2008 International Symposium on Nonlinear Theory and its Applications
NOLTA'08, Budapest, Hungary, September 7-10, 2008

Fuzzy c-Means for Data with Tolerance introducing Penalty Term in Feature Space

Yuchi Kanzawa†, Yasunori Endo‡ and Sadaaki Miyamoto‡

†Faculty of Engineering, Shibaura Institute of Technology
Tokyo, Japan
‡Faculty of Risk Engineering, University of Tsukuba
Ibaraki, Japan
Email: kanzawa@sic.shibaura-it.ac.jp

Abstract—A new fuzzy c-means algorithms for data with tolerance is proposed by introducing a penalty term in feature space. Its idea is derived from the support vector machine introducing a penalty term for "soft margin" in feature space. In the proposed method, the data is allowed to move for minimizing the corresponding objective function but this moveness is controlled by the penalty term.

First, an optimization problem is shown by introducing tolerance with conventional fuzzy c-means algorithm in feature space. Second, Karush-Kuhn-Tucker (KKT) conditions of the optimization problem is considered. Third, an iterative algorithm is proposed by re-expressing the KKT conditions using kernel trick. Fourth, another iterative algorithm is proposed for fuzzy classification function, which shows how prototypical an arbitrary point in the data space is to the obtained each cluster by extending the membership to the whole space. Last, some numerical examples are shown.

1. Introduction

Fuzzy c-means (FCM) [1] is one of the well-known fuzzy clusterings and many FCM variants have been proposed after FCM. FCM is pointed out that it is difficult to classify data with nonlinear borders because FCM uses squared distance between each datum and each cluster center for its dissimilarity. In order to solve this problem of conventional FCM, a new algorithm [2], called K-sFCM, has been proposed using nonlinear transformation from the original pattern space into a higher dimensional feature space with kernel function in Support Vector Machine (SVM) [3].

Soft margin in SVM helps the incompleteness of nonlinearity in the introduced kernel by allowing mislabeling [4]. K-sFCM also has the possibility to misclassify because the nonlinearity of the introduced kernel is incomplete. Taking account into this analogy, "tolerance" for data, which allows to move data in a region, was considered in K-sFCM [5].

In this paper, another type of tolerance for data is introduced with K-sFCM, which does not limit the maximal tolerance like the already proposed method [5], but introduce a penalty term for tolerance with the objective function of K-sFCM like soft margin method in SVM.

The contents of this paper are the followings. In the second section, we define some notation. In the third section, our new algorithm is proposed. In the forth section, we show some numerical examples of our proposed algorithm. In the last section, we conclude this paper.

2. Preliminaries

In this section, we define some notation.

The data set $x = \{x_i \mid x_i \in \mathbf{R}^p, i \in \{1, \dots, N\}\}$ is given. The membership by which x_i belongs to the j-th cluster is denoted by $u_{i,j}$ $(i \in \{1, \cdots, N\}, j \in \{1, \cdots, C\})$ and the set of $u_{i,j}$ is denoted by $u \in \mathbf{R}^{N \times C}$ called the partition matrix. The constraint for u is

$$\sum_{j=1}^{C} u_{i,j} = 1 \quad (0 \le u_{i,j} \le 1).$$

A high-dimensional feature space used in SVM is denoted by $\mathbf{H}$, whereas the original space $\mathbf{R}^p$ is called data space. $\mathbf{H}$ may be an infinite-dimensional metric space. Let the inner product denoted by $\langle \cdot, \cdot \rangle$. The norm of $\mathbf{H}$ for an element $g \in \mathbf{H}$ is given by

$$\|g\|_{\mathbf{H}}^2 = \langle g, g \rangle. \tag{1}$$

A transformation $\Phi : \mathbf{R}^p \to \mathbf{H}$ is employed whereby x_i is mapped into $\Phi(x_i)$. Explicit representation of $\Phi(x)$ is not usable in general but the inner product $\langle \Phi(x), \Phi(y) \rangle$ can be expressed by a kernel function

$$K(x, y) = \langle \Phi(x), \Phi(y) \rangle. \tag{2}$$

A representative kernel function is the radial basis function (RBF) kernel described as

$$K(x, y) = \exp(-\sigma^{-2} \|x - y\|_2^2) \tag{3}$$

with a positive parameter σ. The cluster center set in $\mathbf{H}$ is denoted by $W = \{W_j \mid W_j \in \mathbf{H}, j \in \{1, \dots, C\}\}$. The tolerance for the data x is denoted by $E = \{E_i \mid E_i \in \mathbf{H}, i \in \{1, \dots, N\}\}$. The maximum tolerance is denoted by $\kappa = \{\kappa_i \mid \kappa_i \in \mathbf{R}_+, i \in \{1, \dots, N\}\}$.

In [5], an iterative algorithm derived from the standard FCM are considered on the basis of the following optimization problem:

$$\underset{u,W,E}{\text{minimize}}\, J_{m,\mathrm{k},\mathrm{t}}(u, W, E) \tag{4}$$

$$\text{under} \quad \sum_{j=1}^{C} u_{i,j} = 1, \quad \|E_i\| \le \kappa_i^2, \tag{5}$$

where

$$J_{m,\mathrm{k,t}}(u,W,E)=\sum_{i=1}^{N}\sum_{j=1}^{C}u_{i,j}^{m}\|\Phi(x_i)+E-W_j\|_{\mathbf{H}}^2. \tag{6}$$

3. K-sFCM for Data with Tolerance introducing penalty term

In this section, a new FCM algorithm is proposed by introducing penalty term with K-sFCM. First, the optimization problem is described and its KKT conditions are led. Second, the KKT conditions are reformalized and the iterative algorithm is proposed. Third, another iterative algorithm is also proposed in order to obtain the values of fuzzy classification function.

For x, u, E, W, we consider the following optimization problem:

$$\underset{u,E,W}{\text{minimize}}\, J_{m,\mathrm{k,at}}(u,E,W) \quad \text{under} \quad \sum_{j=1}^{C}u_{i,j}=1, \tag{7}$$

where

$$J_{m,\mathrm{k,at}}(u,E,W)=\sum_{i=1}^{N}\sum_{j=1}^{C}u_{i,j}^{m}\|\Phi(x_i)+E_i-W_j\|_{\mathbf{H}}^2+\sum_{i=1}^{N}\beta_i\|E_i\|. \tag{8}$$

Its Lagrange function $L_{m,\mathrm{k,at}}(u,E,W)$ is as below:

$$\begin{aligned}L_{m,\mathrm{k,t}}(u,E,W)=&\sum_{i=1}^{N}\sum_{j=1}^{C}u_{i,j}^{m}\|\Phi(x_i)+E_i-W_j\|_{\mathbf{H}}^2\\&+\sum_{i=1}^{N}\gamma_i\left(\sum_{j=1}^{C}u_{i,j}-1\right)+\sum_{i=1}^{N}\beta_i\|E_i\|,\end{aligned} \tag{9}$$

where $\gamma=(\gamma_1,\ldots,\gamma_N)$ is KKT vector. From KKT conditions and kernel trick, we obtain the following iterative algorithm:

Algorithm 1 (K-sFCM-AT)

Step 1 *Give the value of m and κ. Select a kernel function $K:\mathbf{R}^p\times\mathbf{R}^p\to\mathbf{R}$. Set the initial cluster centers v_j $(j\in\{1,\cdots,C\})$ in $\mathbf{R}^p$.*

Step 2 *Calculate $Y_{i,j}^{(0)}$, $Z_{j,\tilde{j}}^{(0)}$ and $d_{i,j}^{(0)}$ such that*

$$Y_{i,j}^{(0)}=K(x_i,v_j^{(0)}),\quad Z_{j,\tilde{j}}^{(0)}=K(v_j^{(0)},v_{\tilde{j}}^{(0)}), \tag{10}$$

$$d_{i,j}^{(0)}=K(x_i,x_i)-2Y_{i,j}^{(0)}+Z_{j,j}^{(0)}. \tag{11}$$

Set t be 0.

Step 3 *Calculate $u_{i,j}^{(t)}$, $U_j^{(t)}$, $\mu_i^{(t)}$, $\alpha_i^{(t+1)}$, $Y_{i,j}^{(t+1)}$, $Z_{j,\tilde{j}}^{(t+1)}$ and $d_{i,j}^{(t+1)}$ such that*

$$u_{i,j}^{(t)}=1/\sum_{k=1}^{C}\left(\frac{d_{i,j}^{(t)}}{d_{i,k}^{(t)}}\right)^{1/(m-1)},\quad U_j^{(t)}=\sum_{i=1}^{N}{u_{i,j}^{(t)}}^m, \tag{12}$$

$$\mu_i^{(t)}=\sum_{j=1}^{C}{u_{i,j}^{(t)}}^m,\quad \alpha_i^{(t+1)}=-1/(\mu_i^{(t)}+\beta_i) \tag{13}$$

$$\begin{aligned}Y_{i,j}^{(t+1)}=&{U_j^{(t+1)}}^{-1}\sum_{k=1}^{N}{u_{k,j}^{(t+1)}}^m\\&\cdot\Bigg[\left(1-\alpha_k^{(t+1)}\mu_k^{(t+1)}\right)K(x_i,x_k)\\&+\alpha_k^{(t+1)}\sum_{\ell=1}^{C}{u_{k,\ell}^{(t+1)}}^m Y_{i,\ell}^{(t)}\Bigg],\end{aligned} \tag{14}$$

$$\begin{aligned}Z_{j,\tilde{j}}^{(t+1)}=&{U_j^{(t+1)}}^{-1}{U_{\tilde{j}}^{(t+1)}}^{-1}\sum_{k=1}^{N}\sum_{\ell=1}^{N}{u_{k,j}^{(t+1)}}^m{u_{\ell,\tilde{j}}^{(t+1)}}^m\\&\cdot\Bigg[\left(1-\alpha_k^{(t+1)}\mu_k^{(t+1)}\right)\left(1-\alpha_\ell^{(t+1)}\mu_\ell^{(t+1)}\right)\\&\cdot K(x_k,x_\ell)\\&+\left(1-\alpha_k^{(t+1)}\mu_k^{(t+1)}\right)\alpha_\ell^{(t+1)}\sum_{r=1}^{C}{u_{\ell,r}^{(t+1)}}^m Y_{k,r}^{(t+1)}\\&+\alpha_k^{(t+1)}\left(1-\alpha_\ell^{(t+1)}\mu_\ell^{(t+1)}\right)\sum_{q=1}^{C}{u_{k,q}^{(t+1)}}^m Y_{\ell,q}^{(t+1)}\\&+\alpha_k^{(t+1)}\alpha_\ell^{(t+1)}\sum_{q=1}^{C}\sum_{r=1}^{C}{u_{k,q}^{(t+1)}}^m{u_{\ell,r}^{(t+1)}}^m Z_{q,r}^{(t)}\Bigg],\end{aligned} \tag{15}$$

$$\begin{aligned}d_{i,j}^{(t+1)}=&\left(1-\alpha_i^{(t+1)}\mu_i^{(t+1)}\right)^2K(x_i,x_i)\\&+2\left(1-\alpha_i^{(t+1)}\mu_i^{(t+1)}\right)\\&\cdot\sum_{k=1}^{C}\left(\alpha_i^{(t+1)}{u_{i,k}^{(t+1)}}^m-\delta_{k,j}\right)Y_{i,k}^{(t+1)}\\&+\sum_{k=1}^{C}\sum_{\ell=1}^{C}\left(\alpha_i^{(t+1)}{u_{i,k}^{(t+1)}}^m-\delta_{k,j}\right)\\&\cdot\left(\alpha_i^{(t+1)}{u_{i,\ell}^{(t+1)}}^m-\delta_{\ell,j}\right)Z_{k,\ell}^{(t)}.\end{aligned} \tag{16}$$

Step 4 *Check the stopping criterion. If the criterion is not satisfied, go back to* Step 3.

Fuzzy classification function (FCF) value of K-sFCM-AT for a brand-new data $\tilde{x}\in\mathbf{R}^p$ is obtained by the following algorithm:

Algorithm 2 (FCF for K-sFCM-AT)

Step 1 *Inherit m, C, Z, K from Algorithm 1. Give the maximal tolerance value of $\tilde{\kappa}$ for $\tilde{x}$. Set $a_{j,\tilde{j}}$*

and b_j such that

$$a_{j,\tilde{j}} =\delta_{j,\tilde{j}} - U_j^{-1}\sum_{k}^{N} u_{k,j}^m \alpha_k u_{k,\tilde{j}}^m, \tag{17}$$

$$b_j =U_j^{-1}\sum_{k=1}^{N} u_{k,j}^m (1-\alpha_k\mu_k)K(x,x_k) \tag{18}$$

and solve $Ay = b$. Calculate $\tilde{d}_j$ such that

$$\tilde{d}_j = K(\tilde{x},\tilde{x}) - 2y_j + Z_{j,j}. \tag{19}$$

Step 2 *Calculate $\tilde{u}_j$, $\tilde{\mu}_j$, $\tilde{\alpha}$ and $\tilde{d}_j$ such that*

$$\tilde{u}_j =1/\sum_{k=1}^{C}\left(\frac{\tilde{d}_j}{\tilde{d}_k}\right)^{\frac{1}{m-1}}, \quad \tilde{\mu} = \sum_{j=1}^{C}\tilde{u}_j^m \tag{20}$$

$$\tilde{\alpha} =-1/(\tilde{\mu}+\tilde{\beta}) \tag{21}$$

$$\tilde{d}_j =(1-\tilde{\alpha}\tilde{\mu})^2 K(\tilde{x},\tilde{x}) + 2(1-\tilde{\alpha}\tilde{\mu})\sum_{k=1}^{C}(\tilde{\alpha}\tilde{u}_k - \delta_{k,j})y_k + \sum_{k=1}^{C}\sum_{\ell=1}^{C}(\tilde{\alpha}\tilde{u}_k - \delta_{k,j})(\tilde{\alpha}\tilde{u}_\ell - \delta_{\ell,j})Z_{k,\ell}. \tag{22}$$

Step 3 *Check the stopping criterion. If the criterion is satisfied, $\tilde{u}_j$ is the fuzzy classification function value with respect to $\tilde{x}$. Otherwise, go back to* Step 2.

4. Numerical Examples

In this section, we show some examples of the proposed algorithm 1 and 2. In each example, after ten trials for Algorithm 1 with different initial cluster centers are tested and the solution with the minimal objective function value is selected, Algorithm 2 is applied. For all examples, we employ RBF kernel

$$K(x,y) = \exp(-\sigma^2\|x-y\|_2^2). \tag{23}$$

The first example is classifying the data shown in Fig. 1 into a ring shaped cluster and a ball one. We fix $\sigma^2 = 0.1$ and $m = 2$, and test two different values of $\beta_i \in \{0, 5\}$. The cases of $\beta_i \in \{0, 0.5\}$ produce the correctly classified results shown in Fig. 2 and Fig. 3, respectively. From these figures, we can find that the larger value of β_i, the larger membership for the cluster #1 and the larger size of the range for the cluster #1.

The second example is classifying the data shown in Fig. 4 into two crescents shaped clusters. We fix $\sigma^2 = 0.1$ and $m = 2$ and test two different values of $\beta_i \in \{0, 2\}$. While the case of $\beta_i = 0$ fails shown in Fig. 5, the cases of $\beta_i = 2$ produce the correct result shown in Fig. 6. From these figures, we can find that the tolerance helps the incomplete nonlinearity of the introduced kernel and makes the classification border bended adequately.

5. Conclusion

In this paper, we proposed the fuzzy classification function of the standard fuzzy c-means for data with tolerance using kernel functions, which is another type of tolerance than the already proposed one [5]. First, a certain optimization problem was shown for the fuzzy classification function for fuzzy c-means with penalty term of tolerance for data using kernel functions. Second, Karush-Kuhn-Tucker conditions of the objective function was considered, and the iterative algorithm was proposed for the optimization problem. Another iterative algorithm for fuzzy classification function was also proposed. Last, some numerical examples were shown.

As future works, we will compare the proposed method with the already proposed method [5] from the view point of classification performance.

Note that the fuzzy classification function of K-eFCM-T can be also calculated by similar iterative algorithm, though it is omitted by the sake of pages.

References

[1] Bezdek, J. P.: Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum, New York (1981).

[2] Miyamoto, S. and Suizu, D.: "Fuzzy c-Means Clustering Using Kernel Functions in Support Vector Machines", J. Advanced Computational Intelligence and Intelligent Informatics, vol.7, no.1, pp.25–30, 2003.

[3] Vapnik, V., N.: Statistical Learning Theory, Wiley, New York (1998).

[4] Cotes, C. and Vapnik, V., N.: Support Vector Networks, Machine Learning, vol.20, pp.273–297, (1995). Wiley, New York (1998).

[5] Kanzawa, Y., Endo, Y. and Miyamoto, S.: "Fuzzy c-Means Algorithms for Data with Tolerance using Kernel Functions", IEICE Trans. Fundamentals, (to appear).

[6] Hastie, Tibshirani, and H. Friedman: "The Elements of Statistical Learning: Data Mining, Inference, and Prediction (Springer Series in Statics)", Springer-Verlag, (2001).

[7] S. Sathiya Keerthi and Chih-Jen Lin: "Asymptotic Behaviors of Support Vector Machines with Gaussian Kernel", Neural Computation Vol.15, No.7, pp.1667–1689, (2003).

[8] Miyamoto, S. and Umayahara, K.: "Methods in Hard and Fuzzy Clustering", in: Liu, Z.-Q. and Miyamoto, S. (eds), Soft computing and human-centered machines, Springer-Verlag Tokyo, (2000).

[9] Murata, R., Endo, Y., Haruyama, H. and Miyamoto, S.: "On fuzzy c–means for data with tolerance", J. Advanced Computational Intelligence and Intelligent Informatics, vol.10, no.5, pp.673–681, 2006.

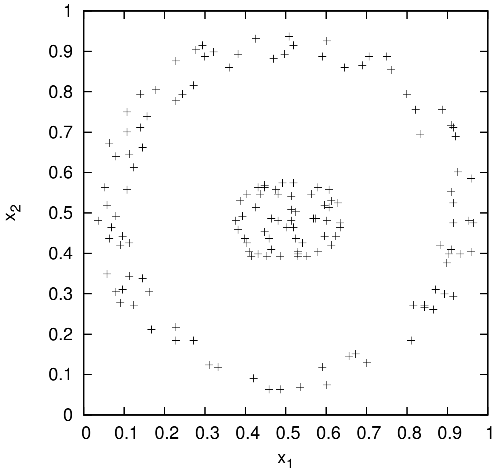

Figure 1: Ring and Ball Data

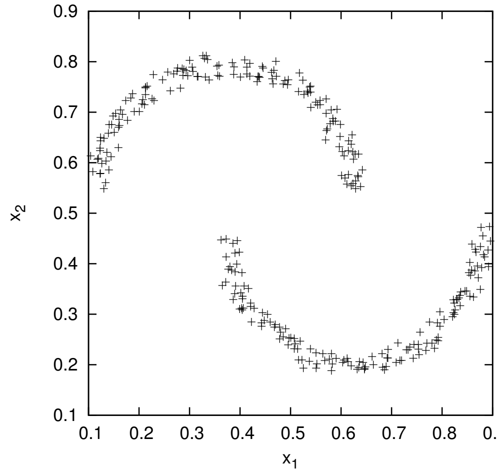

Figure 4: Crescents Data

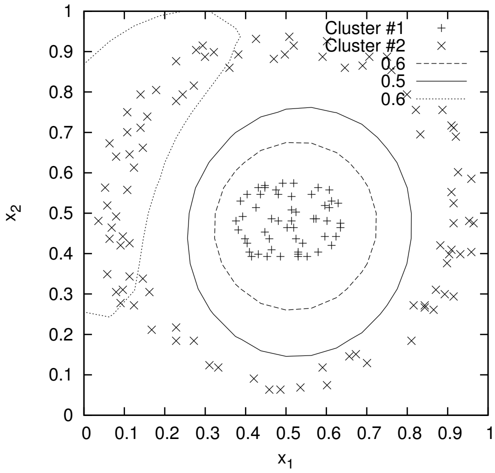

Figure 2: Successful Classification Result of Fig. 1 by K-sFCM-AT 1 with $\sigma^2 = 0.1$, $m = 2$ and $\beta_i = 0$, and its fuzzy classification function surface by Algorithm 2

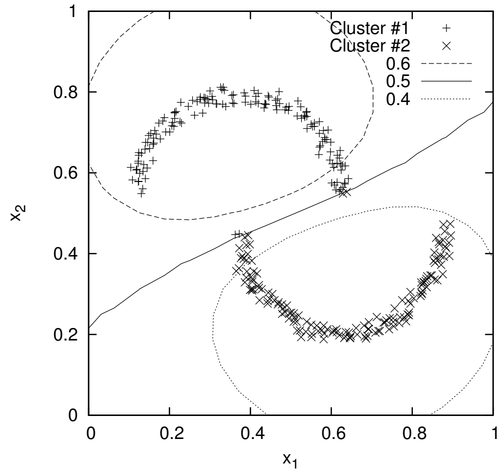

Figure 5: Misclassification Result of Fig. 4 by K-sFCM-AT 1 with $\sigma^2 = 0.1$, $m = 2$ and $\beta_i = 0$, and its fuzzy classification function surface by Algorithm 2

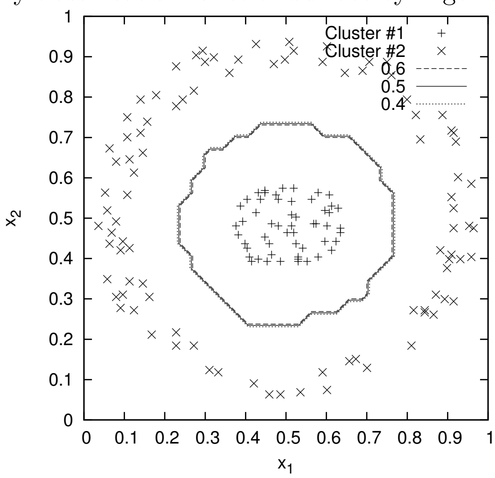

Figure 3: Successful Classification Result of Fig. 1 by K-sFCM-AT 1 with $\sigma^2 = 0.1$, $m = 2$ and $\beta_i = 5$, and its fuzzy classification function surface by Algorithm 2

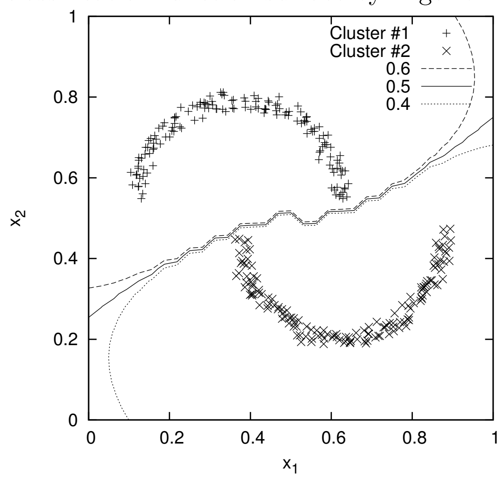

Figure 6: Successful Classification Result of Fig. 4 by K-sFCM-AT 1 with $\sigma^2 = 0.1$, $m = 2$ and $\beta_i = 2$, and its fuzzy classification function surface by Algorithm 2