Applied Soft Computing 10 (2010) 381–389

Contents lists available at ScienceDirect

Applied Soft Computing

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

A new Kernelized hybrid c-mean clustering model with optimized parameters

Meena Tushir [a], Smriti Srivastava [b,*]

[a] Department of Electrical and Electronics Engg., MSIT, New Delhi, India
[b] Department of Instrumentation and Control Engg., NSIT, Sector-3, Dwarka, New Delhi 110075, India

ARTICLE INFO

Article history:
Received 19 June 2008
Received in revised form 22 April 2009
Accepted 2 August 2009
Available online 26 August 2009

Keywords:
Fuzzy clustering
Hybrid clustering
Possibilistic clustering
Kernel method
TS modeling

ABSTRACT

A possibilistic approach was initially proposed for c-means clustering. Although the possibilistic approach is sound, this algorithm tends to find identical clusters. To overcome this shortcoming, a possibilistic Fuzzy c-means algorithm (PFCM) was proposed which produced memberships and possibilities simultaneously, along with the cluster centers. PFCM addresses the noise sensitivity defect of Fuzzy c-means (FCM) and overcomes the coincident cluster problem of possibilistic c-means (PCM). Here we propose a new model called Kernel-based hybrid c-means clustering (KPFCM) where PFCM is extended by adopting a Kernel induced metric in the data space to replace the original Euclidean norm metric. Use of Kernel function makes it possible to cluster data that is linearly non-separable in the original space into homogeneous groups in the transformed high dimensional space. From our experiments, we found that different Kernels with different Kernel widths lead to different clustering results. Thus a key point is to choose an appropriate Kernel width. We have also proposed a simple approach to determine the appropriate values for the Kernel width. The performance of the proposed method has been extensively compared with a few state of the art clustering techniques over a test suit of several artificial and real life data sets. Based on computer simulations, we have shown that our model gives better results than the previous models.

1. Introduction

Clustering [1,2] plays an important role in many engineering fields such as pattern recognition, system and modeling, image analysis, communication, data mining and so on. Clustering methods divide a set of N observations (input vectors) x_1, $x_2, \ldots, x_N$ into c groups $\beta_1, \beta_2, \ldots, \beta_c$ so that members of the same group are more similar to one another than to members of other groups. The number of clusters may be predefined or it may be determined by the method.

Usually the clustering methods assume that each data vector belongs to one and only one class. This approach can be natural for clustering compact and well-separated groups of data. However, frequently clusters overlap and some data vectors belong partially to several clusters. The Fuzzy set theory [1] is a natural way to describe this situation. In this case, a membership degree of a vector x_k to the ith cluster (u_{ik}) is a value from the interval [0,1]. This idea was first introduced by Ruspini [3] and used by Dunn [4] to construct a Fuzzy clustering method based on the criterion function minimization. Bezdek generalized this approach to an infinite family of Fuzzy c-means algorithm using a weighted exponent on the Fuzzy memberships [5,6].

Prototype-based clustering algorithms such as the FCM algorithm minimize the objective function

$$J(X;V,U)=\sum_{i=1}^{c}\sum_{k=1}^{N}(u_{ik})^m d^2(x_k,v_i) \quad \text{Subject to } \sum_{i=1}^{c}u_{ik}=1 \;\; \forall \;\; k \tag{1}$$

where $X=\{x_1, x_2, \ldots, x_N\}$ is a set of vectors in an n-dimensional feature space, $V=(v_1, v_2, \ldots, v_c)$ is a c-tuple of prototypes, $d^2(x_k,v_i)$ is the distance of feature vectors x_k to prototype v_i, N is the total number of feature vectors, C is the number of clusters, u_{ik} is the grade of membership of feature point x_k in cluster v_i and $m \in [1,\alpha)$ is a weighting exponent called fuzzifier.

The updating functions for v_i and u_{ik} are obtained as follows,

$$v_i=\frac{\sum_{k=1}^{N}u_{ik}^m x_k}{\sum_{k=1}^{N}u_{ik}^m}, \; \forall i=1,\ldots,c \tag{2}$$

$$u_{ik}=\frac{1}{\sum_{j=1}^{c}(d_{ij}/d_{kj})^{2/(m-1)}}, \; \forall i=1,\ldots c, \quad \forall k=1,\ldots,N \tag{3}$$

These algorithms usually use a fixed-point iteration scheme to find the solution of the minimization problem. This particular

* Corresponding author. Tel.: +91 11 25500381; fax: +91 11 25099022.
E-mail address: ssmriti@yahoo.com (S. Srivastava).

1568-4946/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.asoc.2009.08.020

scheme makes the FCM type algorithms susceptible to local minima. The issue of global convergence may be addressed by the use of multiple initializations at the cost of increased computations. However, noise in the data sets can exacerbate the situation by creating many spurious minima. Noise can also drastically distort the solution corresponding to global minimum, and there is no way to handle this problem in FCM formulation. One way to reduce the influence of noise points can be if the memberships associated with them are small in all clusters. However due to constraint on U matrix, noise points and outliers will have significantly high membership value and they can severely affect the prototype parameter estimate. This drawback has motivated the researchers to seek alternative formulations. There are a number of useful approaches for controlling the harmful effects of outlying data, including the possibilistic clustering (PCM) approach of Krishnapuram and Keller [8] and Fuzzy noise clustering approach of Dave [7].

The possibilistic approach determines a possibilistic partition, in which a possibilistic membership measures the absolute degree of typicality of a point in a cluster. The PCM algorithm sometimes helps to identify outliers (noise points). Although this approach is sound, but is very sensitive to initializations and sometimes generates coincident clusters. To overcome the problem of identical clusters, some new algorithms have been proposed [9,10] that generate both membership and typicality values when clustering unlabeled data. However it is observed that these algorithms tend to give not so good results for unequal sized clusters. To improve these algorithms, we propose a new Kernel-based hybrid c-means (KPFCM) clustering model [14], which adopts a Kernel induced metric in the data space to replace the original Euclidean norm metric. By replacing the inner product with an appropriate 'Kernel' function, one can implicitly perform a non-linear mapping to a high dimensional feature space in which the data is more clearly separable, thus as shown in this paper, the proposed method is characterized by higher clustering accuracy than the original possibilistic Fuzzy c-means clustering (PFCM) method. Several Kernel-based learning methods for example, support vector machine (SVM), have recently shown remarkable performance in supervised learning [11–13].

The second point that can be raised about this method is the choice of the Kernel width. Choosing appropriate values for Kernel width is one of the key problems in many Kernel-based methods because the values of these parameters have significant impact on the performance. Cross-validation and leave-one-out techniques are generally used in the literature to determine these parameters. Here we propose a simple approach to learn the Kernel width in KPFCM. We have used the same objective function as used for clustering algorithm for simplicity. Then the optimal values of the Kernel width are chosen through optimizing the above-defined objective function.

The remainder of this paper is organized as follows. Section 2 provides background information on the possibilistic Fuzzy c-means clustering. In Section 3, the proposed Kernel-based hybrid c-means clustering model is formulated. Section 4 highlights the potential of the proposed approach through various artificial and real data sets. Concluding remarks are presented in Section 5.

2. Previous works

2.1. Possibilistic c-means clustering

To overcome the noise sensitivity defect of FCM algorithm, Krishnapuram and Keller relaxed the column sum constraint $\sum_{i=1}^{c} u_{ik} = 1 \,\forall\, k$ and proposed a possibilistic approach to clustering (PCM) by minimizing the following objective function,

$$J_{\mathrm{PCM}}(U,V) = \sum_{i=1}^{c}\sum_{k=1}^{N} u_{ik}^{m} d_{ik}^{2} + \sum_{i=1}^{c} \gamma_i \sum_{k=1}^{N} (1-u_{ik})^{m} \tag{4}$$

where γ_i are suitable positive numbers. The first term demands that the distance from the data points to the prototypes be as low as possible, whereas the second term forces u_{ik} to be as large as possible, thus avoiding the trivial solution. The updating of prototypes is same as that in FCM, but the memberships of PCM are updated as follows:

$$u_{ik} = \frac{1}{1 + \left(||x_k - v_i||^2/\gamma_i\right)^{1/m-1}} \tag{5}$$

PCM sometimes helps when the data is noisy. If the initialization of each row is not sufficiently distinct, coincident clusters may result.

2.2. Possibilistic Fuzzy c-means clustering

This approach integrates the Fuzzy approach with possibilistic approach so that it has two types of memberships: (1) a possibilistic (t_{ik}) membership that measures the absolute degree of typicality of a point in any particular cluster and (2) a Fuzzy membership (u_{ik}) that measures the relative degree of sharing of a point among the clusters.

This leads to the following optimization problem,

$$J_{\mathrm{PFCM}}(U,V,T) = \sum_{i=1}^{c}\sum_{k=1}^{N} (a u_{ik}^{m} + b t_{ik}^{\eta}) d_{ik}^{2} + \sum_{i=1}^{c} \gamma_i \sum_{k=1}^{N} (1-t_{ik})^{\eta} \tag{6}$$

subject to the constraint $\sum_{i=1}^{c} u_{ik} = 1 \,\forall\, k$ and $0 \le u_{ik}, t_{ik} < 1$. Here $a > 0$, $b > 0$, $m > 1$ and $\eta > 1$. The constants a and b define the relative importance of Fuzzy membership and typicality values in the objective function. The minimization of the objective function gives the following condition:

$$u_{ik} = \frac{1}{\sum_{j=1}^{c} \left(d_{ik}/d_{jk}\right)^{2/m-1}}, \quad 1 \le i \le c; \quad 1 \le k \le N \tag{7}$$

$$t_{ik} = \frac{1}{1 + \left((b/\gamma_i) d_{ik}^{2}\right)^{1/\eta-1}}, \quad 1 \le i \le c \tag{8}$$

$$v_i = \frac{\sum_{k=1}^{N} (a u_{ik}^{m} + b t_{ik}^{\eta}) x_k}{\sum_{k=1}^{N} (a u_{ik}^{m} + b t_{ik}^{\eta})} \tag{9}$$

Though PFCM is found to perform better than FCM and PCM, however, we found that when two highly unequal sized clusters are given, PFCM fails to give the desired results.

3. Kernel-based hybrid c-means clustering method

3.1. Kernel-based approach

The present work proposes a way of increasing the accuracy of the possibilistic Fuzzy c-means algorithm by exploiting a Kernel function in calculating the distance of the data point from the prototypes; mapping the data points from input space to a high dimensional space in which distance is measured using a Kernel function.

A Kernel function is a generalization of the distance metric that measures the distance between two data points as the data points are mapped into a high dimensional space in which they are more clearly separable. By employing a mapping function $\Phi(x)$, which defines a non-linear transformation: $x \to \Phi(x)$, the non-linearly separable data structure existing in the original data space can

possibly be mapped into a linearly separable case in the higher dimensional feature space.

Given an unlabeled data set $X = \{x_1, \ldots, x_N\}$ in the p-dimensional space R^p, let Φ be a non-linear mapping function from this input space to a high dimensional feature space H:

$$\Phi : R^p \to H, \quad x \to \Phi(x)$$

The key notion in Kernel-based learning is that the mapping function Φ need not be explicitly specified; the dot product in the high dimensional feature space can be calculated through the Kernel function $K(x_i, x_j)$ in the input space R^p:

$$K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$$

Consider the following example. For $p = 2$ and a mapping function Φ,

$$\Phi : R^2 \to H = R^3 \quad (x_{i1}, x_{i2}) \to (x_{i1}^2, x_{i2}^2, \sqrt{2}x_{i1}x_{i2})$$

Then the dot product in the feature space H is calculated as

$$\begin{aligned} \Phi(x_i) \cdot \Phi(x_j) &= (x_{i1}^2, x_{i2}^2, \sqrt{2}x_{i1}x_{i2}) \cdot (x_{j1}^2, x_{j2}^2, \sqrt{2}x_{j1}x_{j2}) \\ &= ((x_{i1}, x_{i2}) \cdot (x_{j1}, x_{j2}))^2 \\ &= (x_i \cdot x_j)^2 = K(x_i, x_j) \end{aligned}$$

where K-function is the square of the dot product in the input space. We see from this example that use of the Kernel function makes it possible to calculate the value of the dot product in the feature space H without explicitly calculating the mapping function Φ. Some examples of Kernel function are as follows:

Example1 (Polynomial Kernel): $K(x_i, x_j) = (x_i \cdot x_j + c)^d$ where, $c \geq 0$, $d \in N$

Example2 (Gaussian Kernel): $K(x_i, x_j) = \exp\left(-\frac{||x_i - x_j||^2}{2\sigma^2}\right)$ where, $\sigma > 0$

Example3 (Radial basis Kernel): $K(x_i, x_j) = \exp\left(-\frac{\sum |x_i^a - x_j^a|^b}{\sigma^2}\right) (0 < b \leq 2)$

Note that RBF function with $a = 1$, $b = 2$, reduces to commonly used Gaussian function.

Example4 (Hyper tangent): $K(x_i, x_j) = 1 - \tanh\left(-\frac{||x_i - x_j||^2}{\sigma^2}\right)$

3.2. Formulation

Our proposed model called Kernel-based hybrid c-means clustering (KPFCM) [14] adopts a Kernel induced metric different from the Euclidean norm in original PFCM. KPFCM minimizes the following objective function:

$$J_{\text{KPFCM}}(U, V, T) = \sum_{k=1}^{N}\sum_{i=1}^{c}(au_{ik}^m + bt_{ik}^\eta)||\Phi(x_k) - \Phi(v_i)||^2 + \sum_{i=1}^{c}\gamma_i\sum_{k=1}^{N}(1 - t_{ik})^\eta \quad (10)$$

where $||\Phi(x_k) - \Phi(v_i)||^2$ is the square of distance between $\Phi(x_k)$ and $\Phi(v_i)$. The distance in the feature space is calculated through the Kernel in the input space as follows:

$$\begin{aligned} ||\Phi(x_k) - \Phi(v_i)||^2 &= (\Phi(x_k) - \Phi(v_i)) \cdot (\Phi(x_k) - \Phi(v_i)) \\ &= \Phi(x_k) \cdot \Phi(x_k) - 2\Phi(x_k)\Phi(v_i) + \Phi(v_i) \cdot \Phi(v_i) \\ &= K(x_k, x_k) - 2K(x_k, v_i) + K(v_i, v_i) \end{aligned}$$

If we adopt the Gaussian function as a Kernel function, i.e. $K(x, y) = \exp(-||x - y||^2/2\sigma^2)$, where σ defined as Kernel width, is a positive number, then $K(x, x) = 1$. Thus Eq. (10) can be written as

$$J_{\text{KPFCM}}(U, V, T) = 2\sum_{k=1}^{N}\sum_{i=1}^{c}(au_{ik}^m + bt_{ik}^\eta)(1 - K(x_k, v_i)) + \sum_{i=1}^{c}\gamma_i\sum_{k=1}^{N}(1 - t_{ik})^\eta \quad (11)$$

Given a set of points X, we minimize $J_{\text{KPFCM}}(U, V, T)$ in order to determine U, V, T. We adopt an alternating optimization approach to minimize $J_{\text{KPFCM}}(U, V, T)$ and need the following theorem:

Theorem 1. The necessary conditions for minimizing J_{KPFCM} under the constraint of U are

$$u_{ik} = \frac{(1/(1 - K(x_k, v_i)))^{1/m-1}}{\sum_{j=1}^{c}(1/(1 - K(x_k, v_j)))^{1/m-1}} \quad (12)$$

$$t_{ik} = \frac{1}{1 + [(2b(1 - K(x_k, v_i)))/\gamma_i]^{1/\eta-1}} \quad (13)$$

$$v_i = \frac{\sum_{k=1}^{N}(au_{ik}^m + bt_{ik}^\eta)K(x_k, v_i)x_k}{\sum_{k=1}^{N}(au_{ik}^m + bt_{ik}^\eta)K(x_k, v_i)} \quad (14)$$

The optimal values of the Kernel width can be obtained through Eq. (11), i.e.

$$\frac{\partial J}{\partial \sigma} = -2\sum_{k=1}^{N}\sum_{i=1}^{c}(au_{ik}^m + bt_{ik}^\eta)K(x_k, v_i)\frac{||x_k - v_i||^2}{\sigma^3} \quad (15)$$

It is suggested to select γ_i [8,10] as

$$\gamma_i = H\frac{2\sum_{k=1}^{N}u_{ik}^m(1 - K(x_k, v_i))}{\sum_{k=1}^{N}u_{ik}^m} \quad (16)$$

Typically, H is chosen as 1.

Proof. We differentiate $J_{\text{KPFCM}}(U, V, T)$ with respect to u_{ik}, t_{ik}, v_i and set the derivatives to zero. Then we get Eqs. (12)–(14). The details are given in Appendix A.

KPFCM algorithm $\begin{bmatrix} \text{inputs} & X, c, m, a, b, \eta \\ \text{outputs} & U, T, V, \sigma \end{bmatrix}$

The general form of Kernel-based hybrid c-means clustering algorithm is given below:

Kernel-based hybrid c-means clustering

Fix the number of clusters C; fix$(m, \eta, a, b) > 1$; Set the learning rate α;
Execute a FCM clustering algorithm to find initial U and V;
Initialize the typicality values t_{ik}^0 randomly;
Initialize the Kernel width $\sigma = \sigma^{(0)}$.
Set iteration count $k = 1$;
Repeat
Update v_i^k using (14).
Compute $\partial J/\partial\sigma$ using (15)
Compute γ_i using (16).
Update t_{ik}^k using (13).
Update u_{ik}^k using (12).
Update the Kernelwidth using $\sigma^{(k+1)} = \sigma^{(k)} + \alpha(\partial J/\partial\sigma)$.
Until a *given stopping criterion is satisfied.*

4. Experimental results

To demonstrate the effectiveness of the proposed method, we applied the Kernel-based hybrid c-means clustering method and three conventional methods (Fuzzy c-means, possibilistic c-means and possibilistic Fuzzy c-means) to a number of widely used data

sets and compared the performance of each method. We used data sets with a wide variety in the shape of clusters, number of data points and count of features of each datum. The real life data sets used in the experiments are well known as Iris data set, wine data set and Wisconsin breast cancer data set [15]. We choose $m = 2$ which is a common choice for Fuzzy clustering. For all data sets we use the following parameters: $\varepsilon = 0.001$, max_iter = 100. The Kernel used in the experiments is the Gaussian Kernel function $K(x, y) = \exp(-||x - y||^2/2\sigma^2)$, where σ is the Kernel width that should be optimized. The initial value for the Kernel width σ is set to $\sigma_0 = (1/c)\left[\sqrt{\sum_{j=1}^{l} ||x_j - \bar{x}||^2}/l\right]$ [16].

4.1. Identical data with noise

The first example involves the data set X_{12} as given in Ref. [10]. In general, FCM performs well for pattern sets that contain partitions of similar volume and similar number of patterns. However when noise is present, we do not get the desired results. For PCM, PFCM and KPFCM we first use FCM for initialization. Fig. 1 shows the clustering results for FCM, PCM, PFCM and our proposed method. The Ideal (true) centroids are

$$V_{\text{ideal}} = \begin{bmatrix} -3.34 & 0 \\ 3.34 & 0 \end{bmatrix}$$

Let $V_{\text{FCM}}^{12}, V_{\text{PCM}}^{12}, V_{\text{PFCM}}^{12}, V_{\text{KPFCM}}^{12}$ be the centroids in Table 1 produced by their respective algorithms. To show the effectiveness of our proposed algorithm, we also compute the error

$$E_* = ||V_{\text{ideal}} - V_*^{12}||^2$$

where * is FCM/PCM/PFCM/KPFCM. $E_{\text{FCM}} = 0.414$, $E_{\text{PCM}} = 1.316$, $E_{\text{PFCM}} = 0.3796$ and $E_{\text{KPFCM}} = 0.0005$. Clearly, from Fig. 1d, we can see that the proposed method is superior to other previous models.

Table 1
Terminal prototype produced by FCM, PCM, PFCM and KPFCM on X_{12}.

FCM ($m = 2$)		PCM ($\eta = 2$)		PFCM ($a = 1$, $b = 1$, $m = 2$, $\eta = 2$)		KPFCM ($a = 1$, $b = 1$, $m = 2$, $\eta = 2$, $\sigma = 0.5$)	
−2.98	0.54	−2.15	0.02	−2.84	0.36	−3.33	0.02
2.98	0.54	2.15	0.02	2.84	0.36	3.33	0.02

4.2. Different volume/equal low number "Square" data with outlier

By changing the volume of clusters in a pattern set we observe the effectiveness of our proposed method in comparison with FCM, PCM and PFCM. The pattern contains two clusters each containing nine patterns with different cluster density with an outlier added at (8, 14). Fig. 2(a) shows the FCM result using $m = 2$ which is generally used when performing FCM. The PCM algorithm finds nearly identical clusters when $m = 2$. The PFCM algorithm finds two distinct centers and gives better results (but not desirable) than FCM and PCM. However our proposed algorithm gives desirable results with cluster centers located almost at their prototypical locations.

4.3. Gaussian random data with noise

(i) In the next experiment, we generate a data set with unequal sized clusters. There are two clusters and the data points in each cluster are normally distributed over two-dimensional space. Their respective means are (3, 4) and (10, 10) and their

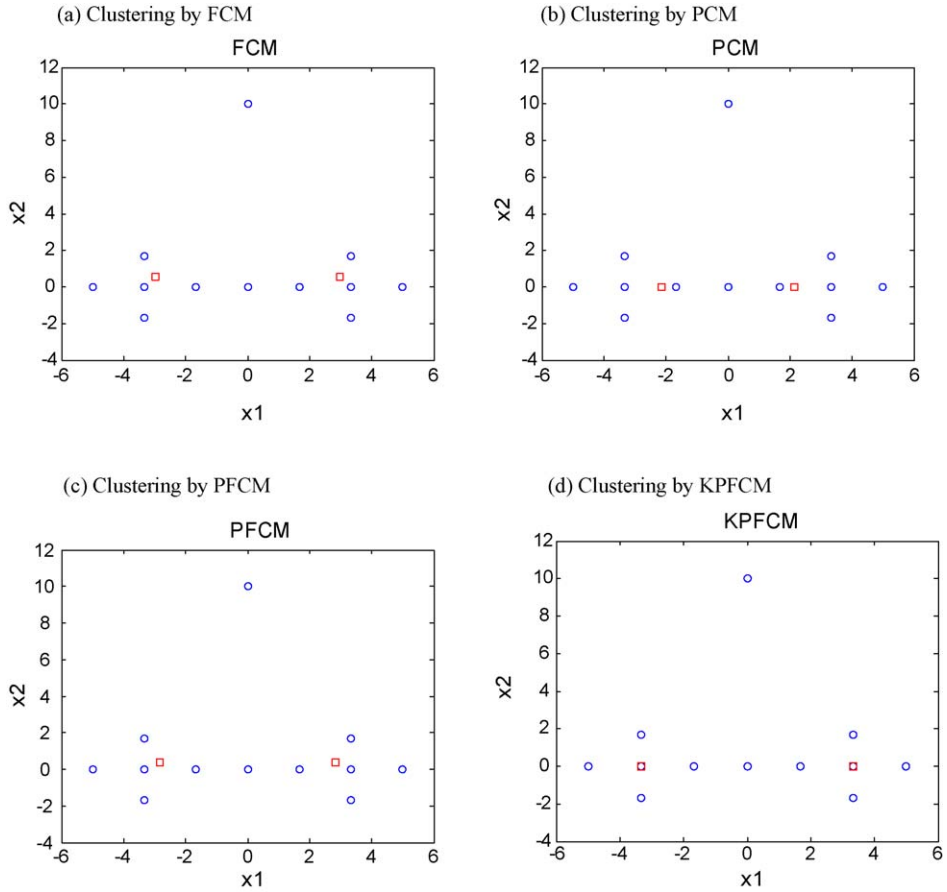

Fig. 1. (a) Clustering by FCM, (b) clustering by PCM, (c) clustering by PFCM, and (d) clustering by KPFCM.

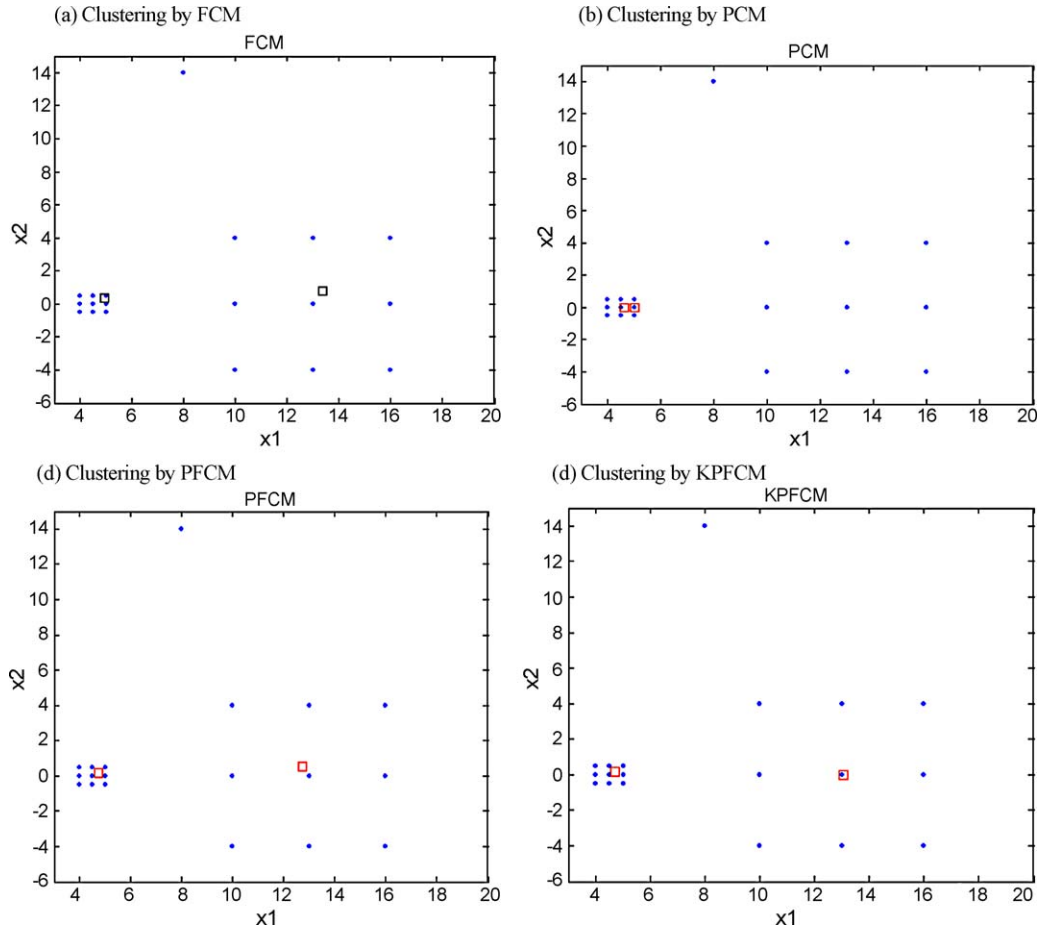

Fig. 2. (a) Clustering by FCM, (b) clustering by PCM, (c) clustering by PFCM, and (d) clustering by KPFCM.

respective covariance matrices are

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} 0.1 & 0 \\ 0 & 2 \end{pmatrix}$$

There are 60 data points in the first cluster and 10 data points in the second cluster. Besides this there are noisy points which are distributed over the region $[-2, 14] \times [0, 18]$. Fig. 3(a) shows the initial cluster centers determined by FCM. The PCM algorithm (Fig. 3b) again finds two identical clusters. We tried several combinations of a, b, m, η for PFCM. Fig. 3(c) shows the best clustering results of PFCM for $a = 1$, $b = 5$, $m = 2$, $\eta = 5$ which is same as FCM. Although the initial clusters determined by these algorithms are not good because of noise, our proposed algorithm can still determine clusters of good quality. Fig. 3(d) shows the clustering results for KPFCM for $a = 1$, $b = 2$, $m = 2$, $\eta = 2$ and $\sigma = 2.5$ which are better than that obtained by PFCM.

(ii) In this example, a Gaussian random number generator was used to create a data set containing two clusters and outliers. The effect of the noise points can be seen as the FCM algorithm shifts the partition space towards the outliers. The results shown by PCM are good when given good initial values from the FCM results. The results shown by our proposed method is better than FCM and PFCM as it properly partitions the feature space and yields prototypes that more closely resemble the actual centroid values for each cluster (Fig. 4).

4.4. High dimensional data sets

We now examine the performance of our proposed clustering algorithm on a number of well-known real data sets namely Iris data set, Wisconsin breast cancer data set and Wine data set. The clustering results were assessed using Huang's accuracy measure (r) [17]

$$r = \frac{\sum_{i=1}^{k} n_i}{n} \tag{17}$$

where n_i is the number of data occurring in both the ith cluster and its corresponding true cluster and n is the number of data points in the data set. According to this measure, a higher value of r indicates a better clustering result with perfect clustering yielding a value $r = 1$.

4.4.1. Iris data set

This is a four-dimensional data set containing 50 samples each of three types of Iris flowers. One of the three clusters (class 1) is well separated from the other two, while classes 2 and 3 have some overlap.

We made several runs of PCM, PFCM and our proposed method when these algorithms are initialized with FCM terminal prototypes. Table 2 compares the algorithms on the quality of the optimum solution as judged by Eq. (17). As indicated in Table 2, the typical result of comparing FCM partitions to the physically correct labels of Iris is 16 errors. PCM gets two coincident clusters making 50 errors. The best result for PFCM as reported in [10] is 13 errors. The number of misclassified data by our proposed algorithm is 11 with an accuracy value of $r = 0.93$.

4.4.2. Breast cancer data set

The Wisconsin breast cancer data is widely used to test the effectiveness of classification. The aim of the classification is to distinguish between benign and malignant cancers based on the

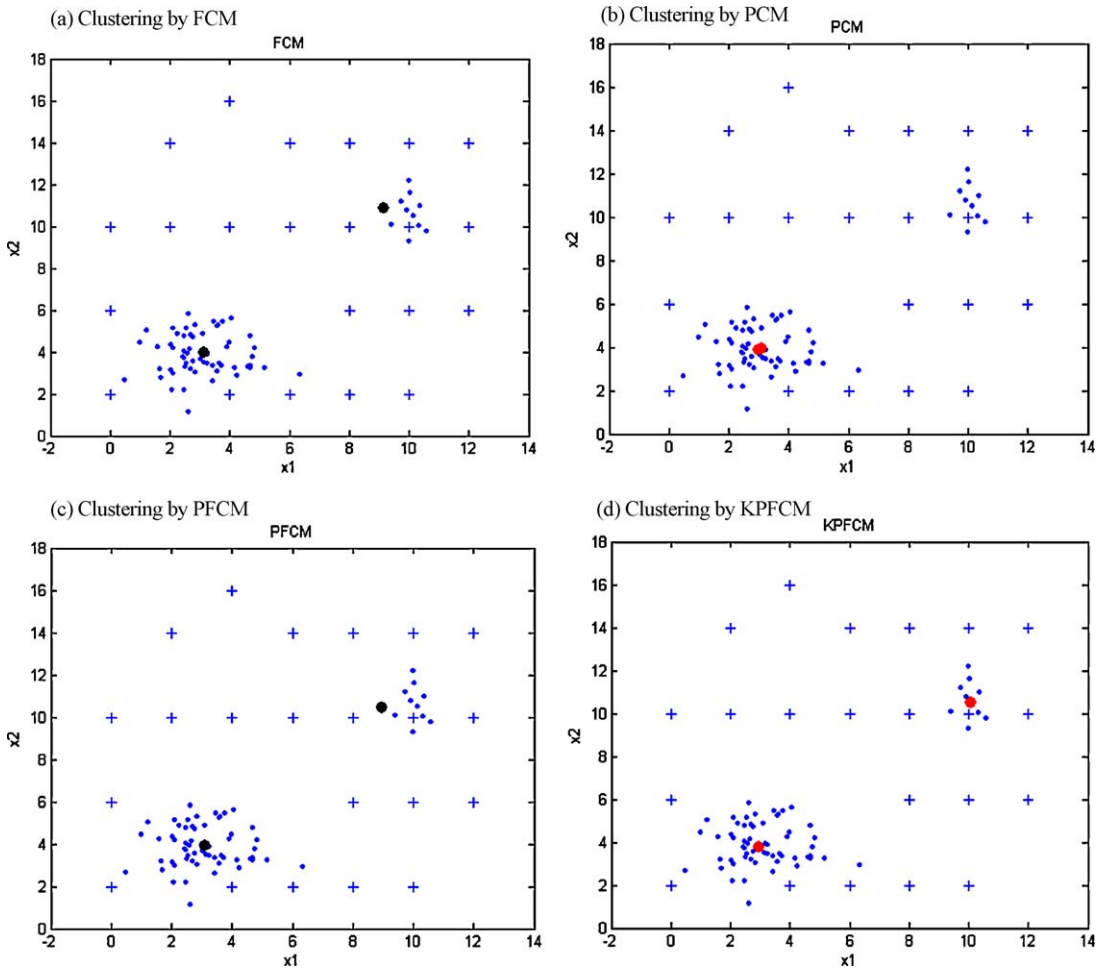

Fig. 3. (a) Clustering by FCM, (b) clustering by PCM, (c) clustering by PFCM, and (d) clustering by KPFCM.

available nine attributes. The original database contains 699 instances; however 16 of these are omitted because these are incomplete. The class distribution is 65.5% benign and 34.5% malignant, respectively. In the case of Wisconsin breast cancer data, the best results are again in the case of our proposed algorithm where from 683, nine-dimensional points "only" 22 are misclustered. The results are slightly better than PFCM where the number of misclassified data points is 23 as shown in Table 3.

4.4.3. Wine data set

The Wine data contains the chemical analysis of 178 wines grown in the same region in Italy but derived from three different cultivators. The problem is to distinguish the three different types based on 13 continuous attributes derived from chemical analysis as shown in Table 4.

Thus as evident, the Kernel-based hybrid c-means clustering (KPFCM) algorithm performed markedly better as compared to other competitive clustering algorithms in terms of number of misclassified data, highlighting the effectiveness and potential of the proposed method.

4.5. KPFCM clustering for identification of TS Fuzzy model

Fuzzy identification is an effective tool for the approximation of uncertain non-linear systems on the basis of measured data [18]. Among the different Fuzzy modeling techniques, the Takagi–Sugeno (TS) model [19] has attracted most attention. This model consists of IF-THEN rules with Fuzzy antecedents and mathematical functions in the consequent part. Fuzzy clustering has been quite extensively used to obtain the antecedent membership functions [20–22], while the parameters of the consequent functions can be estimated by using standard linear least-square methods. This model is of the following form:

$$\text{Rule i}: \text{ If } \; x_1 \text{ is } A_{i1} \text{ and} \ldots \text{ and } x_n \text{ is } A_{in} \;\; \text{THEN} \;\; y_i = c_{i0} + c_{i1} + \ldots + c_{in}x_n \tag{18}$$

Table 2
The number of misclassified data and accuracies using FCM, PCM, PFCM and the proposed methods for the Iris data set.

Methods	Misclassification	Accuracy
FCM	16	0.893
PCM	50	0.667
PFCM	13	0.900
Proposed	11	0.93

Table 3
The number of misclassified data and accuracies using FCM, PCM, PFCM and the proposed methods for the Wisconsin breast cancer data set.

Methods	Misclassification	Accuracy
FCM	30	0.953
PCM	239	0.300
PFCM	23	0.963
Proposed	22	0.965

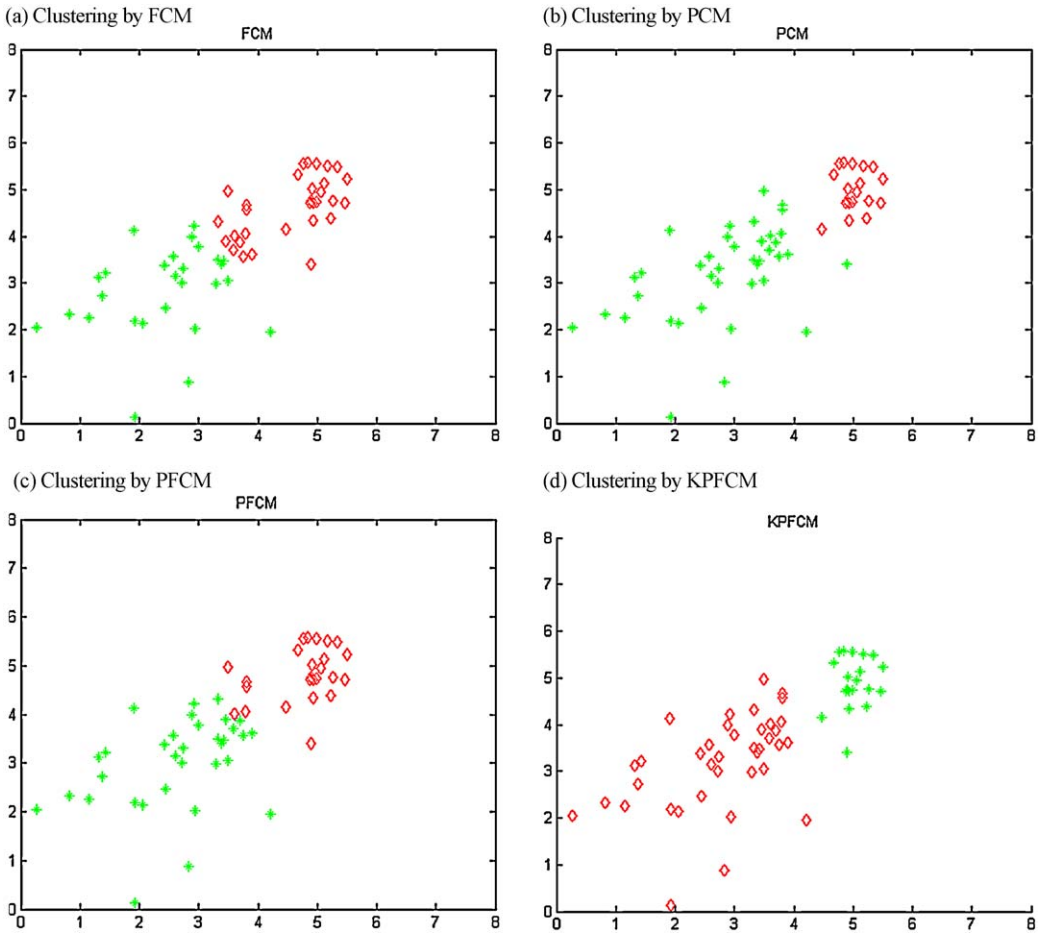

Fig. 4. (a) Clustering by FCM, (b) clustering by PCM, (c) clustering by PFCM, and (d) clustering by KPFCM.

where $i = 1, 2, \ldots, l$, l the number of IF-THEN rules is, $c'_{ik}s(k = 0, 1, \ldots, n)$ are the consequent parameters. y_i is an output from the ith IF-THEN rule, and A_{ij} is a Fuzzy set.

Given an input $(x_1, x_2, \ldots, x_n)$, the final output of the Fuzzy model used is inferred as follows:

$$y = \sum_{i=1}^{l} w_i y_i \tag{19}$$

where y_i is calculated for the consequent equation of the ith implication and the weight w_i implies the overall truth value of the premise of the ith implication for the input, and calculated as

$$w_i = \prod_{k=1}^{n} A_{ik}(x_k) \tag{20}$$

where Gaussian membership functions are used to represent the Fuzzy sets

$$A_{ik}(x_k) = \exp\left(-\frac{(x_k - a_{ik})^2}{{\sigma_{ik}}^2}\right) \tag{21}$$

with a_{ik} being the center and σ_{ik}, the variance of the Gaussian curve.

From (18) and (19)

$$y = \sum_{k=0}^{n}\sum_{i=1}^{l} w_i c_{ik} x_k \tag{22}$$

We use an TSK Fuzzy modeling approach to deal with a model of an operator's control of a chemical plant. The plant is for producing a polymer by the polymerization of some monomers. There are five input candidates, which a human operator might refer to for his control, and one output, i.e. his control.

These are the following:

$u1$: monomer concentration,
$u2$: change of monomer concentration,
$u3$: monomer flow rate,
$u4$, $u5$: local temperatures inside the plant
y: set point for monomer flow rate

70 data points of the above six variables from the actual plant operation are taken from [21].

Training is composed of two phases. In the first phase, we use KPFCM clustering to find the centers $(a_1, a_2, \ldots, a_l)$ and width of the membership functions is calculated as follows:

$$\sigma_{i,j}^2 = \frac{\sum_{k=1}^{n} \mu_{i,k}(x_{j,k} - a_{j,k})^2}{\sum_{k=1}^{n} \mu_{i,k}} \tag{23}$$

In the second phase, gradient descent method is used to minimize the error function

$$E = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(y^* - y)^2}$$

where y and y^* denote outputs of a Fuzzy model and a real system, respectively.

First we find six clusters by our proposed clustering method, which implies six rules in this case. Fig. 5 shows the values of membership functions for the five input variables. Fig. 6(a) shows

Table 4
The number of misclassified data and accuracies using FCM, PCM, PFCM and the proposed methods for the Wine data set.

Methods	Misclassification	Accuracy
FCM	56	0.685
PCM	104	0.415
PFCM	52	0.700
Proposed	48	0.731

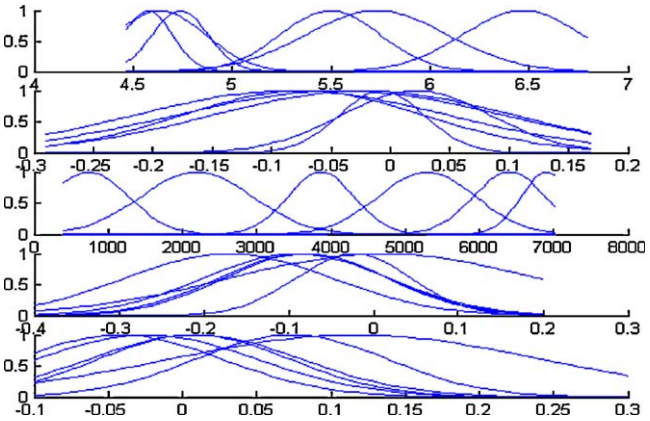

Fig. 5. Membership functions of the TS model for chemical plant based on five inputs.

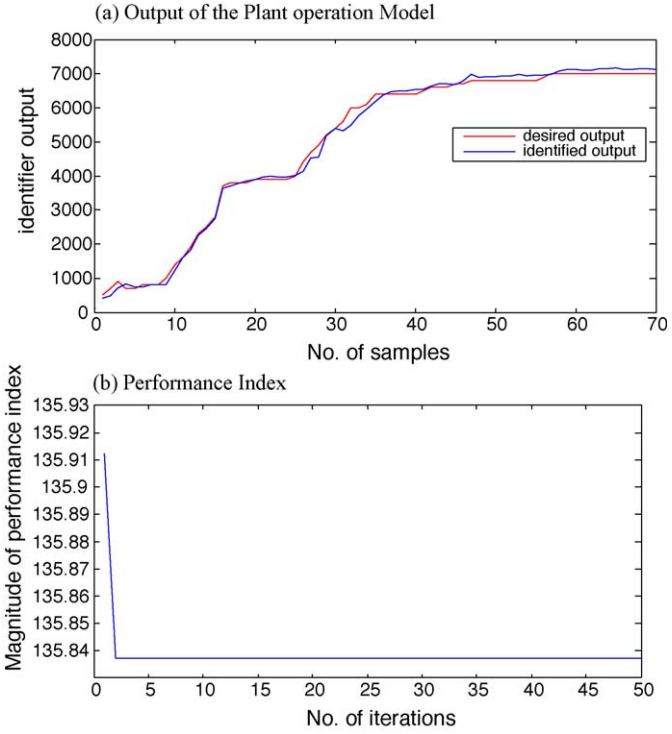

Fig. 6. (a) Output of the plant operation model and (b) performance index.

the actual output and the desired output vs. the number of samples, which clearly indicates that the actual output is following the desired output quite accurately. Fig. 6(b) shows the performance index vs. number of iterations.

5. Conclusions and discussion

A new KPFCM algorithm is proposed that is based on PFCM algorithm in order to deal with some issues in Fuzzy clustering as it is robust to noise and outliers and also tolerates unequal size clusters. We incorporate robustness into PFCM by incorporating a Kernel function. A Kernel function can implicitly map the input data to a high dimensional space in which data classification is easier. Furthermore, one of the major issues in using KPFCM algorithm on a specific problem lies not in performing the clustering itself, but rather in choosing the Kernel function and the values of the associated parameter in this method. We also proposed a simple technique to determine the Kernel width. Several examples are given to verify the applicability of the proposed approach. Compared to PFCM, our proposed method selects more desirable cluster centroids, thereby increasing the clustering accuracy.

The first point which can be raised regarding the proposed method of data partitioning is with regard to the choice of the type of Kernel function chosen in defining the non-linear mapping. This is one of the major questions which is under consideration regarding research being undertaken on Kernel methods. Clearly the choice of Kernel will be data specific, however in the specific case of data partitioning then a Kernel which will have universal approximation qualities such as RBF is most appropriate. So we have done all the tests using Gaussian Kernel which is a simpler form of RBF Kernel.

The other limitation of our proposed method is the determination of number of clusters. This problem can be easily taken care of using a validity measure which can be Kernelized which is the subject of our future research.

Appendix A. Kernel-based hybrid c-means clustering (KPFCM)

A.1. Proof for Kernel-based hybrid c-means clustering

In this appendix we give the proof of the Kernel-based hybrid c-means clustering which is a Kernelized version of possibilistic Fuzzy c-means clustering algorithm. The problem of minimization of objective function J_{KPFCM} {given by Eq. (11)} subjected to the constraint specified by Eq. (1) is solved by minimizing a constraint free objective function defined as:

$$J_{\mathrm{KPFCM}}(U,V,T) = \sum_{k=1}^{n}\sum_{i=1}^{c}(au_{ik}^{m} + bt_{ik}^{\eta})||\Phi(x_k) - \Phi(v_i)||^2 + \sum_{i=1}^{c}\gamma_i\sum_{k=1}^{n}(1-t_{ik})^{\eta} + \sum_{i=1}^{c}\lambda_i\left(\sum_{k=1}^{n}u_{ik} - 1\right)$$

where λ_i ($i = 1, 2, 3, \ldots, c$) are Langrangian multipliers.

By taking the partial derivatives of J_{KPFCM} with respect to u_{ik}, t_{ik}, v_i, yields the solution for the problem.

$$\begin{aligned} ||\Phi(x_k) - \Phi(v_i)||^2 &= (\Phi(x_k) - \Phi(v_i)) \cdot (\Phi(x_k) - \Phi(v_i)) \\ &= \Phi(x_k)\cdot\Phi(x_k) - 2\Phi(x_k)\Phi(v_i) + \Phi(v_i)\cdot\Phi(v_i) \\ &= K(x_k,x_k) - 2K(x_k,v_i) + K(v_i,v_i) \end{aligned}$$

If we use Gaussian function as a Kernel function $K(x, y) = \exp(-||x-y||^2/2\sigma^2)$,

Then $K(x, x) = 1$, hence $(||\Phi(x_k) - \Phi(v_i)||^2 = 2(1 - K(x_k - v_i))$

$$J_{\mathrm{KPFCM}} = 2\sum_{k=1}^{n}\sum_{i=1}^{c}(au_{ik}^{m} + bt_{ik}^{\eta})(1 - K(x_k,v_i)) + \sum_{i=1}^{c}\gamma_i\sum_{k=1}^{n}(1-t_{ik})^{\eta} + \sum_{i=1}^{c}\lambda_i\left(\sum_{k=1}^{n}u_{ik} - 1\right)$$

$$J_{\mathrm{KPFCM}} = 2\sum_{k=1}^{n}\sum_{i=1}^{c}(au_{ik}^{m} + bt_{ik}^{\eta})\left(1 - \exp\left(\frac{-||x_k - v_i||^2}{2\sigma^2}\right)\right) + \sum_{i=1}^{c}\gamma_i\sum_{k=1}^{n}(1-t_{ik})^{\eta} + \sum_{i=1}^{c}\lambda_i\left(\sum_{k=1}^{n}u_{ik} - 1\right) \quad (A.1)$$

A.1.1. Partial derivative of J_{KPFCM} with respect to v_i

The partial derivative of J_{KPFCM} with respect to v_i is:

$$\frac{\partial J}{\partial v_i} = \sum_{k=1}^{n} 2(au_{ik}^m + bt_{ik}^\eta) \times \left(-\exp\left(-\frac{||x_k - v_i||^2}{2\sigma^2}\right)\right) \times \frac{2(x_k - v_i)}{2\sigma^2} \tag{A.2}$$

Equating Eq. (A.2) to zero leads to,

$$\begin{aligned} & \frac{\partial J}{\partial v_i} = 0 \\ \Rightarrow \quad & \sum_{k=1}^{n} (au_{ik}^m + bt_{ik}^\eta) \times K(x_k, v_i) \times (x_k - v_i) = 0 \\ \Rightarrow \quad & \sum_{k=1}^{n} (au_{ik}^m + bt_{ik}^\eta) \times K(x_k, v_i) \times x_k = \sum_{k=1}^{n} (au_{ik}^m + bt_{ik}^\eta) \times K(x_k, v_i) \times v_i \end{aligned}$$

$$v_i = \frac{\sum_{k=1}^{n} (au_{ik}^m + bt_{ik}^\eta) K(x_k, v_i) x_k}{\sum_{k=1}^{n} (au_{ik}^m + bt_{ik}^\eta) K(x_k . v_i)} \tag{A.3}$$

A.1.2. Partial derivative of J_{KPFCM} with respect to t_{ik}

The partial derivative of J_{KPFCM} with respect to t_{ik} is:

$$\frac{\partial J}{\partial t_{ik}} = 2b\eta t_{ik}^{\eta-1}(1 - K(x_k, v_i)) - \gamma_i \eta (1 - t_{ik})^{\eta-1} \tag{A.4}$$

Equating Eq. (A.4) to zero leads to,

$$\begin{aligned} & \frac{\partial J}{\partial t_{ik}} = 0 \\ \Rightarrow \quad & 2bt_{ik}^{\eta-1}(1 - K(x_k, v_i)) = \gamma_i (1 - t_{ik})^{\eta-1} \\ \Rightarrow \quad & t_{ik}^{\eta-1} \left(\frac{2b}{\gamma_i}\right)(1 - K(x_k, v_i)) = (1 - t_{ik})^{\eta-1} \\ \Rightarrow \quad & t_{ik} \left(\frac{2b(1 - K(x_k, v_i))}{\gamma_i}\right)^{1/(\eta-1)} = 1 - t_{ik} \\ \Rightarrow \quad & t_{ik} = \frac{1}{1 + (2b(1 - K(x_k, v_i))/\gamma_i)^{1/(\eta-1)}} \end{aligned} \tag{A.5}$$

A.1.3. Partial derivative of J_{KPFCM} with respect to u_{ik}

The partial derivative of J_{KPFCM} with respect to u_{ik}:

$$\frac{\partial J}{\partial u_{ik}} = 2mau_{ik}^{m-1}(1 - K(x_k, v_i)) + \lambda_i \tag{A.6}$$

Equating Eq. (A.6) to zero leads to the following:

$$\begin{aligned} & \Rightarrow \frac{\partial J}{\partial u_{ik}} = 0 \\ & \Rightarrow 2mau_{ik}^{m-1}(1 - K(x_k, v_i)) + \lambda_i = 0 \\ & \Rightarrow u_{ik} = \left(\frac{-\lambda_i}{2ma}\right)^{(1/(m-1))} \left(\frac{1}{(1 - K(x_k, v_i))}\right)^{(1/(m-1))} \end{aligned} \tag{A.7}$$

To fulfill the constraint (1)

$$\Rightarrow u_{ik} = \frac{u_{ik}}{\sum_{i=1}^{c} u_{ik}} \tag{A.8}$$

In view of Eq. (A.8), Eq. (A.7) can be written as

$$\Rightarrow u_{ik} = \frac{(1/(1 - K(x_k, v_i)))^{1/(m-1)}}{\sum_{j=1}^{c} (1/(1 - K(x_k, v_j)))^{1/(m-1)}} \tag{A.9}$$

References

[1] M.R. Anderberg, Cluster Analysis for Application, Academic Press, New York, 1973 .
[2] E. Backer, A.K. Jain, A clustering performance measure based on fuzzy set decomposition, IEEE Trans. Pattern Anal. Mach Intell. 3 (1) (1981) 66–74.
[3] E.H. Ruspini, A new approach to clustering, Inform. Control 15 (1) (1969) 22–32.
[4] J.C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well-separated cluster, J. Cybern. 3 (3) (1973) 32–57.
[5] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum, New York, 1981.
[6] N.B. Karayiannis, J.C. Bezdek, An integrated approach to fuzzy learning vector quantization and fuzzy c-means clustering, IEEE Trans. Fuzzy Syst. 5 (1997) 622–628.
[7] R. Dave, Characterization and detection of noise in clustering, Pattern Rec. Lett. 12 (11) (1991) 657–664.
[8] R. Krishnapuram, J. Keller, A possibilistic approach to clustering, IEEE Trans. Fuzzy Syst. 1 (2) (1993) 110–998.
[9] N.R. Pal, K. Pal, J.C. Bezdek, A mixed c-means clustering model, in: Proceedings of the IEEE Int. Conf. on Fuzzy Systems, Spain, (1997), pp. 11–21.
[10] N.R. Pal, K. Pal, J. Keller, J.C. Bezdek, A possibilistic Fuzzy c-means clustering algorithm, IEEE Trans. Fuzzy Syst. 13 (4) (2005) 517–530.
[11] N. Christianini, J.S. Taylor, An Introduction to SVMs and other Kernel-based Learning Methods, Cambridge Univ. Press, 2000.
[12] M. Girolami, Mercer Kernel-based clustering in feature space, IEEE Trans. Neural Networks 13 (3) (2002) 780–784.
[13] K.-R. Muller, et al., An introduction to kernel-based learning algorithms, IEEE Trans. Neural Networks 12 (2) (2001) 181–202.
[14] M. Tushir, S. Srivastava, A new kernel based hybrid c-means clustering model, in: Proceedings of IEEE Int. Conf. on fuzzy systems, 23–26 July 2007, London, U.K., (2007), pp. 1–5.
[15] C. Blake, E. Keough, C.J. Merz, UCI Repository of Machine Learning Database, 1998 http://www.ics.uci.edu/~mlearn/Mlrepository.html.
[16] D. Zhang, S. Chen, Z.-H. Zhou, Learning the kernel parameters in kernel minimum distance classifier, Pattern Recogn. Lett. 39 (1) (2006) 133–135.
[17] Z. Huang, M.K. Ng, A fuzzy k-modes algorithm for clustering categorical data, IEEE Trans. Fuzzy Syst. 7 (4) (1999) 446–452.
[18] H. Hellendoorn, D. Driankov, Fuzzy Model Identification: Selected Approaches, Springer, Berlin, Germany, 1997.
[19] T. Takagi, M. Sugeno, Fuzzy identification of systems and its application to modeling and control, IEEE Trans. Syst. Man Cybern. 15 (1) (1985) 116–132.
[20] M. Sugeno, T. Yasukawa, A fuzzy-logic-based approach to qualitative modeling, IEEE Trans. Fuzzy Syst. 1 (1) (1993) 7–31.
[21] R. Babuska, Fuzzy Modeling for Control, Kluwer Academic Publishers, Boston, 1998.
[22] R. Babuska, H.B. Verbruggen, Constructing fuzzy models by product space clustering, in: H. Hellendoorn, D. Driankov (Eds.), Fuzzy Model Identification: Selected Approaches, Springer, Berlin, Germany, 1997, pp. 53–90.