Information Sciences 179 (2009) 3230–3246

Contents lists available at ScienceDirect

Information Sciences

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

A new point symmetry based fuzzy genetic clustering technique for automatic evolution of clusters

Sriparna Saha *, Sanghamitra Bandyopadhyay

Machine Intelligence Unit, Indian Statistical Institute, 203, B.T. Road, Kolkata 700 108, India

ARTICLE INFO

Article history:
Received 3 September 2008
Received in revised form 14 May 2009
Accepted 5 June 2009

Keywords:
Fuzzy clustering
Cluster validity index
Point symmetry
Kd-tree
Genetic algorithm
Variable string length

ABSTRACT

In this paper a fuzzy point symmetry based genetic clustering technique (Fuzzy-VGAPS) is proposed which can automatically determine the number of clusters present in a data set as well as a good fuzzy partitioning of the data. The clusters can be of any size, shape or convexity as long as they possess the property of symmetry. Here the membership values of points to different clusters are computed using the newly proposed point symmetry based distance. A variable number of cluster centers are encoded in the chromosomes. A new fuzzy symmetry based cluster validity index, *FSym*-index is first proposed here and thereafter it is utilized to measure the fitness of the chromosomes. The proposed index can detect non-convex, as well as convex-non-hyperspherical partitioning with variable number of clusters. It is mathematically justified via its relationship to a well-defined hard cluster validity function: the Dunn's index, for which the condition of uniqueness has already been established. The results of the Fuzzy-VGAPS are compared with those obtained by seven other algorithms including both fuzzy and crisp methods on four artificial and four real-life data sets. Some real-life applications of Fuzzy-VGAPS to automatically cluster the gene expression data as well as segmenting the magnetic resonance brain image with multiple sclerosis lesions are also demonstrated.

1. Introduction

Clustering [23] is a core problem in data-mining with innumerable applications spanning many fields [9,15,34,40,50,12,28,35,48]. For identifying clusters from a data set, some measure of similarity or proximity has to be defined. Then this similarity measure is used to assign points to different clusters. It can be noted that symmetry is a very well-known property of any object. Hence it can be expected that clusters also have some form of symmetry. Based on this observation, a new point symmetry (PS) based distance, d_{ps} (PS-distance), is developed in [6]. For reducing the complexity of computing the PS-distance, use of Kd-tree is incorporated in [6]. This proposed distance was then used to develop a genetic algorithm based clustering technique, GAPS [6].

Fuzzy C-means (FCM) [11] is a well-known clustering technique that uses the principles of fuzzy sets to evolve a partition matrix $U(X)$. However, FCM has three major limitations: it requires the *a priori* specification of the number of clusters (K), it often gets stuck at suboptimal solutions based on the initial configuration (recently, proof of its convergence to a local minima or saddle point of the error function has been provided in [27]) and it can detect only hyperspherical shaped clusters. In most of the real-life situations the number of clusters in a data set is not known *a priori*. The real challenge in this situation is to be able to automatically evolve a proper value of K as well as providing the appropriate clustering of a data set.

* Corresponding author. Tel.: +91 3325753112; fax: +91 3325783357.
E-mail addresses: sriparna_r@isical.ac.in (S. Saha), sanghami@isical.ac.in (S. Bandyopadhyay).

0020-0255/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.ins.2009.06.013

A genetic algorithm (GA) based fuzzy clustering technique, Fuzzy-VGA, has been proposed in Ref. [39], where the search capability of genetic algorithm is utilized to overcome the above-mentioned limitations of FCM. Fuzzy-VGA, optimizing the well-known XB-index [49], is able to automatically evolve the appropriate fuzzy clustering for hyperspherical data sets. However for clusters with other than hyperspherical shapes, this algorithm is likely to fail, as it uses, like the FCM, the Euclidean distances of the points from the respective cluster centers for computing the membership values. In order to overcome this limitation, in this article a fuzzy variable string length genetic point symmetry (Fuzzy-VGAPS) based clustering technique is proposed. Here membership values of points to different clusters are computed based on the point symmetry based distance rather than the Euclidean distance. This enables the proposed algorithm to automatically evolve the appropriate clustering of all types of clusters, both convex and non-convex, which have some symmetrical structures. The chromosome encodes the centers of a number of clusters, whose value may vary.

A new fuzzy cluster validity index, named *FSym*-index, is proposed in this article and thereafter it is utilized for computing the fitness of the chromosomes in Fuzzy-VGAPS. The proposed *FSym*-index is based on the newly developed point symmetry distance. As a result, the index is capable of correctly detecting the presence of clusters of different sizes and shapes as long as they possess the symmetry property. A mathematical justification of the proposed index is also established in this article. The superiority of the proposed genetic clustering technique for evolving the appropriate fuzzy partitioning of a data set in comparison to two recently developed automatic clustering techniques and five recent clustering techniques for fixed K are shown for four artificial and four real-life data sets. Analysis of variance (ANOVA) [4] technique has also been used for the purpose of comparison. Results on the eight data sets establish the fact that Fuzzy-VGAPS is capable of detecting the proper number of partitions and the proper partitioning from data sets having similar density clusters of symmetrical shapes irrespective of any geometric structure, size or overlaps. Some real-life applications of Fuzzy-VGAPS to automatically cluster the gene expression data as well as segmenting the magnetic resonance brain image with multiple sclerosis lesions are also demonstrated.

2. The proposed fuzzy cluster validity index

The existing point symmetry based distance is first discussed in this section. Then, the motivation of developing a new cluster validity index based on the point symmetry based distance is stated. Thereafter the cluster validity index is defined and a mathematical justification of the proposed index is provided.

2.1. The point symmetry distance

A new definition of point symmetry based distance (PS-distance), $d_{ps}(\bar{x},\bar{c})$, associated with point $\bar{x}$ with respect to a center $\bar{c}$ is developed in Ref. [6]. It is also shown in Ref. [6] that $d_{ps}(\bar{x},\bar{c})$ is able to overcome some serious limitations of an earlier PS-distance [47]. Let a point be $\bar{x}$. The symmetrical (reflected) point of $\bar{x}$ with respect to a particular center $\bar{c}$ is $2\times\bar{c}-\bar{x}$. Let us denote this by $\bar{x}^*$. Let *knear* unique nearest neighbors of $\bar{x}^*$ be at Euclidean distances of d_i, $i=1,2,\ldots,knear$. Then

$$d_{ps}(\bar{x},\bar{c}) = d_{sym}(\bar{x},\bar{c})\times d_e(\bar{x},\bar{c}), \tag{1}$$

$$= \frac{\sum_{i=1}^{knear} d_i}{knear}\times d_e(\bar{x},\bar{c}), \tag{2}$$

where $d_e(\bar{x},\bar{c})$ is the Euclidean distance between the point $\bar{x}$ and the cluster center $\bar{c}$, and $d_{sym}(\bar{x},\bar{c})$ is a symmetry measure of $\bar{x}$ with respect to $\bar{c}$. Here *knear* is chosen equal to 2.

2.2. Motivation

The two fundamental questions that need to be addressed in any typical clustering scenario are: (i) how many clusters are actually present in the data, and (ii) how real or good the clustering is, i.e., the validity of the clusters formed [20]. Several cluster validity indices have been proposed in the literature, e.g., Davies–Bouldin (DB) index [18], Dunn's index [21], Xie–Beni (XB) index [49], I-index [38], PBMF-index [42], CS-index [17], XB* index [32], index proposed in [33,31], etc., to name just a few. A good review of the cluster validity indices and their categorization can be found in [32]. Most of the validity measures usually assume a certain geometrical structure in the cluster shapes. If clusters of different geometric shapes are present in the same data set, the above methods will not be able to find all of them perfectly. This article presents an attempt in this direction. Here we define a fuzzy cluster validity index named *FSym*-index (symmetry based cluster validity index) that uses a new definition of the point symmetry (PS) distance (d_{ps} [6]). This index is able to detect clusters of any shape or size as long as they possess the symmetry property.

2.3. Definition

In this article, we have proposed a fuzzy symmetry based cluster validity index, *FSym*-index, which measures the goodness of a partitioning in terms of "symmetry" and the separation present in the clusters. Let K cluster centers be denoted by $\bar{c}_i$ where $1\leqslant i\leqslant K$ and $U(X)=[u_{ij}]_{K\times n}$ be a partition matrix for the data. Then *FSym*-index is defined as follows:

$$FSym(K) = \left(\frac{1}{K} \times \frac{1}{\mathcal{E}_K} \times D_K\right), \tag{3}$$

where K is the number of clusters. Here,

$$\mathcal{E}_K = \sum_{i=1}^{K} E_i, \tag{4}$$

such that

$$E_i = \sum_{j=1}^{n} (u_{ij}^m \times d_{ps}(\bar{x}_j, \bar{c}_i)) \tag{5}$$

and

$$D_K = \max_{i,j=1}^{K} \|\bar{c}_i - \bar{c}_j\|. \tag{6}$$

D_K is the maximum Euclidean distance between two cluster centers among all centers. $d_{ps}(\bar{x}_j, \bar{c}_i)$ is the point symmetry distance [6] between the point $\bar{x}_j$ and the cluster center $\bar{c}_i$. This index is inspired by the PBMF-index developed in [42]. The objective is to maximize this index in order to obtain the actual number of clusters.

2.4. Framework of the formulation

In order to obtain the actual number of clusters and to achieve the proper clustering from a data set, *FSym*-index value has to be maximized. As formulated in Eq. (3), *FSym* is a composition of three factors, $1/K$, $1/\mathcal{E}_K$ and D_K. The first factor increases as K decreases; as *FSym* needs to be maximized for optimal clustering, so it will prefer to decrease the value of K. Second factor is the within cluster total symmetrical measure. For clusters which have good symmetrical structure, E_i value is small. This, in turn, indicates that formation of more clusters, which are symmetrical in shape, would be encouraged. Finally the third factor, D_K, measuring the maximum separation between a pair of clusters, increases with the value of K. As these three factors are complementary in nature, so they are expected to compete and balance each other critically for determining the proper partitioning.

2.5. Mathematical justification of FSym-index

In this section, we mathematically justify the new validity index by establishing its relationship to the well-known validity measure proposed by Dunn [21] for hard partitions. This is inspired by a proof of optimality of the Xie–Beni index [49].

Uniqueness and global optimality of the K-partition

The *Dunn-index* [21] is a hard K-partition cluster validity criterion. It has been proved in Ref. [21] that if $Dunn > 1$, unique, compact and separated hard clusters are found. This result has been used for fuzzy cluster validity in Ref. [49] where a hard K-partition, $W(X) = [w_{ij}]_{K\times n}$ is derived from a given fuzzy partition $U(X) = [u_{ij}]_{K\times n}$ as follows: for $j = 1, \ldots, n$, $w_{ij} = 1$ if $i = \text{argmax}_i\{u_{ij}\}$; $w_{ij} = 0$ otherwise. Note that if a particular data set X really has well-separated clusters, then the membership values obtained by any fuzzy clustering technique for this data set should be either significantly close to 1 or to 0. In this section it has been proved that if the Dunn's index corresponding to the optimal solution becomes sufficiently large, the optimal validity function *FSym*, too grows large, indicating that the method has found a unique hard K-partition.

Theorem 1. *Let FSym be the overall FSym-index value of any fuzzy partition for any $K = 2, \ldots, n-1$, and Dunn be the separation index of the corresponding partition. Then we have*

$$FSym \geqslant \frac{Dunn}{n \times K \times 0.5 \times knear \times d_{NN}^{max}},$$

where n is the total number of data points, K is the total number of clusters and knear is the number of nearest neighbors considered while computing d_{ps} as defined in Eq. (2). d_{NN}^{max} is the maximum nearest neighbor distance in the data set. That is $d_{NN}^{max} = \max_{i=1,\ldots,N} d_{NN}(\bar{x}_i)$, where $d_{NN}(\bar{x}_i)$ is the nearest neighbor distance of $\bar{x}_i$.

Proof. Suppose for a particular data set $X = \{\bar{x}_j;\ j = 1, 2, \ldots, n\}$ the fuzzy K-partition is an optimal partition with $\bar{c}_i$ $(i = 1, 2, \ldots, K)$ being the centers of each cluster C_i. Suppose u_{ij} is the fuzzy membership of the data point $\bar{x}_j$ belonging to cluster C_i. Then total symmetrical variation $\mathcal{E}_K$ of the optimal fuzzy K-partition is defined in Eq. (4). Thus,

$$\mathcal{E}_K = \sum_{i=1}^{K} \sum_{j=1}^{n} u_{ij} \times d_{ps}(\bar{x}_j, \bar{c}_i) \tag{7}$$

$$= \sum_{i=1}^{K} \sum_{j=1}^{n} u_{ij} \times \frac{\sum_{t=1}^{knear} d_t}{knear} \times d_e(\bar{x}_j, \bar{c}_i). \tag{8}$$

The total symmetrical variation of the corresponding hard K-partition is

$$E_{K_{hard}} = \sum_{i=1}^{K} \sum_{\bar{x}_j \in C_i} d_{ps}(\bar{x}_j, \bar{c}_i) \tag{9}$$

$$= \sum_{i=1}^{K} \sum_{\bar{x}_j \in C_i} \frac{\sum_{t=1}^{knear} d_t}{knear} d_e(\bar{x}_j, \bar{c}_i). \tag{10}$$

From the definitions of $\mathcal{E}_K$ and $E_{K_{hard}}$ above (as in finding crisp partitioning we are assigning the data points to the clusters with respect to which it has the highest membership value), we can get [49]:

$$\mathcal{E}_K \leqslant E_{K_{hard}}. \tag{11}$$

Let us assume that the reflected point of $\bar{x}_j$ with respect to the cluster center $\bar{c}_i$ lies near any point in the data space. Ideally, a point $\bar{x}_j$ is exactly symmetrical with respect to some $\bar{c}_i$ if the corresponding $d_1 = 0$. However considering the uncertainty of the location of a point as a sphere of radius $d_{NN}^{max}/2$ around it, we can bound d_1 as $d_1 \leqslant \frac{d_{NN}^{max}}{2}$ and $d_2 \leqslant \frac{3 \times d_{NN}^{max}}{2}$ (if d_1 is at a maximum $\frac{d_{NN}^{max}}{2}$ distance away, then d_2 is at a maximum $(\frac{d_{NN}^{max}}{2} + d_{NN}^{max})$ distance away). Similarly $d_3 = \frac{d_{NN}^{max}}{2} + 2 \times d_{NN}^{max} = \frac{5 d_{NN}^{max}}{2} = \frac{(2\times 3-1) d_{NN}^{max}}{2}, \ldots, d_t \leqslant \frac{(2t-1) d_{NN}^{max}}{2}$, where d_t is the tth nearest neighbor distance of $\bar{x}_j^*$. Considering the term $\frac{\sum_{t=1}^{knear} d_t}{knear}$, we can write

$$\frac{\sum_{t=1}^{knear} d_t}{knear} \leqslant \frac{d_{NN}^{max}}{2 \times knear} \left(\sum_{t=1}^{knear} (2 \times t - 1) \right). \tag{12}$$

The right hand side of the inequality may be written as

$$\frac{d_{NN}^{max}}{2 \times knear} \times \frac{(knear \times (2 \times 1 + (knear - 1)2))}{2} = \frac{knear \times d_{NN}^{max}}{2}. \tag{13}$$

So, combining Eqs. (8) and (11)–(13), we can write,

$$\mathcal{E}_K \leqslant \sum_{i=1}^{K} \sum_{\bar{x}_j \in C_i} 0.5 \times knear \times d_{NN}^{max} \times d_e(\bar{x}_j, \bar{c}_i)$$

$$\leqslant 0.5 \times knear \times d_{NN}^{max} \sum_{i=1}^{K} \sum_{\bar{x}_j \in C_i} d_e(\bar{x}_j, \bar{c}_i). \tag{14}$$

Let the center $\bar{c}_i$ be inside the boundary of cluster i, for $i = 1$ to K. Therefore

$$d_e(\bar{x}_j, \bar{c}_i) \leqslant dia(C_i), \tag{15}$$

for $\bar{x}_j \in C_i$ where $dia(C_i) = max_{\bar{x}_k, \bar{x}_j \in u_i} d_e(\bar{x}_k, \bar{x}_j)$. Combining Eqs. (14) and (15), we thus have

$$\mathcal{E}_K \leqslant 0.5 \times knear \times d_{NN}^{max} \sum_{i=1}^{K} \sum_{\bar{x}_j \in C_i} dia(C_i)$$

$$\leqslant 0.5 \times knear \times d_{NN}^{max} \sum_{i=1}^{K} n_i dia(C_i)$$

$$\leqslant 0.5 \times knear \times d_{NN}^{max} \times n \times \max_i dia(C_i).$$

Here n_i denotes the total number of data points in cluster i. So,

$$\frac{1}{\mathcal{E}_K} \geqslant \frac{1}{0.5 \times knear \times d_{NN}^{max} \times n \times \max_i dia(C_i)}.$$

Let us assume that the centroid $\bar{c}_i$ be inside the boundary of cluster i, for $i = 1$ to K. Then, in general, $\min_{i,j, i \neq j} dis(C_i, C_j) \leqslant D_K$ where $dis(C_i, C_j) = \min_{\bar{x}_i \in C_i, \bar{x}_j \in C_j} d_e(\bar{x}_i, \bar{x}_j)$ and $D_K = \max_{i,j=1}^{K} d_e(\bar{c}_i, \bar{c}_j)$. It is because $dis(C_i, C_j)$ denotes minimum distance between any two points belonging to two different clusters where as D_K denotes the maximum separation between any two cluster centers:

$$FSym(K) = \frac{D_K}{K \times \mathcal{E}_K} \geqslant \frac{\min_{i,j} dis(C_i, C_j)}{K \times 0.5 \times knear \times d_{NN}^{max} \times n \times \max_i dia(C_i)},$$

i.e.,

$$FSym(K) \geqslant \frac{\min_{1\leqslant i\leqslant K-1}\left\{\min_{i+1\leqslant j\leqslant K}\frac{dis(C_i,C_j)}{\max_{1\leqslant k\leqslant K}dia(C_k)}\right\}}{K\times 0.5\times knear\times d_{NN}^{max}\times n}. \tag{16}$$

The separation index *Dunn* [21] is defined as

$$Dunn = \min_{1\leqslant i\leqslant K-1}\left\{\min_{i+1\leqslant j\leqslant K}\left\{\frac{dis(C_i,C_j)}{\max_{1\leqslant k\leqslant K}dia(C_k)}\right\}\right\}. \tag{17}$$

So, combining Eqs. (16) and (17), we get

$$FSym(K) \geqslant \frac{Dunn}{K\times 0.5\times knear\times d_{NN}^{max}\times n}.$$

Since the denominator is constant for a given K, *FSym* increases as *Dunn* increases. As mentioned earlier, it has been proved by Dunn [21] that if $Dunn > 1$ the hard K-partition is unique. Thus, if the data set has a distinct substructure and the fuzzy partitioning algorithm has found it, then the corresponding *FSym*-index value is bounded as above. □

Note that the assumption behind the proof is that clusters present in the data set have point symmetry property. Thus the proof is not applicable in the case when the clusters do not have any point symmetry property.

3. Fuzzy-VGAPS clustering: fuzzy variable string length genetic point symmetry based clustering technique

In this section, a variable string length genetic fuzzy clustering technique using the point symmetry based distance is proposed. Here the best partition is considered to be the one that corresponds to the maximum value of the proposed *FSym*-index. In Fuzzy-VGAPS, both the number of clusters as well as the appropriate fuzzy clustering of the data are evolved simultaneously using the search capability of genetic algorithms.

3.1. Motivation

In recent years a large number of clustering techniques have been developed, e.g., Fuzzy-VGA [39], HNGA [44], an evolutionary clustering algorithm [14], tabu search based clustering technique [36], many fuzzy clustering techniques [12,50,48], different variants of K-means clustering techniques [40,9,26], other clustering techniques like [16,37], etc., to name just a few. Among these clustering algorithms only a few [39,44,14] can determine both the number of clusters as well as the appropriate partitioning automatically from different data sets. In the above-mentioned genetic clustering techniques for automatic evolution of clusters, assignment of points to different clusters is done on the lines of K-means clustering algorithm. Consequently, all these approaches are only able to find compact hyperspherical, equisized and convex clusters like those detected by the K-means algorithm [29]. If clusters of different geometric shapes are present in the same data set, the above methods will not be able to find all of them perfectly. This article presents an attempt in this direction. In this paper, a variable string length GA (VGA) based fuzzy clustering technique is proposed. Here membership values of points to different clusters are calculated based on the newly developed point symmetry based distance [6]. The newly proposed *FSym*-index is used as the optimizing criterion. The characteristic features of the proposed clustering technique, referred to as Fuzzy-VGAPS clustering, which distinguishes it from the state-of-the-art approaches are as follows. Use of variable string length GA allows the encoding of a variable number of clusters. The *FSym*-index, used as the fitness function, provides the most appropriate partitioning even when the number of clusters, K, is varied. Moreover, clusters of any shape (e.g., hyperspherical, linear, ellipsoidal, ring shaped, etc.) and size (mixture of small and large clusters, i.e., clusters of unequal sizes) as long as they satisfy the property of point symmetry can be detected. Again use of GA enables the algorithm to come out of local optima, a typical problem associated with local search methods like the K-means and FCM. The basic steps of the proposed clustering technique are provided in Fig. 1. The technique is described below in detail.

3.2. Chromosome representation and population initialization

In Fuzzy-VGAPS clustering, center based encoding is used. Real numbers are encoded in the chromosomes which represent the coordinates of the centers of the partitions. The length of a particular chromosome p is l_p, given by $l_p = D\times M_p$, where D is the dimension of the data and M_p is the number of cluster centers encoded in that chromosome. For example, in three-dimensional space, the chromosome ⟨2.3 1.4 7.6 2.1 3.4 0.01 0.06 2.3 6.7 3.2 11.72 9.5⟩ encodes 4 cluster centers, (2.3,1.4,7.6), (2.1,3.4,0.01), (0.06,2.3,6.7) and (3.2,11.72,9.5). Centers are considered to be indivisible. For a particular chromosome p, the number of cluster centers encoded in it, M_p, is calculated as $M_p = (\text{rand}()\text{mod}M^*) + 2$. Here, rand() is a function returning an integer, and M^* is a soft estimate of the upper bound of the number of clusters. The initial number of clusters will therefore range from two to $M^* + 1$. The M_p centers encoded in a chromosome are randomly selected distinct points from the data set. The selected points are distributed randomly in the chromosome. Thereafter a few iterations of the K-means algorithm (with the number of clusters $(K) = M_p$) is executed with the set of centers encoded in each chromosome.

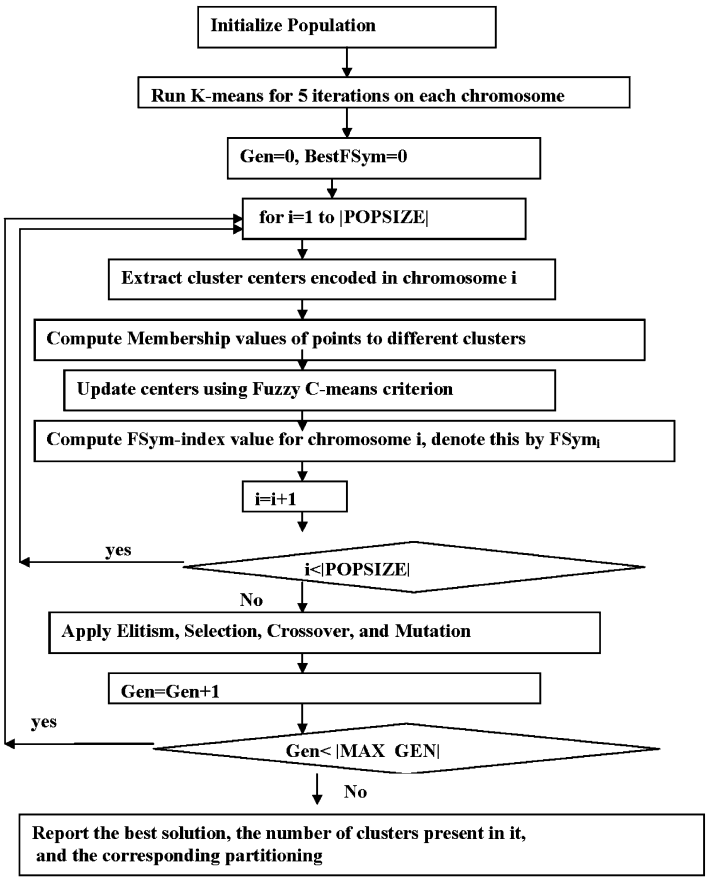

Fig. 1. Flowchart of Fuzzy-VGAPS clustering technique.

The resultant centers are used to replace the centers in the corresponding chromosomes. The use of a few iterations of *K*-means makes the centers more separated initially, than purely randomly selected ones.

3.3. Fitness computation

The fitness computation is performed for each chromosome. This is composed of two steps. Firstly, membership values of n points to different clusters are computed. Next, the *FSym*-index is computed and used as a measure of the fitness of the chromosome.

3.3.1. Computing the membership values

In Fuzzy-VGAPS clustering algorithm, a particular point is assigned to a particular cluster with membership value 1 if that is truly symmetrical with respect to that cluster center. For points whose symmetrical measures to all the cluster centers are greater than some threshold (i.e., points which are not symmetrical with respect to any cluster centers), membership values are calculated using the Euclidean distance. For each point $\bar{x}_j,\ j = 1, 2, \ldots, n$, the membership values to K different clusters are calculated in the following way. Find the cluster center nearest to $\bar{x}_j$ in the symmetrical sense. That is, we find the cluster center k that is nearest to the input pattern $\bar{x}_j$ using the minimum-symmetric-distance criterion: $k = \text{argmin}_{i=1,\ldots,K} d_{ps}(\bar{x}_j, \bar{c}_i)$ where the point symmetry based distance $d_{ps}(\bar{x}_j, \bar{c}_i)$ is computed by Eq. (2). Here, $\bar{c}_i$ denotes the center of the ith cluster. If the corresponding $d_{sym}(\bar{x}_j, \bar{c}_k)$ (as defined in Eq. (1)) is smaller than a pre-specified parameter θ, then the membership u_{ij} is updated using the following criterion:

$$u_{ij} = \begin{cases} 1, & \text{if } i = k, \\ 0, & \text{if } i \neq k. \end{cases}$$

Otherwise, the membership u_{ij} is updated by using the following rule which corresponds to the normal Fuzzy C-means [11] algorithm:

$$u_{ij} = \frac{1}{\sum_{k=1}^{K} \left(\frac{d_{ij}}{d_{kj}}\right)^{\frac{2}{m-1}}}, \tag{18}$$

where, $m \in [1, \infty)$ is a weighting exponent called the fuzzifier. Here we set $m = 2$. d_{ij} represents the Euclidean distance between a pattern $\bar{x}_j$ and the cluster center $\bar{c}_i$. Note that in Eq. (18), the Euclidean distance has been used instead of the symmetry based one. It is because Eq. (18) is derived in [11] under the assumption that the distance measure used, should be a norm. As mentioned earlier, the newly proposed d_{ps} measure is not a norm. Thus, the point symmetry based distance cannot be used in Eq. (18).

The value of θ is determined as in Ref. [6]. It is set equal to d_{NN}^{max}, maximum nearest neighbor distance in the data set.

3.3.2. Updating the centers

The centers encoded in a chromosome are updated using the following equation as in FCM

$$\bar{c}_i = \frac{\sum_{j=1}^{n} (u_{ij})^m \bar{x}_j}{\sum_{j=1}^{n} (u_{ij})^m}, \quad 1 \leqslant i \leqslant K. \tag{19}$$

Centers are updated as above in order to incorporate a limited amount of local search (provided by FCM like center update) for speeding up the convergence of Fuzzy-VGAPS.

3.3.3. Fitness calculation

The fitness of a chromosome indicates the degree of goodness of the solution it represents. The fitness of a chromosome is computed using the newly developed *FSym*-index. The fitness function for chromosome j is defined as $FSym_j$, i.e., the *FSym* index computed for the chromosome. The objective of the GA is to maximize this fitness function.

3.4. Selection

Conventional proportional selection is applied on the population of strings. Here, a string receives a number of copies that is proportional to its fitness in the population. We have used roulette wheel strategy for implementing the proportional selection scheme.

3.5. Crossover

We have used the crossover operation similar to that used in Fuzzy-VGA based clustering [39]. The cluster centers are considered to be indivisible, i.e., the crossover points can only lie in between two cluster centers. The crossover operation designed in [39] ensures that the information exchange takes place in such a way that both the offsprings encode the centers of at least two clusters. The operator is defined as follows [39]: let parent chromosomes P_1 and P_2 encode M_1 and M_2 cluster centers, respectively. τ_1, the crossover point in P_1, is generated as $\tau_1 = \text{rand}() \bmod M_1$. Let τ_2 be the crossover point in P_2, and it may vary in between $[\text{LB}(\tau_2), \text{UB}(\tau_2)]$, where $\text{LB}(\tau_2)$ and $\text{UB}(\tau_2)$ indicate the lower and upper bounds of the range of τ_2, respectively. $\text{LB}(\tau_2)$ and $\text{UB}(\tau_2)$ are given by

$$\text{LB}(\tau_2) = \min\{2, \max\{0, 2 - (M_1 - \tau_1)\}\} \tag{20}$$

and

$$\text{UB}(\tau_2) = \{M_2 - \max\{0, 2 - \tau_1\}\}. \tag{21}$$

Therefore, τ_2 is given by

$$\tau_2 = \begin{cases} \text{LB}(\tau_2) + \text{rand}() \bmod (\text{UB}(\tau_2) - \text{LB}(\tau_2)), & \text{if } (\text{UB}(\tau_2) \geqslant \text{LB}(\tau_2)), \\ 0, & \text{otherwise.} \end{cases}$$

Here, rand() is a function returning an integer. $(x \bmod y)$ returns the remainder of the integer division (x/y), i.e., it returns an integer ranging between 0 to $(y-1)$. It can be verified by some simple calculations that if the crossover points τ_1 and τ_2 are chosen according to the above rules, then none of the offsprings generated would have less than two clusters.

Crossover probability is selected adaptively as in Ref. [46]. The expressions for crossover probabilities are computed as follows. Let f_{max} be the maximum fitness value of the current population, $\bar{f}$ be the average fitness value of the population and f' be the larger of the fitness values of the solutions to be crossed. Then the probability of crossover, μ_c, is calculated as:

$$\mu_c = \begin{cases} k_1 \times \frac{(f_{max} - f')}{(f_{max} - \bar{f})}, & \text{if } f' > \bar{f}, \\ k_3, & \text{if } f' \leqslant \bar{f}. \end{cases}$$

Here, as in Ref. [46], the values of k_1 and k_3 are kept equal to 1.0. Note that, when $f_{max} = \bar{f}$, then $f' = f_{max}$ and μ_c will be equal to k_3. The value of μ_c is increased when the better of the two chromosomes to be crossed is itself quite poor. In contrast when it is a good solution, μ_c is low so as to reduce the likelihood of disrupting a good solution by crossover.

3.6. Mutation

Mutation is applied on each chromosome with probability μ_m. Mutation is of three types:

(1) Each cluster center encoded in a chromosome is replaced with a random variable drawn from a Laplacian distribution, $p(\epsilon) \propto e^{-\frac{|\epsilon-\mu|}{\delta}}$, where the scaling factor δ sets the magnitude of perturbation. Here, μ is the value at the position which is to be perturbed. The scaling factor δ is chosen equal to 1.0. The old value at the position is replaced with the newly generated value. Here, this type of mutation operator is applied for all dimensions independently.
(2) One randomly generated cluster center is removed from the chromosome, i.e., the total number of clusters encoded in the chromosome is decreased by 1.
(3) The total number of clusters encoded in the chromosome is increased by 1. One randomly chosen point from the data set is encoded as the new cluster center.

Any one of the above-mentioned types of mutation is applied randomly on a particular chromosome if it is selected for mutation.

The mutation probability is selected adaptively for each chromosome as in Ref. [46]. Let f_{max} be the maximum fitness value of the current population, $\bar{f}$ be the average fitness value of the population and f be the fitness value of the solution to be mutated. The expression for mutation probability, μ_m, is given below:

$$\mu_m = \begin{cases} k_2 \times \frac{(f_{max}-f)}{(f_{max}-\bar{f})}, & \text{if } f > \bar{f}, \\ k_4, & \text{if } f \leqslant \bar{f}. \end{cases}$$

Here, values of k_2 and k_4 are kept equal to 0.5. This adaptive mutation helps GA to avoid getting stuck at local optimum. When GA converges to a local optimum, i.e., when $f_{max} - \bar{f}$ decreases, μ_c and μ_m both will be increased. As a result, GA will come out of local optimum.

3.7. Termination

In this article, we have executed the algorithm for a fixed number of generations. Moreover, the elitist model of GAs has been used, where the best string seen so far is stored in a location within the population. The best string of the last generation provides the solution to the clustering problem.

4. Implementation results and comparative study

For the experimental results, eight data sets are considered. These are categorized into two groups. The first group consists of four artificial data sets and the second group consists of four real-life data sets obtained from [1]. A short description of the data sets in terms of the number of data points present, dimension of the data set, number of clusters present in the data set is provided in Table 1. The four artificial data sets are displayed in Fig. 2a–d. The first two artificial data sets, *Sym_3_2* and *Ring_3_2* contain a mixture of hyperspherical, ellipsoidal and ring-shaped clusters. *Ring_3_2* also contains some non-convex clusters. The last two artificial data sets, used in [5], contain overlapping clusters.

In the first part of the experimental results, we have shown the superiority of *FSym*-index as compared to PBMF-index [42], a new fuzzy cluster validity index based on relative degree of sharing (Fuzzy-RDS) proposed in [33] and a modified version of XB-index, XB* [32]. In all these cases, Fuzzy-VGAPS is used as the underlying partitioning method. The parameters of the algorithm are as follows. The population size, $P = 100$, number of generations = 50. Mutation and crossover probabilities of the Fuzzy-VGAPS algorithm are selected adaptively as discussed earlier. The number of clusters automatically determined by the proposed Fuzzy-VGAPS clustering optimizing the four indices are shown in Table 1. This table shows that Fuzzy-VGAPS optimizing *FSym*-index is able to determine the appropriate number of clusters for all the data sets. The other indices

Table 1
Results obtained with the different data sets using Fuzzy-VGAPS clustering algorithms optimizing *FSym*-index, PBMF-index, fuzzy cluster validity index, XB*-index. Here AC denotes the actual number of clusters present in the data set.

Name	No. of points	Dimension	AC	Obtained number of clusters using			
				FSym	PBMF	Fuzzy-RDS	XB*
Sym_3_2	600	2	3	3	6	3	2
Ring_3_2	350	2	3	3	8	16	4
Data_10_2	500	2	10	10	11	19	2
Data_9_2	900	2	9	9	7	6	4
Iris	150	4	3	3	3	9	2
Cancer	683	9	2	2	4	17	2
Lungcancer	32	56	3	3	3	3	3
Glass	214	9	6	6	7	12	2

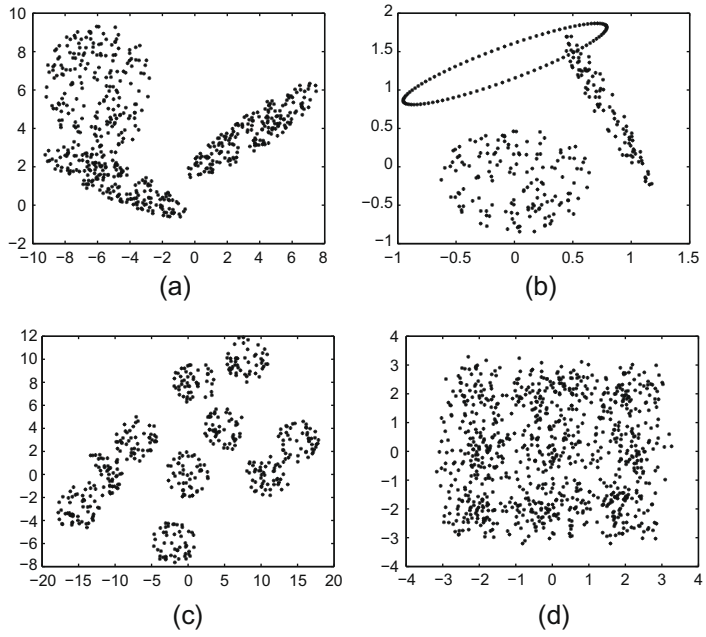

Fig. 2. (a) *Sym_3_2*; (b) *Ring_3_2*; (c) *Data_10_2*; (d) *Data_9_2*.

are able to do so for only two data sets. Thus it can be concluded that Fuzzy-VGAPS optimizing *FSym*-index definitely has an edge over the others in identifying the appropriate number of clusters and the appropriate partitioning for point symmetric type of clusters.

In the second part of the experimental results, Fuzzy-VGAPS clustering technique optimizing *FSym*-index is compared with seven recent clustering techniques:

- Fuzzy-VGA [39], optimizing well-known XB-index [49].
- A crisp algorithm: well-known *K*-means clustering technique with newly developed point symmetry based distance (PSKM). This algorithm follows the basic steps of SBKM proposed in [47] but uses newly developed point symmetry based distance, d_{ps} [6]. It *a priori* assumes the number of clusters present in a data set.
- An adaptation of Fuzzy C-means algorithm using the proposed symmetry distance (PSFCM) to evaluate the advantage provided by the genetic approach. It also assumes *a priori* the number of clusters present in a data set.
- A recently developed fuzzy genetic clustering technique: multistage random sampling genetic algorithm based Fuzzy C-means clustering algorithm (GMFCM) developed in Ref. [19]. It also assumes *a priori* the number of clusters present in a data set.
- A recently developed fuzzy variant of an evolutionary algorithm for clustering (EAC-FCM) [14], which uses a fuzzy cluster validity index, fuzzy silhouette [13] and a fuzzy local search algorithm.
- Hybrid centroid–medoid heuristics (Hybrid) [26]: this is an enhancement of *k*-means clustering technique where local search is used to accelerate convergence without greatly increasing the number of iterations. It also assumes *a priori* the number of clusters present in a data set.
- A shape based clustering technique, SPARCL [16]. It also assumes *a priori* the number of clusters present in a data set.

The parameters of the Fuzzy-VGA clustering technique are as follows: the population size = 100, total number of generations = 20, probability of mutation = 0.01 and probability of crossover = 0.8. Both *K*-means algorithm with point symmetry based distance (PSKM) and Fuzzy C-means with point symmetry based distance (PSFCM) clustering techniques are executed until they converged. Here number of clusters (K) is set equal to the number of clusters identified by *FSym*-index for a particular data set. For GMFCM and EAC-FCM, population size is kept equal to 100. In GMFCM, the other parameters are set as specified in Ref. [19]. For GMFCM again the number of clusters is kept equal to the number of clusters identified by *FSym*-index. In Fuzzy-VGAPS, Fuzzy-VGA and EAC-FCM while generating the initial population the maximum value of the number of clusters is kept equal to $\sqrt{n}$ where n is the total number of points present in the data set, and the fuzzifier $m = 2.0$. For SPARCL, the results are obtained from the respective authors. Thus, it was available for only one run for all data sets except *Lungcancer*. Authors did not execute their code on *Lungcancer* due to its higher dimensional nature. For all data sets, the actual number of clusters present in the data set and those obtained by the three algorithms, Fuzzy-VGA, EAC-FCM and Fuzzy-VGAPS, which can determine K automatically, are shown in Table 2. The results of the other algorithms, i.e., which assume a fixed K, are shown visually.

Table 2
Results obtained with the different data sets using Fuzzy-VGAPS, Fuzzy-VGA and EAC-FCM clustering algorithms.

Name	Actual number of clusters	Obtained number of clusters		
		Fuzzy-VGAPS	Fuzzy-VGA	EAC-FCM
Sym_3_2	3	3	2	16
Ring_3_2	3	3	4	11
Data_10_2	10	10	16	7
Data_9_2	9	9	16	7
Iris	3	3	2	3
Cancer	2	2	2	2
Lungcancer	2	2	5	3
Glass	6	6	6	2

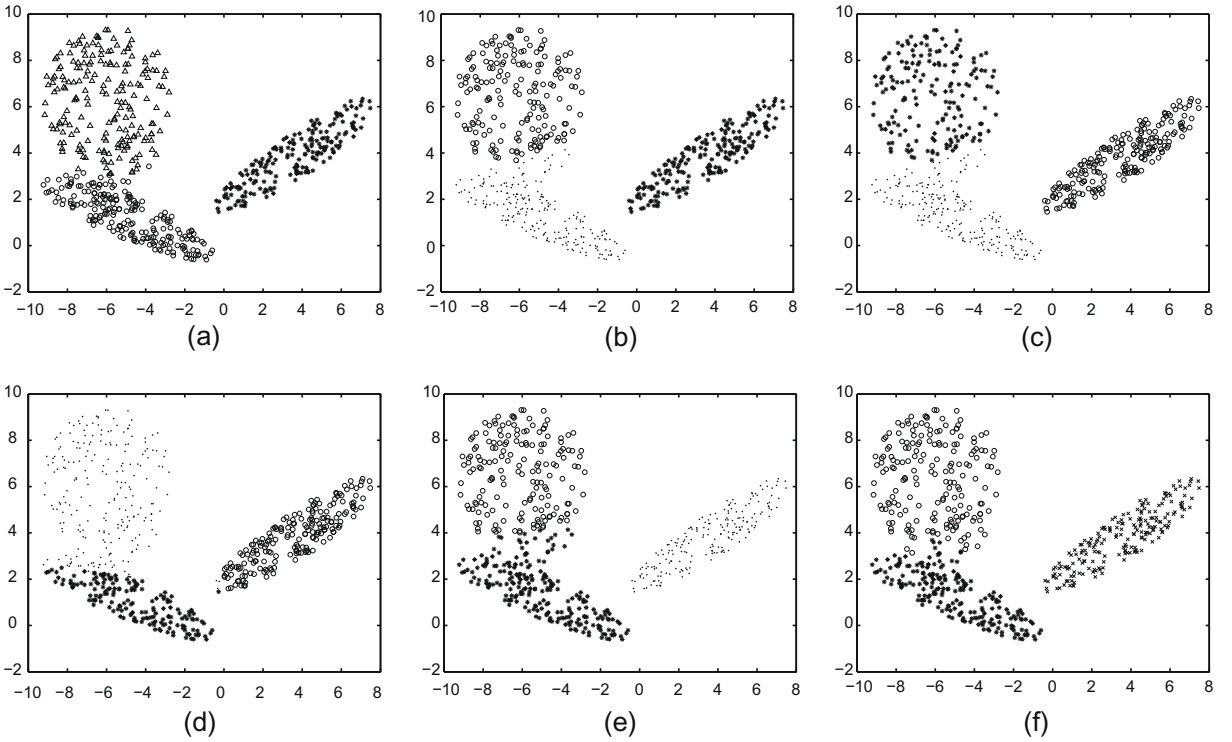

Fig. 3. Clustering results on *Sym_3_2* by (a) Fuzzy-VGAPS providing $K = 3$; (b) PSKM for $K = 3$; (c) PSFCM for $K = 3$; (d) GMFCM for $K = 3$; (e) Hybrid algorithm for $K = 3$; (f) SPARCL algorithm for $K = 3$.

4.1. Discussion of results

- *Sym_3_2:* This data set is designed to test whether the proposed Fuzzy-VGAPS clustering is able to detect automatically some hyper-ellipsoidal shaped clusters. As is evident from Table 2, Fuzzy-VGAPS clustering is able to detect automatically the proper number of partitions and the proper partitioning from this data set as the clusters present here are symmetrical in shape. The corresponding partitioning is shown in Fig. 3a. Fuzzy-VGA is not able to detect the proper partitioning because clusters are not hyperspherical in nature (refer to Table 2). The partitionings obtained by PSKM, PSFCM and GMFCM for $K = 3$ (number of clusters identified by Fuzzy-VGAPS) for this data set are shown in Fig. 3b–d, respectively. EAC-FCM indicates $K = 16$ as the proper number of clusters as clusters present in this data set are not hyperspherical in nature (refer to Table 2). The partitionings indicated by the Hybrid clustering technique [26] and SPARCL clustering technique [16] are shown in Fig. 3e and f, respectively. SPARCL is not able to detect the overlapping clusters correctly.
- *Ring_3_2:* This data set is generated to show that Fuzzy-VGAPS is able to detect automatically different shaped clusters present in a data set as long as they are point symmetric. Fuzzy-VGAPS clustering is able to automatically detect the proper number of partitions (shown in Table 2) and the proper partitioning from this data set as clusters present here have "symmetrical" structure. The partitioning is shown in Fig. 4a. As the data set contains non-convex clusters, Fuzzy-VGA is not able to detect the proper partitioning (refer to Table 2). The partitioning obtained by PSKM, PSFCM and GMFCM for

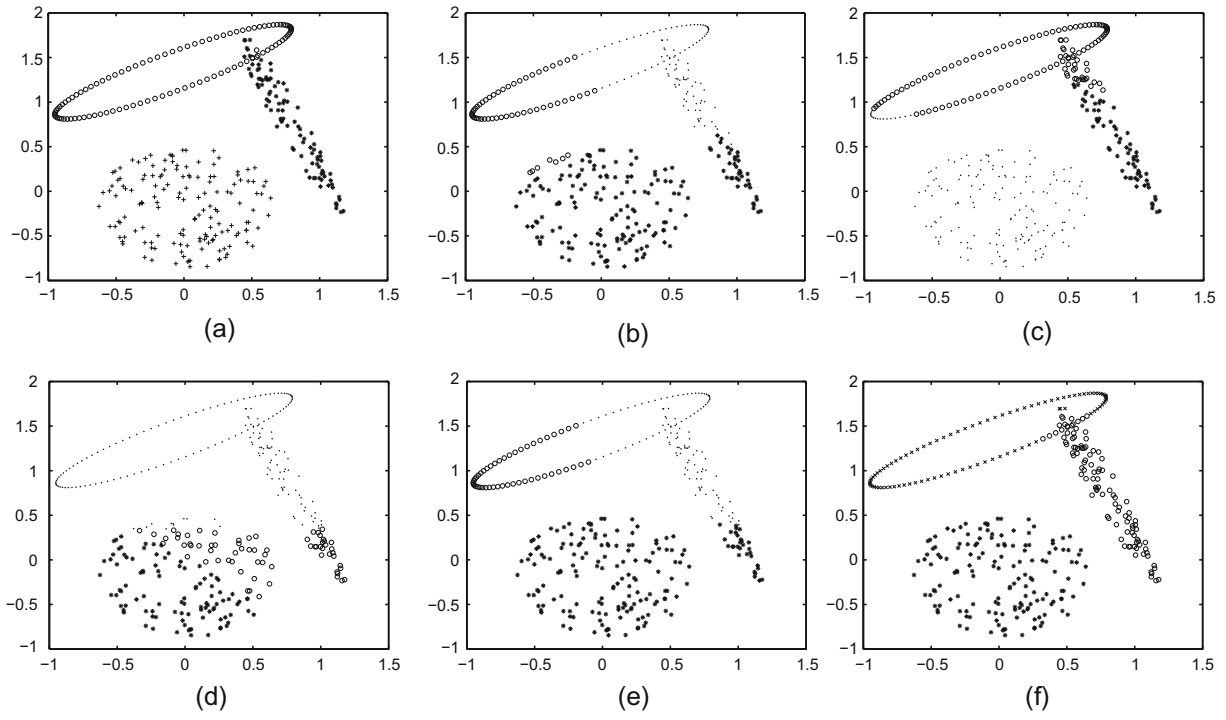

Fig. 4. Clustering results on *Ring_3_2* by (a) Fuzzy-VGAPS providing $K = 3$; (b) PSKM for $K = 3$; (c) PSFCM for $K = 3$; (d) GMFCM for $K = 3$; (e) Hybrid algorithm for $K = 3$; (f) SPARCL algorithm for $K = 3$.

$K = 3$ (number of clusters identified by Fuzzy-VGAPS) for this data set are shown in Fig. 4b–d, respectively. EAC-FCM automatically indicates $K = 11$ as the proper number of clusters as clusters present in this data set are not hyperspherical in nature (refer to Table 2). The partitionings indicated by the Hybrid clustering technique [26], and SPARCL clustering technique [16] for this data set are shown in Fig. 4e and f, respectively. Hybrid clustering technique, in general, fails to detect non-hyperspherical shaped clusters. SPARCL is again not able to detect the overlapping clusters correctly.

- *Data_10_2:* This data set is used to test the performance of the proposed Fuzzy-VGAPS clustering technique for some overlapping clusters. Fuzzy-VGAPS is able to automatically detect the proper partitioning and the proper number of partitions from this data set. The corresponding partitioning is shown in Fig. 5a. This result shows that the fuzzy characteristics of the proposed Fuzzy-VGAPS is used to make the partition more flexible and thus it can handle overlapping clusters. Fuzzy-VGA automatically indicates $K = 16$ as the proper number of clusters (refer to Table 2). The partitionings obtained by PSKM, PSFCM, GMFCM, Hybrid and SPARCL clustering techniques for $K = 10$ (number of clusters identified by Fuzzy-VGAPS) for this data set are shown in Fig. 5b–f, respectively. EAC-FCM indicates $K = 7$ as the proper number of clusters (refer to Table 2).
- *Data_9_2:* This data set is used to show that the proposed Fuzzy-VGAPS is able to automatically detect highly overlapping clusters. Fuzzy-VGAPS is able to detect the proper number of partitions from this data set. The corresponding partitioning is shown in Fig. 6a. This result again clearly illustrates the fuzzy characteristics of the proposed Fuzzy-VGAPS in detecting more overlapping clusters. Fuzzy-VGA automatically indicates $K = 16$ as the proper number of clusters (refer to Table 2). The partitioning obtained by PSKM, PSFCM and GMFCM for $K = 9$ (number of clusters identified by Fuzzy-VGAPS) for this data set are shown in Fig. 6b–d, respectively. EAC-FCM indicates $K = 7$ as the proper number of clusters (refer to Table 2). The partitionings indicated by the Hybrid and SPARCL clustering techniques for this data set are shown in Fig. 6e and f, respectively.
- *Iris:* For real-life data sets, no visualization is possible as these are higher dimensional data sets. For these data sets, the *Minkowski Scores* [7] are reported for each algorithm. This is a measure of the quality of a solution given the true clustering. Let T be the "true" solution and S the solution we wish to measure. Denote by n_{11} the number of pairs of elements that are in the same cluster in both S and T. Denote by n_{01} the number of pairs that are in the same cluster only in S, and by n_{10} the number of pairs that are in the same cluster in T. *Minkowski Score* (MS) is then defined as:

$$\mathrm{MS}(\mathrm{T},\mathrm{S}) = \sqrt{\frac{n_{01}+n_{10}}{n_{11}+n_{10}}}. \tag{22}$$

For MS, the optimum score is 0, with lower scores being "better". The MS-scores and their variances provided by the seven clustering algorithms for these four data sets are reported in Table 3. Statistical analysis of variance (ANOVA) [4] is performed for the real-life data sets on the combined MS values of all the algorithms when each is executed ten times. For

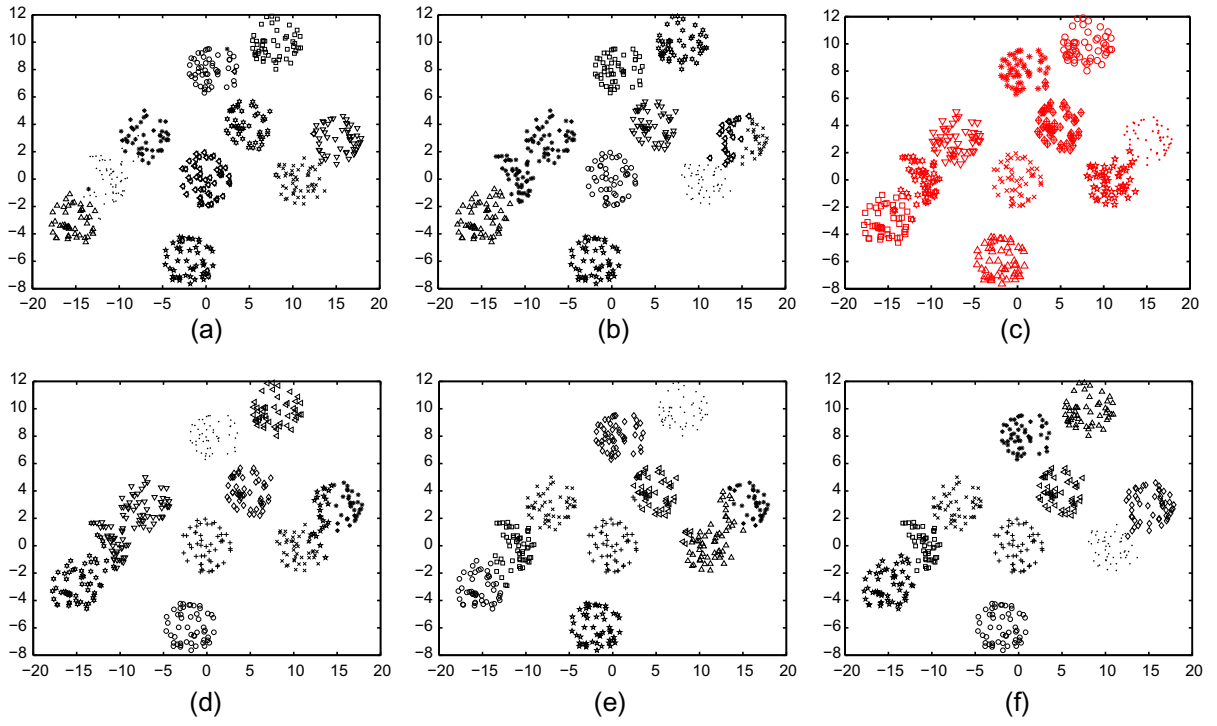

Fig. 5. Clustering results on *Data_10_2* by (a) Fuzzy-VGAPS providing $K = 10$; (b) PSKM for $K = 10$; (c) PSFCM for $K = 10$; (d) GMFCM for $K = 10$; (e) Hybrid algorithm for $K = 10$; (f) SPARCL algorithm for $K = 10$.

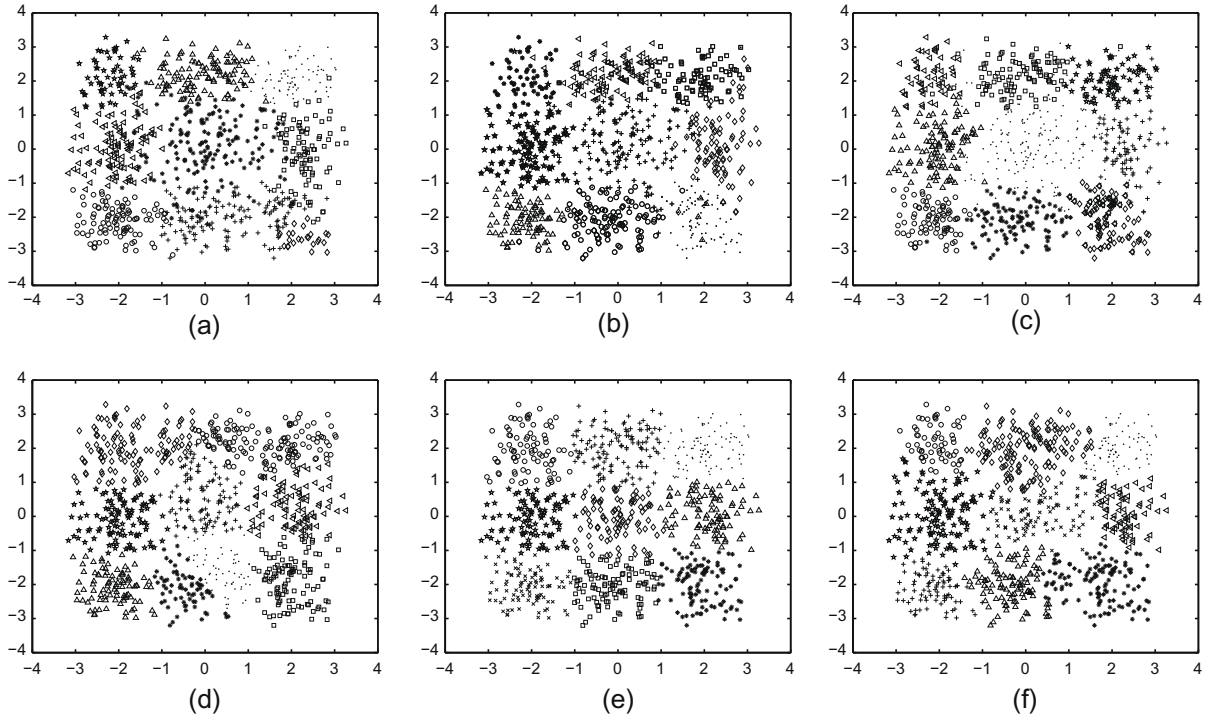

Fig. 6. Clustering result on *Data_9_2* by (a) Fuzzy-VGAPS providing $K = 9$; (b) PSKM for $K = 9$; (c) PSFCM for $K = 9$; (d) GMFCM for $K = 9$; (e) Hybrid algorithm for $K = 9$; (f) SPARCL algorithm for $K = 9$.

SPARCL, the results are obtained from the respective authors. Thus, it was available for only one run for all data sets except *Lungcancer*. Authors did not execute their code on *Lungcancer* due to its higher dimensional nature.

Table 3
Estimated marginal means and variances of Minkowski Scores of seven algorithms obtained for four real-life data sets.

Data set	Minkowski Score							
	Fuzzy-VGAPS	Fuzzy-VGA	PSKM	PSFCM	GMFCM	EAC-FCM	Hybrid	SPARCL
Iris	0.62 ± 0.001	0.85 ± 0.002	0.65 ± 0.02	0.62 ± 0.024	0.52 ± 0.024	0.77 ± 0.033	0.82 ± 0.003	0.57
Cancer	0.32 ± 0.0004	0.39 ± 0.0001	0.37 ± 0.021	0.33 ± 0.0021	0.31 ± 0.0021	0.39 ± 0.00234	0.35 ± 0.00345	0.33
Lungcancer	0.82 ± 0.001	1.13 ± 0.003	0.84 ± 0.0022	0.88 ± 0.0132	0.90 ± 0.0012	1.15 ± 0.0021	1.39 ± 0.0043	–
Glass	1.01 ± 0.001	1.25 ± 0.002	1.11 ± 0.012	1.12 ± 0.023	1.11 ± 0.021	1.32 ± 0.0023	1.14 ± 0.031	1.17

As is evident from Table 2, Fuzzy-VGAPS is able to automatically determine the proper number of clusters from this data set. Fuzzy-VGA automatically determines two clusters from this data set, which is also often obtained for many other methods for *Iris*. The MS-score corresponding to the partitioning provided by Fuzzy-VGAPS compared to that of Fuzzy-VGA (refer to Table 3) is also the minimum. ANOVA tests show that the difference in mean MS-scores of Fuzzy-VGAPS with Fuzzy-VGA clustering is significant. These again reveal that the partitioning provided by Fuzzy-VGAPS is better than that of Fuzzy-VGA. EAC-FCM also automatically determines three clusters from this data set. But the corresponding MS-score (shown in Table 3) is worse than that of Fuzzy-VGAPS. The MS-scores of the partitionings provided by PSKM, PSFCM, GMFCM, Hybrid and SPACL for $K = 3$ (number of partitions identified by Fuzzy-VGAPS) are also provided in Table 3. ANOVA analysis reveals that the performance of both Fuzzy-VGAPS and PSFCM are same for this data set. Table 3 shows that GMFCM performs the best for this data set in terms of MS-score. Hybrid clustering technique attains a higher MS value for this data set (refer to Table 3). But SPARCL attains a smaller MS-score for this particular data set.

- *Cancer:* Here we use the Wisconsin Breast *Cancer* data set, consists of 683 sample points. Each pattern has nine features. There are two categories in the data: malignant and benign. The two classes are known to be linearly separable. For this data set, all the three algorithms, Fuzzy-VGAPS, Fuzzy-VGA and EAC-FCM, are able to determine the proper number of clusters (refer to Table 2). But the MS-score corresponding to Fuzzy-VGAPS is the minimum (refer to Table 3) and ANOVA analysis shows that this improvement is statistically significant. This again shows that the partitioning obtained by Fuzzy-VGAPS clustering is better than that of Fuzzy-VGA and EAC-FCM. PSKM, PSFCM, GMFCM, Hybrid and SPACL algorithms are also executed on this data set with $K = 2$ (number of partitions identified by Fuzzy-VGAPS) and the corresponding MS-scores are also reported in Table 3. Table 3 shows that Fuzzy-VGAPs and PSFCM again perform similarly for this data set and this is also verified after ANOVA analysis. It is also revealed from Table 3 that GMFCM again performs the best for this data set in terms of MS-score. Hybrid and SPARCL clustering techniques attain slightly higher MS values for this data set.
- *Lungcancer:* This data consists of 32 instances having 56 features each. The data describes three types of pathological lung cancers. It can be seen from Table 2 that only Fuzzy-VGAPS is able to find out the proper number of clusters from this data set. The MS-scores, reported in Table 3, again demonstrate the superior performance of Fuzzy-VGAPS over Fuzzy-VGA and EAC-FCM for automatically detecting the proper partitioning from a data set. PSKM, PSFCM, GMFCM, Hybrid and SPACL algorithms are also executed on this data set with $K = 2$ (number of partitions identified by Fuzzy-VGAPS) and the corresponding MS-scores are also reported in Table 3. Table 3 shows that Fuzzy-VGAPS performs the best for this data set in terms of MS-score. ANOVA statistical analysis is also done here. The analysis (results shown in Table 3) shows that the mean MS differences of all the algorithms are statistically significant.
- *Glass:* This is the glass identification data consisting of 214 instances having 9 features (an Id feature has been removed). There are six categories present in this data set. Both Fuzzy-VGAPS and Fuzzy-VGA clustering algorithms are able to detect the proper number of partitions from this data set. But the MS-score (as shown in Table 3) corresponding to the partitioning provided by Fuzzy-VGAPS clustering is less than that of Fuzzy-VGA clustering. ANOVA analysis shows that the mean MS differences of both the algorithms are statistically significant. EAC-FCM automatically determines two clusters from this data set and the MS-score corresponding to this partitioning is provided in Table 3. PSKM, PSFCM, GMFCM, Hybrid and SPACL clustering techniques are also executed on this data set with $K = 6$ (number of partitions identified by Fuzzy-VGAPS) and the corresponding MS-scores are also reported in Table 3. Table 3 shows that Fuzzy-VGAPS again performs the best for this data set in terms of MS-score.

Summary of results

It can be observed from the above results that the proposed Fuzzy-VGAPS is able to determine automatically the proper number of partitions and the proper partitioning from a wide variety of data sets having any type of similar density clusters, irrespective of their geometrical shape and overlapping nature, as long as they possess the characteristic of symmetry. Note that it will fail for data sets where clusters do not have any point symmetry property (where the other algorithms will fail as well). The PSKM and PSFCM clustering algorithms are based on local search and hence they may often get stuck at local optima depending on the choice of the initial cluster centers. The use of genetic algorithm and the newly proposed cluster validity index, *FSym*-index, in Fuzzy-VGAPS enables it to detect automatically the number of clusters present in a data set. The results now clearly illustrate the properties of the proposed Fuzzy-VGAPS. Results on *Data_10_2* and *Data_9_2* show that the fuzzy characteristics of the proposed method make the partitions more flexible and thus enable it to handle more overlapping clusters. Fuzzy-VGA and EAC-FCM, two recently developed fuzzy clustering techniques for automatically determining the number of clusters, succeed for only data sets having hyperspherical clusters. They cannot detect the proper number

of clusters from data sets having some symmetrical clusters other than hyperspheres. But Fuzzy-VGAPS succeeds in such situations. Hybrid clustering technique is also able to detect only hyperspherical shaped clusters but it fails for non-hyperspherical shaped clusters. SPACL clustering technique can detect any shaped clusters only when the clusters are well-separated and also when the dimensionally is low. Thus it fails for overlapping clusters. Moreover, both Hybrid and SPARCL clustering techniques cannot determine the number of clusters automatically from a data set. Based on these observations, and the fact that the property of symmetry is widely evident in real-life situations, application of Fuzzy-GAPS in most clustering tasks seems justified and is therefore recommended.

4.2. Complexity analysis of Fuzzy-VGAPS clustering technique

Below we have analyzed the time complexity of the proposed Fuzzy-VGAPS clustering technique:

- As discussed above Kd-tree data structure has been used in order to find the nearest neighbor of a particular point. The construction of Kd-tree requires $O(N \log N)$ time and $O(N)$ space [3].
- Initialization of GA needs $O(Popsize \times stringlength)$ time where *Popsize* and *stringlength* indicate the population size and the length of each chromosome in the GA, respectively. Note that *stringlength* is $O(M^* \times d)$ where d is the dimension of the data set and M^* is the soft estimate of the upper bound of the number of clusters.
- Fitness computation is composed of three steps:
  (1) In order to find membership values of each point to all cluster centers minimum symmetrical distance of that point with respect to all clusters have to be calculated. For this purpose the Kd-tree based nearest neighbor search is used. If the points are roughly uniformly distributed, then the expected case complexity is $O(c^d + \log N)$, where c is a constant depending on dimension and the point distribution. This is $O(\log N)$ if the dimension d is a constant [8]. Friedman et al. [24] also reported $O(\log N)$ expected time for finding the nearest neighbor. So in order to find minimal symmetrical distance of a particular point, $O(M^* \log N)$ time is needed. Thus total complexity of computing membership values of N points to M^* clusters is $O(M^* N \log N)$.
  (2) For updating the centers total complexity is $O(M^*)$.
  (3) Total complexity for computing the fitness values is $O(N \times M^*)$.
  So the fitness evaluation has total complexity = $O(Popsize \times M^* N \log N)$.
- Selection step of the GA requires $O(Popsize \times stringlength)$ time.
- Mutation and Crossover require $O(Popsize \times stringlength)$ time each.

Thus summing up the above complexities, and considering $stringlength \ll N$, total time complexity becomes $O(M^* N \log N \times Popsize)$ per generation. For maximum *Maxgen* number of generations total complexity becomes $O(M^* N \log N \times Popsize \times Maxgen)$.

Time complexity of Fuzzy-VGA [39] is $O(M^* N \times Popsize \times Maxgen)$. Time complexities of PSKM and PSFCM are $O(KN \times TotalIter)$ where *TotalIter* is the total number of iterations and K is the total number of clusters present in the data set. Similarly the time complexity of EAC-FCM [14] is $O(M^{*2} \times N \times Popsize \times Maxgen \times t)$, where t is the total number of iterations performed by FCM clustering technique.

4.3. Clustering gene expression data

In this section we demonstrate the effectiveness of the proposed clustering algorithm, Fuzzy-VGAPS to automatically cluster a real-life gene expression data set. For this purpose, the Eisen data set (gene expression data of yeast) [22] is considered. The percentage of missing values in the data set is 1.93% which have been estimated using the BPCA method [41]. The data set has been divided into six different subsets based on the type of experiments. Total number of genes in each data set is 2467 and number of experiments (i.e., number of columns) are 18, 14, 11, 14, 15 and 7, respectively. Comparison is made with another popular gene clustering technique, *information based clustering* (*Iclust*) [45]. The biological significance is verified by computing the z-score [25]. Higher values of z-score indicate better results. The z-score values of the partitionings obtained by Fuzzy-VGAPS for these six data sets are, respectively, 13.1, 16.6, 38.9, 25.2, 33.6 and 47.9, while those obtained by *Iclust* are 14.3, 13.4, 24.3, 15.4, 10.1 and 25.7, respectively. In order to quantify the goodness of the obtained partitionings by the two algorithms, Silhouette index [43] values are also calculated. Higher values of Silhouette index [43] indicate better partitioning. Fuzzy-VGAPS attains Silhouette values of −0.5181, 0.1740, 0.1923, 0.1395, 0.1626 and 0.1208, respectively. While *Iclust* attains Silhouette values of −0.1863, −0.1911, −0.2577, −0.1414, −0.4920 and −0.1735, respectively. Thus the values of z-scores and Silhouette indices show that except for the first set, in general Fuzzy-VGAPS performs much better than *Iclust* for clustering gene expression data.

4.4. Application to MR brain image segmentation

Fully automatic brain tissue classification from magnetic resonance images (MRI) is of great importance for research and clinical study of much neurological pathology. The accurate segmentation of MR images into different tissue classes, espe-

Table 4
Minkowski Scores (MS) obtained by Fuzzy C-means (FCM), Expectation Maximization (EM), Fuzzy-VGAPS and Fuzzy-VGA clustering algorithms on simulated MR volumes for brain with multiple sclerosis lesions projected on different z planes. Here #AC, #OC denotes, respectively, the actual number of clusters and the automatically obtained number of clusters (after application of Fuzzy-VGAPS and Fuzzy-VGA).

z plane no.	AC	FCM MS	EM MS	Fuzzy-VGA		Fuzzy-VGAPS	
				OC	MS	OC	MS
1	6	0.59	0.59	2	1.21	10	0.58
2	6	0.75	0.76	2	1.20	10	0.58
3	6	0.74	0.76	2	1.19	7	0.71
4	6	0.87	0.88	5	0.69	5	0.67
5	6	0.76	0.79	2	1.184	6	0.62
6	6	0.80	0.81	2	1.18	3	0.71
7	6	0.79	0.78	0.80	1.17	6	0.70
8	6	0.78	0.77	2	1.16	6	0.71
9	6	0.82	0.83	2	1.16	6	0.68
10	9	0.79	0.81	2	1.17	9	0.65

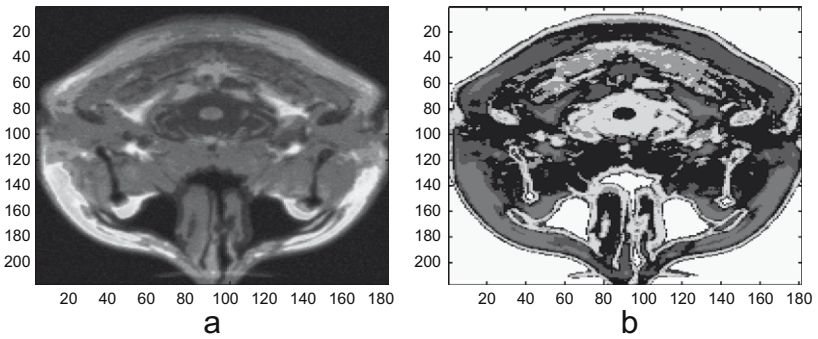

Fig. 7. (a) Original T1-weighted MR image of the brain with multiple sclerosis lesions in $z2$ plane. (b) Segmentation obtained by Fuzzy-VGAPS.

cially gray matter (GM), white matter (WM) and cerebrospinal fluid (CSF), is an important task. In this paper the problem of automatic partitioning MR brain images is posed as the clustering in the intensity space. The proposed Fuzzy-VGAPS clustering technique is applied to automatically segment MR brain images with multiple sclerosis lesions. The proposed algorithm is executed on some simulated MRI volumes for brain with multiple sclerosis lesions obtained from [2]. These images are available in three modalities and correspond to 1 mm slice thickness, 3% noise (calculated relative to the brightest tissue) and with 20% intensity non-uniformity. The image of size 217×181 is available in 181 different z planes. The images contain a total of 11 classes. However, the number of classes varies along the z planes. For this MR brain image, the ground truth information is available to us. In order to measure the segmentation solution quantitatively, we have also calculated *Minkowski Score* (MS) [7].

Fuzzy-VGAPS without adaptive crossover and mutation operators (crossover probability = 0.9, mutation probability = 0.01) is executed on the first 10 z planes of the MR brain images with multiple sclerosis lesions. The automatically determined number of partitions and the corresponding MS-scores are provided in Table 4. It is evident that for most of the planes Fuzzy-VGAPS is able to detect the near-optimal partitionings. Fig. 7a shows the original MS Lesion Brain image in T1 band projected on $z2$ plane just for an illustration. Fig. 7b shows the corresponding automatically segmented image obtained after application of Fuzzy-VGAPS clustering algorithm. The results are then compared with those obtained by Fuzzy-VGA [39]. The number of clusters and the corresponding MS-scores obtained by Fuzzy-VGA after applying it on the first 10 z planes of the MR brain images with multiple sclerosis lesions are also provided in Table 4. Results show that the proposed Fuzzy-VGAPS is more effective than Fuzzy-VGA for automatically segmenting the MR brain images. For the purpose of comparison, we have also executed two popular partitioning techniques Fuzzy C-means [10] and Expectation Maximization (EM) [30] on the above-mentioned brain data sets with K is set as the actual number of clusters present in that particular plane. The corresponding MS-scores are also reported in Table 4. Results show that the MS-score corresponding to the partitioning provided by Fuzzy-VGAPS clustering, in general, is the minimum among all the partitions. This implies the superior performance of Fuzzy-VGAPS for automatically detecting the proper partitioning from the MR brain images.

5. Discussion and conclusions

In this article a variable string length genetic point symmetry based fuzzy clustering technique, Fuzzy-VGAPS, is proposed. It utilizes a new symmetry based fuzzy cluster validity index, *FSym*-index. The characteristic features of the proposed

Fuzzy-VGAPS clustering technique which distinguishes it from the state-of-the-art approaches are as follows. Use of variable string length GA allows the encoding of a variable number of clusters. The *FSym*-index, used as the fitness function, provides the most appropriate partitioning even when the number of clusters, *K*, is varied. Moreover, clusters of any shape (e.g., hyperspherical, linear, ellipsoidal, ring shaped, etc.) and size (mixture of small and large clusters, i.e., clusters of unequal sizes) as long as they satisfy the property of point symmetry can be detected. Again use of GA enables the algorithm to come out of local optima, a typical problem associated with local search methods like *K*-means and FCM. The effectiveness of the proposed Fuzzy-VGAPS clustering technique as compared to seven different clustering techniques, Fuzzy-VGA clustering algorithm, EAC-FCM clustering algorithm, *K*-means clustering technique with point symmetry based distance (PSKM), Fuzzy C-means clustering technique with point symmetry based distance (PSFCM), genetic algorithm based fuzzy C-means clustering technique, GMFCM, a hybrid centroid–medoid clustering technique and a shape based clustering technique, SPARCL are shown for eight data sets. The stability of the clustering technique has also been analyzed by applying it on a particular data set with different initial seeds. For all the experimental data sets it has been found that the proposed technique is stable irrespective of the initial seeds used. Some real-life applications of Fuzzy-VGAPS to automatically cluster the gene expression data as well as segmenting the magnetic resonance brain image with multiple sclerosis lesions are also demonstrated. Note that Fuzzy-VGAPS is not applicable for data sets not containing point symmetric clusters (where the other algorithms will also fail). The point symmetry based distance considered in this paper requires the calculation of nearest neighbors for its computation. Thus for high-dimensional data sets, point symmetry based distance will take much time to compute. Therefore the proposed technique will not be suitable to handle large dimensional data sets efficiently.

Face recognition from images is an interesting future research topic where the properties of symmetry is expected to be beneficial.

Acknowledgements

The authors gratefully acknowledge the valuable comments of the Editor and the anonymous reviewers which helped them in improving the quality of the paper. The authors are also thankful to Vineet Chaoji, Department of Computer Science, Rensselaer Polytechnic Institute for sending the partitioning results of SPACL clustering technique.

References

[1] http://www.ics.uci.edu/~mlearn/MLRepository.html.
[2] BrainWeb: Simulated Brain Database. <http://www.bic.mni.mcgill.ca/brainweb>.
[3] M.R. Anderberg, Computational Geometry: Algorithms and Applications, Springer, 2000.
[4] T.W. Anderson, S.L. Scolve, Introduction to the Statistical Analysis of Data, Houghton Mifflin, 1978.
[5] S. Bandyopadhyay, S.K. Pal, Classification and Learning Using Genetic Algorithms: Applications in Bioinformatics and Web Intelligence (Natural Computing Series), Springer-Verlag New York Inc., Secaucus, NJ, USA, 2007.
[6] S. Bandyopadhyay, S. Saha, GAPS: a clustering method using a new point symmetry based distance measure, Pattern Recognition 40 (2007) 3430–3451.
[7] A. Ben-Hur, I. Guyon, Detecting Stable Clusters Using Principal Component Analysis in Methods in Molecular Biology, Humana Press, 2003.
[8] J.L. Bentley, B.W. Weide, A.C. Yao, Optimal expected-time algorithms for closest point problems, ACM Transactions on Mathematical Software 6 (4) (1980) 563–580.
[9] M. Bereta, T. Burczyński, Immune *K*-means and negative selection algorithms for data analysis, Information Sciences 179 (10) (2009) 1407–1425.
[10] J.C. Bezdek, Fuzzy Mathematics in Pattern Classification, Ph.D. Thesis, Cornell University, Ithaca, NY, 1973.
[11] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum, New York, 1981.
[12] C. Borgelt, Accelerating fuzzy clustering, Information Sciences (2008), doi:10.1016/j.ins.2008.09.017.
[13] R.J.G.B. Campello, Eduardo R. Hruschka, A fuzzy extension of the silhouette width criterion for cluster analysis, Fuzzy Sets and Systems 157 (2007) 2858–2875.
[14] R.J.G.B. Campello, Eduardo R. Hruschka, Vinícius S. Alves, On the efficiency of evolutionary fuzzy clustering, Journal of Heuristics 15 (1) (2009) 43–75.
[15] R.J.G.B. Campello, E.R. Hruschka, On comparing two sequences of numbers and its applications to clustering analysis, Information Sciences 179 (8) (2009) 1025–1039.
[16] V. Chaoji, M.A. Hasan, S. Salem, M.J. Zaki, SPARCL: efficient and effective shape-based clustering, in: Proceedings of the IEEE International Conference on Data Mining, 2008.
[17] H.C. Chou, M.C. Su, Eugene Lai, A new cluster validity measure and its application to image compression, Pattern Analysis and Applications 7 (2004) 205–220.
[18] D.L. Davies, D.W. Bouldin, A cluster separation measure, IEEE Transactions on Pattern Analysis and Machine Intelligence 1 (1979) 224–227.
[19] Y.-Y. Dong, Y.-J. Zhang, C.-L. Chang, Multistage random sampling genetic-algorithm-based fuzzy C-means clustering algorithm, in: Proceedings of the Third International Conference on Machine Learning and Cybernetics, Shanghai, 2004, pp. 2069–2073.
[20] R.C. Dubes, A.K. Jain, Clustering techniques: the user's dilemma, Pattern Recognition 8 (1976) 247–260.
[21] J.C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters, Journal of Cybernetics 3 (1973) 32–57.
[22] M.B. Eisen, P.T. Spellman, P.O. Brown, D. Botstein, Cluster analysis and display of genome-wide expression patterns, Proceedings of National Academy of Sciences 95 (1998) 14863–14868.
[23] B.S. Everitt, S. Landau, M. Leese, Cluster Analysis, Arnold, London, 2001.
[24] J.H. Friedman, J.L. Bently, R.A. Finkel, An algorithm for finding best matches in logarithmic expected time, ACM Transactions on Mathematical Software 3 (3) (1977) 209–226.
[25] F.D. Gibbons, F.P. Roth, Judging the quality of gene expression-based clustering methods using gene annotation, Genome Research 12 (2002) 1574–1581.
[26] N. Grira, M.E. Houle, Best of both: a hybridized centroid–medoid clustering heuristic, in: ICML'07: Proceedings of the 24th International Conference on Machine Learning, New York, NY, USA, ACM, 2007, pp. 313–320.
[27] L. Gröll, J. Jäkel, A new convergence proof of fuzzy C-means, IEEE Transactions Fuzzy Systems 13 (5) (2005) 717–720.
[28] C.-C. Hsu, C.-L. Chen, Y.-W. Su, Hierarchical clustering of mixed data based on distance hierarchy, Information Sciences 177 (20) (2007) 4474–4492.

[29] A.K. Jain, P.W. Duin, M. Jianchang, Statistical pattern recognition: a review, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (1) (2000) 4–37.
[30] A.K. Jain, M.N. Murthy, P.J. Flynn, Data clustering: a review, ACM Computing Reviews (1999).
[31] D.-W. Kim, K.H. Lee, D. Lee, Fuzzy cluster validation index based on inter-cluster proximity, Pattern Recognition Letters 24 (15) (2003) 2561–2574.
[32] M. Kim, R.S. Ramakrishna, New indices for cluster validity assessment, Pattern Recognition Letters 26 (15) (2005) 2353–2363.
[33] Y.-I. Kim, D.-W. Kim, D. Lee, K.H. Lee, A cluster validation index for GK cluster analysis based on relative degree of sharing, Information Sciences 168 (1–4) (2004) 225–242.
[34] C.-H. Lee, O.R. Zaöane, H.-H. Park, J. Huang, R. Greiner, Clustering high dimensional data: a graph-based relaxed optimization approach, Information Sciences 178 (23) (2008) 4501–4511.
[35] Y. Liu, Z. Yi, H. Wu, M. Ye, K. Chen, A tabu search approach for the minimum sum-of-squares clustering problem, Information Sciences 178 (12) (2008) 2680–2704.
[36] Y. Liu, Z. Yi, H. Wu, M. Ye, K. Chen, A tabu search approach for the minimum sum-of-squares clustering problem, Information Sciences 178 (12) (2008) 2680–2704.
[37] B. Long, Z. (Mark) Zhang, P.S. Yu, Combining multiple clusterings by soft correspondence, in: ICDM'05: Proceedings of the Fifth IEEE International Conference on Data Mining, Washington, DC, USA, IEEE Computer Society, 2005, pp. 282–289.
[38] U. Maulik, S. Bandyopadhyay, Performance evaluation of some clustering algorithms and validity indices, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (12) (2002) 1650–1654.
[39] U. Maulik, S. Bandyopadhyay, Fuzzy partitioning using a real-coded variable-length genetic algorithm for pixel classification, IEEE Transactions Geoscience and Remote Sensing 41 (5) (2003) 1075–1081.
[40] C.D. Nguyen, K.J. Cios, GAKREM: a novel hybrid clustering algorithm, Information Sciences 178 (22) (2008) 4205–4227.
[41] S. Oba, M. Sato, I. Takemasa, M. Monden, K. Matsubara, S. Ishii, A bayesian missing value estimation method for gene expression profile data, Bioinformatics 19 (2003) 2088–2096.
[42] M.K. Pakhira, U. Maulik, S. Bandyopadhyay, Validity index for crisp and fuzzy clusters, Pattern Recognition 37 (3) (2004) 487–501.
[43] P. Rousseeuw, Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, Journal of Computational and Applied Mathematics 20 (1987) 5365.
[44] W. Sheng, S. Swift, L. Zhang, X. Liu, A weighted sum validity function for clustering with a hybrid niching genetic algorithm, IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics 35 (6) (2005) 1156–1167.
[45] N. Slonim, G.S. Atwal, G. Tkačik, W. Bialek, Information-based clustering, Proceedings of the National Academy of Sciences of the United States of America 102 (51) (2005) 18297–18302.
[46] M. Srinivas, L.M. Patnaik, Adaptive probabilities of crossover and mutation in genetic algorithms, IEEE Transactions on Systems, Man and Cybernetics 24 (4) (1994) 656–667.
[47] M.C. Su, C.-H. Chou, A modified version of the *K*-means algorithm with a distance based on cluster symmetry, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (6) (2001) 674–680.
[48] Y.-J. Wang, H.-S. Lee, A clustering method to identify representative financial ratios, Information Sciences 178 (4) (2008) 1087–1097.
[49] X.L. Xie, G. Beni, A validity measure for fuzzy clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (1991) 841–847.
[50] Z. Xu, J. Chen, J. Wu, Clustering algorithm for intuitionistic fuzzy sets, Information Sciences 178 (19) (2008) 3775–3790.