A Similarity-Based Robust Clustering Method

Miin-Shen Yang and Kuo-Lung Wu

Abstract—This paper presents an alternating optimization clustering procedure called a similarity-based clustering method (SCM). It is an effective and robust approach to clustering on the basis of a total similarity objective function related to the approximate density shape estimation. We show that the data points in SCM can self-organize local optimal cluster number and volumes without using cluster validity functions or a variance-covariance matrix. The proposed clustering method is also robust to noise and outliers based on the influence function and gross error sensitivity analysis. Therefore, SCM exhibits three robust clustering characteristics: 1) robust to the initialization (cluster number and initial guesses), 2) robust to cluster volumes (ability to detect different volumes of clusters), and 3) robust to noise and outliers. Several numerical data sets and actual data are used in the SCM to show these good aspects. The computational complexity of SCM is also analyzed. Some experimental results of comparing the proposed SCM with the existing methods show the superiority of the SCM method.

Index Terms—Robust clustering algorithm, fuzzy clustering, alternating optimization algorithm, total similarity, noise.

1 Introduction

Cluster analysis is a branch in statistical multivariate analysis and an unsupervised learning in pattern recognition [9], [17], [19]. It is a method for clustering a data set into most similar groups in the same cluster and most dissimilar groups in different clusters. Generally, we may roughly divide clustering methods into the following categories: hierarchical clustering [15], [19], mixture-model clustering [25], [26], learning-network clustering [13], [20], [23], [28], objective-function-based clustering, and partition clustering, etc. [2], [19], [29]. However, most of these clustering methods are less to include the property of robustness. A robust idea is important in pattern recognition [7], [11], [27], [34]. By combining the robust statistic with the M-estimator [7], [16], many clustering algorithms may be robust to noise and outliers. However, robustness in clustering, in general, involves three aspects: 1) robust to the initialization (cluster number and initial guesses), 2) robust to cluster volumes (ability to detect different volumes of clusters), and 3) robustness to noise and outliers (ability to tolerate noise and outliers).

Clustering methods for solving an unknown cluster number can be classified into three main categories: hierarchical clustering, validity measure, and progressive clustering. In hierarchical clustering, the cluster number need not be known a priori and initialization problems do not arise. If the clusters are separated well, the Agglomerative Hierarchical Clustering (AHC) with single linkage can easily detect the optimal cluster number. However, hierarchical clustering considers only local neighbors in each step. The global shape and size of clusters are always ignored. The methods for considering the global shape and size of clusters are the well-known prototype-based clustering algorithms [2], [14], [21], [29]. Because the cluster number in these algorithms needs to be specified a priori, these clustering methods need to use some validity measures [3], [8], [24], [30]. For a given cluster-number range, the validity measure is evaluated for each given cluster number and then an optimal number is chosen for these validity measures. This method for searching an optimal cluster number is dependent on the selected clustering algorithm, whose performance is still dependent on the initial cluster centers.

Another method involves performing a progressive clustering [21] by initializing an overspecified cluster number. After convergence, spurious clusters are eliminated and compatible clusters are merged. The problem with this method is to define the spurious and compatible clusters. An analogical method that combines the concepts of progressive clustering and agglomerative hierarchical clustering was developed by Frigui and Krishnapuram [10], [11]. Moreover, although overspecification of the cluster number can reduce the initial cluster center effects, they can not guarantee that all clusters in the data set will be found. An alternative version of progressive clustering was proposed to extract one cluster at a time, such as Jolion et al. [18], Yager and Filev [31], and Chiu [5]. The performance of this approach depends on the validity measures for the individual and global extracted clusters.

Using the Euclidean norm in prototype-based clustering algorithm has the tendency to produce equal volumes of clusters. A widely used method for detecting different volumes of clusters is to adopt the variance-covariance matrix, such as Gustafson and Kessel [14] and Gath and Geva [12]. In unsupervised clustering, the performance still depends on the initialization technique and the chosen validity measures. Dave and Krishnapuram [7] gave a unified view of methods to make clustering algorithms robust by adopting a weight function for the prototype update equation. Algorithms with this technique can

- *M.-S. Yang is with the Department of Applied Mathematics, Chung Yuan Christian University, Chung-Li, Taiwan 32023, ROC. E-mail: msyang@math.cycu.edu.tw.*
- *K.-L. Wu is with the Department of Information Management, Kun Shan University of Technology, Yung-Kang, Tainan, Taiwan 71023, ROC. E-mail: klwu@mail.ksut.edu.tw.*

Manuscript received 31 July 2002; revised 19 May 2003; accepted 26 Oct. 2003.
Recommended for acceptance by S. Pal.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 117041.

analyze the robust properties against noise and outliers using many statistical properties such as the breakdown point, local-shift sensitivity, gross error sensitivity, and influence function [16]. In unsupervised clustering, it is difficult to separate outliers from clusters or to identify actual outliers. The use of a validity measure or a progressive clustering depends up on the constructed validity function.

In this study, we embed an initialization technique into the proposed similarly clustering algorithm (SCA). The data points with our proposed clustering method can self-organize the cluster number and cluster structures. By processing the final states of the data points into the Agglomerative Hierarchical Clustering (AHC) algorithm, a local optimal cluster number c^* and the volumes of these c^* clusters are determined simultaneously. The proposed clustering method is called the similarity-based clustering method (SCM) which is robust to initializations and cluster volumes. The organization of this paper is as follows: In Section 2, the proposed clustering method and its objective function are described. We give a tool for analyzing the important parameter γ and then create the correlation comparison algorithm (CCA). To consider the robustness to initializations such as the cluster number and initial cluster centers, we constructed the SCA and its classification procedure. The method including CCA, SCA, and AHC is called the SCM. In Section 3, some theoretical studies for robustness are given. We use the gross error sensitivity and influence function to discuss the robust properties to noise and outliers in the SCA. In Section 4, we use the statistical hypothesis testing to analyze the rule for searching for good CCA parameters. In Section 5, several numerical examples are presented to show the effectiveness of the proposed SCM algorithm. In Section 6, we make the comparisons of SCM with the best-known fuzzy c-means (FCM) [1] and possibilistic c-means (PCM) algorithms. We also analyze the computational complexity of these algorithms. Conclusions and discussion are made in Section 7.

2 A Similarity-Based Clustering Method and Its Objective Function

In general, most clustering algorithms are procedures that minimize the total dissimilarity for example, k-means [9], [15], fuzzy c-means (FCM) [2], [32], and possibilistic c-means (PCM) [22], etc. Let the data set be $X = \{x_1, \ldots, x_n\}$ where x_j is a feature vector in the s-dimensional Euclidean space R^s and c is the specified number of clusters. We use the Euclidean norm $\|x_j - z_i\|^2$ as the dissimilarity measure between x_j and the ith cluster center z_i. If we want to cluster X into c clusters, we may find z_i to minimize the total dissimilarity objective function. This forms the main k-means structure. FCM extends k-means to be a fuzzy clustering with fuzzy sets as membership functions. PCM relaxes the membership function restrictions in FCM to produce a possibilistic-type clustering algorithm. However, these clustering algorithms are still procedures that minimize the total dissimilarity measure.

We consider $S(x_j, z_i)$ as the similarity measure between x_j and the ith cluster center z_i. Our goal is to find z_i to maximize the total similarity measure $J_s(\mathbf{z})$ with

$$J_s(\mathbf{z}) = \sum_{i=1}^{c} \sum_{j=1}^{n} f(S(x_j, z_i)), \tag{1}$$

where f is a monotone increasing function and $z = (z_1, \cdots, z_c)$. Since $f(S(x_j, z_i))$ is still a reasonable similarity measure, $J_s(z)$ is a meaningful clustering objective function. The problem here is selecting a useful and reasonable $S(x_j, z_i)$ and f. In our similarity-based clustering method (SCM), we use the similarity relation $S(x_j, z_i)$ given in Zadeh [33] with

$$S(x_j, z_i) = \exp\left(-\frac{\|x_j - z_i\|^2}{\beta}\right), \tag{2}$$

where β is the normalized term. Note that the distance measure here is chosen with the Euclidean norm $\|x_j - z_i\|^2$. However, any suitable distance measure can be used to replace the Euclidean norm. Throughout this paper, we use the Euclidean norm. Let the monotone increasing function $f(\cdot)$ be

$$f(\cdot) = (\cdot)^{\gamma} \ , \ \gamma > 0. \tag{3}$$

Our clustering method can then be set up by maximizing the total similarity measure $J_s(z)$ with

$$J_s(\mathbf{z}) = \sum_{i=1}^{c} \sum_{j=1}^{n} \left(\exp - \frac{\|x_j - z_i\|^2}{\beta}\right)^{\gamma}. \tag{4}$$

We mentioned that using the power parameter γ in the monotone increasing function (3) is important because it can take over the effect of the normalized parameter β so that we can assign the estimate of β as the sample variance. That is, β could be defined by

$$\beta = \frac{\sum_{j=1}^{n} \|x_j - \overline{x}\|^2}{n} \quad \text{where} \quad \overline{x} = \frac{\sum_{j=1}^{n} x_j}{n}. \tag{5}$$

We then analyze the effect of parameter γ and propose a tool to acquire a good estimate. According to the analysis of γ, we know that γ can determine the location of peaks in the objective function $J_s(\mathbf{z})$. A good γ estimate will induce a good clustering result. Thus, β is no longer sensitive to the result.

We know that to maximize the total similarity measure $J_s(\mathbf{z})$ is a way to find the peaks of the objective function $J_s(\mathbf{z})$. The parameter γ can determine the location of the peaks of $J_s(\mathbf{z})$. We shall first propose a tool for analyzing this parameter. To analyze the effect of γ, let $\widetilde{J}_s(x_k)$ be the total similarity of the data point x_k to all data points with

$$\widetilde{J}_s(x_k) = \sum_{j=1}^{n} \left(\exp - \frac{\|x_j - x_k\|^2}{\beta}\right)^{\gamma}, \quad k = 1, \cdots, n. \tag{6}$$

This function can be seen closely related to the density shape of the data points in the neighborhood of x_k. A large value for $\widetilde{J}_s(x_k)$ means that the data point x_k is close to some cluster centers and has many data points around it. This

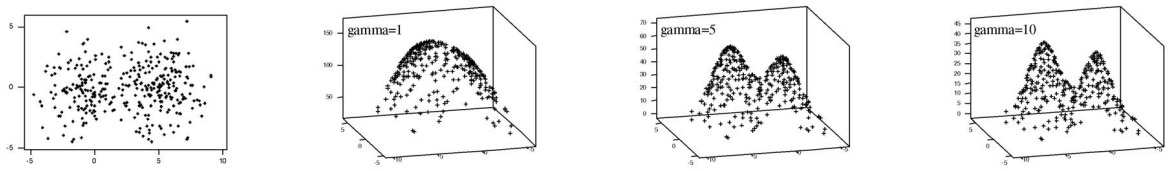


Fig. 1. (a) Two-clusters data set. (b), (c), and (d) are 3D plots of (6) (the approximate density shapes) with gamma=1, 5, and 10, respectively.

TABLE 1
The Values of CCA

data set	5 and 10	10 and 15	15 and 20	selected gamma
Figure 1	0.992	0.997	0.998	5
Example 1	0.922	0.973	0.990	10
Example 4	0.677	0.836	0.990	15
Example 5	0.711	0.955	0.990	15
Example 6	0.974	0.994	0.998	5
IRIS data	0.922	0.985	0.994	10
Example 7	0.966	0.985	0.992	10

function is equivalent to the mountain function proposed by Yager and Filev [31] and modified by Chiu [5]. Based on $\widetilde{J}_s(x_k)$, we provide a tool for analyzing the effect of parameter γ and give a method for a better choice of γ.

First, we use the data set shown in Fig. 1a to see the influence of γ on (6). In Figs. 1b, 1c, and 1d, we show the 3D plots for (6) with $\gamma = 1, 5$ and 10, respectively. Note that the "+" sign means the value of $\widetilde{J}_s(x_k)$ with respect to the data point x_k, $k = 1, \ldots, n$. According to Fig. 1b, the objective function $J_s(\mathbf{z})$ seems to have only one peak when $\gamma = 1$ and the peaks will be separated when γ increase to 5 and 10 as shown in Figs. 1c and 1d. (6) can show that the data set has two peaks (clusters) after $\gamma = 5$. This means that when $\gamma = 1$, the objective function $J_s(z)$ will have only one optimizer (peak) and when $\gamma = 5$ or 10, the objective function $J_s(\mathbf{z})$ actually detects the data set as two separate clusters. The question here is which selection of γ can represent the actual density shape of the data set. Although the 3D plots of $\widetilde{J}_s(x_k)$ shown as Fig. 1 can help us to make a decision, this kind of subjective impression is restricted in low dimensional cases. Thus, a more precise method must be considered.

We can see that Figs. 1b and 1c show differences when $\gamma = 1$ and $\gamma = 5$. But, Figs. 1c and 1d show similarities. The reason is that the values of $\widetilde{J}_s(x_k)$ in the data have a very high relationship between the $\gamma = 5$ and $\gamma = 10$ cases. In the statistical sense, the correlation between the values of $\widetilde{J}_s(x_k)$ when $\gamma = 5$ and $\gamma = 10$ is very close to one. If the pairwise comparisons of these correlations are larger than a given threshold, (6) with this γ value will provide a good approximate density shape for the data set shown in Figs. 1c and 1d. We mention that the function $\widetilde{J}_s(x_k)$ in (6) is closely related to the density shape of the data points in the neighborhood of x_k where the parameter γ is highly related to its neighborhood radius (or boundary). Thus, to increase γ is equivalent to decreasing the neighborhood radius of the data point x_k. The values of correlation comparison in (6) are larger than a given threshold, which means that the approximate density shapes are not altered when we decrease the neighborhood radius of the data points. Therefore, the approximate density shapes are stable with such a γ value and it will be a good estimate for the exact density shape.

To observe the change in the SCM objective function globally, not locally, we may increase γ by $\gamma = 1, 5, 10, \cdots$ etc. A large increased shift for γ in our correlation comparison procedure may lose a good estimate for the parameter γ. This is not suggested. However, a small increased shift for γ will decrease the neighborhood radius slowly for the data points where the shapes of $\tilde{J}_s(x_k)$ may make no significant difference so that the procedure becomes a time consumer. Of course, an ideal choice of the shift of γ should be so dependent on the data set that it is difficult to find this ideal shift of γ. In general, the shift of five is recommended to be adopted, and the correlations of "$\gamma=1, \gamma=5$," "$\gamma = 5, \gamma = 10$," "$\gamma = 10, \gamma = 15$" will increase when γ increases. However, a too large γ gives a very small neighborhood radius for the data points and each data point will become an individual cluster. Thus, approximating the density shape with too large a γ will overestimate the true density shape and the objective function will have too many peaks. In another direction, the objective function will have only one peak when γ is small even though the data set actually has many clusters. Approximating the density shape with too small a γ will underestimate the true density shape. In general, the SCM objective function will have only one peak when γ is small, say for example, $\gamma < 5$. Thus, we may start the correlation comparison with $\gamma = 5$. In our experiments, we suggest a threshold around $0.97 \sim 0.999$. The statistical hypothesis testing for the threshold of γ will be discussed in Section 4. In our simulation, we chose 0.97 for the threshold. The correlations for the data set in Fig. 1 are shown in the first row of Table 1. The next example will provide us more information about the effect of γ.

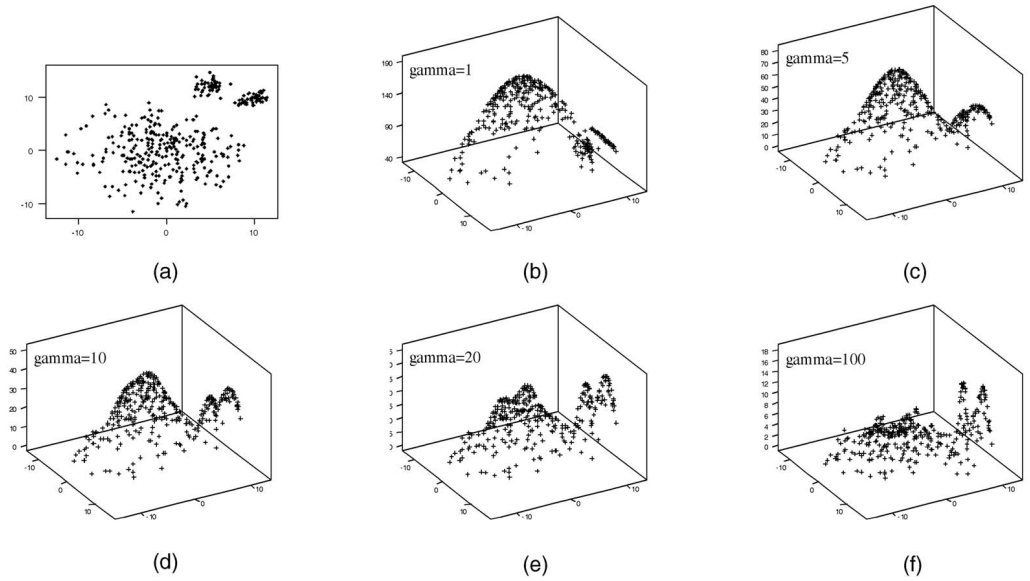


Fig. 2. (a) Three-clusters data set. (b), (c), (d), (e), and (f) are the approximate density shapes with gamma=1, 5, 10, 20, and 100, respectively.

Example 1. In the data set of Fig. 2a, there is one large cluster and two small clusters. The correlation comparison of different γ are illustrated in the second row of Table 1. Under the suggested threshold, $\gamma = 10$ will be a good estimate. Fig. 2d also gives us the same information. In Fig. 2b, only one cluster will be found when $\gamma = 1$. As $\gamma = 5$ in Fig. 2c, the two small clusters are very close. When $\gamma = 10$, the three distinguishable peaks can be shown in Fig. 2d. The case of a large γ with $\gamma = 20$ is plotted in Fig. 2e and the area of the large cluster of this data seems to have more than one peak. This phenomenon can be shown obviously in Fig. 2f when $\gamma = 100$. The SCM objective function will have so many peaks (a large number and confusion clusters) in this very large γ situation.

We propose a tool to select γ using a correlation comparison procedure with "$\gamma=5, \gamma=10$," "$\gamma=10, \gamma=15$," "$\gamma=15, \gamma=20$," $\cdots etc.$ Under this method, we can easily find a good estimate of γ via the density shape estimation concept and the underestimate (small γ) and overestimate (large γ) are undesirable. In order to execute the correlation comparison procedure as a computer program, (6) is rewritten as

$$\widetilde{J}_s(x_k)_{\gamma_m} = \sum_{j=1}^{n} \left(\exp - \frac{\|x_j - x_k\|^2}{\beta} \right)^{\gamma_m}, \quad k = 1, \cdots, n, \tag{7}$$

where $\gamma_m = 5m,\ m = 1,\ 2,\ 3,\ \cdots$. The correlation comparison algorithm (CCA) can then be summarized as follows:

Correlation Comparison Algorithm (CCA)

Step 1. Set $m = 1$ and $\epsilon_1 = 0.97$.
Step 2. Calculate the correlation of the values of $\widetilde{J}_s(x_k)_{\gamma_m}$ and $\widetilde{J}_s(x_k)_{\gamma_{(m+1)}}$.
Step 3. IF the correlation is greater than or equal to the specified ϵ_1,
THEN choose γ_m to be the estimate of γ;
ELSE $m = m + 1$ and GOTO Step 2

After we estimate the approximate density shape (parameter γ) of the data set using the SCM objective function, the next step is to find a z_i that maximizes $J_s(\mathbf{z})$ (i.e., the peak of the SCM objective function). We differentiate $J_s(\mathbf{z})$ with respect to all z_i using

$$\frac{dJ_s(\mathbf{z})}{dz_i} = \sum_{j=1}^{n} 2\frac{\gamma}{\beta}(x_j - z_i) \left(\exp - \frac{\|x_j - z_i\|^2}{\beta} \right)^{\gamma} \tag{8}$$

and set (8) to zero. The necessary condition that maximizes $J_s(\mathbf{z})$ is

$$z_i = \frac{\sum_{j=1}^{n} x_j \left(\exp - \frac{\|x_j - z_i\|^2}{\beta} \right)^{\gamma}}{\sum_{j=1}^{n} \left(\exp - \frac{\|x_j - z_i\|^2}{\beta} \right)^{\gamma}}. \tag{9}$$

This necessary condition can be decomposed into two conditions. First, we take the similarity relations $S(x_j, z_i)$ with

$$S_{ij} = S(x_j, z_i) = \exp \left(- \frac{\|x_j - z_i\|^2}{\beta} \right) \tag{10}$$

and then the necessary condition (9) becomes

$$z_i = \frac{\sum_{j=1}^{n} S_{ij}^{\gamma}\, x_j}{\sum_{j=1}^{n} S_{ij}^{\gamma}}. \tag{11}$$

Here, β is assigned to be the sample variance as (5). Note that z_i in (9) cannot be solved directly. However, we can used the fixed-point iterative method to approximate it. Let the right side of (9) be $f(z_i)$. The first step is to specify the initial value $z_i^{(0)}$ and then compute $f(z_i^{(0)})$ and set it to be $z_i^{(1)}$. Repeat the steps until the $(l+1)$th solution $z_i^{(l+1)}$ is very close to the lth solution. This forms the proposed similarity clustering algorithm (SCA). The combination of (10) and (11) is equivalent to the necessary condition (9). Thus, after

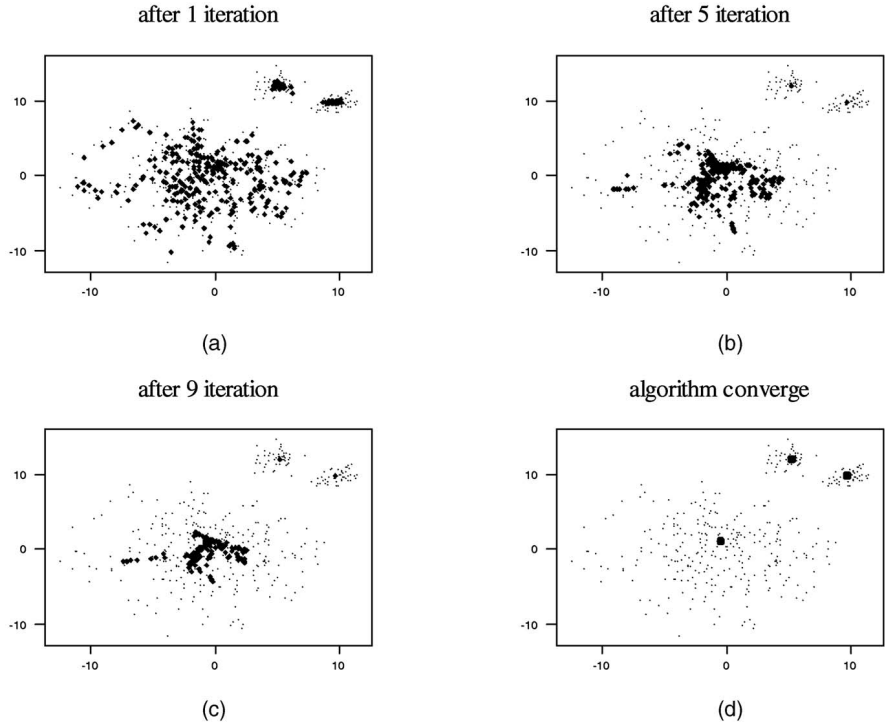


Fig. 3. The states of the data points (initial cluster centers) in SCA after 1, 5, and 9 iterations and convergence.

the CCA is implemented to get a good estimate γ, the SCA will be used to find the peaks (modes) of the approximate density shape and is then summarized as follows:

Similarity Clustering Algorithm (SCA)

Initialize $z_i^{(0)}$, $i = 1, \ldots, c$ and give ϵ_2;

Set iteration counter $\ell = 0$;

Step 1. Estimate $S_{ij}^{(\ell+1)}$ by (10);

Step 2. Estimate $z_i^{(\ell+1)}$ by (11);

Increment ℓ ; Until $\max_i \|z_i^{(\ell+1)} - z_i^{(\ell)}\| < \epsilon_2$.

Suppose that the data set has only one peak on the SCM objective function, all random initial centers will be then centralized to that unique peak (optimizer). This property can be used to solve the cluster validity problem by finding all peaks of the SCM objective function. We can randomly give more initial cluster centers to process SCA. These initial centers will be then centralized to the peaks of the SCM objective function. This idea can be found in some progressive clustering methods [10], [11], [21]. The problem here is what kind of the initialization can guarantee that all peaks (clusters) will be found out simultaneously. This problem is equivalent to what kind of the initialization technique can solve the validity problem correctly. Intuitively, if we let all data points to be the initial centers (i.e., $z^{(0)} = (z_1^{(0)}, \cdots, z_n^{(0)}) = (x_1, \cdots, x_n)$), all peaks (clusters) will be found and the number of peaks will be the optimal cluster number c^*. The following is a simple example.

Example 2. We process SCA with the data shown in Fig. 2a by initializing $z^{(0)} = (z_1^{(0)}, \cdots, z_n^{(0)}) = (x_1, \cdots, x_n)$. All initial values are centralized to three peaks (cluster centers) as shown in Fig. 3d. We show the states of these initial cluster centers after 1, 5, and 9 iterations in Figs. 3a, 3b, and 3c, respectively. The convergent (final) states of all initial cluster centers are shown in Fig. 3d. The optimal cluster number c^* is three for this data set. This result coincides to Fig. 2d. By specifying $z^{(0)} = (z_1^{(0)}, \cdots, z_n^{(0)}) = (x_1, \cdots, x_n)$, SCA is robust to the initializations (cluster number and initial guesses).

The final SCA result, of course, has n final cluster centers (the final states of all n initial data points) and the optimal cluster number c^* for the above example can be observed by the view of sight as shown in Fig. 3d. However, a precise method to determine the optimal cluster number c^* from these final n cluster centers should be provided. We choose this method by processing the Agglomerative Hierarchical Clustering (AHC) with the final states of all cluster centers to find the optimal c^*. The chosen dissimilarity measure for AHC is the Euclidean norm which had been used in our total similarity measure $J_s(z)$. In general, single linkage method is used as a linkage way in our AHC. Note that, there are many methods to process AHC such as the complete linkage method, unweighted pair group method using arithmetic average, weighted pair group method using centroid, and Ward's method using a minimum variance, etc. [9], [19]). Since the final states of n initial cluster centers ($\mathbf{z}^{(0)} = (z_1^{(0)}, \cdots, z_n^{(0)}) = (x_1, \cdots, x_n)$) are centralized to c^* cluster centers, all above AHC methods may present the same clustering results. The Hierarchical Clustering tree of the final states of all data points (all initial cluster centers) of Example 2 is shown in Fig. 4. The increase in y-coordinate represents the distance between clusters. Fig. 4 shows that there are three well separated clusters in the final states of the data points and hence $c^* = 3$. Three cluster centers can be selected randomly from three groups of the final state of $(x_1, \cdots, x_n)$, respectively.

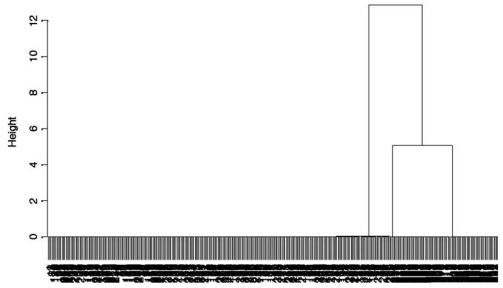


Fig. 4. The Hierarchical Clustering tree of the final states of the data points of Example 2. The increase in Y-coordinate represents the distance between clusters. There are three well-separated clusters in the final states of the data points. Since too many data points need to be merged, the labels of data points (in the bottom of the figure) are not clear.

The Hierarchical Clustering tree was created using the statistical package "S-plus." This method can also be applied to a high dimensional data set. We did not set a threshold to merge clusters such as many progressive clustering methods do, because the Hierarchical Clustering tree can take over the effect of the selected threshold. The AHC results can also provide us the classification result with different volumes of clusters which will be discussed below.

Because we processed SCA with $\mathbf{z}^{(0)} = (z_1^{(0)}, \cdots, z_n^{(0)}) = (x_1, \cdots, x_n)$, the final n cluster centers will present the final states of all data points. This procedure provides us with a method to classify the data using their final states. Suppose $z_1^{(0)}$ and $z_n^{(0)}$ will converge to the same peak, this provides us the information that $z_1^{(0)}$ and $z_n^{(0)}$ should belong to the same cluster and so should x_1 and x_n. Hence, when $\mathbf{z}^{(0)} = (z_1^{(0)}, \cdots, z_n^{(0)}) = (x_1, \cdots, x_n)$ is centralized to c^* clusters, the data set will also be classified into those c^* clusters simultaneously. By processing the final states of $\mathbf{z}^{(0)} = (z_1^{(0)}, \cdots, z_n^{(0)}) = (x_1, \cdots, x_n)$ into AHC, the optimal cluster number c^* and the identified clusters will be found simultaneously.

Example 3. The identified clusters for Example 2 using above procedure are shown in Fig. 5. Different volumes of clusters can be detected using the above classification procedure. This result is difficult to be achieved by many clustering algorithms such as k-means or fuzzy c-means. Detecting different volumes of clusters is an important problem in pattern recognition. To solve this problem, most used methods adopt a variance-covariance matrix to measure the cluster volumes such as Gustafson and Kessel [14] and Gath and Geva [12], etc. However, this will make the algorithm complex. In our idea, data points via the SCA can selfly organize the cluster number and the structures. This characteristic presents the superiority of the proposed procedure over these existing clustering methods. We accomplish this objective simply with a new idea, cooperating with AHC.

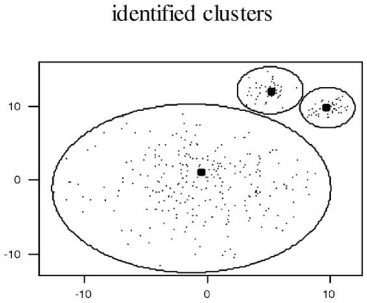


Fig. 5. The identified clusters by processing AHC with the final states of the data points in Example 2.

So far, we can completely cluster the data set of Fig. 2a into three different volume clusters shown in Fig. 5. Our process includes that first, process CCA to estimate the approximate density shape of the data set, second, process SCA with $\mathbf{z}^{(0)} = (z_1^{(0)}, \cdots, z_n^{(0)}) = (x_1, \cdots, x_n)$ and finally, process AHC with the final states of $\mathbf{z}^{(0)} = (z_1^{(0)}, \cdots, z_n^{(0)}) = (x_1, \cdots, x_n)$ to find the optimal cluster number c^* and identify these c^* clusters. This forms the structure of SCM which is shown in Fig. 6 and summarized as follows:

Similarity-Based Clustering Method (SCM)

Step 1. Estimate γ using CCA.
Step 2. Self-organize the data using SCA.
Step 3. Process AHC with the final states of the data points.
Step 4. Find c^* according to the Hierarchical Clustering tree.
Step 5. Identify these c^* clusters.

SCM can obtain the cluster number c^* and identify these c^* clusters for an unknown cluster number data set. However, in some situations, we may wish to fix the cluster number (i.e., c^* equals to a given positive integer c). In this situation, we can adjust SCM by processing AHC with a fixed cluster number c and these c clusters will then be obtained. In next section, we have the theoretical analysis on the robust properties of SCM to noise and outliers.

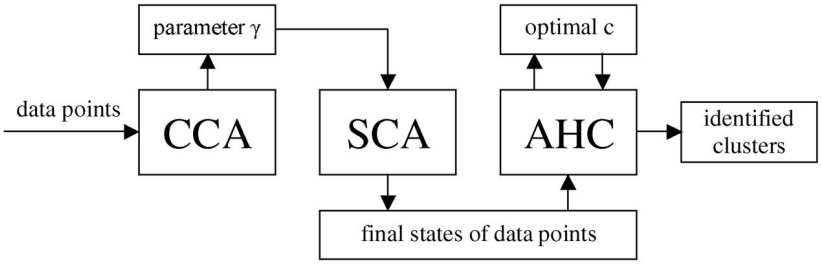
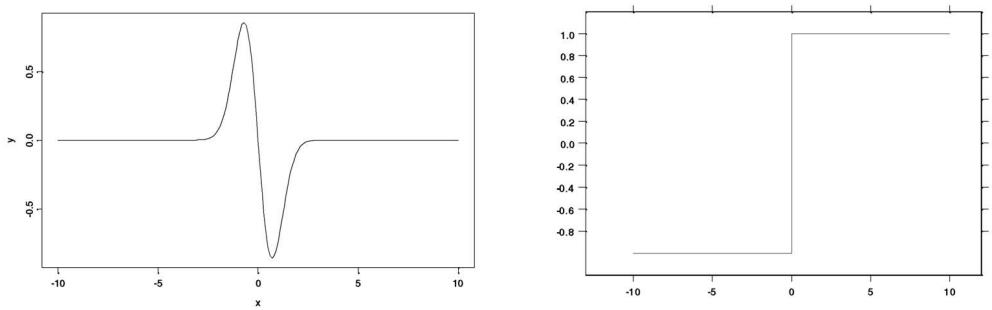


Fig. 6. The structure of SCM.



(a) (b)

Fig. 7. (a) φ function of our estimate. (b) φ function of the median.

3 The Robust Properties to Noise and Outliers

A good clustering method should have the ability to tolerate noise and detect outliers in the data set. Many criteria such as the breakdown point, local-shift sensitivity, gross error sensitivity, and influence function [16] can be used to measure the robustness. We used the gross error sensitivity and influence function to show that our weighted cluster center update equation is robust to noise and outliers. Let $\{x_1, \cdots, x_n\}$ be an observed data set of real numbers and θ is an unknown parameter to be estimated. An M-estimator [16] is generated by minimizing the form

$$\sum_{j=1}^{n} \rho(x_j; \theta), \tag{12}$$

where ρ is an arbitrary function that can measure the loss of x_j and θ (i.e., the dissimilarity measure of x_j and θ). Here, we have an interesting location estimate that minimizes

$$\sum_{j=1}^{n} \rho(x_j - \theta) \tag{13}$$

and the M-estimator is generated by solving the equation

$$\sum_{j=1}^{n} \varphi(x_j - \theta) = 0, \tag{14}$$

where $\varphi(x_j - \theta) = (\partial/\partial\theta)\rho(x_j - \theta)$. If we take the loss function of x_j and θ with $\rho(x_j - \theta) = (x_j - \theta)^2$, the M-estimator is the sample mean, which is equivalent to the classical least square (LS) estimator and if $\rho(x_j - \theta) = | x_j - \theta |$, the M-estimator is the median.

The influence function or influence curve (IC) can help us to assess the relative influence of individual observations toward the value of an estimate. The M-estimator has been shown that its influence function is proportional to its φ function [16]. In the location problem, we have the influence function of an M-estimator with

$$\mathrm{IC}(x; F, \theta) = \frac{\varphi(x - \theta)}{\int \varphi'(x - \theta) dF_X(x)}, \tag{15}$$

where $F_X(x)$ denotes the distribution function of X. If the influence function of an estimator is unbounded, an outlier might cause trouble. Many important robustness measures can be observed from the influence function. One of the important measures is the gross error sensitivity γ^*, defined by

$$\gamma^* = \sup_x | \mathrm{IC}(x; F, \theta) | . \tag{16}$$

This quantity can interpret the worst approximate influence that the addition of an infinitesimal point mass can have on the value of the associated estimator. Let

$$\rho(x - z) = 1 - \exp(-\beta^{-1}(x_j - z)^2)^{\gamma}. \tag{17}$$

Minimize (13) with (17) is equivalent to minimize

$$n - \sum_{j=1}^{n} \exp(-\beta^{-1}(x_j - z)^2)^{\gamma} \tag{18}$$

and also equivalent to maximize

$$\sum_{j=1}^{n} \exp(-\beta^{-1}(x_j - z)^2)^{\gamma} \tag{19}$$

which is the SCM objective function with one cluster. Our estimate of z ((9)) resulting from (8) is equivalent to an M-estimator with the similarity measure (2) replaced by a dissimilarity measure (17) and the φ function of our estimate is

$$\varphi(x_j - z) = \frac{-2\beta^{-1}(x_j - z)}{\exp(\beta^{-1}(x_j - z)^2)}. \tag{20}$$

Since the influence function $\mathrm{IC}(x_j; F, z)$ is a proportion to $\varphi(x_j - z)$ according to (15), we need only to analyze the term $\varphi(x_j - z)$. By applying the L'Hospital's rule, we have

$$\lim_{x_j \to \infty} \varphi(x_j - z) = \lim_{x_j \to -\infty} \varphi(x_j - z) = 0. \tag{21}$$

Thus, we have $\mathrm{IC}(x_j; F, z) = 0$ when x_j tends to positive or negative infinity. We can also obtain the maximum and minimum values of $\varphi(x - z)$ by solving

$$(\partial/\partial x_j)\varphi(x_j - z) = 0. \tag{22}$$

According to above, the function $\varphi(x_j - z)$ with (17) is bounded and continuous, as shown in Fig. 7a. We also show the φ function of the median in Fig. 7b. The influence of an extremely large or small x_j on the median is a constant. However, the influence of an extremely large or small x_j on

our estimator is very small according to (21). In fact, (21) also shows that an extremely large or small x_j can be thought of a new observation that may be from an unknown new population and have no influence (i.e., $\mathrm{IC}(x_j; F, z) = 0$) on our estimator. The use of $\rho(x-\theta) = (x-\theta)^2$, that is corresponding to the LS method, has $\varphi(x-\theta) = 2(\theta - x)$. The influence function of LS estimator is unbounded and the gross error sensitivity $\gamma^* = \infty$ and, hence, it is not robust to the outliers and noisy points. By solving (22), we can find the location of x_j which has a maximum influence on z (i.e., maximum $\mathrm{IC}(x_j; F, z)$). Our estimator has a bounded and continuous influence function and also a finite gross error sensitivity (according to (19) and (20)). Hence, it is robust from the robust statistical point of view.

We may also use the data set of Fig. 2a in Examples 1, 2, and 3 to demonstrate the robustness of SCM. The data set of Fig. 2a is originally generated by three random bivariate normal distributions. The largest cluster is generalized from the bivariate normal distribution with the largest variance so that it can be thought of a cluster with noisy points when it is compared to other two compact clusters. The clustering result of SCM shown in Fig. 5 exhibits the ability to tolerate noise and outliers in this example. Since it is unnecessary to specify the initial cluster number and centers in SCM, it is also robust to the initializations. More examples and comparisons are made in Sections 5 and 6.

4 The Statistical Hypothesis Testing for the Threshold of CCA

We showed that the SCM objective function can be seen as a density shape estimation of the data set. The parameter γ is close related to the kernel width and to increase γ is equivalent to decrease the neighborhood radius. A small γ will underestimate the true density shape. This phenomenon can be explained by (11) with

$$\lim_{\gamma\to 0}\{z_i\} = \lim_{\gamma\to 0}\left\{\frac{\sum_{j=1}^n S_{ij}^{\gamma} x_j}{\sum_{j=1}^n S_{ij}^{\gamma}}\right\} = \frac{\sum_{j=i}^n x_j}{n} = \overline{x}. \tag{23}$$

This means that when γ tends to zero, the SCM objective function will have only one peak on the sample mean $\overline{x}$. In fact, a small γ has almost one peak in the SCM objective function such as the case of $\gamma < 5$. A large γ will overestimate the density shape, which can also be explained by denoting $\hat{S} = max\{S_{i1}, \cdots, S_{in}\}$, $S'_{ij} = S_{ij}/\hat{S}$, and then

$$\begin{aligned}\lim_{\gamma\to\infty}\{z_i\} &= \lim_{\gamma\to\infty}\left\{\frac{\sum_{j=1}^n S_{ij}^{\gamma} x_j}{\sum_{j=1}^n S_{ij}^{\gamma}}\right\} \\ &= \lim_{\gamma\to\infty}\left\{\frac{\sum_{j=1}^n (S'_{ij})^{\gamma} x_j}{\sum_{j=1}^n (S'_{ij})^{\gamma}}\right\} = \frac{\sum_{S'_{ij}=1} x_j}{\sum_{S'_{ij}=1} 1}.\end{aligned} \tag{24}$$

This means that when γ tends to infinity, the data point which is the closest to the initial center will be the peak so that almost all of data points are the peaks.

The CCA value is larger than a given threshold means that the approximate density is not altered when the neighborhood radius decreases. Therefore, we say that the approximate density shape is stable with such γ value and we take this value to be our estimate of γ. The value of the threshold denotes the degree of stability that can be accepted. It also indicates the size limit of the correlation value that can be accepted for the stability of the density shape. Now, suppose that we have a random sample of size n from a bivariate normal distribution with the correlation coefficient ρ. Let R denote the sample correlation coefficient (i.e., the values of CCA). If we want to test the statistical hypothesis

$$\begin{cases} H_0: & \rho \le \rho_0 \\ H_1: & \rho > \rho_0 \end{cases}, \tag{25}$$

where $-1 < \rho_0 < 1$, the test statistic [4] is

$$Z_R = \frac{1}{2} ln\left(\frac{1+R}{1-R}\right), \tag{26}$$

which is called the Fisher's z-transformation and approximates the normal distribution with the mean

$$E(Z_R) = \frac{1}{2} ln\left(\frac{1+\rho}{1-\rho}\right) \tag{27}$$

and the variance

$$Var(Z_R) = \frac{1}{n-3}. \tag{28}$$

We reject H_0 if

$$\frac{\sqrt{n-3}}{2}\left[ln\left(\frac{1+R}{1-R}\right) - ln\left(\frac{1+\rho_0}{1-\rho_0}\right)\right] > Z_\alpha, \tag{29}$$

where Z_α denotes the value of the standard normal distribution with right tail probability α. The above rejection rule is equivalent to reject H_0 if

$$R > \frac{\left(\frac{1+\rho_0}{1-\rho_0}\right)\exp(2Z_\alpha/\sqrt{n-3}) - 1}{\left(\frac{1+\rho_0}{1-\rho_0}\right)\exp(2Z_\alpha/\sqrt{n-3}) + 1}. \tag{30}$$

The right side of (30) is a monotone decreasing function of n and equals to ρ_0 when n tends to infinity. It is reasonable to restrict n to the set $\{4, 5, \cdots\}$ and the right side of (30) will then belong to the interval

$$\left[\rho_0, \frac{\left(\frac{1+\rho_0}{1-\rho_0}\right)\exp(2Z_\alpha) - 1}{\left(\frac{1+\rho_0}{1-\rho_0}\right)\exp(2Z_\alpha) + 1}\right]. \tag{31}$$

If we select ρ_0 and Z_α, (30) can give us a threshold for CCA according to the statistical hypothesis testing concept. The threshold will depend on the sample size n. The value of ρ_0 denotes the degree of correlation (stability of density shape) that can be accepted. If we do not consider the effect of n, (31) can give us an interval estimate of the threshold for CCA. The interval is $[0.97, 0.9988]$ when ρ_0 is 0.97 and the significance level α is 0.05. This is why we suggest a threshold of around 0.97 to 0.999.

5 Numerical Examples

In this section, we present more examples with numerical data and real data.

Example 4. This is a 16 group data set as shown in Fig. 8a. Data points in each group are uniformly generated in each rectangle. In this data set, any one validity measure

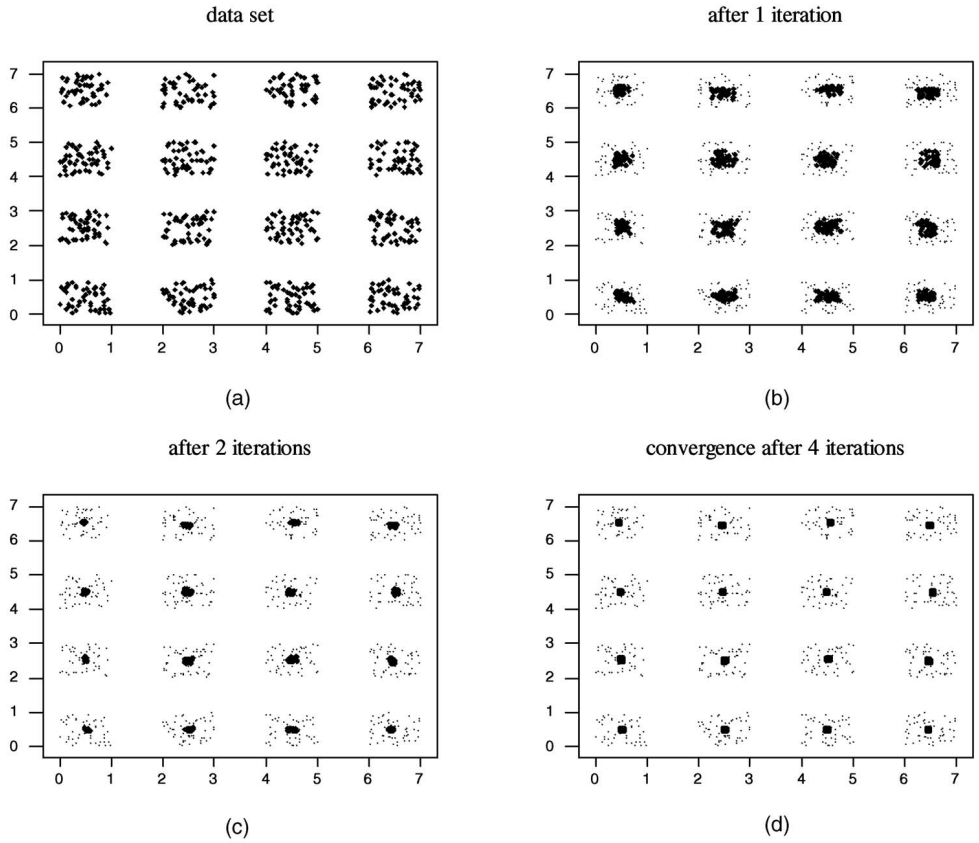


Fig. 8. (a) The 16-clusters data set in Example 4. (b), (c), and (d) are the states of the data points in SCA after 1, 2, and 4 (convergence) iterations, respectively.

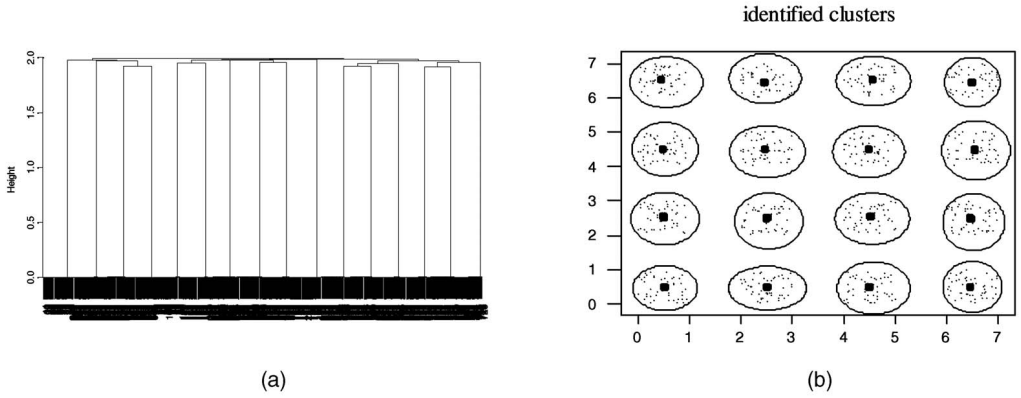


Fig. 9. The results of AHC of Example 4. (a) The Hierarchical Clustering tree. (b) The identified clusters.

should have an optimal solution with $c^* = 16$. However, this result is obtained only when the selected algorithm partitions the data into 16 well-separated clusters. If the initializations are not properly chosen (for example, no centers are initialized in each of these 16 rectangles), we can not ensure that good partitions will be found by the clustering algorithms. Whether a fix cluster number algorithm or a progressive clustering method, the clustering problem will become whatever the initialization techniques can guarantee that meets our need. This is a difficult problem especially in a high dimensional case. In SCM, this problem will not arise. After a good density shape estimate γ is estimated, SCM will find all of the clusters. The CCA result for this data set is shown in the third row of Table 1. $\gamma = 15$ will be a good estimate. The data point states in SCA after 1, 2, and 4 (convergence) iterations are shown in Figs. 8b, 8c, and 8d, respectively. The AHC result is shown in Fig. 9. The Hierarchical Clustering tree in Fig. 9a shows that $c^* = 16$. Fig. 9b shows these 16 identified clusters. SCM does not have an initialization problem and can produce a good result in this example.

Example 5. This is a three dimensional data set with large cluster number and different volume as shown in Fig. 10a. The CCA result is shown in the fourth row of Table 1. $\gamma = 15$ will be a good estimate. The data point states in SCA after 1, 5, and 14 (convergence) iterations

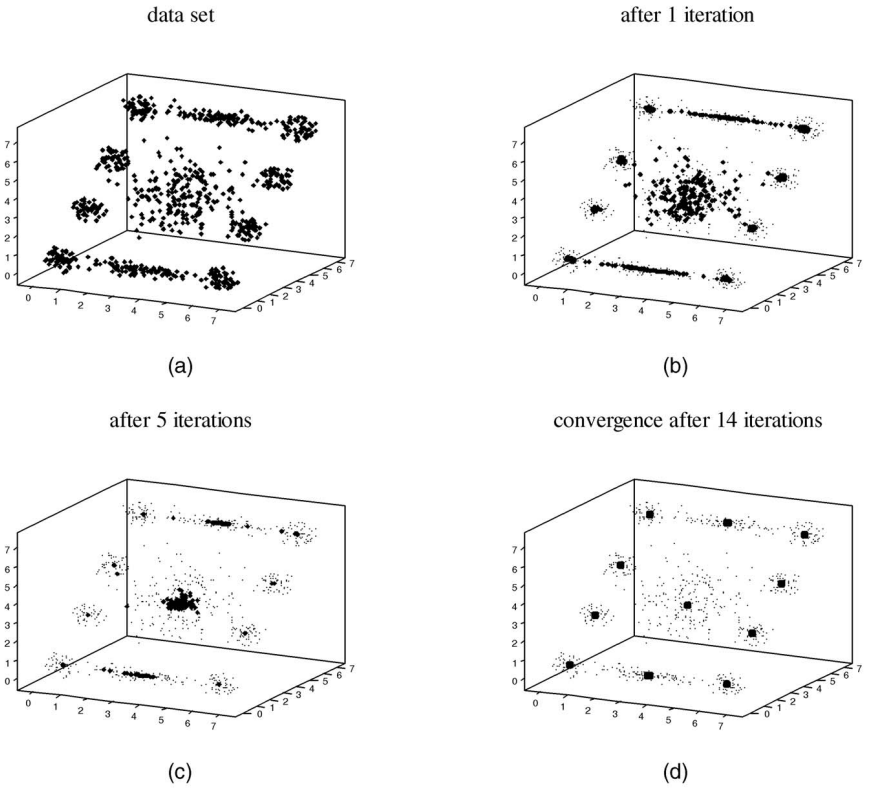


Fig. 10. (a) The data set in Example 5. (b), (c), and (d) are the states of the data points in SCA after 1, 5, and 14 (convergence) iterations, respectively.

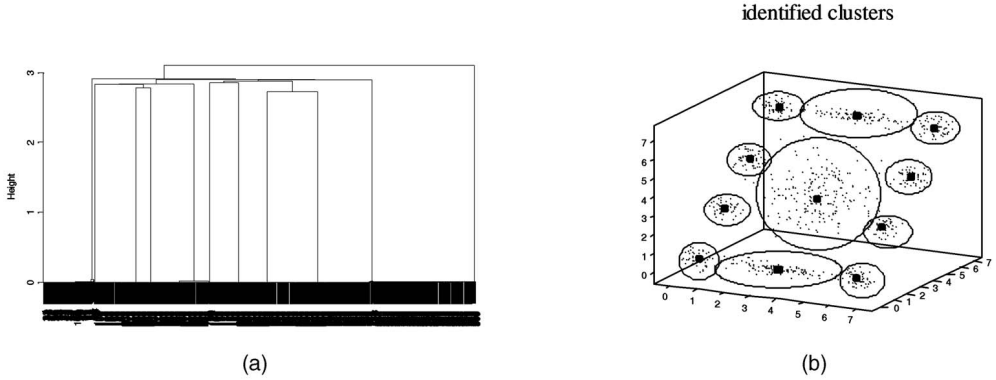


Fig. 11. The AHC results of Example 5. (a) The Hierarchical Clustering tree. (b) The identified clusters.

are shown in Figs. 10b, 10c, and 10d, respectively. The AHC result is shown in Fig. 11. Fig. 11a shows that $c^* = 11$. Fig. 11b shows the identified cluster with their correct volumes. In this example, correct cluster volumes are found by SCM without using any distance transformations or variance-covariance matrix. Moreover, this result is also independent of the initializations.

Example 6. This is a four cluster data set that looks like the word "TV" as shown in Fig. 12a. We randomly draw a lot of uniform noisy points. The CCA result is shown in the fifth row of Table 1. $\gamma = 5$ will be a good estimate. The data point states in SCA after 5, 10, and 30 (convergence) iterations are shown in Figs. 12b, 12c, and 12d, respectively. The AHC result is shown in Fig. 13. Fig. 13a shows that $c^* = 4$. Fig. 13b shows the four identified clusters with four different symbols. We can see that the final states on the right side of the "V" are not very centralized to a point as Fig. 12d shows. The reason is that the location of the peak is very flat. This situation occurs frequently, especially in a high dimensional environment. We use the IRIS data to illustrate this phenomenon. The CCA result of the IRIS data is shown in the sixth row of Table 1. $\gamma = 10$ is used in the IRIS data. Fig. 14a shows the final states of the data points in SCA and Fig. 14b shows their Hierarchical Cluster tree. The peak area of the two overlapped clusters is very flat and the final states of the data points for these two overlapped clusters are not very centralized to a point. According to the Hierarchical Clustering tree in Figs. 13a and 14b, we can easily decide the optimal cluster number for Fig. 12a and the IRIS data. We

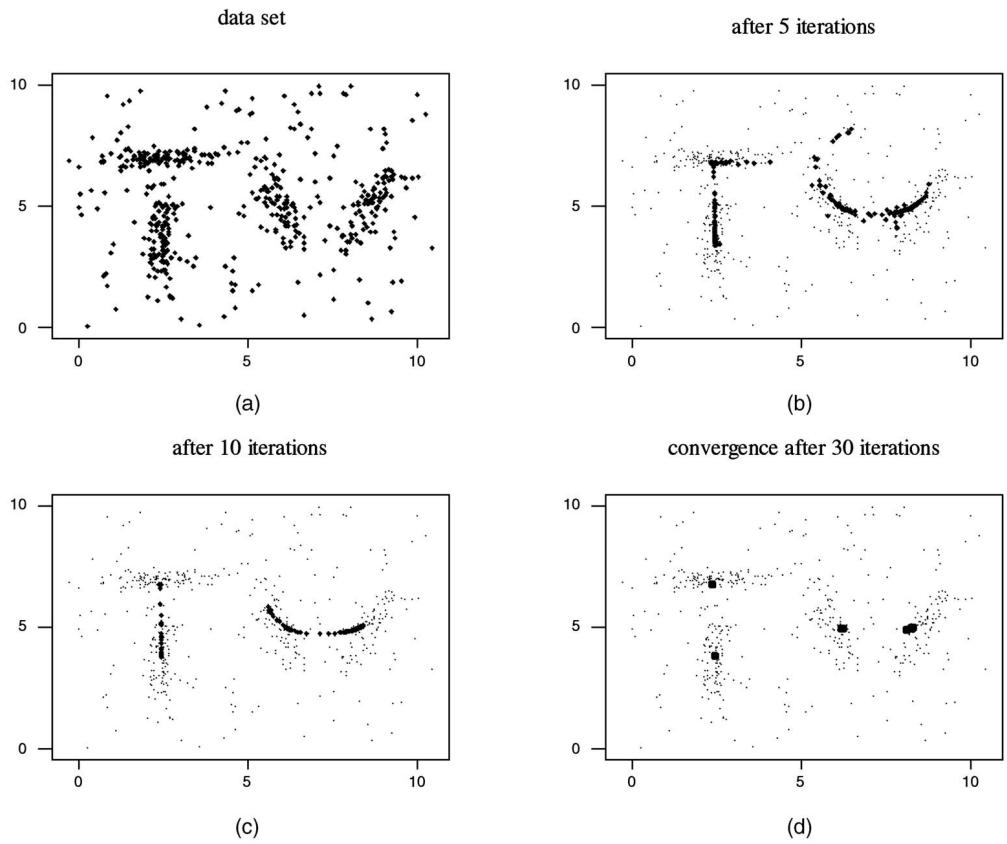


Fig. 12. (a) The data set with uniform noisy pints in Example 6. (b), (c), and (d) are the states of the data points in SCA after 5, 10, and 30 (convergence) iterations, respectively.

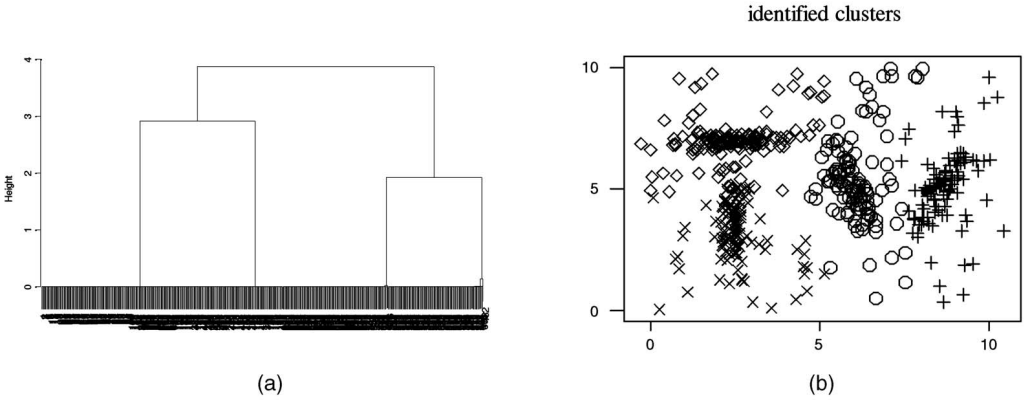


Fig. 13. The AHC results of Example 6. (a) The Hierarchical Clustering tree. (b) The identified four clusters with four different symbols.

do not need to set a threshold to merge clusters, because the Hierarchical Clustering tree can take over the effect of the threshold and produce an optimal partition simultaneously. In next section, we will have the comparisons of SCM with the best-known fuzzy c-means (FCM) [2], [32] and the possibilistic c-means (PCM) [22] algorithms. We then make the analysis on computational complexity.

6 Comparisons and Computational Complexity

In Sections 2 and 3, we discussed how the SCM achieves three robust aspects mentioned in Section 1. In Section 5, we used several numerical and real data to demonstrate the effectiveness of SCM. In this section, we first consider the well-known FCM algorithm. We know that the cluster number c in FCM is a priori. There are many validity indexes proposed to solve this cluster validity problem [3], [8], [24], [30]. Here, we simply choose three popular indexes which are partition coefficient (PC) [3], partition entropy (PE) [2], and Xie and Beni (XB) [30] indexes. We implement FCM with PC, PE, and XB indexes for the IRIS data set. These indexes results with FCM for IRIS are shown in Fig. 15. All indexes show that $c = 2$ is optimal for the IRIS data set. From Example 6 in Section 5, SCM also gave the optimal cluster number $c = 2$. Although the IRIS data set

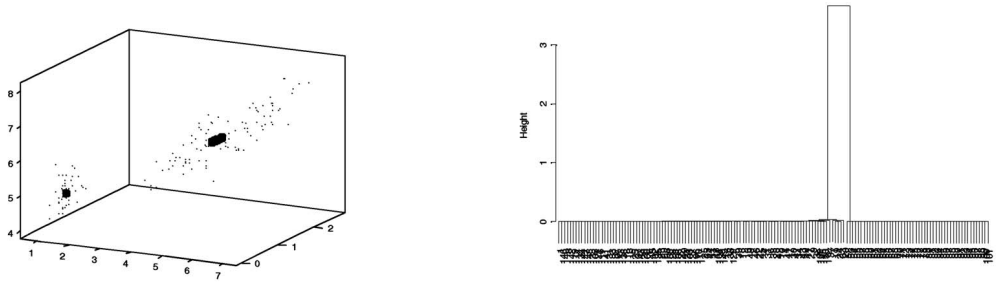


Fig. 14. (a) The final states of the IRIS data set in SCA. (b) The Hierarchical Clustering tree of the final states of the data points.

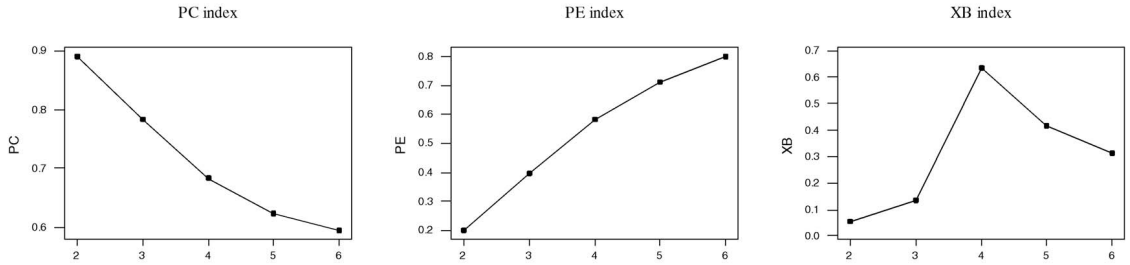


Fig. 15. The validity indexes for the IRIS data set by processing the FCM algorithm.

has three clusters, there have two overlapped clusters. For unsupervised clustering methods, $c = 2$ is supposed to be a reasonable cluster number estimate for the IRIS data.

Although FCM is a very useful clustering method, its membership functions $\mu_{ij} = \mu_i(x_j)$ with $\mu_{ij} \in [0, 1]$ and $\sum_{i=1}^{c} \mu_{ij} = 1$ do not always correspond well to the degree of belonging of the data and may be inaccurate in a noisy environment [6], [7]. To improve this weakness of FCM and produce memberships to explain the degree of belonging for the data, Krishnapuram and Keller [22] relaxed the restriction of $\sum_{i=1}^{c} \mu_{ij} = 1$ and then created a possibility c-means (PCM) algorithm which used a possibility type membership function to describe convex fuzzy sets and had the objective function

$$J_{PCM}(U, X) = \sum_{i=1}^{c}\sum_{j=1}^{n}({\mu_{ij}}^m\|x_j - z_i\|^2 + \eta_i(1 - \mu_{ij})^m). \quad (32)$$

Note that the determination of the normalization parameter η_i is quite important. It is recommended to select η_i as

$$\eta_i = K\frac{\sum_{j=1}^{n}{\mu_{ij}}^m\|x_j - z_i\|^2}{\sum_{j=1}^{n}{\mu_{ij}}^m} \text{ or } \eta_i = \frac{\sum_{x_j \in (\pi_i)_\alpha}\|x_j - z_i\|^2}{|x_j \in (\pi_i)_\alpha|}, \quad (33)$$

where $K \in (0, \infty)$ is typically chosen to be one and $x_j \in (\pi_i)_\alpha$ if $\{\mu_{ij} \geq \alpha | j = 1, \cdots, n\}$. The PCM result is sensitive to the parameters η_i and m. In some situations, PCM will produce coincident clusters even though the clusters are well-separated [1]. We now use the proposed total similarity measure $J_s(z)$ of (1) to explain the reasons for these unreasonable results produced by the PCM algorithm.

Suppose we consider another similarity measure function as

$$S(x_j, z_i) = \frac{1}{1 + \frac{\|x_j - z_i\|^2}{\eta_i}}, \quad (34)$$

where η_i is constant. We can set up the second kind of total similarity objective function model with

$$J_s(z) = \sum_{i=1}^{c}\sum_{j=1}^{n}\left(\frac{1}{1 + \frac{\|x_j - z_i\|^2}{\eta_i}}\right)^{\gamma}. \quad (35)$$

The necessary condition for maximizing (35) is

$$z_i = \frac{\sum_{j=1}^{n}\left(1 + \frac{\|x_j - z_i\|^2}{\eta_i}\right)^{-(\gamma+1)} x_j}{\sum_{j=1}^{n}\left(1 + \frac{\|x_j - z_i\|^2}{\eta_i}\right)^{-(\gamma+1)}}. \quad (36)$$

Equation (34) can be used as a membership function with $S(x_j, z_i) = 1$ as $\|x_j - z_i\|^2 = 0$ and $S(x_j, z_i) = 0$ as $\|x_j - z_i\|^2$ tends to infinity. Now, we decompose the necessary condition (36) into two optimization steps as

$$\mu_{ij} = \left(1 + \frac{\|x_j - z_i\|^2}{\eta_i}\right)^{-1} \quad (37)$$

and

$$z_i = \frac{\sum_{j=1}^{n}\mu_{ij}^{\gamma+1}\, x_j}{\sum_{j=1}^{n}\mu_{ij}^{\gamma+1}}. \quad (38)$$

If we take $\gamma = 1$ here, this two-steps optimization algorithm will be equivalent to PCM algorithm in the case of $m = 2$. That is, we can have the second model of the PCM objective function with $m = 2$. In Sections 2 and 4, we had analyzed that the small γ may have coincident clusters. Identically, the objective function (35) may have just only one peak

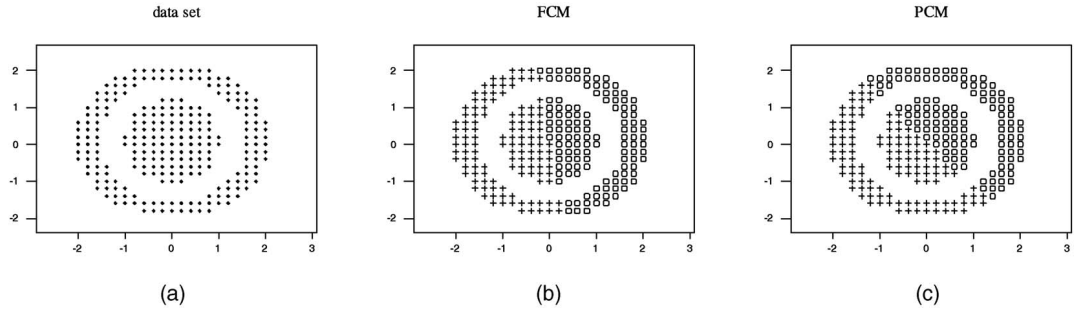


Fig. 16. (a) The data set. (b) The FCM clustering result with $c = 2$. (c) The PCM clustering with $c = 2$.

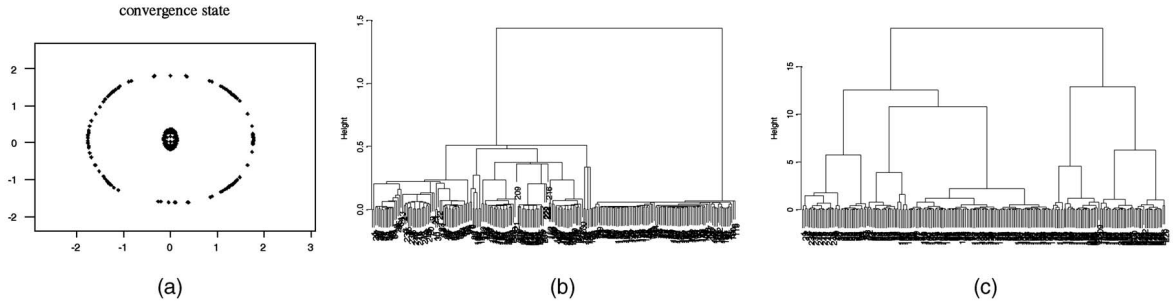


Fig. 17. (a) The final states of the data points in SCA. (b) The Hierarchical Clustering tree with the single linkage method. (c) The Hierarchical Clustering tree with the Ward's method.

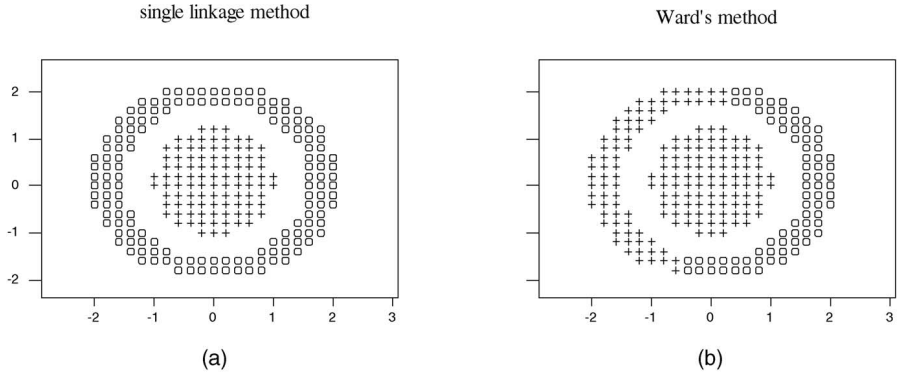


Fig. 18. The identified clusters of single linkage and Ward's methods.

when $\gamma = 1$. Barni et al. [1] demonstrated that PCM has the undesirable tendency to produce coincident clusters. Here, we give another perspective to this coincident tendency. Note that the parameter γ in SCM plays quite a different role than the fuzzifier m in PCM and FCM. In SCM, we adjust the parameter γ to search for a good density shape estimate of the data set. In PCM and FCM, the change in m will produce a different degree of belonging. Below, we provide a data set to make the comparison of SCM with FCM and PCM algorithms. We also analyze the computational complexity.

Example 7. Fig. 16a shows a data set that contains two sets A and B in a two dimensional Euclidean space. Data points in the set A fall in the unit disk centered at the origin. Data points in the set B have the distances from the origin approximately in 1.5 and 2. We implement FCM and PCM with a priori $c = 2$ for this data set where the clustering results are shown in Figs. 16b and 16c. The CCA value with the estimate $\gamma = 10$ of this data set are shown in the seventh row of Table 1. The final states of all data points in SCA are shown in Fig. 17a. We use the single linkage and Ward's methods to process AHC for these SCA final states. The Hierarchical Clustering trees of the single linkage and Ward's methods are shown in Figs. 17b and 17c, respectively. The single linkage method shows that $c = 2$ is a cluster number estimate for this data set and the clustering result is shown in Fig. 18a where the sets A and B are well separated. However, the clustering result of Ward's method is shown in Fig. 18b where the sets A and B can not be well separated. In fact, most clustering algorithms based on a dissimilarity (distance) of the data point from the cluster center are likely to fail when applied to the data set shown in Fig. 16a. This demonstrates that the SCM clustering results may depend on the choice of different AHC method. In general, if the final states of the data points in SCA are centralized to c^* cluster centers as shown in Figs. 3d, 8d, and 10d, most chosen AHC methods may obtain the same result. However, if the final states of the data points in SCA are not centralized to c^* cluster centers as shown in Fig. 17a, different AHC methods may provide different results.

Overall, we suggest that the single linkage method is chosen as a linkage way to process AHC in our SCM so that similar data points (final states of the data point in SCA) can be easily merged into the same cluster. In this example, SCM with the single linkage AHC method can give good clustering results. However, the SCM clustering results for this data set with Ward's AHC method are not good. It depends on the choice of AHC methods.

Compared with other clustering method, SCM requires more computational time. In SCA, for each correlation comparison step, we need to compute (7) for all n data points in an s-dimensional space. Its computational complexity is $O(n^2st_1)$ where t_1 is the number of iterations in CCA. In our experiments, a good estimate of γ always falls in the interval $[5, 20]$ and the CCA can reach the threshold 0.97 when $\gamma \in [5, 20]$. Thus, in general, t_1 is a small positive integer. Since all data points are specified to be the initial values in SCA, the computational complexity of SCA is $O(n^2st_2)$, where t_2 is the number of iterations in SCA. After the final states of all data points are organized by SCA, we process AHC to find the final clustering results with the optimal cluster number c^*. For a given data point, we need to check $(n-1)$ points to find a point to merge in the AHC step. Thus, the computational complexity of AHC is $O(\sum_{k=1}^{n-1} k)$ which is equivalent to $O(n^2)$. In FCM and PCM, the computational complexity is $O(ncst_c)$, where t_c is the number of iterations when the cluster number is c. For solving the validity problem, we need process FCM or PCM with $c = 2$ to $c = c_{max}$, where c_{max} is the possible maximum number of clusters. Thus, the computational complexity of FCM and PCM for finding the optimal cluster number c^* is $O(n(c_{max}-1)st^*)$, where $t^* = \sum_{c=2}^{c_{max}} t_c$. Since we combine three schemes with CCA, SCA, and AHC to create SCM, it requires more computational time than other existing clustering methods. However, the computer runs still fast for SCM and the reward is three robust aspects.

7 Conclusions and Discussion

We proposed a similarity-based clustering method (SCM) in this paper. The SCM objective function is equivalent to the approximate density shape estimation. The SCM structure includes a CCA process to estimate the approximate density shape of the data set, a second process, SCA, with all data points to self-organize their final states and finally process AHC with the final states of the data points to find a cluster number estimate c^* and identify these c^* clusters. In SCM, only the density shape estimation parameter γ needs to be specified. We used a correlation comparison technique to estimate γ. The threshold for searching γ can be obtained using the statistical hypothesis testing methodology. The threshold will depend on the degree of stability that can be accepted by users. With a good parameter estimate γ and a proper chosen AHC method, SCM can achieve robust clustering results.

To have SCM robust to initializations (cluster number and initial guesses), we process SCA with $\mathbf{z}^{(0)} = (z_1^{(0)}, \cdots, z_n^{(0)}) = (x_1, \cdots, x_n)$ and use a AHC method to find a cluster number estimate c^* and then identify these c^* clusters. In general, the final partition results will depend on the distance of the data point from the cluster center and also on the cluster center location so that the SCM clustering results may depend on the choice of AHC methods. We gave several examples to demonstrate these SCM properties. Most data sets with ellipsoidal cluster shapes of different volumes and also with noisy points can be successfully processed using SCM with different AHC method. These were shown in Examples 4, 5, and 6 with Figs. 8, 9, 10, 11, 12, 13, and 14. However, if the data sets are with different cluster shapes as shown in Fig. 16 of Example 7 so that the final states of the data points in SCA are not centralized to c^* cluster centers, the SCM clustering results may depend on the chosen AHC method. In general, we recommend using SCM with the single linkage AHC method. In fact, detecting different shapes of clusters is another difficult clustering problem. The robustness to different cluster shapes should be another robust clustering characteristic that will be a further research topic.

Finally, we mention that SCM requires more computational time than general clustering methods such as FCM and PCM, etc. This is because SCM includes three schemes with CCA, SCA, and AHC to finish three robust aspects. The computational complexity in SCM will become large, especially in a high-dimensional space. However, a fast clustering method based on SCM objective function can randomly initialize a lot of cluster centers to process SCA and then find the cluster number estimate using an agglomerative method. The numerical comparisons of SCM with FCM and PCM and also the computational complexity analysis had been discussed in this paper. It demonstrated the superiority of the SCM method. Overall, the proposed SCM is a well-structured robust clustering procedure based on a simple total similarity objective function.

Acknowledgments

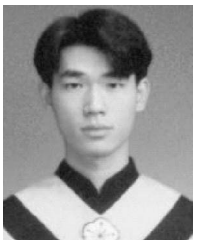
The authors would like to thank the anonymous referees for their helpful comments and suggestions to improve the presentation of the paper. This work was supported in part by the National Science Council of Taiwan, ROC, under Grant NSC-90-2118-M-033-001.

References

[1] M. Barni, V. Cappellini, and A. Mecocci, "Comments on: A Possibilistic Approach to Clustering," *IEEE Trans. Fuzzy Systems,* vol. 4, pp. 393-396, 1996.
[2] J.C. Bezdek, *Pattern Reccognition with Fuzzy Objectiv Function Algorithm.* Plenum Press, 1981.
[3] J.C. Bezdek, "Cluster Validity with Fuzzy Sets," *J. Cybernetics,* vol. 3, pp. 58-73, 1974.
[4] P.J. Bickel and K.A. Doksum, *Mathematical Statistics,* second ed. Prentice-Hall, 2001.
[5] S.L. Chiu, "Fuzzy Model Identification Based on Cluster Estimation," *J. Intelligent and Fuzzy Systems,* vol. 2, pp. 267-278, 1994.
[6] R.N. Dave, "Characterization and Detection of Noise in Clustering," *Pattern Recognition Letters,* vol. 12, pp. 657-664, 1991.
[7] R.N. Dave and R. Krishnapuram, "Robust Clustering Methods: A Unified View," *IEEE Trans. Fuzzy Systems,* vol. 5, pp. 270-293, 1997.
[8] D.L. Davies and D.W. Bouldin, "A Cluster Separation Measure," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 1, pp. 224-227, 1979.
[9] R.O. Duda and P.E. Hart, *Pattern Classification and Scene Analysis.* Wiley, 1973.
[10] H. Frigui and R. Krishnapuram, "Clustering by Competitive Agglomeration," *Pattern Recognition,* vol. 30, pp. 1223-1232, 1997.

[11] H. Frigui and R. Krishnapuram, "A Robust Competitive Clustering Algorithm with Applications in Computer Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 21, pp. 450-465, 1999.
[12] I. Gath and A.B. Geva, "Unsupervised Optimal Fuzzy Clustering," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 11, pp. 773-781, 1989.
[13] S. Grossberg, "Adaptive Pattern Classification and Universal Recoding, I: Parallel Development and Coding of Neural Feature Detectors," *Biological Cybernetics,* vol. 23, pp. 121-134, 1976.
[14] E.E. Gustafson and W.C. Kessel, "Fuzzy Clustering with a Fuzzy Matrix," *Proc. IEEE Conf. Design and Control,* pp. 761-766, 1979.
[15] J.A. Hartigan, *Clustering Algorithms.* Wiley, 1975.
[16] P.J. Huber, *Robust Statistics.* Wiley, 1981.
[17] A.K. Jain, R.P.W. Duin, and J. Mao, "Statistical Pattern Recognition: A Review," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, pp. 4-37, 2000.
[18] J.M. Jolion, P. Meer, and S. Bataouche, "Robust Clustering with Applications in Computer Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 13, pp. 791-802, 1991.
[19] L. Kaufman and P.J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis.* Wiley, 1990.
[20] T. Kohonen, "Learning Vector Quantization," *Neural Network,* vol. 1, p. 303 1988.
[21] R. Krishnapuram, H. Frigui, and O. Nasraoui, "Fuzzy and Possibilistic Shell Clustering Algorithm and Their Application to Boundary Detection and Surface Approximation," *IEEE Trans. Fuzzy Systems,* vol. 3 pp. 29-60, 1995.
[22] R. Krishnapuram and J.M. Keller, "A Possibilistic Approach to Clustering," *IEEE Trans. Fuzzy Systems,* vol. 1, pp. 98-110, 1993.
[23] R.P. Lippmann, "An Introduction to Computing with Neural Nets," *IEEE ASSP Magazine,* pp. 4-22, Apr. 1987.
[24] N.R. Pal and J.C. Bezdek, "On Cluster Validity for Fuzzy c-Means Model," *IEEE Trans. Fuzzy Systems,* vol. 1, pp. 370-379, 1995.
[25] G.J. McLachlan and K.E. Basford, *Mixture Models: Inference and Applications to Clustering.* New York: Marcel Dekker, 1988.
[26] G.J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions.* John Wiley and Sons, 1997.
[27] C.V. Stewart, "Minpran: A New Robust Estimator for Computer Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, pp. 925-938, 1995.
[28] E.C.K. Tsao, J.C. Bezdek, and N.R. Pal, "Fuzzy Kohonen Clustering Net Works," *Pattern Recognition,* vol. 27, pp. 757-764, 1994.
[29] K.L. Wu and M.S. Yang, "Alternative c-Means Clustering Algorithms," *Pattern Recognition,* vol. 35, pp. 2267-2278, 2002.
[30] X.L. Xie and G. Beni, "A Validity Measure for Fuzzy Clustering," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 13, pp. 841-847, 1991.
[31] R.R. Yager and D.P. Filev, "Approximate Clustering Via the Mountain Method," *IEEE Trans. Systems, Man and Cybernetics,* vol. 24, pp. 1279-1284, 1994.
[32] M.S. Yang, "A Survey of Fuzzy Clustering," *Mathematical and Computer Modelling,* vol. 18, pp. 1-16, 1993.
[33] L.A. Zadeh, "Similarity Relations and Fuzzy Orderings," *Information Sciences,* vol. 3, pp. 177-200, 1971.
[34] X. Zhuang, T. Wang, and P. Zhang, "A Highly Robust Estimator Through Partially Likelihood Function Modeling and Its Application in Computer Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 14, pp. 19-35, 1992.

Miin-Shen Yang received the BS degree in mathematics from the Chung Yuan Christian University, Chung-Li, Taiwan, in 1977, the MS degree in applied mathematics from the National Chiao-Tung University, Hsinchu, Taiwan, ROC, in 1980, and the PhD degree in statistics from the University of South Carolina, Columbia, in 1989. During 1989, he joined the faculty member of the Chung Yuan Christian University as an associate professor. Since 1994, he has been a professor at the Chung Yuan Christian University, where he is currently the chairman of the Department of Applied Mathematics. During 1997-1998, he was a visiting professor in the Department of Industrial Engineering, University of Washington, Seattle. His current research interests include applications of statistics, fuzzy clustering, pattern recognition, and neural fuzzy systems.



Kuo-Lung Wu received the BS degree in mathematics in 1997, the MS and PhD degrees in applied mathematics in 2000 and 2003 from the Chung Yuan Christian University, Chung-Li, Taiwan, ROC, respectively. Since 2003, he has been an assistant professor of Department of Information Management at Kun Shan University of Technology, Yung-Kang, Tainan, Taiwan, ROC. He is a member of the Phi Tau Phi Scholastic Honor Society of the Republic of China. His research interests include fuzzy theorem, cluster analysis, pattern recognition, and neural networks.