Combining Multiple Clusterings Using Evidence Accumulation

Ana L.N. Fred[*] and Anil K. Jain[+]

[*] Instituto Superior Técnico / Instituto de Telecomunicações

Av. Rovisco Pais, 1049-001 Lisboa, Portugal

email: afred@lx.it.pt

[+] Dept. of Computer Science and Engineering

Michigan State University, USA

email: jain@cse.msu.edu

Abstract

We explore the idea of evidence accumulation (EAC) for combining the results of multiple clusterings. First, a clustering ensemble - a set of object partitions, is produced. Given a data set (n objects or patterns in d dimensions), different ways of producing data partitions are: (1)- applying different clustering algorithms, and (2)- applying the same clustering algorithm with different values of parameters or initializations. Further, combinations of different data representations (feature spaces) and clustering algorithms can also provide a multitude of significantly different data partitionings. We propose a simple framework for extracting a consistent clustering, given the various partitions in a clustering ensemble. According to the EAC concept, each partition is viewed as an independent evidence of data organization, individual data partitions being combined, based on a voting mechanism, to generate a new $n \times n$ similarity matrix between the n patterns. The final data partition of the n patterns is obtained by applying a hierarchical agglomerative clustering algorithm on this matrix. We have developed a theoretical framework for the analysis of the proposed clustering combination strategy and its evaluation, based on the concept of mutual information between data partitions. Stability of the results is evaluated using bootstrapping techniques. A detailed discussion of an evidence accumulation-based clustering algorithm, using a split and merge strategy based on the K-means clustering algorithm, is presented. Experimental results of the proposed method on several synthetic and real data sets are compared with other combination strategies, and with individual clustering results produced by well known clustering algorithms.

Keywords

Cluster analysis, combining clustering partitions, cluster fusion, evidence accumulation, robust clustering, K-means algorithm, single-link method, cluster validity, mutual information.

I. Introduction

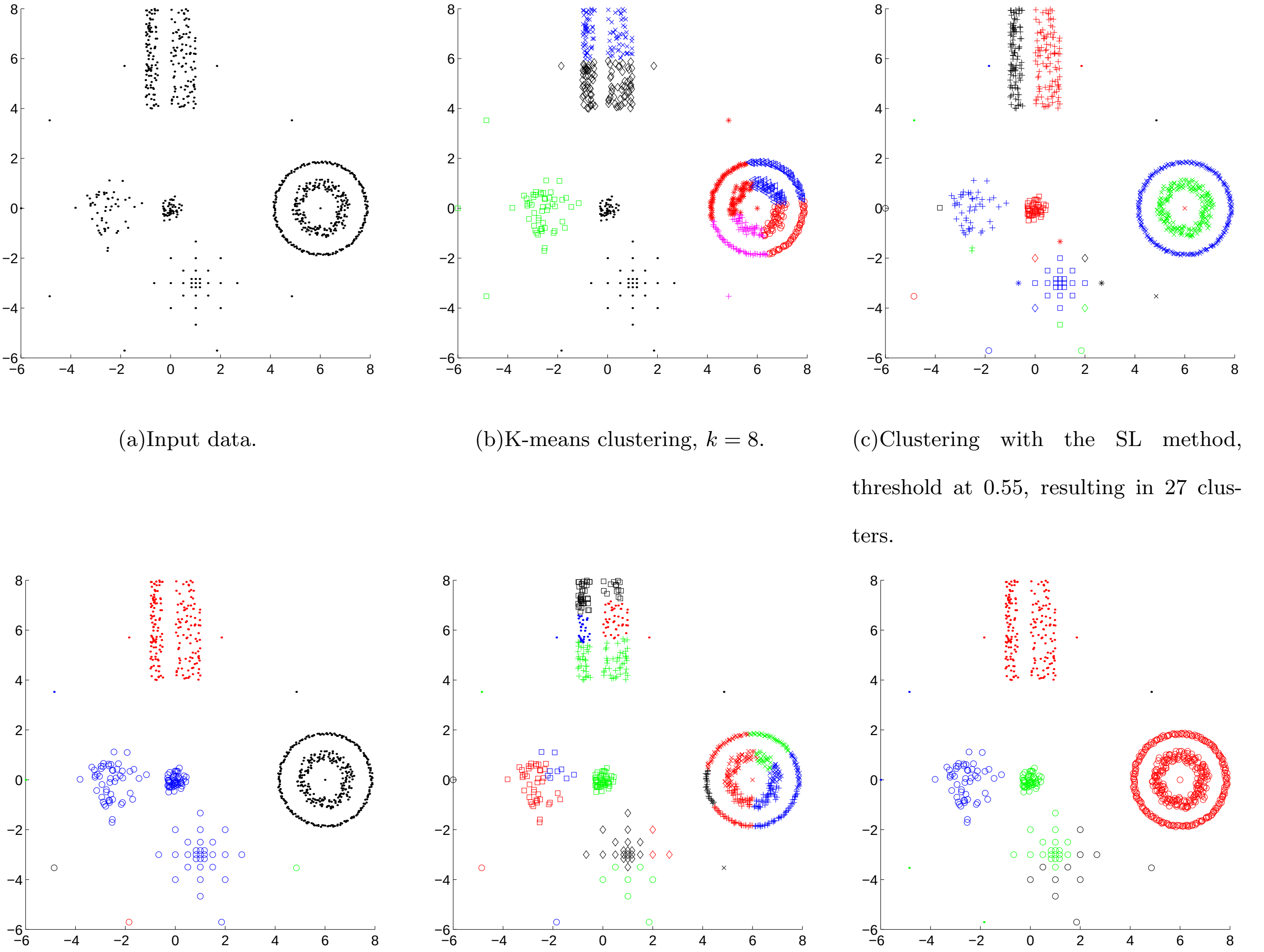
Data clustering or unsupervised learning is an important but an extremely difficult problem. The objective of clustering is to partition a set of unlabelled objects into homogeneous groups or clusters. A number of application areas use clustering techniques for organizing or discovering structure in data, such as data mining [1], [2], information retrieval [3], [4], [5], image segmentation [6], and machine learning. In real world problems, clusters can appear with different shapes, sizes, data sparseness, and degree of separation. Further, noise in the data can mask the true underlying structure present in the data. Clustering techniques require the definition of a similarity measure between patterns, which is not easy to specify in the absence of any prior knowledge about cluster shapes. Additionally, quantitative evaluation of the quality of clustering results is difficult due to the subjective notion of clustering.

A large number of clustering algorithms exist [7], [8], [9], [10], [11], yet no single algorithm is able to identify all sorts of cluster shapes and structures that are encountered in practice. Each algorithm has its own approach for estimating the number of clusters [12], [13], imposing a structure on the data [14], [15], [16], and validating the resulting clusters [17], [18], [19], [20], [21], [22]. Model-based techniques assume particular cluster shapes that can be given a simple and compact description. Examples of model-based techniques include: parametric density approaches, such as mixture decomposition techniques [23], [24], [25], [26]; prototype-based methods, such as central clustering [14], square-error clustering [27], K-means [28], [8] or K-medoids clustering [9]; and shape fitting approaches [15], [6], [16]. Model order selection is sometimes left as a design parameter to be specified by the user, or it is incorporated in the clustering procedure [29], [30], [25]. Most of the above techniques utilize an optimization procedure tuned to a particular cluster shape, or emphasize cluster compactness. Fisher et al. [31] proposed an optimization-based clustering algorithm, based on a pairwise clustering cost function, emphasizing cluster connectedness. Non-parametric density based clustering methods attempt to identify high density clusters separated by low density regions [5] [32], [33]. Graph-theoretical approaches [34] have mostly been explored in hierarchical methods, that can be represented graphically as a tree or dendrogram [7], [8]. Both agglomerative [28], [35] and divisive approaches [36] (such as those based on the minimum spanning tree - MST [28]) have been proposed; different algorithms are obtained depending on the definition of similarity measures between patterns and between clusters [37]. The single-link (SL) and the complete-link (CL) hierarchical methods [7], [8] are the best known techniques in this class, emphasizing, respectively, connectedness and compactness of patterns in a cluster. Prototype-based hierarchical methods, which define similarity between clusters based on cluster representatives, such as the centroid, emphasize compactness. Variations of the prototype-based hierarchical clustering include the use of multiple prototypes per cluster, as in the CURE algorithm [38]. Other hierarchical agglomerative clustering algorithms follow a split and merge technique, the data being initially split into a large number of small clusters, merging being based on inter-cluster similarity; a final partition is selected among the clustering hierarchy by thresholding techniques or based on measures of cluster validity [39], [5], [40], [41], [42], [43]. Treating the clustering problem as a graph partitioning problem, a recent approach, known as *spectral clustering*, applies spectral graph theory for clustering [44], [45], [46].

Among the various clustering methods, the K-means algorithm, which minimizes the squared-error criteria, is one of the simplest clustering algorithm. It is computationally efficient and does not require the user to specify many parameters. Its major limitation, however, is the inability to identify clusters with arbitrary shapes, ultimately imposing hyper-spherical shaped clusters on the data. Extensions of the basic K-means algorithm include: use of Mahalanobis distance to identify hyper-ellipsoidal clusters [28]; introducing fuzzy set theory to obtain non-exclusive partitions [20]; and adaptations to straight line fitting [47].

While hundreds of clustering algorithms exist, it is difficult to find a single clustering algorithm that can handle all types of cluster shapes and sizes, or even decide which algorithm would be the best one for a particular data set [48], [49]. Figure 1 illustrates how different algorithms, or even the same algorithm with different parameters, produce very distinct results. Considering that clustering is an important tool for data mining and exploratory data analysis, it is wise to apply several different clustering algorithms to the given data and then determine the best algorithm for the data.

Inspired by the work in sensor fusion and classifier combination [50], [51], [52], a clustering combination approach has been proposed [53], [54], [55]. Fred and Jain introduce the concept of evidence accumulation clustering, that maps the individual data partitions in a clustering ensemble into a new similarity measure between patterns, summarizing inter-pattern structure perceived from these clusterings. The final data partition is obtained by applying the single-link method to this new similarity



(a)Input data.

(b)K-means clustering, $k = 8$.

(c)Clustering with the SL method, threshold at 0.55, resulting in 27 clusters.

(d)Clustering with the SL method, forcing 8 clusters.

(e)Clustering with the CL method, threshold at 2.6, resulting in 22 clusters.

(f)Clustering with the CL method, forcing 8 clusters.

Fig. 1. Results of clusterings using different algorithms (K-means, single-link – SL, and complete-link – CL) with different parameters. Each cluster identified is shown in a different color/pattern.

matrix. The results of this method show that, the combination of "weak"clustering algorithms such as the K-means, which impose a simple structure on the data, can lead to the identification of true underlying clusters with arbitrary shapes, sizes and densities. Strehl and Ghosh [56] explore the concept of consensus between data partitions and propose three different combination mechanisms. The first step of the consensus functions is to transform the data partitions into a hyper-graph representation. The hyper-graph-partitioning algorithm (HGPA) obtains the combined partition by partitioning the hyper-graph into k unconnected components of approximately the same size, by cutting a minimum number of hyper-edges. The meta-clustering algorithm (MCLA) applies a graph-based clustering to hyper-edges in the hyper-graph representation. Finally, CSPA uses a pair-wise similarity, as defined by Fred and Jain [55], and the final data partition is obtained by applying the METIS algorithm of Karypis and Kumar to the induced similarity measure between patterns.

In this paper we further explore the concept of evidence accumulation clustering (EAC). A formal definition of the problem of combining data partitions is given in section II. Assuming no restrictions on the number of clusters in the data partitions to be combined, or on how these data partitions are produced, we introduce the EAC framework in section III. The proposed voting mechanism for clustering combination performs a mapping of the data partitions into a new similarity matrix between patterns, to which any clustering algorithm can be applied. We explore the single-link (SL) and the average-link (AL) methods for this purpose. An improvement of the well-known SL method, by exploring nearest neighbor relationships, leads to a more efficient algorithm than the one introduced in [53], [54], [55], and extends the range of applicability of the proposed technique to larger data sets. Section IV addresses performance evaluation issues. We use the concept of mutual information for measuring the consistency between data partitions (section IV-A). This leads to a theoretical framework for the analysis of clustering combination results, and an optimality criteria (section IV-B) that focuses on consistency and robustness properties. Stability of clustering combination solutions is evaluated based on perturbation/variance analysis, using a bootstrap technique. An evidence accumulation-based algorithm based on the K-means algorithm, is presented and discussed in section V. Experimental results (section VI) on both synthetic and real data, illustrate the versatility and robustness of the proposed methods, as compared to individual clusterings produced by well known clustering algorithms, and compared to other ensemble combination methods.

II. Problem Formulation

Let $X = \{\mathrm{x_1}, \mathrm{x_2}, \ldots, \mathrm{x_n}\}$ be a set of n objects, and let $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ be the representation of these patterns; x_i may be defined, for instance, over some $d-$dimensional feature space, $x_i \in \mathcal{R}^d$, such as when adopting vector representations, or $x_i = x_{i1}x_{i2}\ldots x_{im_i}$ may be a string, m_i being the string length, when using string descriptions. A clustering algorithm takes $\mathcal{X}$ as input and organizes the n patterns into k clusters, according to some similarity measure between patterns, forming a data partition P. Different clustering algorithms will, in general, produce different partitions for the same data set, either in terms of cluster membership and/or the number of clusters produced. Different clustering results can also be produced by the same clustering algorithm by using different algorithmic parameter values or different initializations, or by exploring different pattern representations or feature spaces.

Consider N partitions of the data X, and let $\mathbb{P}$ represent the set of N partitions, which we define as a *clustering ensemble*:

$$\mathbb{P} = \{P^1, P^2, \ldots, P^N\} \tag{1}$$

$$\begin{aligned} P^1 &= \left\{C_1^1, C_2^1, \ldots, C_{k_1}^1\right\} \\ &\vdots \\ P^N &= \left\{C_1^N, C_2^N, \ldots, C_{k_N}^N\right\} \end{aligned} \tag{2}$$

where C_j^i is the jth cluster in data partition P^i, which has k_i clusters, and n_j^i is the cardinality of C_j^i, with $\sum_{j=1}^{k_i} n_j^i = n, \quad i = 1, \ldots, N$.

The problem is to find an "optimal"data partition, P^*, using the information available in N different data partitions in $\mathbb{P} = \{P^1, P^2, \ldots, P^N\}$. We define k^* as the number of clusters in P^*. Ideally, P^* should satisfy the following properties:

(a) Consistency with the clustering ensemble $\mathbb{P}$;

(b) Robustness to small variations in $\mathbb{P}$;

(c) Goodness of fit with ground truth information (true cluster labels of patterns), if available.

The first property focuses on the agreement of the combined data partition, P^*, with the individual partitions, $P^i, i = 1, \ldots, N$. By robustness we assume that the number of clusters and the cluster membership in P^* are essentially invariant to small perturbations in $\mathbb{P}$. The last requirement assumes the knowledge of the ground truth, which, considering the unsupervised nature of the clustering process, seems to be a contradiction. However, true cluster labels of patterns, if available, are used only as an additional validation tool for the proposed methods.

III. Evidence Accumulation Clustering

In order to address the cluster ensemble combination problem, we propose the concept of evidence accumulation clustering. We make *no assumptions* on the number of clusters, k_i, in each data partition, P^i, and on the number of clusters k^* in the combined data partition, P^*. It is expected that the combined data partition, P^*, will better explain natural groupings of the data compared to the individual clustering results, P^i.

The idea of evidence accumulation clustering is to combine the results of multiple clusterings into a single data partition, by viewing each clustering result as an independent evidence of data organization. This requires us to address the following three issues: (1) how to collect evidence or generate the clustering ensemble?, (2) how to combine the evidence?, and (3) how to extract a consistent data partitioning from the combined evidence?

A. Producing Clustering Ensembles

Clustering ensembles can be generated by following two approaches: (1) choice of data representation, and (2) choice of clustering algorithms or algorithmic parameters. In the first approach, different partitions of the objects under analysis may be produced by: (a) employing different pre-processing and/or feature extraction mechanisms, which ultimately lead to different pattern representations (vectors, strings, graphs, etc.) or different feature spaces; (b) exploring sub-spaces of the same data representation, such as using sub-sets of features; (c) perturbing the data, such as in bootstrapping techniques (like bagging), or sampling approaches, as, for instance, using a set of prototype samples to represent huge data sets. In the second approach, we can generate clustering ensembles by: (i) applying different clustering algorithms; (ii) using the same clustering algorithm with different parameters or initializations; (iii) exploring different dissimilarity measures for evaluating inter-pattern relationships, within a given clustering algorithm.

A combination of these two main mechanisms for producing clustering ensembles leads to exploration of distinct views of inter-pattern relationships. From a computational perspective, clustering results produced in an "independent way"facilitate efficient data analysis by utilizing distributed computing, and reuse of results obtained previously.

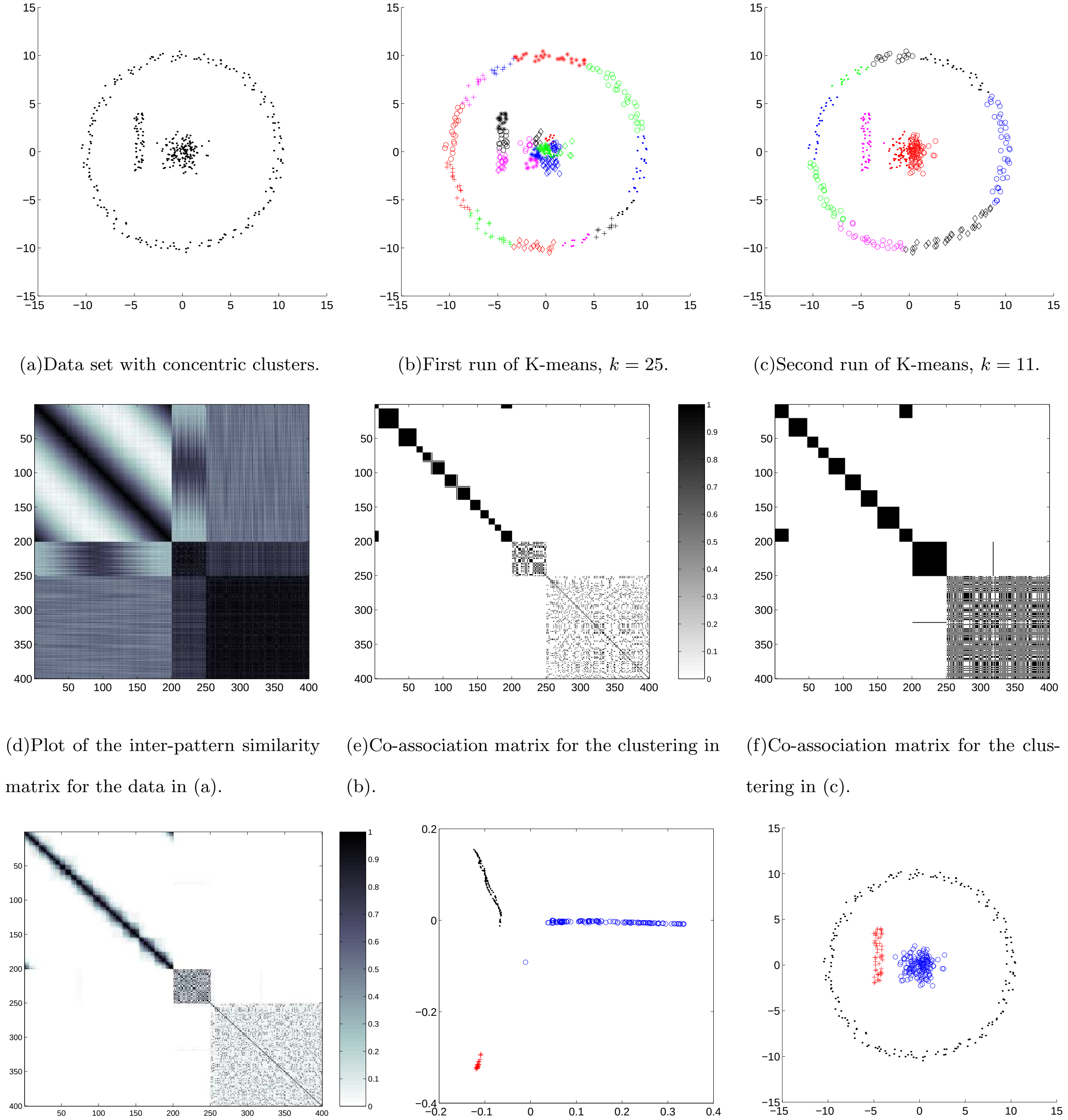
B. Combining Evidence: The Co-Association Matrix

In order to cope with *partitions with different numbers of clusters*, we propose a voting mechanism to combine the clustering results, leading to a new measure of similarity between patterns. The underlying assumption is that patterns belonging to a "natural"cluster are very likely to be co-located in the same cluster in different data partitions. Taking the co-occurrences of pairs of patterns in the same cluster as votes for their association, the N data partitions of n patterns are mapped into a $n \times n$ co-association matrix:

$$\mathcal{C}(i,j) = \frac{n_{ij}}{N},$$

where n_{ij} is the number of times the pattern pair (i,j) is assigned to the same cluster among the N partitions.

The process of evidence accumulation is illustrated using the three-cluster data set containing 400 patterns in figure 2(a): outer ring (200 patterns); rectangular shaped cluster (50 patterns); 2-D gaussian cluster (150 patterns). A clustering ensemble with 30 partitions ($N = 30$) was produced by running the K-means algorithm with random initialization with k randomly chosen in the interval [10, 30].



(a)Data set with concentric clusters.

(b)First run of K-means, $k = 25$.

(c)Second run of K-means, $k = 11$.

(d)Plot of the inter-pattern similarity matrix for the data in (a).

(e)Co-association matrix for the clustering in (b).

(f)Co-association matrix for the clustering in (c).

(g)Co-association matrix based on the combination of 30 clusterings.

(h)2-D multi-dimensional scaling of the co-association matrix in (g).

(i)Evidence accumulation data partition.

Fig. 2. Individual clusterings and combination results on concentric clusters using the K-means algorithm.

Figures 2(b) and 2(c) show results of two runs of the K-means algorithm; the corresponding matrices of associations are given in figures 2(e) and 2(f), respectively; result of the combination, by plotting the final co-association matrix, is shown in fig. 2(g). For comparison, the plot of the similarity between the original patterns, based on the Euclidean distance, is represented in figure 2(d): $sim(x_i, x_j) = \max_{x_k, x_l} \{d_E(x_k, x_l)\} - d_E(x_i, x_j)$, with d_E denoting the Euclidean distance. Matrix coordinates in figures 2(d) to 2(g) indicate pattern indices: 1-200 - outer ring; 201-250 - bar-shaped cluster; 251-400 - gaussian cluster. In figures 2(d) to 2(g), colors are in a gradation from white (zero similarity) to dark black (highest similarity), as shown on the color bars. In plots 2(e) and 2(f) the white spots within the lower right square are associated with the splitting of the gaussian cluster into several small clusters. It can be seen that, although individual data partitions are quite different, neighboring patterns fall in the same cluster in most of the partitions. As a result, the true structure of the clusters becomes more evident in the co-association matrix: notice the more clear separation between the clusters (large white zones) in figure 2(g) as compared to the original similarity matrix in figure 2(d). It is interesting to analyze the representation of the ring-shaped cluster in figure 2(g): the black spots connect neighboring patterns; the large white zone corresponds to the similarity between non-neighboring patterns. By moving along the ring, one can always find a path of highly similar patterns.

The evidence accumulation mechanism thus maps the partitions in the clustering ensemble into a new similarity measure between patterns (summarized in the co-association matrix $\mathcal{C}$), intrinsically performing a non-linear transformation of the original feature space into a new representation. Figure 2(h) illustrates this transformation by showing the 2-dimensional plot of the result of applying multi-dimensional scaling over the matrix $\mathcal{C}$ in figure 2(g). Black dots correspond to the outer ring in figure 2(a), blue circles correspond to the inner gaussian data, and red crosses are the transformation of the rectangular shaped cluster.

C. Recovering Natural Clusters

The core of the evidence accumulation clustering technique is the mapping of partitions into the co-association matrix, $\mathcal{C}$. This corresponds to a non-linear transformation of the original feature space into a new representation, summarized in the similarity matrix, $\mathcal{C}$, induced by inter-pattern relationships present in the clustering ensemble. We can now apply any clustering algorithm over this new similarity matrix in order to find a consistent data partition. We herein emphasize neighborhood relationship and apply the single link and the average-link methods to the matrix $\mathcal{C}$; the decision on the number of clusters is based on cluster lifetime, as illustrated in figure 3. We define $k-$cluster lifetime as the range of threshold values on the dendrogram that lead to the identification of k clusters. Lifetimes of $2-$, $3-$, and $4-$cluster partitions are represented in figure 3 as l_2, l_3 and l_4, respectively. For instance, the lifetime of the $3-$cluster solution, $l_3 = 0.3600$, is computed as the difference between the minimum (0.4) and the maximum (0.76) threshold values that lead to the separation of patterns into three clusters. The 1-cluster lifetime is a special case of the above, defined as the difference between the minimum threshold value that leads to the 1-cluster solution (0.94) and the maximum distance value (1.0).

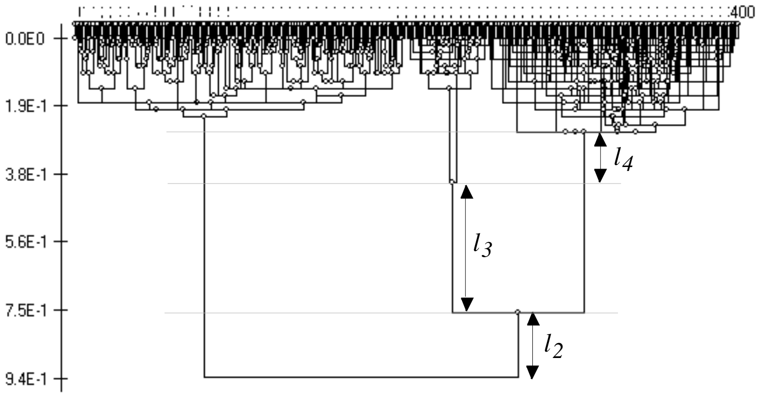


Fig. 3. Dendrogram produced by the SL method using the similarity matrix in figure 2(g). Distances ($1 - \textit{similarity}$) are represented along the graph ordinate. From the dendrogram, the following cluster lifetimes are identified: 2-clusters: $l_2 = 0.18$; 3-clusters: $l_3 = 0.36$; 4-clusters: $l_4 = 0.14$; 5-clusters: 0.02. The 3-cluster partition (shown in fig. 2(i)), corresponding to the longest lifetime, is chosen (threshold on the dendrogram is between 0.4 and 0.76).

D. Outline of the Evidence Accumulation Clustering Algorithm

A well known difficulty of the single-link method is its quadratic space and time complexities, related to the processing of a $n \times n$ proximity matrix for large n. To circumvent this, we pre-compute a $n \times p$ matrix which stores the indices of the p nearest neighbors for each of the n patterns. The SL algorithm is now applied to the corresponding $n \times p$ similarity matrix. The nearest neighbor matrix can be computed [57] as a pre-processing step. In the rest of the paper the value $p = 20$ will be used, a value high enough to ensure correct results with the SL method. The proposed evidence accumulation method, when using the SL method for obtaining the final data partition, is summarized in table I. When using the AL method, the $n \times n$ proximity matrix is used instead, and the AL is used in step 2 to compute the dendrogram.

TABLE I

Data clustering using Evidence Accumulation (using SL).

Input: n - number of patterns $n \times p$ nearest neighbor matrix p - nearest neighbor index

N - number of clusterings $\mathbb{P} = \left\{P^1, \ldots P^N\right\}$ - clustering ensemble

Output: P^* - Combined data partition.

Initialization: Set the $n \times p$ co-association matrix, $\mathcal{C}(.,.)$, to a null matrix.

1. For each data partition $P^l \in \mathbb{P}$ do:

1.1. Update the co-association matrix: for each pattern pair (i, j) in the pth neighbor list, that belongs to the same cluster in P^l, set $\mathcal{C}(i,j) = \mathcal{C}(i,j) + \frac{1}{N}$.

2. Compute the SL dendrogram of $\mathcal{C}$; the final partition, P^*, is chosen as the one with the highest lifetime.

IV. Figures of Merit for the Evaluation of Clustering Combination Results

According to the problem formulation in section II, the quality of combination results can be evaluated in terms of consistency with the clustering ensemble $\mathbb{P}$, robustness to small variations in $\mathbb{P}$, and, whenever possible, goodness of fit with the ground truth information. These requirements assume a measure of similarity or agreement between the data partitions. We follow an information theoretic approach, exploring the concept of *mutual information* to define the similarity between data partitions. Next, objective functions, optimality criteria, and associated figures of merit are proposed and used to evaluate the performance of combination methods.

A. Measuring the Consistency of Data Partitions Using Mutual Information

A partition P^a describes a labelling of the n patterns in the data set X, into k_a clusters. Taking frequency counts as approximations for probabilities, the entropy [58] of the data partition P^a is expressed by $H(P^a) = -\sum_{i=1}^{k_a} \frac{n_i^a}{n} \log\left(\frac{n_i^a}{n}\right)$, where n_i^a represents the number of patterns in cluster $C_i^a \in P^a$. The agreement between two partitions P^a and P^b is measured by the mutual information $I(P^a, P^b)$, as proposed by Strehl and Ghosh [56]

$$I(P^a, P^b) = \sum_{i=1}^{k_a} \sum_{j=1}^{k_b} \frac{n_{ij}^{ab}}{n} \log\left(\frac{\frac{n_{ij}^{ab}}{n}}{\frac{n_i^a}{n} \cdot \frac{n_j^b}{n}}\right), \tag{3}$$

with n_{ij}^{ab} denoting the number of shared patterns between clusters C_i^a and C_j^b, $C_i^a \in P^a$ and $C_j^b \in P^b$. From the definition of mutual information [58], it is easy to demonstrate that $I(P^a, P^b) \leq \left(H(P^a) + H(P^b)\right)/2$. We define the *normalized mutual information* (NMI) between two partitions P^a and P^b as $NMI(P^a, P^b) = \frac{2 \cdot I(P^a, P^b)}{H(P^a) + H(P^b)}$, which, after simplification, leads to

$$NMI(P^a, P^b) = \frac{-2 \sum_{i=1}^{k_a} \sum_{j=1}^{k_b} n_{ij}^{ab} \log\left(\frac{n_{ij}^{ab} \cdot n}{n_i^a \cdot n_j^b}\right)}{\sum_{i=1}^{k_a} n_i^a \log\left(\frac{n_i^a}{n}\right) + \sum_{j=1}^{k_b} n_j^b \log\left(\frac{n_j^b}{n}\right)}. \tag{4}$$

Note that $0 \leq NMI(.,.) \leq 1$. Equation (4) differs from the mutual-information based similarity between partitions proposed in [56] in terms of the normalizing term; the entropy terms $H(P^a)$ and $H(P^b)$ have been replaced in [56] by the upper bounds $\log(k_a)$ and $\log(k_b)$, respectively.

We define $NMI(P, \mathbb{P})$ as the *average normalized mutual information* between P, an arbitrary data partition, and the elements in the clustering ensemble, $\mathbb{P}$:

$$NMI(P, \mathbb{P}) = \frac{1}{N} \sum_{i=1}^{N} NMI(P, P^i). \tag{5}$$

We further define the *average agreement between partitions* in a clustering ensemble $\mathbb{P}$ by

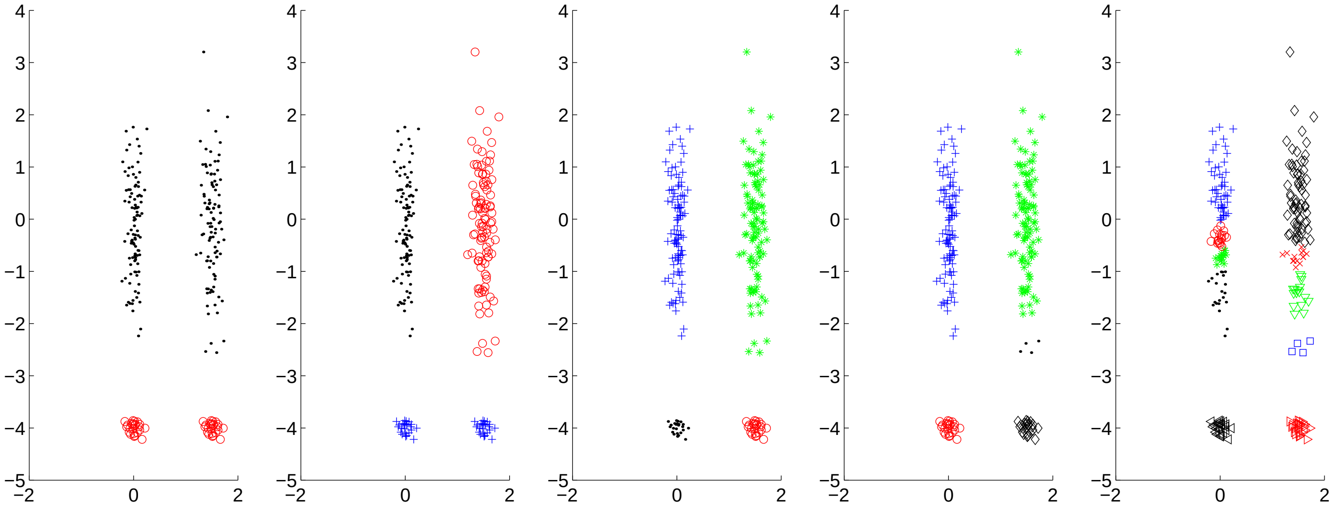
$$NMI(\mathbb{P},\mathbb{P}) = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} NMI(P^i, P^j) / \binom{N}{2}. \tag{6}$$

B. Objective Functions and Optimality Criteria

Let $\check{\mathbb{P}}^{\mathbf{k}} = \left\{ \check{P}^{1^{\mathbf{k}}}, \ldots, \check{P}^{m^{\mathbf{k}}} \right\}$, $m = \frac{1}{\mathbf{k}!} \sum_{l=1}^{\mathbf{k}} \binom{\mathbf{k}}{l} (-1)^{\mathbf{k}-l} l^n$, represent the set of all possible partitions of the n patterns in X into k clusters. We define *k-cluster consensus partition*, $P^{*^{\mathbf{k}}}$, $P^{*^{\mathbf{k}}} \in \check{\mathbb{P}}^{\mathbf{k}}$, as the k-cluster partition that best fits the clustering ensemble $\mathbb{P}$ in the sense of maximizing the objective function $NMI(\check{P}^{\mathbf{k}}, \mathbb{P})$:

$$P^{*^{\mathbf{k}}} = \arg\max_i \left\{ NMI(\check{P}^{i^{\mathbf{k}}}, \mathbb{P}) \right\}. \tag{7}$$

For each value of $\mathbf{k}$, the criterion in equation (7) satisfies the property (a) in section II. The data partition, $P^{*^{\mathbf{k}}}$, can be seen as a $\mathbf{k}$-cluster prototype representation of the set of partitions, $\mathbb{P}$.



(a)P^1: Nc=2. (b)P^2: Nc=3. (c)P^3: Nc=4. (d)P^4: Nc=5. (e)P^5: Nc=10.

Fig. 4. Five different partitions of the “cigar”data set: Nc indicates the number of clusters in the partition.

The average normalized mutual information function tends to assign higher values of similarity to partitions with equal or similar number of clusters, when compared to partitions with different number of clusters. This is illustrated in figures 4 and 5, concerning a data set formed by 4 gaussian clusters, which we refer to as “cigar”data. We generated three different clustering ensembles, by applying the

K-means algorithm with random initialization, and $N = 50$. The three different clustering ensembles $\mathbb{P}^k$ correspond to three different values of k, the number of clusters specified for the K-means algorithm: (i) $k = 10$, (ii) $k = 20$, and (iii) random selection of k within the interval $[4, 20]$ ($\mathbb{P}^{4-20}$). We produced 8 plausible reference partitions, $P^i, i = 1, \ldots 8$, with the following numbers of clusters: Nc = 2, 3, 4, 5, 10, 15, 20, and 50. For Nc from 2 to 5, these partitions correspond to the two dense clusters relatively well separated from the remaining clusters; for higher values of Nc, each of the two dense gaussian clusters forms a cluster, and each of the sparse gaussians is split into several clusters using the K-means algorithm. Figure 4 presents the first 5 of these partitions. As shown in figure 5, $NMI(P^i, \mathbb{P}^k)$ tends to take its maximum value for situations where Nc (the number of clusters in P^i) is similar to k (blue and red curves), having a steep increase as Nc approaches k, and smoothly decreasing with higher Nc values. The black curve, corresponding to $NMI(P^i, \mathbb{P}^{4-20})$, is similar to the curve for $NMI(P^i, \mathbb{P}^{10})$; this suggests that the average normalized mutual information values for clustering ensembles with uniform compositions of the number of clusters, approaches the corresponding similarity measure of a clustering ensemble with a fixed number of clusters, given by the average number of clusters in the clustering ensemble.

It is apparent from figure 5 that the absolute value of $NMI(P^i, \mathbb{P})$ cannot be used for identifying the correct number of clusters underlying a clustering ensemble, as it is biased towards the average number of clusters in $\mathbb{P}$. Therefore, the objective function in equation (7) can only be used for finding a consensus clustering, as proposed in [56], under the assumption that the number of clusters in the combined partition, k^*, is known. The optimal partition, summarizing the overall inter-pattern relationships and that is consistent with the clustering ensemble $\mathbb{P}$, should be obtained by searching amongst the set of $P^{*^{\mathbf{k}}}$ solutions of equation (7), for various values of $\mathbf{k}$. The appropriate value for k^* is identified based on the robustness property, mentioned earlier.

In order to address the robustness property (b) in section II, we perturb the clustering ensemble $\mathbb{P}$, using a bootstrap technique, and compute the variance of the resulting NMI values. Let $\mathbb{P}^B =$

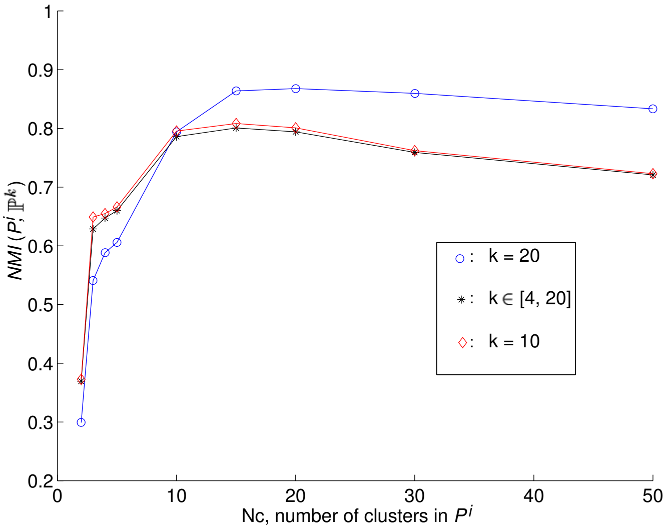


Fig. 5. Plot of $NMI(P^i, \mathbb{P}^k)$ for the partitions, P^i, with the number of clusters in P^i ranging from 2 to 50. $\mathbb{P}^k$ refers to a clustering ensemble formed by 50 data partitions produced by the K-means algorithm, for fixed k ($k = 10$ - red curve; $k = 20$ - blue curve), and for k randomly selected within the interval $[4, 20]$ (dark line).

$\left\{\mathbb{P}^{b_1}, \ldots, \mathbb{P}^{b_B}\right\}$ be B bootstrap clustering ensembles produced by sampling with replacement from $\mathbb{P}$, and let $\mathbb{P}^{*_B} = \{P^{*_{b_1}}, \ldots, P^{*_{b_B}}\}$ be the corresponding set of combined data partitions. The mean value of the average normalized mutual information between **k**-cluster combined partitions, $P^{*^{\mathbf{k}}_B}$, and the bootstrap clustering ensembles is given by

$$\overline{NMI(P^{*^{\mathbf{k}}_b}, \mathbb{P}^b)} = \frac{1}{B} \sum_{i=1}^{B} NMI(P^{*^{\mathbf{k}}_{b_i}}, \mathbb{P}^{b_i}), \tag{8}$$

and the corresponding variance is defined as follows

$$var\{NMI(P^{*^{\mathbf{k}}_b}, \mathbb{P}^b)\} = \frac{1}{B-1} \sum_{i=1}^{B} \left(NMI(P^{*^{\mathbf{k}}_{b_i}}, \mathbb{P}^{b_i}) - \overline{NMI(P^{*^{\mathbf{k}}_b}, \mathbb{P}^b)} \right)^2. \tag{9}$$

It is expected that a robust data partition combination technique will be stable with respect to minor clustering ensemble variations. This leads to the following minimum variance criterion. Find a partition P^* such that

$$P^* = P^{*^{\mathbf{k}}} : \min_{\mathbf{k}} \left\{ var\{NMI(P^{*^{\mathbf{k}}_b}, \mathbb{P}^b) \right\} \text{ is achieved.} \tag{10}$$

Let us define the *variance of NMI between the bootstrap clustering ensembles* as

$$var\{NMI(\mathbb{P}^b, \mathbb{P}^b)\} = \frac{1}{B-1} \sum_{i=1}^{B} \left(NMI(\mathbb{P}^{b_i}, \mathbb{P}^{b_i}) - \overline{NMI(\mathbb{P}^b, \mathbb{P}^b)} \right)^2, \tag{11}$$

with $\overline{NMI(\mathbb{P}^b, \mathbb{P}^b)} = \frac{1}{B} \sum_{i=1}^{B} NMI(\mathbb{P}^{bi}, \mathbb{P}^{bi})$. Minimization of the variance criterion in equation (10) implies the following inequality:

$$var\{NMI(P^{*}{}_{b}^{\mathbf{k}}, \mathbb{P}^b)\} \leq var\{NMI(\mathbb{P}^b, \mathbb{P}^b)\}. \tag{12}$$

In the following, standard deviations (std) will be used instead of variances.

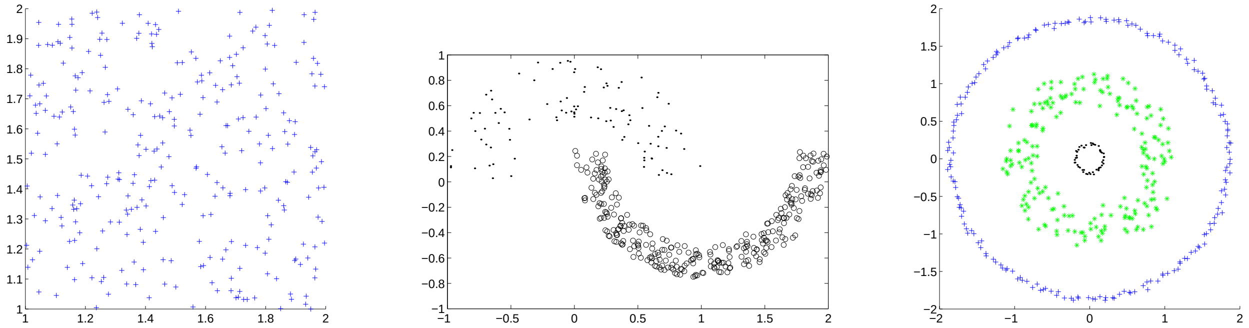
V. Combining Data Partitions Based on the K-Means Algorithm

We now consider clustering ensembles produced by the K-means algorithm. We follow a split and merge strategy. First, the data is split into a large number of small, spherical clusters, using the K-means algorithm with an arbitrary initialization of cluster centers. Multiple runs of the K-means lead to different data partitions $(P^1, P^2, \ldots, P^N)$. The clustering results are combined using the evidence accumulation technique described in section III, leading to a new similarity matrix between patterns, $\mathcal{C}$. The final clusters are obtained by applying the single-link method (or the AL method - in this section, we shall be using the SL; application of the AL is reported in the next section) on this matrix, thus merging small clusters produced in the first stage of the method. The final data partition is chosen as the one with the highest lifetime, yielding P^*.

Two different algorithms are considered, differing in the way k is selected to produce the K-means clustering ensemble:

- Algorithm I *Fixed k*: N data partitions are produced by a random initialization of the k cluster centers.
- Algorithm II *Random k*: the value of k for each run of the K-means is chosen randomly within an interval $[k_{min}, k_{max}]$. N data partitions are produced by a random initialization of the k cluster centers.

These algorithms will be evaluated using the following data sets: (i) random data in a 5-dimensional hyper-cube (figure 6(a)); (ii) half-rings data set (figure 6(b)); (iii) three concentric rings (figure 6(c)); (iv) cigar data set (figure 4(c)); (v) Iris data set.



(a)2-D view of 5-D random data set (300 patterns).

(b)Half-rings shaped clusters. Upper cluster has 100 patterns and the lower cluster has 300 patterns.

(c)3-rings data set. Number of patterns in the three rings are 200, 200 and 50.

Fig. 6. Test data sets.

A. Evidence Accumulation Combination Results

Both the fixed k and the random k versions of the K-means based evidence accumulation algorithms were applied on the test data sets, with $N = 50$.

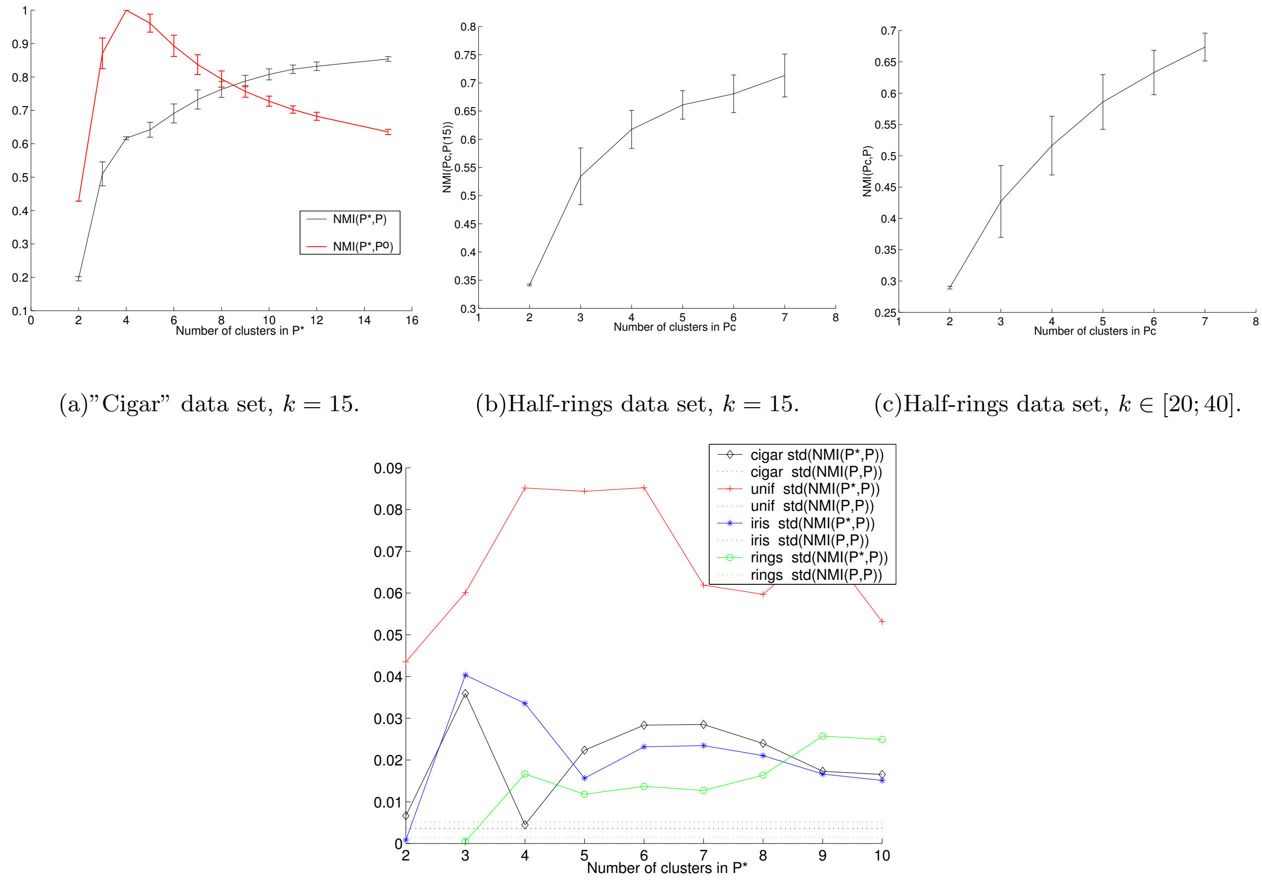
For the random data set in figure 6(a), a single cluster was identified no matter what fixed k value or interval for k value was considered. Algorithms I and II correctly identified the “natural”clusters in the three synthetic data sets, as plotted in figures 4(c), 6(b) and 6(c). Based on perceptual evaluation, the resulting clusters are considered the optimal partitions for these data. As expected, a direct application of the K-means algorithm to these four data sets leads to a very poor performance; the K-means algorithm can therefore be seen as a weak clusterer for these data sets. These examples show that the performance of weak clusterers, such as the K-means for data sets with complex shaped clusters, can be significantly improved by using the evidence accumulation strategy.

The Iris data set consists of three types of Iris plants (Setosa, Versicolor and Virginica), with 50 instances per class, represented by 4 features. This data, extensively used in classifier comparisons, is known to have one class (Setosa) linearly separable from the remaining two classes, while the other two classes partially overlap in the feature space. The most consistent combined partition obtained consists of two clusters, corresponding to a merging of the two overlapping classes Virginica and Versicolor, and a single cluster for the Setosa type. This example illustrates the difficulty of the K-means based evidence accumulation method using the SL in handling touching clusters. Section VI proposes a technique to address this situation.

B. Optimality Criteria and Adequacy of the Highest Lifetime Partition Criterion

The results presented in the previous section are concerned with combined partitions extracted from the dendrogram produced by the SL method according to the highest lifetime partition criterion, defined in section III-C. We now investigate the optimality of these results and the ability of this criterion on deciding the number of clusters in the combined partition.

The typical evolution of $\overline{NMI(P^{*^{\mathbf{k}}_{b}}, \mathbb{P}^{b})}$ and of $std\{NMI(P^{*^{\mathbf{k}}_{b}}, \mathbb{P}^{b})\}$ is illustrated in figure 7(a) (curve and error bars in black, referred to as $NMI(P^{*}, P)$) for the cigar data set; statistics were computed over $B = 100$ bootstrap experiments, and $P^{*^{\mathbf{k}}_{b}}$ partitions were obtained by forcing $\mathbf{k}-$cluster solutions using the SL method on the co-association matrices produced by combining $N = 50$ data partitions, with fixed $k = 15$. While the average normalized mutual information grows with increasing $\mathbf{k}$ (with a maximum at the number of clusters in the clustering ensemble, $\mathbf{k} = 15$), the variance is a good indicator of the "natural" number of clusters, having a minimum value at $\mathbf{k} = 4$; the partition lifetime criterion for extracting the combined partition from the dendrogram produced by the SL method, leads precisely to this number of clusters, corresponding to the partition in figure 4(c). This also corresponds to the perceptual evaluation of the data, which we represent as P^{o}. The curve and error bars in red represent $\overline{NMI(P^{*^{\mathbf{k}}_{b}}, P^{o})} \equiv NMI(P^{o}, \mathbb{P}^{*^{\mathbf{k}}_{B}})$ and $std\{NMI(P^{*^{\mathbf{k}}_{b}}, P^{o})\}$, respectively. Now, zero variance is achieved for the 2-cluster and the 4-cluster solutions, meaning that a unique partition is produced as the corresponding $\mathbf{k}-$cluster consensus partition; the maximum agreement with perceptual evaluation of the data, $NMI(P^{o}, \mathbb{P}^{*^{k}_{B}}) = 1$, is obtained for $\mathbf{k} = 4$, which coincides with the minimum variance of



(a)"Cigar" data set, $k = 15$.

(b)Half-rings data set, $k = 15$.

(c)Half-rings data set, $k \in [20; 40]$.

(d)Variance plots for several data sets.

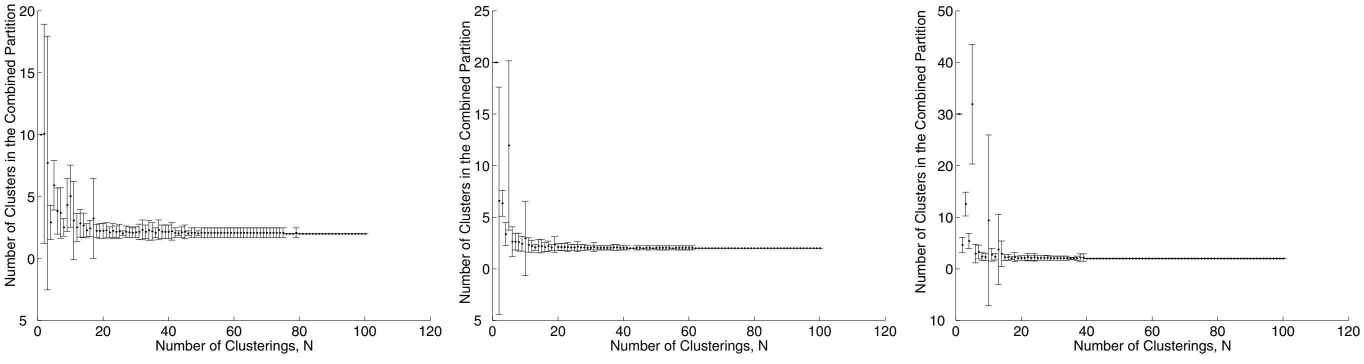
Fig. 7. Evolution of $\overline{NMI(P^{*k}_{b}, \mathbb{P}^b)}$ and of $std\{NMI(P^{*k}_{b}, \mathbb{P}^b)\}$ as a function of k, the number of clusters in the combined data partition, P^*, for various data sets, with $N = 50$ and $B = 100$.

$NMI(P^{*\mathbf{k}}_{b}, \mathbb{P}^b)$. Figures 7(b) and 7(c), corresponding to the half-rings data set, show that the same type of behavior is observed independently of using the fixed k or the random k version of the K-means evidence accumulation algorithm; in this case, minimum variance occurs for $\mathbf{k} = 2$, the number of natural clusters in this data set, which also coincides with the value provided by the highest lifetime partition criterion.

Figure 7(d) further corroborates these consistency and robustness results, showing plots of $std\{NMI(P^{*\mathbf{k}}_{b}, \mathbb{P}^b)\}$ (solid line curves) and of $std\{NMI(\mathbb{P}^b, \mathbb{P}^b)\}$ (dashed lines) for several data sets. It is interesting to note that, in the absence of a multiple clusters structure, the $std\{NMI(P^{*\mathbf{k}}_{b}, \mathbb{P}^b)\}$ curve for the random data set (in red) has high values, for $\mathbf{k} \geq 2$, compared to $std\{NMI(\mathbb{P}^b, \mathbb{P}^b)\}$, not obeying the inequality in equation (12); the evidence accumulation algorithm identifies a single cluster in this situation (figure 6(a)). With the remaining data sets, the evidence accumulation clustering decision corresponds to the minimum of $std\{NMI(P^{*^{\mathbf{k}}_{b}}, \mathbb{P}^b)\}$, which falls below $std\{NMI(\mathbb{P}^b, \mathbb{P}^b)\}$, thus obeying the inequality (12).

C. On the Selection of Design Parameters

The K-means based evidence accumulation method has two design parameters: N, the number of clusterings in the clustering ensemble, and k, the number of clusters in the K-means algorithm. The value of N is related to the convergence of the algorithm. Typically, convergence is achieved for values of N near 50, when an appropriate value of k is selected; with complex data sets (complexity meaning a departure from spherical clusters), we use $N = 200$, a value high enough to ensure convergence of the method. Figure 8 shows convergence curves for the half-rings data set. Each plot shows the mean and standard deviation of k^*, the number of clusters in the combined partition P^*, as a function of N, for three different values of k, computed over 25 experiments. As seen, faster convergence is achieved with higher values of k.



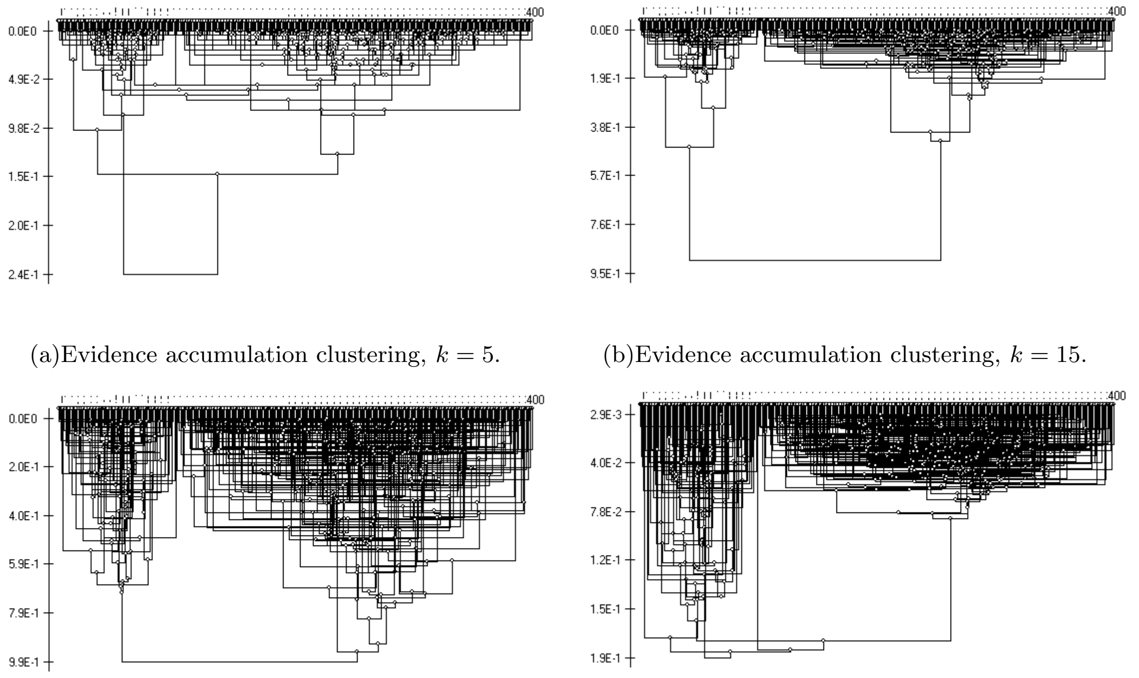
(a)$k = 10$.

(b)$k = 20$.

(c)$k = 30$.

Fig. 8. Convergence curves for the K-means based evidence accumulation combination algorithm, using a fixed k value, on the half-rings data set. Average and standard deviation values were estimated over 25 different clustering ensembles.

The influence of k on the fixed-k version of the algorithm is illustrated in figure 9, showing the



(a)Evidence accumulation clustering, $k = 5$.

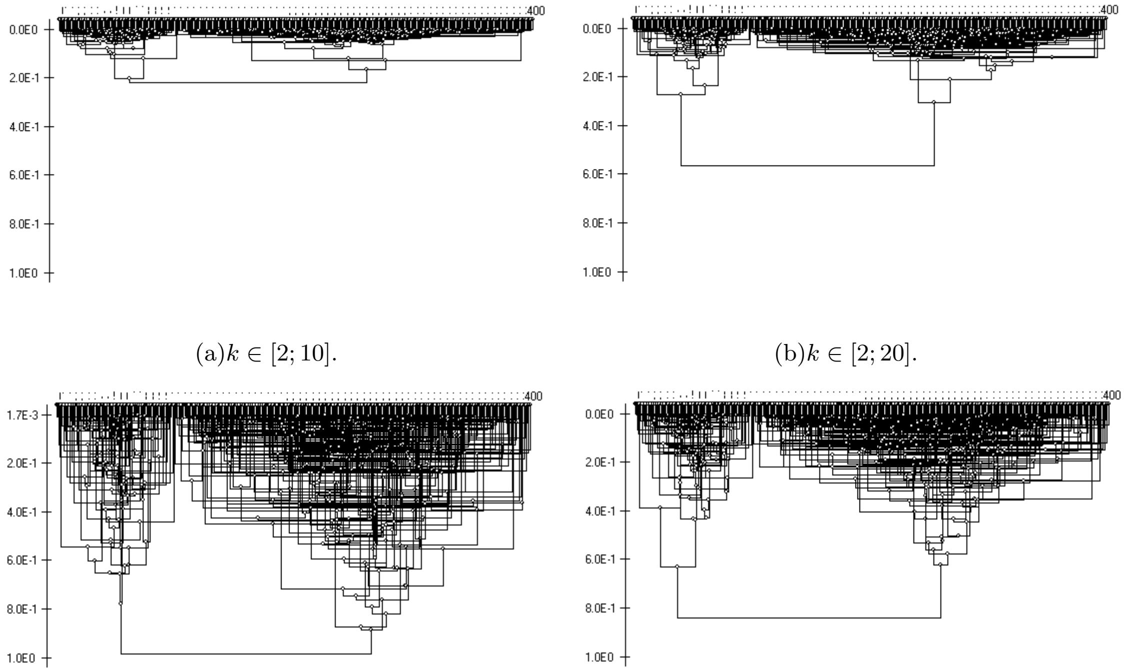
(b)Evidence accumulation clustering, $k = 15$.

(c)Evidence accumulation clustering, $k = 80$.

(d)Dendrogram produced by the SL method directly applied to the input data of figure 6(b), using the Euclidean distance.

Fig. 9. Half-rings data set. Vertical axis on dendrograms (a) to (c) corresponds to distances, $d(i,j)$, with $d(i,j) = 1 - \mathcal{C}(i,j)$.

dendrograms produced by the single-link method over the co-association matrix, $\mathcal{C}$, for several values of k, and $N = 200$. Small values of k are not able to capture the complexity of the data, while large values may produce an over-fragmentation of the data (in the limit, each pattern forming its own cluster). By using the mixture decomposition method in [25], the data set is decomposed into 10 gaussian components. This should be a lower bound on the value of k to be used with the K-means, as this algorithm imposes spherical shaped clusters, and therefore a higher number of components may be needed for evidence accumulation. This is in agreement with the dendrograms in figures 9(a)- 9(c). Although the two-cluster structure starts to emerge in the dendrogram for $k = 5$ (Fig. 9(a)), a clear cluster separation is present in the dendrogram for $k = 15$ (fig. 9(b)). As k increases, similarity values between pattern pairs decrease, and links in the dendrograms progressively begin to form at higher levels, causing the two natural clusters to be less clearly defined (see fig. 9(c) for $k = 80$); as a result, the number of clusters obtained in the combined partition increases with k.



(a)$k \in [2; 10]$.

(b)$k \in [2; 20]$.

(c)$k \in [60; 90]$.

(d)$k \in [2; 80]$.

Fig. 10. Combining 200 K-means clusterings, with k randomly selected within an interval $[k_{min}, k_{max}]$. Each figure shows the dendrogram produced by the SL method over the co-association matrix, when using the indicated range for k.

Algorithm II, with k randomly chosen in an interval, provides more robust solutions, being less sensitive to the values of k_{min} and k_{max}. Figure 10 shows the combination of clustering ensembles produced by K-means clustering with k uniformly distributed in the interval $[k_{min}, k_{max}]$. Several intervals were considered, with $N = 50$ and $N = 200$. The dendrograms in figures 10(a) and 10(b) show how the inability of the K-means clustering to capture the true structure of data with low values of k is progressively overcome as wider ranges of k are considered. Low values of k, characterized by high similarity values in the co-association matrix, $\mathcal{C}$, contribute to a scaling up of similarity values, where as high values of k produce random, high granularity data partitions, scaling down the similarity values (see figure 10(c)). Thus, using a wide range for values of k has an averaging effect and leads to identification of the true structure, as seen in figure 10(d).

In conclusion, there is a minimum value of k that is needed in order for the clustering structure to capture the structure of the data, using the K-means algorithm. Algorithm I should use a k value higher than this minimum. Although algorithm II is more robust than algorithm I, it is important to ensure that the range (k_{min}, k_{max}) is not completely below the minimum k value. Therefore, in order to adequately select these parameters, one could either: (i) gather *a priori* information about the minimum k value, such as the one provided by an algorithm performing gaussian mixture decomposition; or (ii) test several values for k, k_{min} and k_{max}, and then analyze the stability of the results. The mutual information indices defined previously can help us in this task.

VI. Experimental Results

Results of the application of the algorithms I and II on several synthetic data sets were already presented in section V-A. In this section we further apply the evidence accumulation clustering (EAC) paradigm, using either the single-link algorithm (EAC-SL) or the average-link algorithm (EAC-AL), on the following data sets: (1) *Complex image* - data set of figure 1; 739 patterns distributed in 8 clusters (the 10 points of the outer circle are considered one cluster); (2) *Cigar data* - 4 clusters (see figure 4); (3) *Half-rings* - 2 clusters in figure 6(b); (4) *3-rings* – figure 6(c), three clusters; (5) *Iris data* - three clusters; (6) Wisconsin *Breast-cancer* (683 patterns represented by 9 integer-valued attributes, with two class labels - benign and malignant), available at the UCI Machine Learning Repository; (7) *Optical digits* - from a total of 3823 samples (each with 64 features) available at the UCI repository (we used a subset composed of the first 100 samples of all the digits); (8) *Log Yeast* and (9) *Std Yeast* consist of the logarithm and the standardized version (normalization to zero mean and unitary variance), respectively, of gene expression levels of 384 genes over two cell cycles of yeast cell data [59]; (10) *Textures* - consists of 4,000 patterns in a 19-dimensional feature space, representing an image with 4 distinct textures, distributed in a 2 × 2 grid (fig. 11) [25].

The first set of experiments with the split-and-merge strategy described in section V aims at: (i)

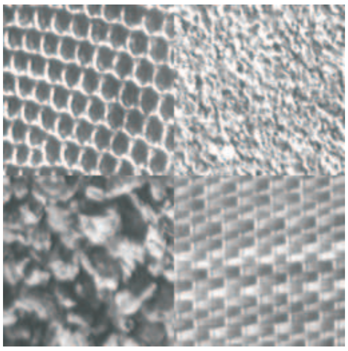


Fig. 11. Texture image.

assessing the improvement of the combination technique over single run of clustering algorithms; (ii) comparing the EAC with other combination methods. For these purposes, we have compared our experimental results with: (a) single run of the well-known clustering algorithms K-means (KM), SL, and AL, and the spectral clustering algorithm (SC) by Ng et al. [45]; (b) the cluster ensemble combination methods by Strehl and Ghosh [56], CSPA, HPGA and MCLA. The spectral clustering algorithm requires the setting of a scaling parameter, σ, in addition to the number of clusters, k, to partition the data. For each value of k, we run the algorithm for values of σ in the interval $[0.08; 3.0]$, in steps of 0.02; results herein presented correspond to the selection of σ according to the minimum square error criterion described in [45]. Combination methods were applied to clustering ensembles obtained by K-means clustering, with random initialization of cluster centers, and a random selection of k in large intervals.

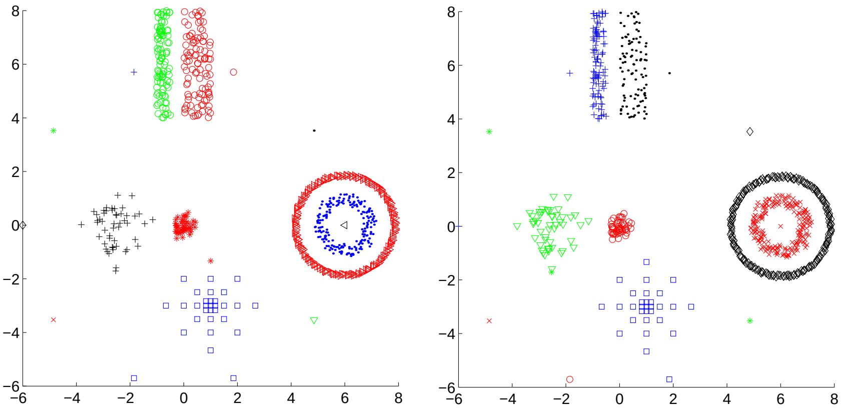
In order to evaluate the final partitioning we computed the error rate by matching the clustering results with ground truth information, taken as the known labelling of the data of the real data sets, or taken as the perceptual grouping of the synthetic data sets. The K-means algorithm is known to be sensitive to initialization. For each value of k we performed 50 runs of the K-means algorithm with random initialization of cluster centers, and retained only the result that corresponded to the best match obtained over these 50 experiments (minimum error rates attained) for comparison purposes, although a large variance on the error rates was observed.

	Single run of algorithms				Combination methods				
Data Set	KM	SL	AL	SC	EAC-SL	EAC-AL	CSPA	HPGA	MCLA
Complex image	41.5	52.5	48.4	1.2	18.7	12.5	41.3	34.6	-
Cigar	31.0	39.6	12.8	0.0	0.0	29.2	39.6	27.2	39.6
Half-rings	17.5	24.3	5.3	0.0	0.0	0.0	25.0	28.5	25.3
3-Rings	59.1	0.0	56.4	0.0	0.0	0.0	22.2	22.9	44.4
Iris	11.3	32.0	9.3	32.0	25.3	10.0	2.0	2.7	2.0
Breast Cancer	3.4	34.9	5.7	5.1	35.4	2.9	15.2	12.0	15.0
Optical Digits	20.3	89.4	24.3	12.3	60.0	21.0	16.0	22.0	12.0
Log Yeast	63.0	65.1	71.4	52.3	63.0	59.0	66.0	68.0	68.0
Std Yeast	26.3	63.8	34.1	29.2	52.0	33.0	47.0	43.0	46.0

TABLE II

Error rates (in percentage) for different clustering algorithms (KM: K-means; SL: single-link; AL: average-link; SC: spectral clustering) and combination methods (EAC-SL and EAC-AL are the proposed combination methods; CSPA, HPGA and MCLA are graph-based combination methods by Strehl and Gosh).

Table II summarizes the error rates obtained on the different data sets, when assuming k, the number of clusters, is known. Combination methods combined 50 ($N = 50$) partitions; experiments with $N = 100$ and $N = 200$ led to similar results. The dash in the MCLA column means that the computation was interrupted due to high time/space complexity. Comparing the results of the proposed EAC paradigm (columns EAC-SL and EAC-AL) with the graph-based combination approaches (last three columns), we can see that evidence accumulation has the best overall performance, achieving minimum error rates for all the data sets, except for the Iris and the Optical Digits cases. The difference in the results is particularly significant in situations of arbitrary shaped clusters (the first four data sets), demonstrating the versatility of the EAC approach in addressing the cluster shape issue. Furthermore, the lifetime criterion associated with the EAC technique, provides an automatic way to decide on the number of clusters, when k is not known and clusters are well separated. Comparison of the EAC-SL and the EAC-AL shows that EAC-AL is more robust, producing, in general, better results, especially in situations of touching clusters (last 5 data sets), a situation that is poorly handled by the SL method. In situations of well separated clusters, however, best performance is achieved by the EAC-SL method (notice the 29% error rate of the EAC-AL in the cigar data set, as compared to 0% error with the EAC-SL). Although the EAC-SL appears to perform worse for the complex image data set, this is because we forced the algorithm to find an 8-cluster partition. If we apply the lifetime criterion for deciding the number of clusters, a 13-cluster partition is obtained corresponding to a 1.2% error rate; the data partition obtained is plotted in fig. 12(a), showing that the basic clusters are correctly identified, the 10 patterns in the outer circle being either merged into a nearby cluster, or isolated as a single cluster.



(a)Cluster combination using EAC-SL - 13 clusters.

(b)Spectral clustering - 8 clusters.

Fig. 12. Clustering of the data set in figure 1(a) using EAC-SL and SC: the same error rate is achieved.

It can be seen from table II that the EAC-AL has an overall better performance than the single run of the KM, SL and AL algorithms. The best performing single run algorithm is the spectral clustering; the basic philosophy of this method consists of mapping the original space into a new, more compact feature space, by means of eigenvector decomposition, followed by clustering (K-means clustering, in the algorithm by Ng et al.), a computationally expensive procedure ($O(n^3)$ time complexity, with n being the number of samples). Both the SC and the EAC-SL are able to correctly identify the well separated clusters in data sets 2 to 4. Concerning the touching clusters, the EAC-AL wins in data sets Iris and Breast Cancer, and has a lower performance for the last three data sets. The difficulty of the EAC-SL to handle touching clusters is however notorious.

Data Set	k^*	KM (av)	EAC-SL	EAC-AL	CSPA	HPGA	MCLA
Complex image	8	59.4	59.4	59.4	62.0	72.9	-
Cigar	4	37.6	12.4	26.3	43.1	54.0	38.4
Half-rings	2	25.5	25.5	25.0	25.5	50.0	25.5
3-Rings	3	65.0	55.1	63.8	65.4	63.56	65.4
Iris	3	18.1	11.1	11.1	13.3	37.3	11.2
Breast Cancer	2	3.9	4.0	4.0	17.3	49.9	3.8
Optical Digits	10	27.4	56.6	23.2	18.1	40.7	18.5
Log Yeast	3	69.3	66.6	68.5	72.5	74.0	69.9
Std Yeast	3	38.8	44.1	31.8	42.2	45.3	43.1

TABLE III

Combining data partitions produced by K-means clustering with a fixed number of clusters equal to the "natural"number of clusters k^*.

In a second set of experiments we tried to evaluate the behavior of the several combination techniques, assuming the number of clusters is known, when the partitions in the clustering ensemble each have k^* clusters, with k^* being the "natural"number of clusters for the data set. The corresponding results are summarized in table III, showing average error rates computed over 20 repetitions of the combination experiments, using N=50. Abandoning the split and merge strategy, that was analyzed and justified in section V, the overall performance of combination techniques degraded, as expected, in particular in situations of complex shaped clusters. The issue in these experiments is to find a consensus data partition by combining the partitions in the clustering ensemble, the underlying philosophy of Strehl and Gosh's methods. It can be seen from table III that, once again, the proposed EAC technique produces, in general, better results than the graph-based combination approaches. Furthermore, the error rates obtained with the EAC-AL method are systematically lower or comparable to the average error rates obtained with the individual clusterings in the clustering ensembles (third column of table III). With the EAC-SL, combination results outperform average error rates computed over individual partitions in the clustering ensemble when dealing with well separated clusters; this result is not always observed in situations of touching clusters. The graph-based combination approaches, however, produced, most of the times, poorer performances than the average performance of single run of K-means.

We further evaluated how these evidence accumulation algorithms would perform when the original data are represented by a set of prototypes, thus reducing their computational complexity and extending their applicability to very large data sets. In order to improve the robustness of the EAC-SL to touching clusters, we propose a noise removal technique that combines sampling with density-based selection of prototypes. We start by decomposing the data set into a large number of small and compact clusters using the K-means algorithm, and representing each cluster by its centroid. Each centroid is seen as a prototype of a region in the feature space. The next phase is to remove noisy patterns based on density analysis. We build on the ideas of Jarvis and Patrick [60] and Ertoz- et al. [61] to obtain a set of "core"points from the available data. Jarvis and Patrick defined a shared nearest neighbor graph by the process of k-nearest neighbor sparsification: a link between patterns i and j is defined if and only if both i and j are included in their q nearest neighbor lists. The strength or weight of this link is defined by $w(i,j) = \sum_{l=1}^{shared\_neighbors} (k+1-m_l)(k+1-n_l)$, where m_l and n_l are the positions of the shared neighbor l in the i and j lists, respectively. The weight $w(i,j)$ is a measure of the similarity between pattern pairs based on the connectivity. By adopting the graph theoretic point of view, we estimate point densities by the sum of the similarities of a point's nearest neighbors, that is, its total link strength defined as [61]: $ls(i) = \sum_{j \in \{q-nearest\_neighbors\ of\ i\}} w(i,j)$. We use $q = 20$, as the number of neighbors. The higher the density $ls(i)$ (strong connectivity), the more likely it is that point i is a core or representative point. On the contrary, the lower the density, the more likely the point is a noise point or an outlier. Points that obey the criterion $ls(i) \geq (ls_{av} - \alpha ls_{std})$, where ls_{av} and ls_{std} are the average and the standard deviation, respectively, of ls, are defined as the core points; α is chosen to eliminate a small fraction of the patterns. We refer to this technique for selecting prototypes as the *shared nearest neighbor* (SNN) technique. The data set representatives thus obtained are then clustered using the techniques described previously. The final data partition is obtained by assigning the original patterns to clusters according to the nearest neighbor rule: assign pattern x_i to the cluster to which its nearest core point belongs.

		Single run of algorithms				Combination methods				
Data Set	#prot	KM	SL	AL	SC	EAC-SL	EAC-AL	CSPA	HPGA	MCLA
Textures	301	4.5	47.6	29.5	8.4	36.2	8.0	10.1	10.7	8.1
Textures, $k = 4$	301	4.5	47.6	29.5	8.4	10.1	8.1	9.25	28.8	8.1
Iris	122	11.3	32.0	14.0	32.0	10.0	10.0	8.7	14.7	16.0

TABLE IV

CLUSTERING USING PROTOTYPES. THE COLUMN "#PROT"INDICATES THE NUMBER OF PROTOTYPES USED.

Table IV shows some of the results obtained. For the Texture data set we selected 301 core points out of 4000 in the original data using the SNN technique; the first row in table IV concerns the construction of clustering ensembles using $k \in [2; 20]$, $N = 200$, while in the second row the clustering ensemble used a fixed $k = 4$, which is equal to the true number of classes. Once again, the proposed EAC paradigm produces better results than the graph-based combination techniques. While the EAC-

AL outperforms the SL, AL and the SC algorithms, selection of the best single run of the K-means algorithm, with $k = 4$, over 50 experiments leads to the best error rate, suggesting a compact structure of the clusters; this is further corroborated by the significant reduction in the error rate by the EAC-SL method when using a fixed-k composition of the clustering ensemble (10% error), as compared to the mixed-k composition (36.2%). As shown in the last row of table IV, the performance of the EAC-SL is also significantly improved by the sampling technique on the Iris data set, achieving similar results to the EAC-AL method (10% error). The same happens with the Breast Cancer database: by applying the SNN technique, 199 prototypes were used leading to an error rate of 3.1% with the EAC-SL method, a results comparable to the EAC-AL, and better than all the remaining algorithms and combination methods. The spectral kernel method described in [46] achieves a 20.35% error rate when using a Gaussian kernel, and 2.71% error with a linear kernel on the Breast Cancer data.

Concerning computational time complexity, the split-and-merge strategy of cluster ensemble combination comprises two phases: (1) building the cluster ensemble using the K-means algorithm, which takes $O(nkN)$ time, with n being the number of samples, k the number of clusters in each data partition, and N is the number of partitions in the clustering ensemble; (2) combining the data partitions. The proposed EAC combination method, associated with the SL or the AL algorihms, is $O(n^2N)$; compared to the CSPA ($O(n^2Nk)$), HPGA ($O(nNk)$), and MCLA ($O(nk^2N^2)$) combination methods, HPGA is the fastest. Except for the spectral clustering method ($O(n^3)$), ensemble methods are computationally more expensive than the single run clustering algorithms analyzed (K-means, SL, and AL); however, as demonstrated empirically, they provide increased robustness and performance, being able to identify complex cluster structures, not adequately handled by these algorithms.

VII. Conclusions

We have proposed a method for combining various clustering partitions of a given data set in order to obtain a partition that is better than individual partitions. These individual partitions could have been obtained either by applying the same clustering algorithm with different initialization of parameters or by different clustering algorithms applied to the given data. Our evidence accumulation technique maps the clustering ensemble into a new similarity measure between patterns, by accumulating pairwise pattern associations. Different clustering algorithms could be applied on the new similarity matrix. We have explored the evidence accumulation clustering approach with the single-link and average-link hierarchical agglomerative algorithms, to extract the combined data partition. Furthermore, we have introduced a theoretical framework, and optimality criteria, for the analysis of clustering combination results, based on the concept of mutual information, and on variance analysis using bootstrapping. A K-means based evidence accumulation technique was analyzed in light of the proposed optimality criteria. Results obtained on both synthetic and real data sets illustrate the ability of the evidence accumulation technique to identify clusters with arbitrary shapes and arbitrary sizes. Experimental results were compared with individual runs of well known clustering algorithms, and also with other cluster ensemble combination methods. We have shown that the proposed evidence accumulation clustering performs better compared to other combination methods. It is expected that the application of the evidence accumulation technique to more powerful clustering methods than K-means, can lead to even better partitions of complex data sets.

Acknowledgments

We would like to thank Martin H. Law and André Lourenço for their help in applying the spectral clustering method on the data sets. This work was partially supported by the Portuguese Foundation for Science and Technology (FCT), Portuguese Ministry of Science and Technology, and FEDER, under grant POSI/33143/SRI/2000, and ONR grant no. N00014-01-1-0266.

References

[1] D. Fasulo, "An analysis of recent work on clustering," Tech. Rep., University of Washington, Seatle. http://www.cs.washington.edu/homes/dfasulo/clustering.ps. http://citeseer.nj.nec.com/fasulo99analysi.html, 1999.

[2] D. Judd, P. Mckinley, and A. K. Jain, "Large-scale parallel data clustering," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 2, pp. 153–158, 1997.

[3] S. K. Bhatia and J. S. Deogun, "Conceptual clustering in information retrieval," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 28, no. 3, pp. 427–536, 1998.

[4] C. Carpineto and G. Romano, "A lattice conceptual clustering system and its application to browsing retrieval," *Machine Learning*, vol. 24, no. 2, pp. 95–122, 1996.

[5] E. J. Pauwels and G. Frederix, "Finding regions of interest for content-extraction," in *Proc. of IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases VII*, San Jose, January 1999, vol. SPIE Vol. 3656, pp. 501–510.

[6] H. Frigui and R. Krishnapuram, "A robust competitive clustering algorithm with applications in computer vision," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 450–466, 1999.

[7] A.K. Jain, M. N. Murty, and P.J. Flynn, "Data clustering: A review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, September 1999.

[8] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, Wiley, second edition, 2001.

[9] L. Kaufman and P. J. Rosseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, John Wiley & Sons, Inc., 1990.

[10] B. Everitt, *Cluster Analysis*, John Wiley and Sons, 1993.

[11] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, Academic Press, 1999.

[12] A. K. Jain and J. V. Moreau, "Bootstrap technique in cluster analysis," *Pattern Recognition*, vol. 20, no. 5, pp. 547–568, 1987.

[13] R. Kothari and D. Pitts, "On finding the number of clusters," *Pattern Recognition Letters*, vol. 20, pp. 405–416, 1999.

[14] J. Buhmann and M. Held, "Unsupervised learning without overfitting: Empirical risk approximation as an induction principle for reliable clustering," in *International Conference on Advances in Pattern Recognition*, Sameer Singh, Ed. 1999, pp. 167–176, Springer Verlag.

[15] D. Stanford and A. E. Raftery, "Principal curve clustering with noise," Tech. Rep., University of Washington, http://www.stat.washington.edu/raftery, 1997.

[16] Y. Man and I. Gath, "Detection and separation of ring-shaped clusters using fuzzy clusters," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 8, pp. 855–861, 1994.

[17] R. Dubes and A. K. Jain, "Validity studies in clustering methodologies," *Pattern Recognition*, vol. 11, pp. 235–254, 1979.

[18] T. A. Bailey and R. Dubes, "Cluster validity profiles," *Pattern Recognition*, vol. 15, no. 2, pp. 61–83, 1982.

[19] M. Har-Even and V. L. Brailovsky, "Probabilistic validation approach for clustering," *Pattern Recognition*, vol. 16, pp. 1189–1196, 1995.

[20] N. R. Pal and J. C. Bezdek, “On cluster validity for the fuzzy c-means model,” *IEEE Trans. Fuzzy Systems*, vol. 3, pp. 370–379, 1995.

[21] A. Fred and J. Leitão, “Clustering under a hypothesis of smooth dissimilarity increments,” in *Proc. of the 15th Int’l Conference on Pattern Recognition*, Barcelona, 2000, vol. 2, pp. 190–194.

[22] A. Fred, “Clustering based on dissimilarity first derivatives,” in *Proc. of the 2nd Intl. Workshop on Pattern Recognition in Information Systems*, J. I nesta and L. Micó, Eds. 2002, pp. 257–266, ICEIS PRESS.

[23] G. McLachlan and K. Basford, *Mixture Models: Inference and Application to Clustering*, Marcel Dekker, New York, 1988.

[24] S. Roberts, D. Husmeier, I. Rezek, and W. Penny, “Bayesian approaches to gaussian mixture modelling,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, 1998.

[25] M. Figueiredo and A. K. Jain, “Unsupervised learning of finite mixture models,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 381–396, 2002.

[26] J. D. Banfield and A. E. Raftery, “Model-based gaussian and non-gaussian clustering,” *Biometrics*, vol. 49, pp. 803–821, September 1993.

[27] B. Mirkin, “Concept learning and feature selection based on square-error clustering,” *Machine Learning*, vol. 35, pp. 25–39, 1999.

[28] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice Hall, 1988.

[29] H. Tenmoto, M. Kudo, and M. Shimbo, “MDL-based selection of the number of components in mixture models for pattern recognition,” in *Advances in Pattern Recognition*, Adnan Amin, Dov Dori, Pavel Pudil, and Herbert Freeman, Eds. 1998, vol. 1451 of *Lecture Notes in Computer Science*, pp. 831–836, Springer Verlag.

[30] H. Bischof and A. Leonardis, “Vector quantization and minimum description length,” in *International Conference on Advances on Pattern Recognition*, Sameer Singh, Ed. 1999, pp. 355–364, Springer Verlag.

[31] B. Fischer, T. Zoller, and J. Buhmann, “Path based pairwise data clustering with application to texture segmentation,” in *Energy Minimization Methods in Computer Vision and Pattern Recognition*, M. Figueiredo, J. Zerubia, and A. K. Jain, Eds. 2001, vol. 2134 of *LNCS*, pp. 235–266, Springer Verlag.

[32] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, New York, 1990.

[33] E. Gokcay and J. C. Principe, “Information theoretic clustering,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 158–171, 2002.

[34] C. Zahn, “Graph-theoretical methods for detecting and describing gestalt structures,” *IEEE Trans. Computers*, vol. C-20, no. 1, pp. 68–86, 1971.

[35] Y. El-Sonbaty and M. A. Ismail, “On-line hierarchical clustering,” *Pattern Recognition Letters*, pp. 1285–1291, 1998.

[36] M. Chavent, “A monothetic clustering method,” *Pattern Recognition Letters*, vol. 19, pp. 989–996, 1998.

[37] A. Fred and J. Leitão, “A comparative study of string dissimilarity measures in structural clustering,” in *Int’l Conference on Advances in Pattern Recognition*, Sameer Singh, Ed., pp. 385–394. Springer, 1998.

[38] S. Guha, R. Rastogi, and K. Shim, “CURE: An efficient clustering algorithm for large databases,” in *Proc. of 1998 ACM-SIGMOID In. Conf. on Management of Data*, 1998.

[39] E. W. Tyree and J. A. Long, "The use of linked line segments for cluster representation and data reduction," *Pattern Recognition Letters*, vol. 20, pp. 21–29, 1999.

[40] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 17, pp. 790–799, 1995.

[41] D. Comaniciu and P. Meer, "Distribution free decomposition of multivariate data," *Pattern Analysis and Applications*, vol. 2, pp. 22–30, 1999.

[42] G. Karypis, E-H Han, and V. Kumar, "CHAMELEON: A hierarchical clustering algorithm using dynamic modeling," *IEEE Computer*, vol. 32, no. 8, pp. 68–75, 1999.

[43] P. Bajcsy and N. Ahuja, "Location- and density-based hierarchical clustering using similarity analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 9, pp. 1011–1015, 1998.

[44] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[45] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds., Cambridge, MA, 2002, MIT Press.

[46] N. Cristianini, J. Shawe-Taylor, and J. Kandola, "Spectral kernel methods for clustering," in *Advances in Neural Information Processing Systems 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. 2002, MIT Press, Cambridge, MA.

[47] P-Y. Yin, "Algorithms for straight line fitting using k-means," *Pattern Recognition Letters*, vol. 19, pp. 31–41, 1998.

[48] Chris Fraley and Adrian E. Raftery, "How many clusters? Which clustering method? Answers via model-based cluster analysis," *The Computer Journal*, vol. 41, no. 8, pp. 578–588, 1998.

[49] R. Dubes and A. K. Jain, "Clustering tecnhiques: the user's dilemma," *Pattern Recognition*, vol. 8, pp. 247–260, 1976.

[50] J. Kittler, M. Hatef, R. P Duin, and J. Matas, "On combining classifiers," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[51] T. Dietterich, "Ensemble methods in machine learning," in *Multiple Classifier Systems*, Kittler and Roli, Eds., vol. 1857 of *Lecture Notes in Computer Science*, pp. 1–15. Springer, 2000.

[52] L. Lam, "Classifier combinations: Implementations and theoretical issues," in *Multiple Classifier Systems*, Kittler and Roli, Eds., vol. 1857 of *Lecture Notes in Computer Science*, pp. 78–86. Springer, 2000.

[53] A. Fred, "Finding consistent clusters in data partitions," in *Multiple Classifier Systems*, Josef Kittler and Fabio Roli, Eds., vol. LNCS 2096, pp. 309–318. Springer, 2001.

[54] A. Fred and A. K. Jain, "Data clustering using evidence accumulation," in *Proc. of the 16th Int'l Conference on Pattern Recognition*, 2002, pp. 276–280.

[55] A. Fred and A. K. Jain, "Evidence accumulation clustering based on the k-means algorithm," in *Structural, Syntactic, and Statistical Pattern Recognition*, T. Caelli et al., Ed., vol. LNCS 2396, pp. 442–451. Springer-Verlag, 2002.

[56] A. Strehl and J. Ghosh, "Cluster ensembles - a knowledge reuse framework for combining multiple partitions," *Journal of Machine Learning Research*, vol. 3(Dec), pp. 583–617, 2002.

[57] B. Kamgar-Parsi and L. N. Kanal, "An improved branch and bound algorithm for computing k-nearest neighbors," *Pattern Recognition Letters*, vol. I, pp. 195–205, 1985.

[58] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley, 1991.

[59] A. Raftery K. Yeung, C.Fraley and W.Ruzzo, "Model-based clustering and data transformation for gene expression data," Tech. Rep. UW-CSE-01-04-02, Dept. of Computer Science and Engineering, University of Washington, 5 2001.

[60] R. A. Jarvis and E. A. Patrick, "Clustering using a similarity measure based on shared nearest neighbors," *IEEE Transactions on Computers*, vol. C-22, no. 11, 1973.

[61] L. Ertoz, M. Steinbach, and V. Kumar, "A new shared nearest neighbor clustering algorithm and its applications," in *Workshop on Clustering High Dimensional Data and its Applications at 2nd SIAM International Conference on Data Mining, http://www-users.cs.umn.edu/ kumar/papers/papers.html*, 2002.