ARTICLE IN PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Pattern Recognition Letters xxx (2004) xxx–xxx

Pattern Recognition Letters

www.elsevier.com/locate/patrec

# A kernel-based subtractive clustering method

Dae-Won Kim [a,*], KiYoung Lee [b], Doheon Lee [a], Kwang H. Lee [a,b]

[a] *Department of BioSystems and Advanced Information Technology Research Center, KAIST, 373-1 Guseong-dong, Yuseong-gu, Daejeon 305-701, Korea*

[b] *Department of Electrical Engineering & Computer Science, KAIST, Guseong-dong 373-1, Yuseong-gu, Daejeon 305-701, Republic of Korea*

Received 30 March 2004

**Abstract**

In this paper the conventional subtractive clustering method is extended by calculating the mountain value of each data point based on a kernel-induced distance instead of the conventional sum-of-squares distance. The 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. Use of the 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. Application of the conventional subtractive method and the kernel-based subtractive method to well-known data sets showed the superiority of the proposed approach.


*Keywords:* Clustering; Mountain method; Subtractive method; Kernel function

## 1. Introduction

Clustering has emerged as a popular technique for pattern recognition, image processing, and, most recently, data mining. Data clustering, also known as cluster analysis, classifies a collection of unlabeled data patterns into homogeneous clusters based on a similarity measure (Jain and Dubes, 1998; Jain et al., 1999). A variety of clustering algorithms have been proposed, including the hierarchical, *k*-means, and fuzzy *c*-means algorithms (Bezdek et al., 1999).

Yager and Filev developed the mountain method for estimating cluster centroids (Yager and Filev, 1994). This simple method estimates the cluster centroids by constructing and destroying

* Corresponding author. Tel.: +82 42 869 4353/5561; fax: +82 42 869 8680.

*E-mail addresses:* dwkim@bisl.kaist.ac.kr, dwkim@if.kaist.ac.kr (D.-W. Kim).

0167-8655/$ - see front matter © 2004 Published by Elsevier B.V.
doi:10.1016/j.patrec.2004.10.001

the mountain function on a grid space. However, although the mountain method is effective for low-dimensional data sets, it becomes prohibitively inefficient when applied to high-dimensional data. To reduce the computational complexity of this method, Chiu suggested calculating the mountain function on the data points rather than the grid points, an approach known as the subtractive method (Chiu, 1995). Velthuizen improved the implementation of the mountain and subtractive methods to allow large data sets to be clustered effectively (Velthuizen et al., 1997), and Pal extended these algorithms to detect circular shell-shaped clusters (Pal and Chakraborty, 2000). Yao used an entropy function in place of the mountain function; under this approach, a data point with minimum entropy is selected as a candidate cluster centroid (Yao et al., 2000).

A kernel function measures the distance between two data points as the data points are mapped into a high dimensional feature space in which the data is linearly separable (Muller et al., 2001; Girolami, 2002). Several kernel-based learning methods, for example support vector machine (SVM), have recently shown remarkable performance in supervised learning (Scholkopf and Smola, 2002; Zhang and Chen, 2003; Muller et al., 2001; Girolami, 2002; Vapnik, 1998; Wu and Xie, 2003; Zhang and Rudnicky, 2002). In the present work, we introduce a kernel-based subtractive method in which the kernel function is incorporated into the calculation of the mountains. The kernel-induced mountain values increase the separability of data by working in a high dimensional space; thus, as shown in this paper, the proposed method is characterized by higher clustering accuracy than the original subtractive method.

The remainder of this paper is organized as follows. Section 2 provides background information on the mountain function and discusses the issues associated with conventional methods for estimating cluster centroids. In Section 3, the proposed kernel-based subtractive method is formulated. Section 4 highlights the potential of the proposed approach through various experimental examples. Concluding remarks are presented in Section 5.

## 2. Previous works

### 2.1. The mountain method

In the original mountain method, proposed by Yager and Filev (1994), a grid is created in the data space, and then a potential function, referred to as the mountain function, is calculated on each grid point. The grid points with higher mountain values are selected as the cluster centroids.

Let us consider an unlabeled data set $X = \{x_1, \ldots, x_n\}$ in the $p$-dimensional space $R^p$. Let $x_{jk}$ be the $k$-th coordinate of the $j$-th data point for $1 \leqslant j \leqslant n$ and $1 \leqslant k \leqslant p$. The $p$-dimensional space $R^p$ is restricted to a $p$-dimensional hypercube $I_1 \times I_2 \times \cdots \times I_p$ where the intervals $I_k$, $1 \leqslant k \leqslant p$ are defined by the ranges of the coordinates $x_{jk}$. Obviously, the hypercube contains the data set $X$. Then the intervals $I_k$ are subdivided into $r_k$ equidistant points. This discretization forms a $p$-dimensional grid in the hypercube with grid points $v_i$ for $1 \leqslant i \leqslant N$ where $N(= r_1 \times \cdots \times r_p)$ is the number of grid points.

Let $d(v_i, x_j)^2 = \|v_i - x_j\|^2$ be the square of distance between a grid point $v_i$ and a data point $x_j$. Of the distance measures proposed to date, the Euclidean distance is the most widely used (Bezdek et al., 1999; Yager and Filev, 1994). The mountain function at a grid point $v_i$ is defined as

$$M(v_i) = \sum_{j=1}^{n} \mathrm{e}^{-\alpha\|v_i - x_j\|^2} \tag{1}$$

where $\alpha$ is a positive constant. A higher value of the mountain function indicates that $v_i$ has more data points $x_j$ in its vicinity. Thus, it is reasonable to select a $v_i$ with a high value of the mountain function $M(v_i)$ as a cluster centroid.

After calculating the mountain function for each grid point, the cluster centroids are selected by destroying the mountains. Let $M_1^*$ be the maximum value of the mountain function:

$$M_1^* = \mathrm{Max}_i[M(v_i)] \tag{2}$$

and let $v_1^*$ be the grid point whose mountain value is $M_1^*$. Then $v_1^*$ is selected as the first cluster centroid. To find other cluster centroids, we must first eliminate the effects of the cluster centroids that have already been identified. To achieve this, a

value inversely proportional to the distance of the grid point from the found centroids is subtracted from the previous mountain function; this process is carried out using the equation:

$$\widehat{M}^{j}(v_i) = \widehat{M}^{j-1}(v_i) - M^{*}_{j-1} \sum_{j=1}^{n} \mathrm{e}^{-\beta \|v_i - v^{*}_{j-1}\|^2} \tag{3}$$

where $\widehat{M}^{j}$ is the new mountain function, $\widehat{M}^{j-1}$ is the old mountain function, $M^{*}_{j-1}$ is the maximum value of $\widehat{M}^{j-1}$, $v^{*}_{j-1}$ is the newly found centroid, and $\beta$ is a positive constant. From Eq. (3), we see that the mountain values of grid points closer to the newly found centroid are decreased to a much greater extent than those further away. Thus, the procedure to approximate the cluster centroids is as follows:

Step 1. Initialize the parameters $\alpha$, $\beta$ and the intervals $I_k$, $1 \leqslant k \leqslant p$.
Step 2. Quantize the intervals and determine the grid.
Step 3. Compute the mountain functions $M(v_i)$ for each $v_i$, $1 \leqslant i \leqslant n$.
Step 4. Choose the grid point $v_i$ for which $M(v_i)$ is highest as a cluster centroid.
Step 5. Destroy and recompute the mountain function.
Step 6. If the number of centroids found is equal to the pre-specified number of clusters, then stop; otherwise go to Step 4.

### 2.2. The subtractive method

The clustering performance of the mountain method strongly depends on the grid resolution, with finer grids giving better performance. As the grid resolution is increased, however, the method becomes computationally expensive. Moreover, the mountain method becomes computationally inefficient when applied to high dimensional data because the number of grid points required increases exponentially with the dimension of data.

Chiu suggested an improved version of the mountain method, referred to as the subtractive method, in which each data point is considered as a potential cluster centroid (Chiu, 1995). Under this method, the mountain function is calculated on data points rather than grid points. The computational load of this method presumably still increases with increasing dimension of data, just not at the same rate for the original mountain method.

The mountain function at a data point $x_i$ is defined as

$$M(x_i) = \sum_{j=1}^{n} \mathrm{e}^{-\alpha \|x_i - x_j\|^2} \tag{4}$$

where $\alpha$ is a positive constant and $\|x_i - x_j\|^2$ is the square of distance between $x_i$ and $x_j$. Using this mountain function, cluster centroids are selected in a manner similar to that used in the original mountain method. Let $M^{*}_1$ be the maximum value of the mountain function

$$M^{*}_1 = \mathrm{Max}_i[M(x_i)] \tag{5}$$

and let $x^{*}_i$ be the data point whose mountain value is $M^{*}_1$; this data point is selected as the first cluster centroid. The modified mountain function used to find subsequent cluster centroids is defined as

$$\widehat{M}^{j}(x_i) = \widehat{M}^{j-1}(x_i) - M^{*}_{j-1} \sum_{j=1}^{n} \mathrm{e}^{-\beta \|x_i - x^{*}_{j-1}\|^2} \tag{6}$$

where $x^{*}_{j-1}$ is the newly found centroid and $\beta$ is a positive constant. The procedure for the subtractive method is similar to that of the mountain method except for the interval and the grid being eliminated.

The mountain and subtractive methods can both be used either as (1) stand-alone clustering methods by assigning each data point to a specific cluster based on the distances between the data point and the centroids or (2) supporting tools to estimate the initial cluster centroids for other clustering methods such as the $k$-means and fuzzy $c$-means methods.

## 3. Kernel-based subtractive clustering method

### 3.1. Kernel-based approach

The reduction in computational complexity achieved in going from the grid-based formulation of the mountain method to the point-based sub-

tractive clustering method can, in systems with far fewer data points than grid points, be accompanied by lower accuracy. The present work proposes a way of increasing the accuracy of the subtractive method by exploiting a kernel function in calculating the mountain value of each data point; mapping the data points from input space to a high dimensional space in which distance is measured using a kernel function, and each mountain value is calculated. The mountains of the proposed kernel-based method calculated in a high dimensional space are much more informative than those of the conventional subtractive method calculated in the original space; leading to more accurate selection of the cluster centroids.

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 the data are more clearly separable (Muller et al., 2001; Girolami, 2002).

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

$$\Phi : R^p \rightarrow H \quad x \mapsto \Phi(x) \tag{7}$$

Let us consider the dot product $(x_i \cdot x_j)$, often referred to as the inner product, which is used as a similarity measure in a variety of machine learning methods. By applying the nonlinear mapping function $\Phi$, the $x_i \cdot x_j$ in the input space is mapped to $\Phi(x_i) \cdot \Phi(x_j)$ in the feature space, which is thought to be a more general similarity measure (Scholkopf and Smola, 2002).

The key notion in kernel-based learning is that the mapping function $\Phi$ 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 $R^p$ (Scholkopf and Smola, 2002)

$$K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j) \tag{8}$$

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

$$\Phi : R^2 \rightarrow H = R^3 \quad (x_{i1}, x_{i2}) \mapsto (x_{i1}^2, x_{i2}^2, \sqrt{2}x_{i1}x_{i2}) \tag{9}$$

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 $\Phi$.

Three commonly used kernel functions (Scholkopf and Smola, 2002) are the polynomial kernel function,

$$K(x_i, x_j) = (x_i \cdot x_j + c)^d \tag{10}$$

where $c \geqslant 0$, $d \in N$; the Gaussian kernel function,

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \tag{11}$$

where $\sigma > 0$; and the sigmoidal kernel function

$$K(x_i, x_j) = \tanh(\kappa(x_i \cdot x_j) + \vartheta) \tag{12}$$

where $\kappa > 0$ and $\vartheta < 0$.

### 3.2. Formulation

Given a data point $x_i \in R^p$ $(1 \leqslant i \leqslant n)$ and a nonlinear mapping $\Phi : R^p \rightarrow H$, the mountain function at a data point $x_i$ is defined as

$$M(x_i) = \sum_{j=1}^{n} e^{-\alpha\|\Phi(x_i) - \Phi(x_j)\|^2} \tag{13}$$

where $\alpha$ is a positive constant and $\|\Phi(x_i) - \Phi(x_j)\|^2$ is the square of distance between $\Phi(x_i)$ and $\Phi(x_j)$. Thus a higher value of $M(x_i)$ indicates that $x_i$ has more data points $x_j$ near to it in the feature space. The distance in the feature space is calculated through the kernel in the input space as follows:

$$\begin{aligned}\|\Phi(x_i) - \Phi(x_j)\|^2 &= (\Phi(x_i) - \Phi(x_j)) \cdot (\Phi(x_i) - \Phi(x_j)) \\ &= \Phi(x_i) \cdot \Phi(x_i) - 2\Phi(x_i)\Phi(x_j) \\ &\quad + \Phi(x_j)\Phi(x_j) \\ &= K(x_i, x_i) - 2K(x_i, x_j) + K(x_j, x_j)\end{aligned} \tag{14}$$

Therefore, Eq. (13) can be rewritten as

$$M(x_i) = \sum_{j=1}^{n} e^{-\alpha(K(x_i,x_i)-2K(x_i,x_j)+K(x_j,x_j))} \quad (15)$$

The cluster centroid selection procedure is similar to that of the subtractive method. After calculating the mountain values, the data point $x_i^*$ whose mountain value is $M_1^* = \mathrm{Max}_i[M(x_i)]$ is selected as the first cluster centroid. To eliminate the effects of the previously identified centroids, the mountain function to find subsequent centroids is modified as follows:

$$\widehat{M}^j(x_i) = \widehat{M}^{j-1}(x_i) - M_{j-1}^* \sum_{j=1}^{n} e^{-\beta\|\Phi(x_i)-\Phi(x_{j-1}^*)\|^2} \quad (16)$$

$$= \widehat{M}^{j-1}(x_i) - M_{j-1}^* \times \sum_{j=1}^{n} e^{-\beta(K(x_i,x_i)-2K(x_i,x_{j-1}^*)+K(x_{j-1}^*,x_{j-1}^*))} \quad (17)$$

where $x_{j-1}^*$ is the newly found centroid and $\beta$ is a positive constant.

The procedure for the kernel-based subtractive method is as follows:

Step 1. Given the number of clusters, $k$, and the chosen values of $\alpha$, $\beta$, choose a kernel function $K$.

Step 2. Compute the mountain function. For each $x_i$:

$$M(x_i) = \sum_{j=1}^{n} e^{-\alpha(K(x_i,x_i)-2K(x_i,x_j)+K(x_j,x_j))} \quad (18)$$

Step 3. Choose the data point $x_i$ whose mountain function is highest as a cluster centroid.

Step 4. Destroy and recompute the mountain function. For each $x_i$:

$$\widehat{M}^j(x_i) = \widehat{M}^{j-1}(x_i) - M_{j-1}^* \times \sum_{j=1}^{n} e^{-\beta(K(x_i,x_i)-2K(x_i,x_{j-1}^*)+K(x_{j-1}^*,x_{j-1}^*))} \quad (19)$$

Step 5. If the number of centroids found is equal to $k$, then stop; otherwise go to Step 3.

## 4. Experimental results

To demonstrate the effectiveness of the proposed method, we applied the kernel-based subtractive method and three conventional methods (the $k$-means, fuzzy $c$-means, and subtractive methods) to a number of widely used data sets and compared the performance of each method. The subtractive and proposed methods are used as stand-alone clustering methods to clearly show the effects of the cluster centroids selected.

In these experiments, the $k$-means and fuzzy $c$-means methods were run 100 times with the initial centroids randomly selected from the data set. The parameters of the $k$-means and fuzzy $c$-means methods were set to a termination criterion $\epsilon = 0.001$, and weighting exponent $m = 2.0$. The parameters of the subtractive and kernel-based subtractive methods were set to $\alpha = 5.4$ and $\beta = 1.5$ as suggested by Pal and Chakraborty (2000). The Gaussian kernel was used. The various methods were applied to the same five data sets, referred to as the X30 (Bezdek and Pal, 1998), BENSAID (Bensaid et al., 1996), DUNN (Dunn, 1974), IRIS (Bezdek et al., 1999), and ELLIPSE data sets. Figs. 1–4 show scatterplots of the clustering results of the subtractive and the proposed methods. Tables 1–5 display the clustering results of the centroids, the mountain values, and the number of misclassified data of the subtractive method and the proposed method for the five data sets. Table 6 summarizes the results obtained when the $k$-means, fuzzy $c$-means, subtractive, and proposed clustering methods were applied to the five data sets; for each data set, the highest accuracy value is marked in bold face.

The clustering results were assessed using Huang's accuracy measure ($r$) (Huang and Ng, 1999):

$$r = \frac{\sum_{i=1}^{k} a_i}{n} \quad (20)$$

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

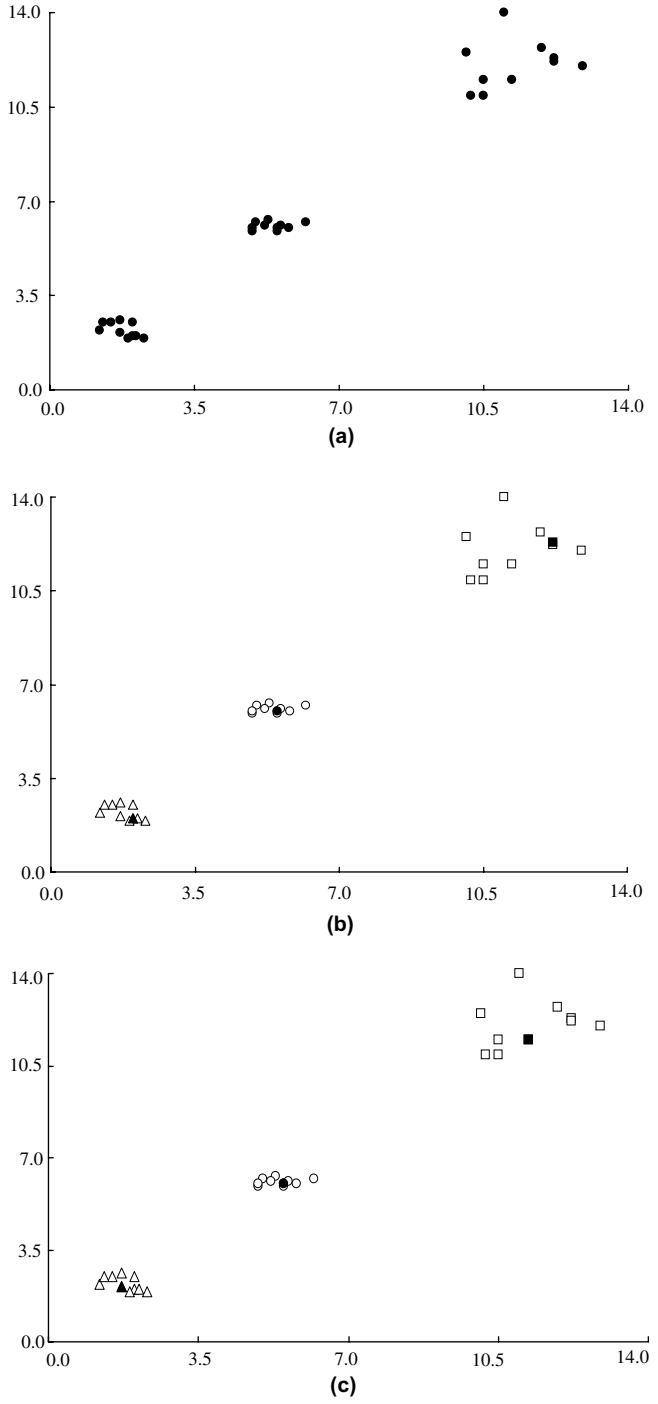


Fig. 1. Comparison of the results: (a) data set "X30"; (b) clustering using the subtractive method; and (c) clustering using the proposed method.

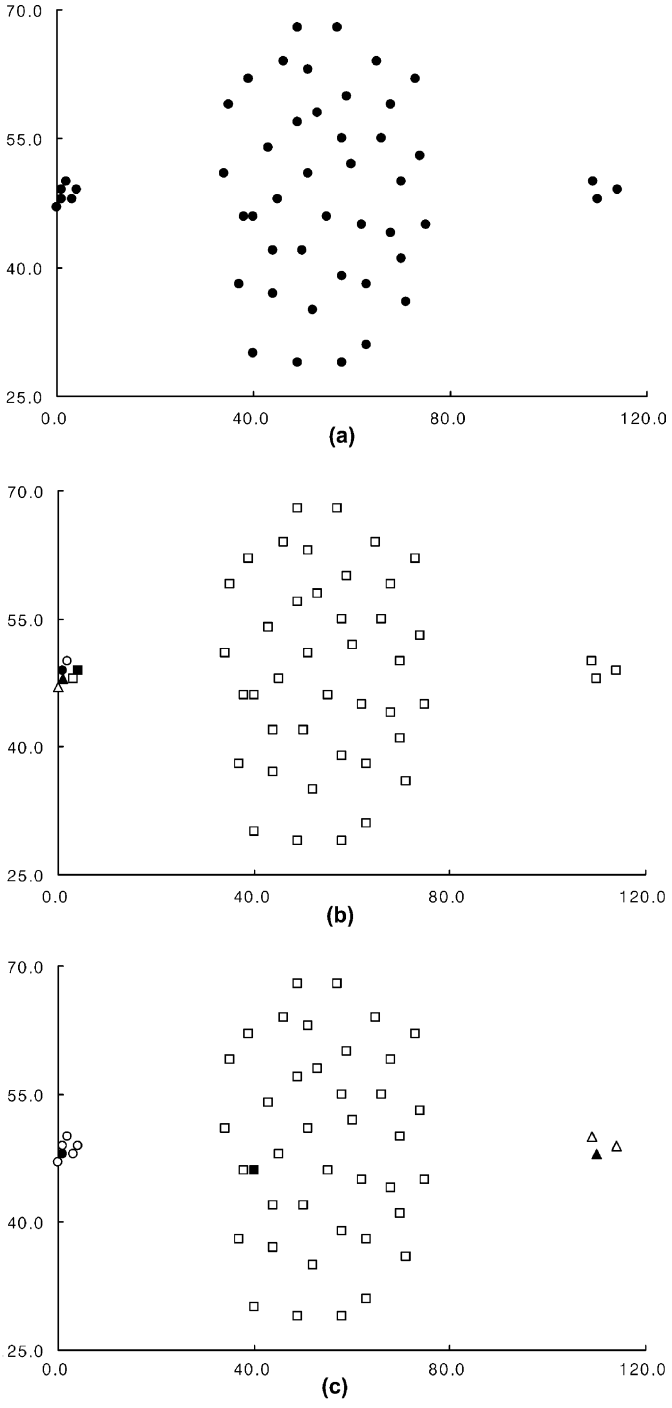


Fig. 2. Comparison of the results: (a) data set "BENSAID"; (b) clustering using the subtractive method; and (c) clustering using the proposed method.

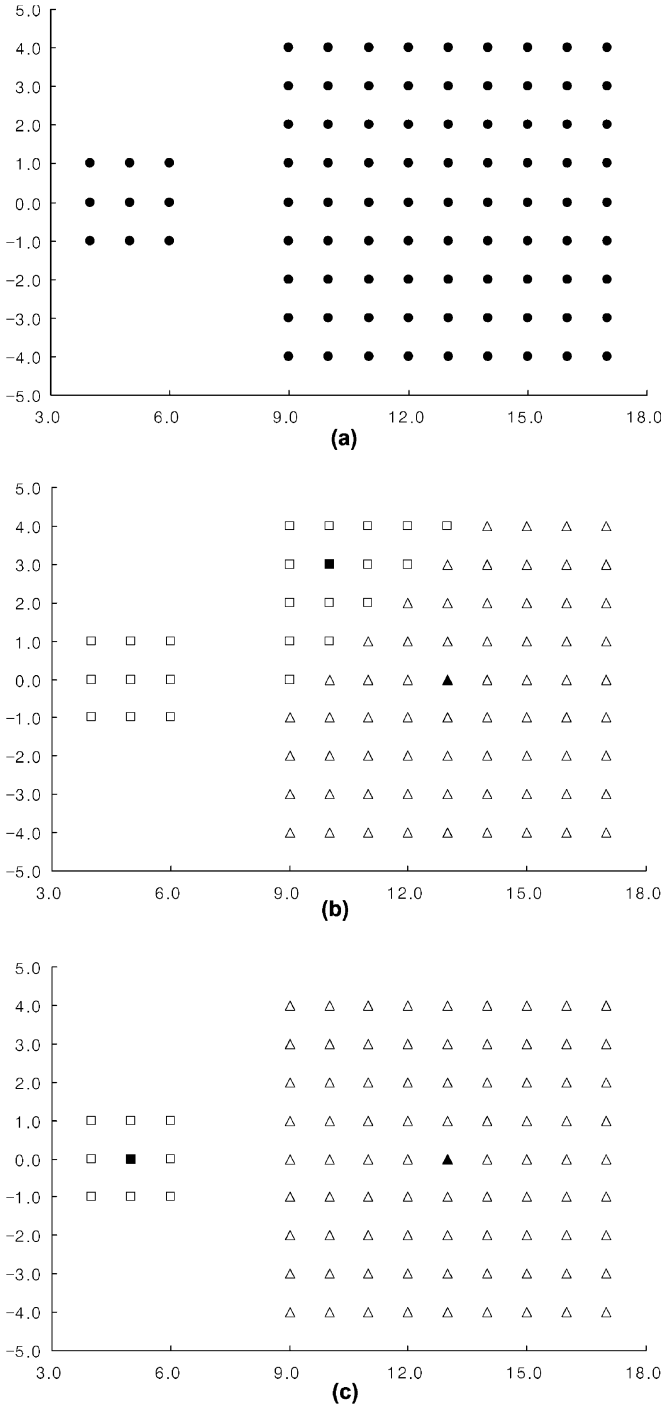


Fig. 3. Comparison of the results: (a) data set "DUNN"; (b) clustering using the subtractive method; and (c) clustering using the proposed method.

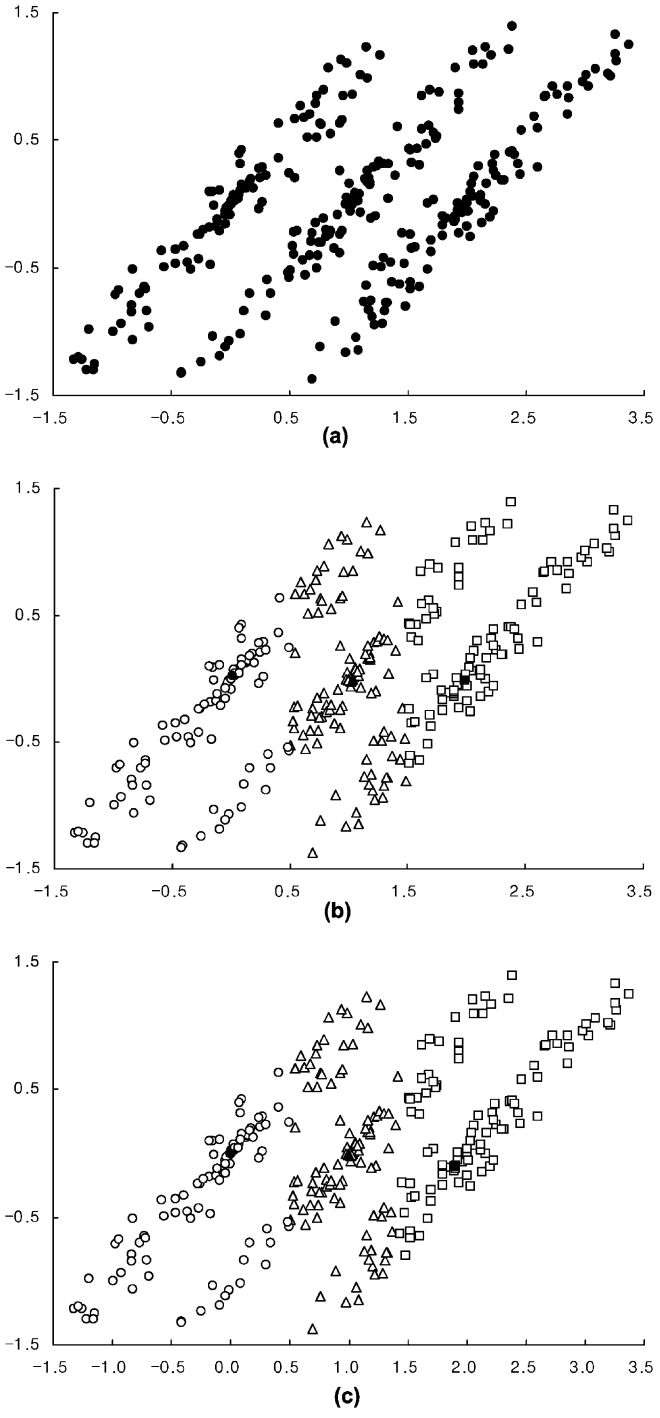


Fig. 4. Comparison of the results: (a) data set "ELLIPSE"; (b) clustering using the subtractive method; and (c) clustering using the proposed method.

Table 1
The centroids, their mountain values, and the number of misclassified data and accuracies using the subtractive and proposed methods for the X30 data set

| Methods | Centroids | | Mountain | Misclassification | Accuracy |
|---|---|---|---|---|---|
| Subtractive | 5.5 | 6.0 | 1.72 | 0 | 1.000 |
| | 2.0 | 2.0 | 1.55 | | |
| | 12.2 | 12.3 | 0.67 | | |
| Proposed | 5.5 | 6.0 | 8.69 | 0 | 1.000 |
| | 1.7 | 2.1 | 6.42 | | |
| | 11.2 | 11.5 | 4.62 | | |

Table 2
The centroids, their mountain values, and the number of misclassified data and accuracies using the subtractive and proposed methods for the BENSAID data set

| Methods | Centroids | | Mountain | Misclassification | Accuracy |
|---|---|---|---|---|---|
| Subtractive | 1.0 | 48.0 | 0.0050 | 7 | 0.857 |
| | 1.0 | 49.0 | 0.0030 | | |
| | 4.0 | 49.0 | 0.0004 | | |
| Proposed | 1.0 | 48.0 | 2.37 | 0 | 1.000 |
| | 40.0 | 46.0 | 0.33 | | |
| | 110.0 | 48.0 | 0.26 | | |

Table 3
The centroids, their mountain values, and the number of misclassified data and accuracies using the subtractive and proposed methods for the DUNN data set

| Methods | Centroids | | Mountain | Misclassification | Accuracy |
|---|---|---|---|---|---|
| Subtractive | 13.0 | 0.0 | 0.02 | 15 | 0.833 |
| | 10.0 | 3.0 | 0.02 | | |
| Proposed | 13.0 | 0.0 | 14.54 | 0 | 1.000 |
| | 5.0 | 0.0 | 4.50 | | |

Table 4
The centroids, their mountain values, and the number of misclassified data and accuracies using the subtractive and proposed methods for the IRIS data set

| Methods | Centroids | | | | Mountain | Misclassification | Accuracy |
|---|---|---|---|---|---|---|---|
| Subtractive | 4.90 | 3.10 | 1.50 | 0.10 | 6.48 | 71 | 0.527 |
| | 5.70 | 2.90 | 4.20 | 1.30 | 3.40 | | |
| | 5.10 | 3.50 | 1.40 | 0.20 | 3.11 | | |
| Proposed | 5.00 | 3.40 | 1.50 | 0.20 | 34.13 | 10 | 0.933 |
| | 6.20 | 2.80 | 4.80 | 1.80 | 30.69 | | |
| | 6.00 | 2.90 | 4.50 | 1.50 | 2.07 | | |

Fig. 1(a) shows the X30 data set (Bezdek and Pal, 1998), which contains $n = 30$ data points. This data set has three compact, well-separated clusters with 10 points per cluster. Fig. 1(b) and (c) show the clustering results obtained using the subtractive and proposed methods respectively. The clus-

Table 5
The centroids, their mountain values, and the number of misclassified data and accuracies using the subtractive and proposed methods for the ELLIPSE data set

| Methods | Centroids | | Mountain | Misclassification | Accuracy |
|---|---|---|---|---|---|
| Subtractive | 1.030 | −0.008 | 20.385 | 89 | 0.703 |
| | 0.011 | 0.021 | 15.532 | | |
| | 1.992 | −0.011 | 11.810 | | |
| Proposed | 0.998 | −0.006 | 3.943 | 85 | 0.717 |
| | −0.002 | −0.002 | 3.356 | | |
| | 1.897 | −0.108 | 2.509 | | |

Table 6
Clustering accuracy achieved by each clustering method for the five data sets

| Data set | *K*-means | Fuzzy *c*-means | Subtractive | Proposed |
|---|---|---|---|---|
| X30 | 0.838 | **1.000** | **1.000** | **1.000** |
| BENSAID | 0.815 | 0.769 | 0.857 | **1.000** |
| DUNN | 0.689 | 0.700 | 0.833 | **1.000** |
| IRIS | 0.833 | 0.893 | 0.527 | **0.933** |
| ELLIPSE | 0.644 | 0.658 | 0.703 | **0.717** |

ter centroids are marked in black. Both methods successfully identified the centroids. Table 1 lists the selected centroids, mountain values, number of misclassified data, and accuracies of the two methods. The subtractive and proposed methods both achieved perfect accuracy values of $r = 1.000$. The *k*-means and fuzzy *c*-means methods gave classification accuracies of $r = 0.838$ and $r = 1.000$ respectively (Table 6).

Fig. 2(a) shows a scatterplot of the BENSAID data set (Bensaid et al., 1996). This data set comprises 49 data points in two dimensional space, and consists of three clusters. Fig. 2(b) and (c) show the selected centroids and clustering results obtained using the subtractive and proposed methods respectively. The subtractive method did not clearly identify the cluster centroids (Fig. 2(b)), whereas the proposed method successfully selected the centroids (Fig. 2(c)). This is evident in the accuracy values, which were $r = 1.000$ for the proposed method compared to $r = 0.857$ for the subtractive method. Thus the proposed method was 14.3% more accurate than the subtractive method (Table 2). The *k*-means and fuzzy *c*-means methods gave accuracies of $r = 0.815$ and $r = 0.769$ respectively (Table 6).

Fig. 3(a) shows the third data set, Dunn (1974), which consists of 90 data points distributed in two clusters. The subtractive method failed to give correct clustering results (Fig. 3(b)); the centroid marked as a black rectangle was incorrectly chosen, leading to the misclassification of the top-left data of the right-side cluster. In contrast, the proposed method successfully identified the centroids of the two clusters (Fig. 3(c)). The accuracies of the subtractive and proposed methods were $r = 0.833$ and $r = 1.000$ respectively (Table 3), indicating that use of the kernel-based approach enhanced the accuracy by 16.7%. The *k*-means and fuzzy *c*-means methods gave lower accuracies of $r = 0.689$ and $r = 0.700$ respectively (Table 6). The tests on the BENSAID and DUNN sets demonstrate that the proposed method more clearly classified these systems with unequal-sized clusters compared to the other methods.

The fourth data set, the IRIS (Bezdek et al., 1999), has $n = 150$ data points in a four-dimensional space that are grouped in three physical clusters, two of which are overlapped. As indicated in Table 4, the subtractive method showed a low level of accuracy ($r = 0.527$) for this data set, with 71 data misclassified. In contrast, the proposed method gave an accuracy value of $r = 0.933$, and misclassified only 10 data points. In this case, the proposed approach was 47.3% more accurate than the subtractive method. The *k*-means and fuzzy *c*-means

methods showed the classification accuracies of $r = 0.833$ and $r = 0.893$ respectively (Table 6).

To test the performance of the kernel-based method for non-spherical clusters, we applied the four clustering methods to the ELLIPSE data set, shown in Fig. 4(a), which contains 300 data points. The optimal number of clusters for this data set is three. Both methods identified three centroids, one from each cluster (Fig. 4(b) and (c)). In Table 5, the subtractive method gave an accuracy of $r = 0.703$, and the proposed method was 1.4% more accurate, with a clustering accuracy of $r = 0.717$. The $k$-means and fuzzy $c$-means methods showed similar accuracies of $r = 0.644$ and $r = 0.658$ respectively (Table 6). Although the proposed method provided a better clustering result than the other methods, its accuracy for this data set was lower than the four other data sets considered. This indicates that the proposed approach is limited in its ability to classify non-spherical clusters.

In the test calculations, the proposed kernel-based approach gave markedly better clustering performance than the other three methods considered, highlighting the effectiveness and potential of the proposed method.

## 5. Conclusions

The conventional subtractive method is capable of efficiently clustering data; however, its precision and ability to correctly classify data are compromised by its use of data points as the cluster centroids. To address these shortcomings of the subtractive method, we developed a new kernel-based subtractive method in which a kernel function is used to calculate the mountains. A kernel function can implicitly map the input data to a high dimensional space in which data classification is easier. Compared to the conventional subtractive method, the kernel-induced mountain computation selects more desirable cluster centroids, thereby increasing the clustering accuracy. To test the performance of the proposed clustering method, it was applied to various data sets. The proposed method showed good clustering performance for most data sets, with the exception of non-spherical clusters. In future work, we plan to improve the mountain function to detect non-spherical clusters by considering the covariance information of clusters or employing the Gustafson–Kessel (GK) clustering algorithm (Bezdek et al., 1999).

## Acknowledgement

This work was supported by the Korean Systems Biology Research Grant (M1-0309-02-0002) from the Ministry of Science and Technology. We would like to thank Chung Moon Soul Center for BioInformation and BioElectronics and the IBM SUR program for providing research and computing facilities.

## References

Bensaid, A.M., et al., 1996. Validity-guided clustering with applications to image (re)segmentation. IEEE Trans. Fuzzy Systems 4 (2), 112–123.

Bezdek, J.C., Pal, N.R., 1998. Some new indexes of cluster validity. IEEE Trans. Systems, Man, Cybernet.-Part B: Cybernetics 28 (3), 301–315.

Bezdek, J.C., et al., 1999. Fuzzy Models and Algorithms for Pattern Recognition and Image Processing. Kluwer Academy Publishers, Boston.

Chiu, S.L., 1995. Extracting fuzzy rules for pattern classification by cluster estimation. In: The 6th Internat. Fuzzy Systems Association World Congress, p. 1–4.

Dunn, J., 1974. A fuzzy relative of the ISODATA process and its use in detecting compact well separated clusters. J. Cybernet. 3, 32–57.

Girolami, M., 2002. Mercer kernel-based clustering in feature space. IEEE Trans. Neural Networks 13 (3), 780–784.

Huang, Z., Ng, M.K., 1999. A fuzzy $k$-modes algorithm for clustering categorical data. IEEE Trans. Fuzzy Systems 7 (4), 446–452.

Jain, A.K., Dubes, R.C., 1998. Algorithms for Clustering. Prentice-Hall.

Jain, A.K., Murty, M.N., Flynn, P.J., 1999. Data clustering: a review. ACM Comput. Surveys 31 (3), 264–323.

Muller, K.-R., et al., 2001. An introduction to kernel-based learning algorithms. IEEE Trans. Neural Networks 12 (2), 181–202.

Pal, N.R., Chakraborty, D., 2000. Mountain and subtractive clusteing method: improvements and generalization. Internat. J. Intell. Systems 15, 329–341.

Scholkopf, B., Smola, A.J., 2002. Learning with Kernels. The MIT Press, Cambridge.

Vapnik, V.N., 1998. Statistical Learning Theory. Wiley, New York.

Velthuizen, R.P., et al., 1997. An investigation of mountain method clustering for large data sets. Pattern Recognition 30 (7), 1121–1135.

Wu, Z.D., Xie, W.W., 2003. Fuzzy *c*-means clustering algorithm based on kernel method. In: The fifth Internat. Conf. Comput. Intell. Multimedia Appl., p. 1–6.

Yager, R.R., Filev, D.P., 1994. Approximate clustering via the mountain method. IEEE Trans. Systems, Man, Cybernet. 24 (8), 1279–1284.

Yao, J., et al., 2000. Entropy-based fuzzy clustering and fuzzy modeling. Fuzzy Sets and Systems 113, 381–388.

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

Zhang, R., Rudnicky, A.I., 2002. A large scale clustering scheme for kernel *k*-means. In: The Sixteenth Internat. Conf. Pattern Recognition, p. 289–292.