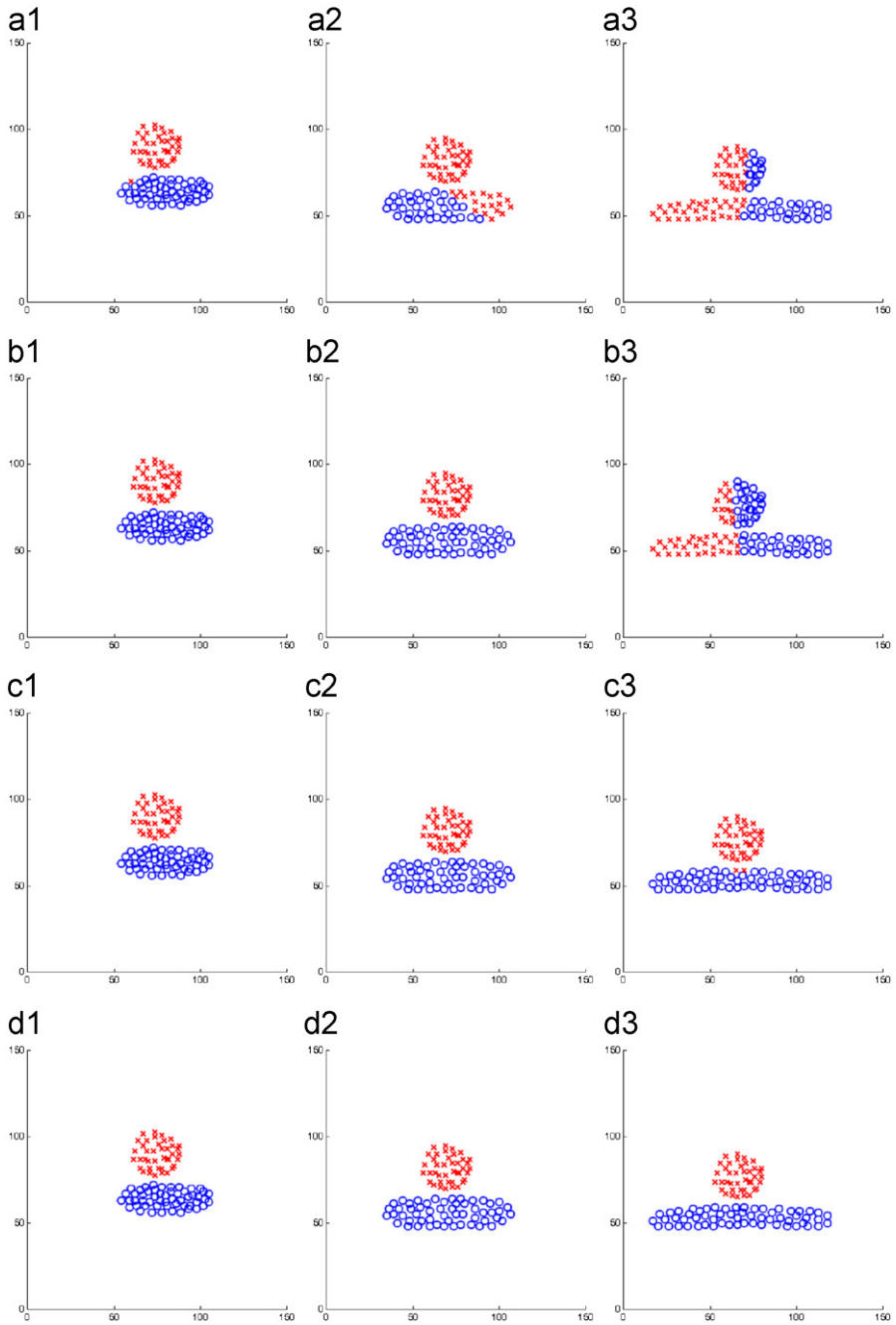

Fig. 3. Effect of changes in cluster shape: clustering results of data sets with roundness ratio $R_{roundness}=1/3$, 1/4 and 1/8 from (a1)–(a3) FCM; (b1)–(b3) FCM-σ; (c1)–(c3) KFCM; (d1)–(d3) KFCM-σ. (a1) $R_{roundness}=1/3$ FCM, (a2) $R_{roundness}=1/4$ FCM, (a3) $R_{roundness}=1/8$ FCM, (b1) $R_{roundness}=1/3$ FCM-σ, (b2) $R_{roundness}=1/4$ FCM-σ, (b3) $R_{roundness}=1/8$ FCM-σ, (c1) $R_{roundness}=1/3$ KFCM, (c2) $R_{roundness}=1/4$ KFCM, (c3) $R_{roundness}=1/8$ KFCM, (d1) $R_{roundness}=1/3$ KFCM-σ, (d2) $R_{roundness}=1/4$ KFCM-σ, (d3) $R_{roundness}=1/8$ KFCM-σ.

generate quadratic discriminant functions to separate the data sets and, therefore, work successfully for the core-ring and sun–moon data sets. The data set with the interlacing of two half-rings requires a cubic polynomial to separate these two half-ring clusters.

Fig. 5 further demonstrate the clustering results of the two kernel-based clustering methods for a 3-class data set. Fig. 5 shows the data set that contains two cores within a circular ring. The KFCM and KFCM-σ algorithms with the proposed bandwidth setting rule can also reliably partition the data set into three correct clusters, as seen in Fig. 5(c) and (d), respectively. We have also evaluated numerous linearly non-separable data sets of different shapes and densities in the experiments. The proposed kernel-based clustering approach can well partition all the data sets as long as the clusters in a data set can be separated by quadratic functions.

3.2. Real data sets

Four real data sets, Astroparticle, Splice, Statlog (Australian credit approval) and Iris, from public data banks are used to evaluate the performance of the four clustering algorithms. The data sets of Astroparticle and Splice come from LIBSVM (A Library for Support Vector Machine). Statlog/Australian and Iris are obtained from UCI (UC Irvine Machine Learning Repository).

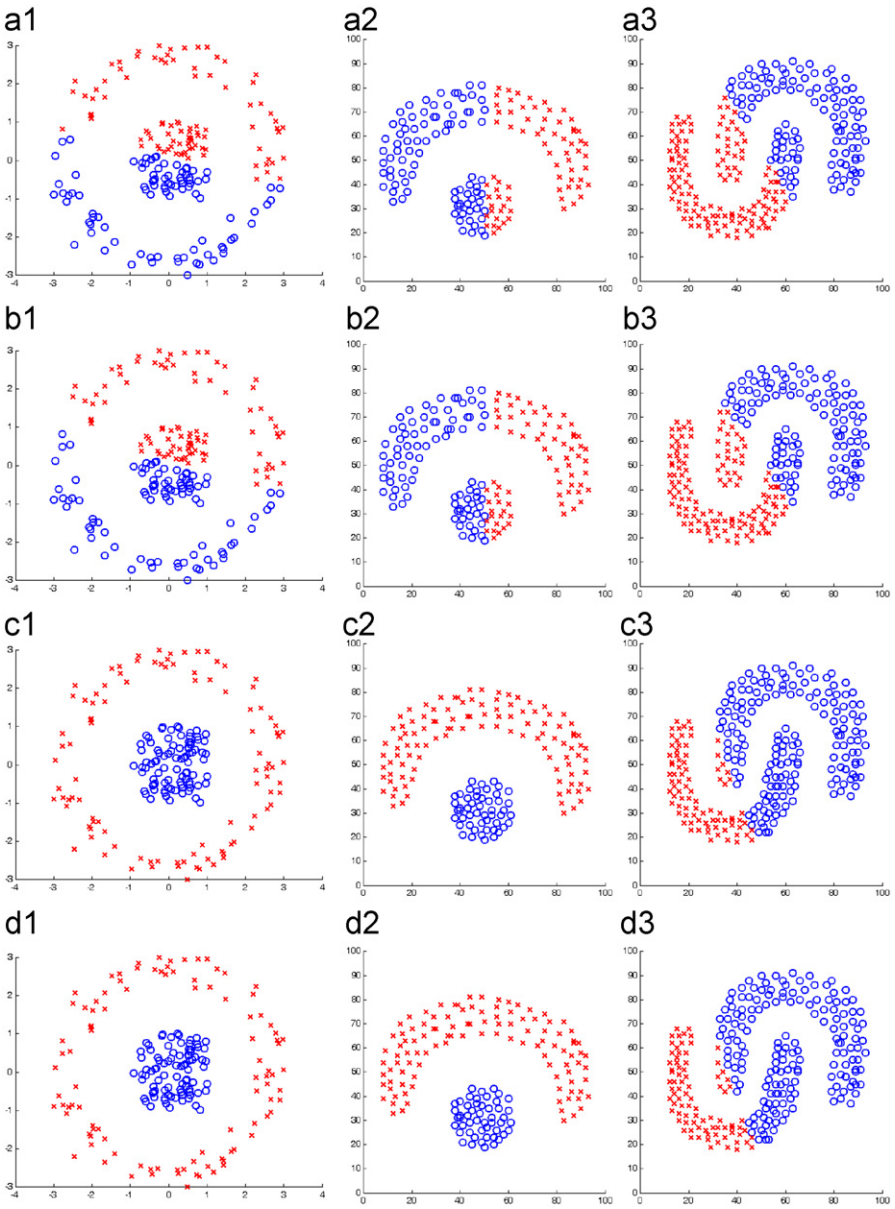

Fig. 4. Clustering results of linearly non-separable data sets from (a1)–(a3) FCM; (b1)–(b3) FCM-σ; (c1)–(c3) KFCM; (d1)–(d3) KFCM-σ. (a1) FCM, (a2) FCM, (a3) FCM, (b1) FCM-σ, (b2) FCM-σ, (b3) FCM-σ, (c1) KFCM, (c2) KFCM, (c3) KFCM, (d1) KFCM-σ, (d2) KFCM-σ, (d3) KFCM-σ.

The Astroparticle data set is a classification application in astroparticle physics. It involves two classes and 4 features. The Statlog/Australian data set concerns credit card applications. The credit card is approved or declined based on 14 features of the applicant. The purpose of the Splice data set is to recognize two classes of splice junctions in a DNA sequence, where each class is represented by a high dimensionality of 60 features. The Iris data set contains 3 classes (Setosa, Versicolor and Virginica) of iris plant. Each type of iris plant is described by 4 features. The detailed data characteristics of the four data sets are listed in Table 1. All the parameter settings are the same as those used for the artificial data sets in Section 3.1. The original feature values are directly used for clustering without data normalization. The proposed bandwidth setting rule is used for individual data sets.

Each of the four clustering algorithms is replicated 30 times with different random initial weights w_{ic} for each data set. In each replication, all four clustering algorithms use the same initial weight values so that the effect of initialization can be eliminated. Table 2 summarizes the means and standard deviations (S.D.) of the recognition rates in percentage of the four data sets from the four clustering algorithms. Based on the 30 replications of each data set, the resulting standard deviation of the recognition rates is very small, ranging from 0% to 3.9% for the proposed FCM-σ and KFCM-σ methods. The test results also reveal the proposed FCM-σ that directly works on the observed data space can effectively improve the recognition rates up to 10% for the Astroparticle and Statlog data sets, compared to those of the conventional FCM. The FCM-σ and FCM methods show similar clustering results for the Splice and Iris data sets.

The proposed KFCM-σ algorithm that works on the mapped feature space is far superior to the conventional FCM. By analyzing the 95% confidence interval for the difference of recognition rates between KFCM-σ and FCM, the improvement is 21.8% for Astroparticle, 12.0% for Statlog, 0.45–3.3% for Splice and 4.0% for

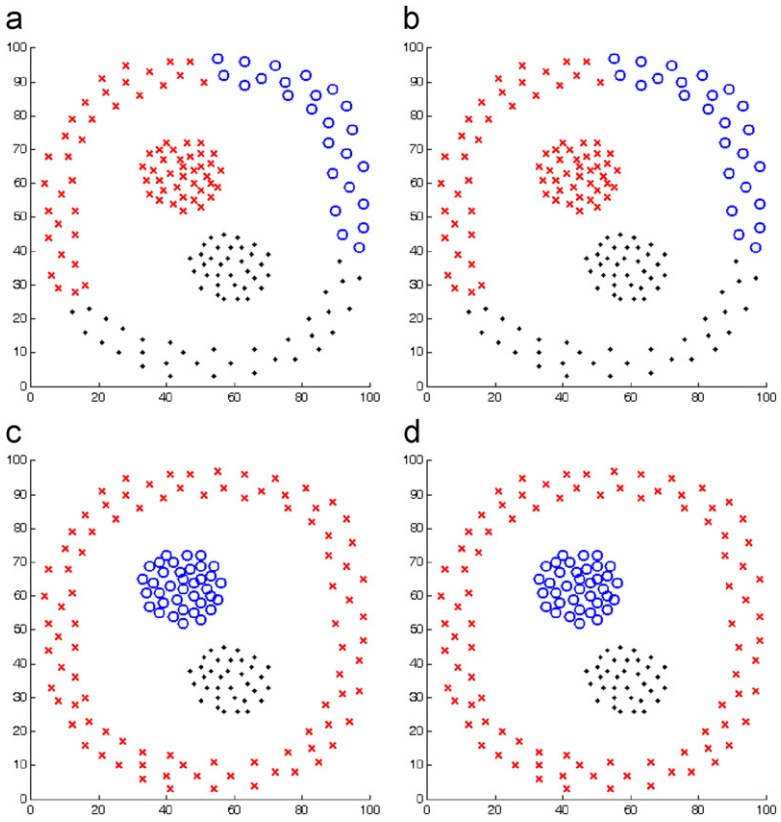

Fig. 5. Clustering results from (a) FCM, (b) FCM-σ, (c) KFCM, and (d) KFCM-σ for a 3-class data set. The kernel parameter value is given by the bandwidth setting rule. (a) FCM, (b) FCM-σ, (c) KFCM, (d) KFCM-σ.

Table 1
Data characteristics of real data sets.

Data set	Number of clusters	Number of features	Number of data points (samples in each cluster)	Description	Source
Astroparticle	2	4	3089 (2000/1089)	Astronomical application	LIBSVM
Statlog (Australian credit approval)	2	14	690 (307/383)	Credit card approval	UCI
Splice	2	60	1000 (517/483)	Splice junctions in DNA sequence	LIBSVM
Iris	3	4	150 (50/50/50)	Iris plant	UCI

Table 2
Means and standard deviations (S.D.) of the recognition rates[a] for the four real data sets from the four clustering methods.

Data set	Clustering method							
	FCM		FCM-σ		KFCM		KFCM-σ	
	Mean (%)	S.D. (%)	Mean (%)	S.D. (%)	Mean (%)	S.D. (%)	Mean (%)	S.D. (%)
Astroparticle	66.0	0	75.4	0	85.8	0	88.0	0
Statlog (Australian credit approval)	56.1	0	66.1	0	62.6	0	68.1	0
Splice	63.3	1.8	65.3	3.5	63.5	4.0	65.2	3.9
Iris	89.3	0	89.3	0	92.0	0	93.3	0

[a] Based on 30 replications.

Iris. The improvement of the proposed KFCM-σ with respect to KFCM is less significant. It gives 2.2% for Astroparticle, 5.5% for Statlog, −0.7–1.27% for Splice and 1.3% for Iris.

In order to evaluate the effectiveness of the bandwidth selection rule proposed in the paper, we have also tested the four real data sets (Astroparticle, Statlog/Australian, Splice and Iris) by an exhaustive search of all possible bandwidth values with a very small resolution in a wide range and found the one with maximum recognition rates based on the known classes of all data points in the data set. The proposed simple bandwidth selection rule is also compared with the bandwidth setting rule introduced by Zelnik-Manor and Perona [10] for the KFCM-σ clustering. For the bandwidth that is adaptively determined by the distance of a point to its Kth nearest neighbor in the data set, Zelnik-Manor and Perona [10] used $K=7$ in their paper. Von Luxburg [11] suggested the Kth neighbor in the order of $\log(N)+1$, where N is the total

number of data points. Table 3 summarizes the mean recognition rates of 30 replications for the KFCM-σ algorithm with various bandwidth settings. It shows that the proposed bandwidth selection rule outperforms Zelnik-Manor and Perona's bandwidth setting for all four real data sets. The simple bandwidth selection rule also generates good recognition rates very close to those obtained from the exhaustive search for Statlog/Australian, Splice and Iris data sets. For Astroparticle data set, the difference is only 4.7%. The standard deviation of the recognition rates from Zelnik-Manor and Perona's bandwidth setting varies between 3.6% and 9.9%, whereas the proposed bandwidth selection rule results in small standard deviations between 0% and 3.9%.

3.3. Color image segmentation

The performance of the four clustering algorithms on color image segmentation is also evaluated. The original RGB tristimulus values are used as the feature vector for each pixel in the image. Parameter settings are the same as those defined in Section 3.1. The color images used for testing are of size 100×100 pixels. Each test image thus involves 10,000 data points in the data set.

Fig. 6(a) shows the color image of a blouse with a shade of claret. Random background noise appears in the vicinity of the blouse. Fig. 6(b) depicts the RGB distribution in 3D perspective. No visible clusters are shown in the plot. Fig. 6(c)–(f) shows the two-cluster segmentation results of the four clustering algorithms, FCM, FCM-σ, KFCM and KFCM-σ, respectively. The colors shown in the segmented images are based on the mean RGB values of the assigned data points in each cluster. The conventional FCM segments the blouse from the background with a few noisy blobs. The remaining three clustering algorithms well segment the blouse with accurate shape and present no noise. Fig. 7(a) further demonstrates a color image that contains 8 ellipse-shaped objects of analogous colors. Fig. 7(b) displays the RGB distribution in the 3D color space. Fig. 7(c)–(f) presents the two-cluster segmentation results of the four clustering algorithms. The conventional FCM detects only 6 out of the 8 ellipses and generates some noise in the segmented objects. The FCM-σ method detects 7 ellipses and misses the last one. The two kernel-based clustering algorithms perform equally well for segmenting the 8 color-shaded ellipses.

Table 3
Effect of bandwidth setting on recognition rates for the KFCM-σ clustering.

Data set	Bandwidth setting			
	Proposed h (%)	Zelnik-Manor and Perona [10] (%)		Exhaustive search (%)
		$K=7$	$K=1+\log N$	
Astroparticle	88.0	58.3	61.4	92.7
Statlog (Australian credit approval)	68.1	53.8	60.5	68.4
Splice	65.2	61.5	58.9	66.4
Iris	93.3	71.3	76.1	93.3

Table 4 lists the computation times of the two color images from the four clustering algorithms. The termination criterion of the E–M procedure is tightly given by $|J^t - J^{t-1}| \le 10^{-10}$, where J^t is the objective value at iteration t, for all comparative methods. The clustering algorithms were implemented on a Pentium Core2 Duo, 3.0 GHz personal computer. The results indicate that the conventional FCM and the proposed FCM-σ are computationally very fast. The computation times of the two kernel-based algorithms are exponentially increased for an image of 10,000 pixel points. For the implementation of kernel-based clustering algorithms, the kernel function K_{ij} should be pre-calculated and stored in a look-up-table with i (data point $\boldsymbol{x}_i$) and j (data point x_j) as row and column addresses, and then the calculation of K_{ij} can be eliminated in every iteration of the E–M procedure.

The experiments above have revealed that the proposed FCM with the new distance metric can significantly improve the clustering, compared to the conventional FCM. The proposed KFCM can further improve the clustering and achieve nonlinear separation of clusters. In terms of the computational efficiency, the computational complexity of FCM and the proposed FCM-σ is only in the order of $N \cdot C \cdot P$, where N is the total number of data points, C is the number of clusters and P is the number of dimensions of the feature vector. Since the kernel-based clustering algorithms need evaluate the kernel values between all possible pairs of two data points, the computational complexity of KFCM and KFCM-σ is in the order of $N^2 \cdot C \cdot P$. The computational load is very high when N is distinctly large, especially when the kernel methods are applied to image segmentation.

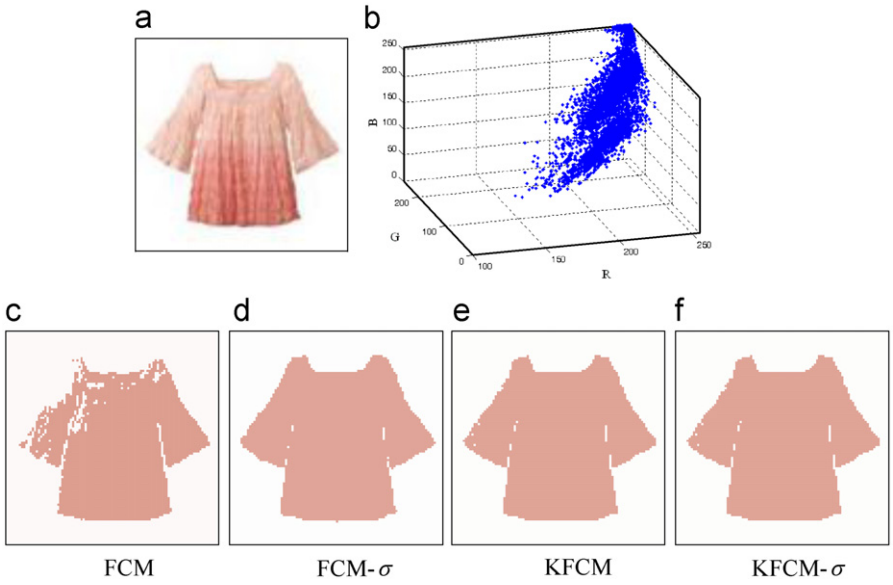

Fig. 6. (a) Blouse image; (b) RGB distribution in 3D perspective; (c)–(f) segmentation results from FCM, FCM-σ, KFCM, and KFCM-σ, respectively.

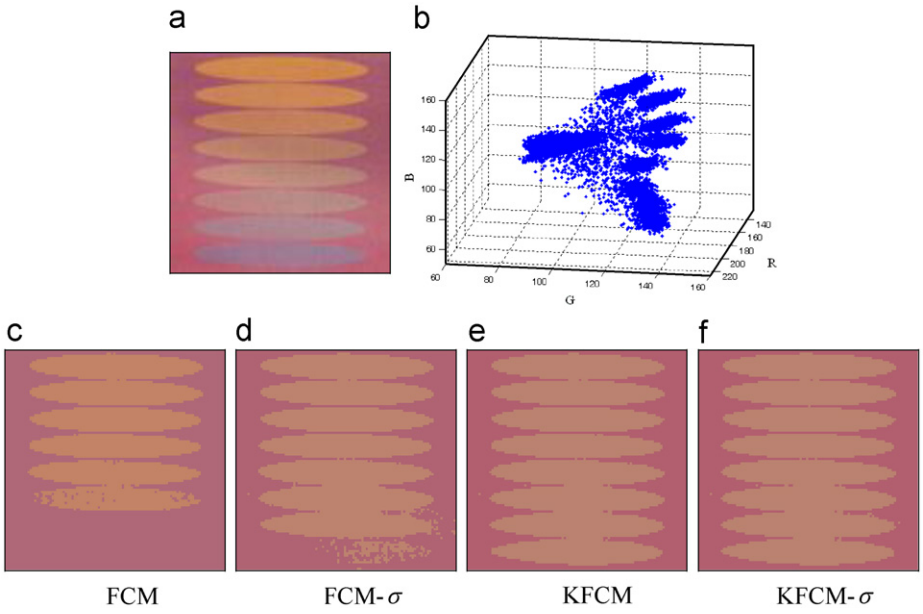

Fig. 7. (a) Image of 8 ellipses with analogous colors; (b) RGB distribution in 3D perspective; (c)–(f) segmentation results from FCM, FCM-σ, KFCM, and KFCM-σ, respectively.

Table 4
Computation times (s) of 100×100 images for color segmentation.

Color image	Clustering method			
	FCM	FCM-σ	KFCM	KFCM-σ
Blouse image Fig. 6(a)	0.4	0.6	463.5	508.3
8-ellipses image Fig. 7(a)	0.8	1.1	278.7	290.8

4. Conclusions

This paper has presented a new distance metric that incorporates the distance variation of data points in each cluster to FCM in the observed data space and KFCM in the mapped feature space. The proposed FCM with the new distance metric can significantly improve the clustering effectiveness with the same computational load, compared to the conventional FCM. The proposed KFCM with the new distance metric slightly outperforms the KFCM in feature space. The simple bandwidth setting rule for the two RBF kernel-based clustering methods is computationally fast. It works well for the 2D artificial data sets with non-circular distribution and uneven density. It also performs extremely well for linearly non-separable data. The proposed kernel FCM is far superior to the conventional FCM and shows a slight improvement over the KFCM on the real data sets in public data libraries. Finding bandwidth selection rules/criteria that have a theoretical justification is important for future research.

The experiments on the 2D artificial data sets have shown that the kernel-based clustering methods can well partition nonlinearly distributed data, but only up to quadratic functions. It is worthy of further investigation for extending the kernel-based clustering to a higher polynomial function. A good computational strategy is also required in the future to use the kernel-based clustering for data sets with a huge number of data points, especially for color segmentation in a very large image.

References

[1] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithm, Plenum Press, New York, 1981.
[2] K.-L. Wu, M.-S. Yang, Alternative *C*-means clustering algorithms, Pattern Recognition 35 (2002) 2267–2278.
[3] D.-Q. Zhang, S.-C. Chen, Clustering incomplete data using kernel-based fuzzy *C*-means algorithm, Neural Processing Letters 18 (2003) 155–162.
[4] D.-Q. Zhang, S.-C. Chen, A novel kernelized fuzzy *C*-means algorithm with application in medical image segmentation, Artificial Intelligence in Medicine 32 (2004) 37–50.
[5] X.Z. Wang, Y.D. Wang, L.J. Wang, Improving fuzzy *C*-means clustering based on feature-weight learning, Pattern Recognition Letters 25 (2004) 1123–1132.
[6] W.-L. Hung, M.-S. Yang, D.-H. Chen, Bootstrapping approach to feature-weight selection in fuzzy *C*-means algorithms with an application in color image segmentation, Pattern Recognition Letters 29 (2008) 1317–1325.
[7] M. Girolami, Mercer kernel-based clustering in feature space, IEEE Transactions on Neural Networks 13 (2002) 780–784.
[8] D.-W. Kim, K. Lee, D. Lee, K.H. Lee, Evaluation of the performance of clustering algorithms in kernel-induced feature space, Pattern Recognition 38 (2005) 607–611.
[9] A.Y. Ng., M.I. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm, in: T.G. Dietterich, S. Becker, Z. Ghahramani (Eds.), Advances in Neural Information Processing Systems 14, MIT Press, Cambridge, MA, , 2002.
[10] L. Zelnik-Manor, P. Perona, Self-tuning spectral clustering, Advances in Neural Information Processing Systems 17 (2004) 1601–1608.
[11] U. von Luxburg, A tutorial on spectral clustering, Statistics and Computing 17 (2007) 395–416.
[12] A.L.N. Fred, A.K. Jain, Combining multiple clusterings using evidence accumulation, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (2005) 835–850.
[13] B. Scholkopf, A.J. Smola, Learning with Kernels, Cambridge, MIA Press, MA, 2002.
[14] B. Scholkopf, A. Smola, K.R. Muller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Computation 10 (1998) 1299–1319.

Du-Ming Tsai received the B.S. degree in Industrial Engineering from the Tunghai University, Taiwan in 1981, and the M.S. and Ph.D. degrees in Industrial Engineering from Iowa State University, Ames, Iowa in 1984 and 1987, respectively. From 1988 to 1990, he was a Principal Engineer of Digital Equipment Corporation, Taiwan branch, where his work focused on process and automation research and development. Currently he is a Professor of Industrial Engineering and Management at the Yuan-Ze University, Taiwan. His research interests include automated visual inspection, object recognition and texture analysis.

Chung-Chan Lin received the B.S. degree in Industrial Engineering and Systems Management from the Feng Chia University, Taiwan in 2008 and the M.S. degrees in Industrial Engineering and Management from the Yuan-Ze University in 2010.