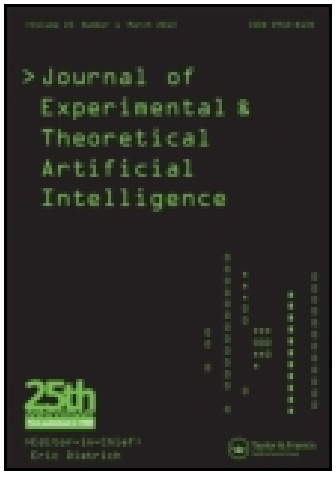


Journal of Experimental & Theoretical Artificial Intelligence

Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/teta20

To improve the quality of cluster ensembles by selecting a subset of base clusters

Hosein Alizadeh[a], Behrouz Minaei-Bidgoli[a] & Hamid Parvin[a]

[a] Computer Engineering Department, Iran University of Science and Technology, 1684613114 Narmak, Tehran, Iran

Published online: 23 Jul 2013.

To cite this article: Hosein Alizadeh, Behrouz Minaei-Bidgoli & Hamid Parvin (2014) To improve the quality of cluster ensembles by selecting a subset of base clusters, Journal of Experimental & Theoretical Artificial Intelligence, 26:1, 127-150, DOI: 10.1080/0952813X.2013.813974

To link to this article: http://dx.doi.org/10.1080/0952813X.2013.813974

Journal of Experimental & Theoretical Artificial Intelligence, 2014
Vol. 26, No. 1, 127–150, http://dx.doi.org/10.1080/0952813X.2013.813974

To improve the quality of cluster ensembles by selecting a subset of base clusters

Hosein Alizadeh*, Behrouz Minaei-Bidgoli[1] and Hamid Parvin[2]

Computer Engineering Department, Iran University of Science and Technology, 1684613114 Narmak, Tehran, Iran

(*Received 13 October 2012; final version received 16 May 2013*)

Conventional clustering ensemble algorithms employ a set of primary results; each result includes a set of clusters which are emerged from data. Given a large number of available clusters, one is faced with the following questions: (a) can we obtain the same quality of results with a smaller number of clusters instead of full ensemble? (b) If so, which subset of clusters is more efficient to be used in the ensemble? In this paper, these two questions are going to be answered. We explore a clustering ensemble approach combined with a cluster stability criterion as well as a dataset simplicity criterion to discover the finest subset of base clusters for each kind of datasets. Also, a novel method is proposed in order to accumulate the selected clusters and to extract final partitioning. Although it is expected that by reducing the size of ensemble the performance decreases, our experimental results show that our selecting mechanism generally lead to superior results.

Keywords: clustering ensemble; cluster stability; extended EAC; dataset simplicity

1. Introduction

Data clustering or unsupervised learning is an essential and also ill-posed, non-polynomial-hard problem. The objective of clustering is to group a set of unlabelled objects into homogeneous groups or clusters. These clusters are created so as to maximise the similarity of data with the same cluster labels and minimise it for data with different cluster labels (Dudoit & Fridlyand, 2003; Faceli, de Carvalho, & de Souto, 2007; Jain & Dubes, 1988). In the last few years, researches in clustering have tended to combinational methods. The main idea behind clustering ensemble is that different lookings at data can generate diverse partitions of the same data. By combining the resulting partitions, it is possible to obtain an efficient consensus partitioning even when the clusters are not dense and well separated (Jain, 2009). In fact, cluster ensemble methods attempt to find better and more robust clustering solutions by fusing information from several primary partitionings (Fred & Lourenco, 2008).

Generally, there are two main steps in clustering ensemble. The first step is creating some primary partitionings that should be as diverse as possible to give more information about the underlying patterns in the data (Fred & Jain, 2002). Therefore, the primary partitionings are better to be weak in order to produce more diverse partitionings. In fact, diversity leads to exploration of distinct views of inter-pattern relationships (Fred & Jain, 2005; Kuncheva, 2005). Many methods have been suggested for creating diversity in primary results: using different clustering algorithms

*Corresponding author. Email: halizadeh@iust.ac.ir

(Fred & Jain, 2005) and choosing different initialisation (Fred & Jain, 2002), different algorithm parameters (Alizadeh, Minaei-Bidgoli, & Parvin, 2013), subset of features (Strehl & Ghosh, 2002) and resampling over data (Kaufman & Rosseeuw, 1990; Minaei-Bidgoli, Parvin, Alinejad, Alizadeh, & Punch, 2011). In this paper, some of these methods are used for providing diverse primary results including: resampling, different initialization, and different algorithm parameters.

The second main step is to obtain consensus partition from the ensembles. This consensus partition has to minimise a consensus criterion. The different consensus criteria are employed based on the method selected to combine the base partitionings. In general, it can be a direct searching for a partitioning that maximises average normalised mutual information between the consensus partition and the base partitionings. It is discussed that even a greedy search which is very fast among informed search methods to find the consensus partition is computationally poor comparing to heuristic-based methods (Strehl & Ghosh, 2002).

Heuristic clustering ensemble algorithms commonly belong to one derivation of three main classes: instance-based clustering ensemble, cluster-based clustering ensemble and hybrid approach. The first algorithm usually obtains co-association matrix from base partitions along with aggregating them in the consensus partitioning (Fred & Jain, 2002, 2005, 2006; Fred & Lourenco, 2008). The second algorithm is clustering the clusters obtained in the first phase such as Hypergraph Partitioning Algorithm (HGPA), Meta-clustering Algorithm (MCLA) and so on (Strehl & Ghosh, 2002). The third algorithm, hybrid approach, combines the advantages of both instance-based and cluster-based approaches, such as Spectral Graph Bipartitioning method which partitions both instances and clusters (obtained in the first phase) simultaneously in the second phase (Fern & Brodley, 2004). Weighted Bipartite Partitioning Algorithm (Al-Razgan & Domeniconi, 2006) and Weighted Subspace Bipartite Partitioning Algorithm (Domeniconi & Al-Razgan, 2009) are the latest modifications of the hybrid approach. This approach maps the problem of 'finding a consensus partition' into a 'bipartite graph partitioning' problem. In Fern and Brodley (2004), this mapping has been done by 0/1 weight values and it is extended to the range of [0,1] later (Al-Razgan & Domeniconi, 2006; Domeniconi & Al-Razgan, 2009).

Since the instance and cluster-based clustering ensemble approaches are more popular (Alizadeh et al., 2013; Azimi & Fern, 2009; Fred & Jain, 2005; Gionis, Mannila, & Tsaparas, 2007; Minaei-Bidgoli, Topchy, & Punch, 2004; Minaei-Bidgoli et al., 2011; Punera & Ghosh, 2008; Strehl & Ghosh, 2002; Topchy, Jain, & Punch, 2005) than the others, they are consequently used in this paper. So, they are explained in detail in Section 2.

It has not been exposed whether reduction/change of primary clusters involved in the ensemble has any meaningful relation with the structure of datasets or not. This paper follows a procedure to empirically reveal this relation. It is explained in Section 3.

Specifically, our contributions are as follows:

(1) We analyse the effect of participating a subset of primary clusters specific to the dataset in the final ensemble. Moreover, we suggest which subset of clusters is better to be used in the ensemble. Consequently, we can reduce size of ensemble by choosing them instead of the full ensemble.
(2) We propose a procedure that is to adaptively choose a better subset of primary clusters rather than all of them.
(3) We provide a novel measure to classify datasets in terms of their simplicities.
(4) We suggest a modification on evidence accumulation clustering (EAC) method to accumulate the base results in a co-association matrix.
(5) We demonstrate the effectiveness of the proposed technique by applying it to several standard datasets. Furthermore, we take hypergraph partitioning algorithms such as

HGPA, Cluster-based Similarity Partitioning Algorithm (CSPA) and MCLA as consensus functions to have a robust comparison.

The remainder of the paper is organised as follows. Section 2 reviews the related literature. Section 3 presents the proposed procedure of reducing the size of clustering ensemble. The performance of the proposed method is evaluated in Section 4. Also, Section 4 provides a discussion about the effect of using each subset of primary clusters and the final results. A summarisation of our contributions and conclusion remarks are stated in Section 5.

2. Related work

In most of recent studies in the field of clustering ensemble, an equal weight is given to each ensemble member. In addition, either all clusters in each partition are selected to contribute to the combined solution, or all of them are filtered from the ensemble (Fred & Jain, 2006). Strehl and Ghosh (2002) have proposed a full search method for the best ensemble based on a selection criterion.

2.1 Previous clustering ensembles

Strehl and Ghosh (2002) have introduced the concept of consensus clustering. They have proposed three hypergraph-based consensus functions. They transform the data partitions into a hypergraph representation as the first step. Although the vertices of hypergraph are datapoints, its hyperedges are clusters. Then minimum-cut hypergraph algorithms can be employed to partition vertices or datapoints. The minimum k-cut of this hypergraph into k partitions gives the final consensus partition. Efficient heuristics to solve the k-way minimum-cut partitioning problem are known, some with computational complexity on the order of $O(|\epsilon|)$, where ϵ is the number of hyperedges. Three hypergraph algorithms, CSPA, HGPA and MCLA, are more described as follows.

2.1.1 CSPA

In CSPA, the original feature space is first mapped to a co-association feature space which is mentioned above. Then a minimum-cut hypergraph algorithm like METIS is applied on the new spaced datapoints. As before, this method presumes that 'the more two datapoints placed in same cluster in primary partitions, the more probable those datapoints inherently belongs to the same clusters'. The CSPA is the simplest heuristic among them. Its computational complexity is $O(kN^2M)$, where k is the number of clusters, N is the number of datapoints and M is the number of partitions. The following two methods are computationally less expensive.

2.1.2 HGPA

The HGPA algorithm presumes that 'vertices are datapoints and clusters emerged in primary partitions are its hyperedges'. Now again a minimum-cut hypergraph algorithm like METIS is applied on the hypergraph to split the vertices (datapoints) of the hypergraph to k distinct components. Its computational complexity is $O(kNM)$, where again k is the number of clusters, N is the number of datapoints and M is the number of partitions.

2.1.3 MCLA

MCLA first partitions the cluster obtained in the primary partitions and then it uses a vote-based mechanism to produce the consensus partition. Another name for this method is 'cluster

clustering'. The cluster clustering is performed using METIS. Its computational complexity is $O(k^2NM^2)$, where still k, N and M are fixed as the previous ones. The reader is referred to Strehl and Ghosh (2002) for further details about hypergraph-based methods.

The other famous name in the field of cluster ensemble is evidence accumulate clustering (EAC) which is first introduced in Fred (2001) and Fred and Jain (2002). It is an instance-based clustering ensemble approach. Fred and Jain has first presented the concept of EAC that uses the normalised similarity matrix, in which its (i, j)th entity accumulated the number of falling data i and data j into a same cluster. Indeed, the matrix maps the ensemble partitions to a new feature space that is the abstract of original ensemble. This data structure is called co-association matrix. Then a hierarchical-based clustering, like single linkage, can be employed to obtain the final results from the new feature space. The results of this method show that the combination of 'weak' clustering algorithms such as the k-means, which imposes a simple structure on the data, can lead to the identification of true underlying clusters with arbitrary shapes, sizes and densities. The reader is referred to Fred and Jain (2005) for further details about the EAC method. A newer version of EAC method, called Robust Adaptive Clustering is recently introduced by Mok, Huang, Kwok, and Au (2012). There are many other cluster ensemble methods in the literature review (Christou, 2011; Singh, Mukherjee, Peng, & Xu, 2010; Vega-Ponz & Ruiz-Shulcloper, 2011).

More recently, Topchy, Jain, and Punch (2004) have offered a probabilistic model of consensus partitions using a finite mixture of multinomial distributions in a space of clusterings. They have defined the consensus partition as a solution of a maximum-likelihood problem and using the EM algorithm has solved it. They have shown that even if missing values exist, the quality of the overall consensus partition is acceptable (Jain & Dubes, 1988). In their mixture model approach to clustering ensemble, the data should be adapted to a mixture of Gaussians, there are k mixture components, one for each cluster.

Fred and Jain (2006) have offered a new clustering ensemble method which learns the pairwise similarities between points in order to facilitate a proper partitioning of data without the *a priori* knowledge of the number of clusters and of the shape of the clusters. This method which is based on cluster stability evaluates the primary clustering results instead of final clustering.

A Bayesian version of the multinomial mixture model described was proposed by Wang, Shan, and Banerjee (2009). As in the simple mixture model, they assume that the distributions of consensus clusters are in the form of the multinomial one. They claim that Bayesian cluster ensembles are a mixed-membership model for learning cluster ensembles, and are applicable to all the primary variants of the problem. They have used a special version of expectation maximisation and Gibbs' sampling for inference and parameter estimation. After their work, a non-parametric version of Bayesian cluster ensemble has also been proposed (Wang, Domeniconi, & Laskey, 2010), in that it facilitates the number of final clusters to adapt with data. It is worthy to be mentioned that although both of the Bayesian cluster ensemble models presented above have been used only for hard partitional clustering, they can easily be used for overlapping clustering as well.

2.2 Cluster ensemble selection

Our work can be categorised as the problem of cluster ensemble selection. The problem of cluster ensemble selection investigates that 'given a large set of clustering solutions, can we obtain the same or better quality of results with a smaller number of clusters instead of using all available clusters?' This problem is following the practice of supervised ensemble learning (Caruana, Niculescu-Mizil, Crew, & Ksikes, 2004).

The problem of cluster ensemble selection is first introduced in Hadjitodorov, Kuncheva, and Todorova (2006) and further studied in Fern and Lin (2008), Azimi and Fern (2009) and Parvin, Minaei-Bidgoli, and Alizadeh (2011). In the proposed method of Hadjitodorov et al. (2006), first multiple cluster ensembles are generated. Then, the ensemble with the median diversity is chosen to produce the final clustering.

In contrast, Fern and Lin (2008) have sought to select a small subset from a large given ensemble to form the final ensemble. They have designed their ensemble selection methods based on quality and diversity, the two factors that have been shown to influence cluster ensemble performance. They have utilised the Normalised Mutual Information (NMI) criterion in order to evaluate the primary partitionings. Then, they have shown by comprehensive experimental results that using quality or diversity alone may not consistently achieve improved performance. They empirically evaluated their performance in comparison with both full ensembles and a random selection strategy and shown that selecting a subset of partitionings based on the both diversity and quality factors yields statistically significant performance improvement in comparison with the two rival strategies.

Moreover, Azimi and Fern (2009) has proposed an adaptive clustering ensemble algorithm which selects a subset of partitions adaptively. They have used the common NMI measure to evaluate whole partitions and to select those which are similar to a reference clustering ensemble result. The similarity between Azimi and Fern's method with this paper's scheme is that both methods select only a subset of primary results to participate in the final ensemble. On the other hand, the difference is that they evaluates whole primary partitions, whereas we assess the individual clusters. Therefore, a promising cluster among a weak partition cannot be detected in Azimi and Fern's method and consequently cannot be used in the final solution.

Considering the other point of view of the cluster ensemble selection, Parvin et al. (2011) have proposed a new method for clustering data so as to assign a weight vector to the feature space of the data. In this method, calculating the data variance through every feature, the feature in which variance is higher participates in combination with greater weight. They have also proved the convergence of their suggested algorithm.

One of the key problems addressed in this work, 'can we obtain the same quality of results with a smaller number of clusters instead of full ensemble?', is partially addressed by Bayesian clustering ensembles (Wang et al., 2009, 2010). In Bayesian clustering ensembles, the contribution of each base clustering to the consensus clustering is not equal. Base clusterings with redundant information would not affect the consensus clustering much. Our paper answers this key question in a different way that is discussed in Bayesian clustering ensembles. Next, we try to discover the relation between the average quality of base results and the quality of final consensus results.

2.3 Evaluating clusters

The main problem in the clustering ensemble based on a subset of selected primary clusters/partitions is how to evaluate each cluster/partition. Due to the unsupervisedness of clustering problem, its validation is inherently vague. Indeed, considering a special criterion in clustering problem results in the final partition, with high-quality score at that special criterion. In other words, employing a special criterion of clustering results in a partitioning which tries to optimise the score at that criterion. So presenting a suitable criterion to evaluate a partition in a specific domain is a complicated problem. Baumgartner et al. (2000) have presented a resampling-based technique to validate the results of exploratory fuzzy clustering analysis. Since the concept of cluster stability was introduced as a means to assess the validity of a data partition, it has been

incrementally used in the literature (Fred & Jain, 2006; Lange, Braun, Roth, & Buhmann, 2002; Law, Topchy, & Jain, 2004; Shamir & Tishby, 2007). This idea which is based on resampling method is initially described in Breckenridge (1989) and later generalised in different ways in Fridlyand and Dudoit (2001) and Levine and Domany (2001). Roth, Lange, Braun, and Buhmann (2002) and Roth, Braun, Lange, and Buhmann (2002) have proposed a resampling-based technique for cluster validation. The basic element in this method which is a complementary version of the past methods is cluster stability. The stability measures the similarity between obtained partitions from two individual clustering algorithms. The great values of the stability mean that applying the clustering algorithm several times on the dataset probably yields the same results (Luxburg & Ben-David, 2005; Rakhlin & Caponnetto, 2007). Roth and Lange (2003) have presented a new algorithm for data clustering which is based on feature selection. In this method, the resampling-based stability measure is used for setting the algorithm parameters. There are several cluster validation methods which are based on the stability concept (Lange, Roth, Braun, & Buhmann, 2004). Ben-Hur, Elisseeff, and Guyon (2002) have proposed a technique to exploit the stability measurements of clustering solutions obtained by perturbing the dataset. In this technique, the stability is characterised by the distribution of pairwise similarities between clusterings obtained from sub-samples of the data. First, the co-association matrix is acquired using the resampling method. Then, Jaccard coefficient is calculated from this matrix as the stability measure. Furthermore, Castro and Yang (Estivill-Castro & Yang, 2003) have offered a method by which support vector machines are used to evaluate the separation in the clustering results. By filtering noise and outliers, this method can identify the robust and potentially meaningful clustering results. More recently, Alizadeh, Parvin, and Parvin (2012) have proposed a new criterion called 'Max' for assessing individual clusters. They have also extended it to compensate some drawbacks and developed a new version called Alizadeh–Parvin–Moshki–Minaei criterion (Alizadeh, Minaei-Bidgoli, & Parvin, in press).

3. Proposed method

The main idea of the proposed method is to reveal a subset of better performing primary clusters in the ensemble. Full ensemble refers to the case that all primary clusters participate in the ensemble, and therefore final partitioning is made from all of them. This paper investigates whether change of participating clusters in the ensemble affects the performance of final results or not? To answer this question, a scenario is planed which is demonstrated in Figure 1. The following sections explain the components of the proposed procedure in detail.

3.1 Generating primary results

At first, primary results are generated to reach at a satisfactory level of diversity in clusters. First, B primary partitions are provided by using k-means (a well-known clustering algorithm; Jain & Dubes, 1988) by means of different parameters for creating diversity methods.

3.2 Evaluating clusters

In this step, the obtained clusters are evaluated and sorted by their goodness value. Some of traditional ways to evaluate individual clusters are based on sum of squared error, cohesion and separation and silhouette coefficient (Jain & Dubes, 1988; Tan, Steinbach, & Kumar, 2005). A new method for cluster validation, first noted in Lange et al. (2002) and Law et al. (2004) which has found high popularity in the literature, is cluster stability (Jain, 2009; Law et al., 2004;

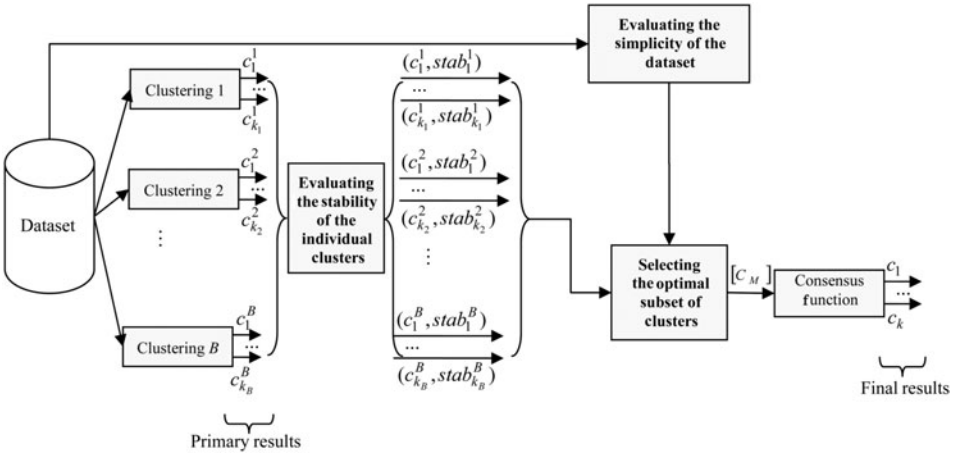


Figure 1. The procedure of the proposed method.

Luxburg & Ben-David, 2005; Shamir & Tishby, 2008). Because of the popularity of the cluster stability, it is used as cluster goodness in this paper.

Definition 1: Cluster stability. Cluster stability reflects the variation in the clustering results under perturbation of the data which are provided by resampling. On the other hand, a stable cluster is one that if we apply the clustering algorithm several times on the resampled data, it will probably be replicated more and more. In fact, cluster stability is measured as the amount of variation in the clustering solution over different subsamples drawn from the input data (Jain, 2009). Figure 2 taken from Law et al. (2004) gives an intuition of stable clusters in '2 Gaussian' data versus unstable ones in '2 Spiral' data.

Notation of cluster stability: Assume that the problem is computing the stability of cluster C_i. In this method first a set of partitionings over resampled datasets called reference set is generated. In this notation D is a resampled data and P^D is a partitioning over D. Now, the problem is answering this question: 'How many times the cluster C_i is repeated in the reference partitions?' Here the answer is notated NMI(C_i,P^D), the Normalised Mutual Information between the cluster C_i and a reference partition P^D. Most previous works only compare a partition with another partition (Meila, 2003; Strehl & Ghosh, 2002). The stability used in Law et al. (2004) evaluates the similarity between a cluster and a partition by transforming the cluster C_i to a partition and employing common partition-to-partition methods. In this method, P^a stands for a partition consisting of two clusters, $P^a = \{C_i, D/C_i\}$, where D/C_i denotes the set of datapoints in D that are not in C_i. P^D is transformed to P^b, a partition of two clusters, by $P^b = \{C*, D/C*\}$, where $C*$ denotes the union of all 'positive' clusters in P^D and others are in $D/C*$. A cluster C_j in P^D is positive if there are more than half of its datapoints in C_i. Now, the NMI(P^a, P^b) which is representative for NMI(C_i, P^D) is calculated as definition in Fred and Jain (2005):

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

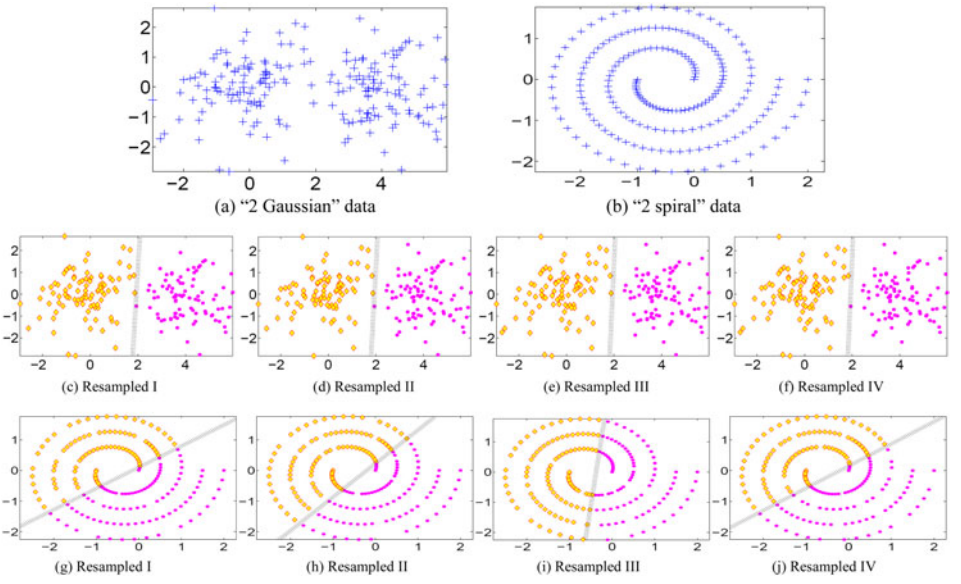


Figure 2. Results of k-means with $k = 2$ for different resampled versions of two datasets. Dotted lines in the figures correspond to the cluster boundaries. The partitions of '2 Gaussian' dataset are almost the same for different resampled versions, suggesting that k-means with $k = 2$ gives good clusters. The same cannot be said for the '2 spiral' dataset.

where n is the total number of samples, k_a stands for the number of clusters in the partition a, n_{ij}^{ab} denotes the number of shared patterns between the clusters $C_i^a \in p^a$ and $C_i^a \in p^b$, and n_i^a is the number of samples in the cluster i extracted from the partition a. This is done between the cluster C_i and all the partitions available in the reference set.

It is worthy to note that the cluster stability value is sensible to the mechanism under which the points are chosen. To choose the base clusters, Alizadeh et al. (2012, in press) show another mechanism resulted in a consensus partitioning somehow different from the one produced by the NMI-based selected clusters. However, choosing this mechanism to match clusters (to find the 'positive' cluster) is intuitively the 'best' approximation (Law et al., 2004).

Figure 3 demonstrates deriving P^a and P^b by an example. Figure 3(a),(b) is results of k-means with $k = 4$ over the artificial data. These two partitionings in Figure 3(a),(b) is modified in partitionings in Figure 3(c),(d), respectively. Now, the NMI is calculated according to these new partitions.

The total stability of cluster C_i is defined as follows:

$$\text{Stability}\,(C_i) = \frac{1}{M}\sum_{j=1}^{M} \text{NMI}_j, \tag{2}$$

where NMI_j stands for the stability of cluster C_i with respect to the jth partition in reference set and M is the number of partitions available in reference set. This procedure is applied for each cluster of every primary partition. The procedure of evaluating the stability of an individual cluster is summarised in Algorithm 1.

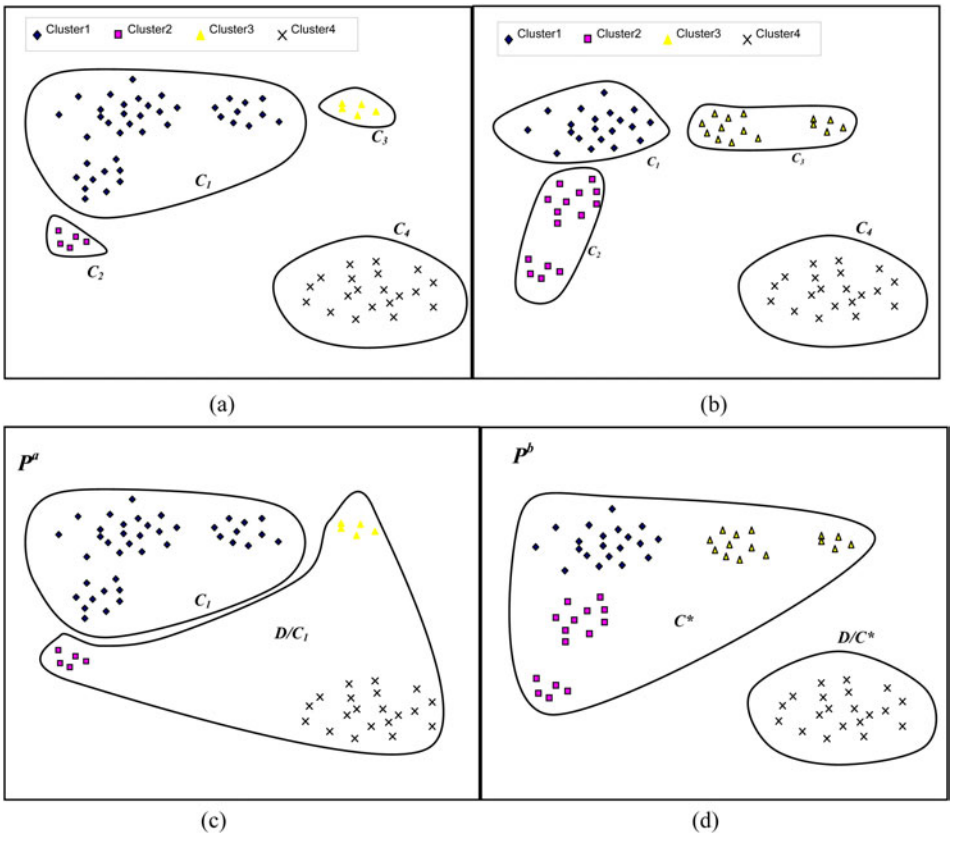


Figure 3. The problem of evaluating the cluster stability of C_1 in (a) with respect to (b) which is a partition from reference set. The new partitioning $P^a = \{C_1, D/C_1\}$ and $P^b = \{C*, D/C*\}$ are, respectively, shown in (c) and (d).

Algorithm 1. Evaluating cluster stability

Input: an individual cluster C_i from a primary partitioning.
Output: Stability(C_i)
(1) Build $P^a = \{C_i, D/C_i\}$;
(2) For $j := 1$ to M do
  (a) Apply a clustering algorithm over resampled data set and generate partitioning P^D;
  (b) Build $P^b = \{C*, D/C*\}$ from P^D;
  (c) Compute NMI[j] = NMI(P^a, P^b) as Equation (1);
  end For;
(3) Stability(C_i) = average of NMI[j] as Equation (2);

Note that according to the above procedure for evaluating individual clusters, the reference set is a distinct set of partitioning generated over resampled data with the size of M. To decrease the complexity of the problem, one can use the set of primary partitionings as the reference set.

Figure 4 shows the stability of the clusters in two artificial datasets. Since the base clustering algorithm is usually able to recognise the clusters in Figure 4(a), we expect the stability value to be very high for each cluster. Figure 4(b) has both higher and lower stable clusters.

3.3 Assigning subsets of base clusters to each dataset

After calculating stability values for all clusters, a specific subset of primary clusters is suggested for each class of datasets. Here, the goal is investigating which subset of the base clusters will be more efficient to participate in the ensemble. In other words, we are going to discover probable relationship between special subset of clusters and simplicity of the dataset. To study this relationship first the primary clusters are grouped into subsets according to their stabilities. Then, the datasets are classified into hard, medium and easy classes according to a new measure named simplicity which will be discussed later. After that, we determine the correspondence between special type of the dataset and a predefined subset of the primary clusters. Although one can partition them into a great number of subsets, the task of dedicating a special subset to a particular dataset will be difficult to choose. Therefore, we preferred using threefold subsets in our work. In addition, the effect of using different proportions of the primary clusters in the final ensemble is discussed in the experiments.

After calculating the stability criterion, the results of the whole primary clusters are sorted based on their stability value. Then, three different subsets of primary clustering results are investigated. So that the first, the second and the third subsets are, respectively, containing 33% of primary clusters having the lowest, the medium and the highest stability values. These subsets are shown in Figure 5.

In the rest of this issue, we discuss how the proposed method defines and calculates a simplicity criterion for datasets. Moreover, a specific method for choosing the more efficient subsets of primary clusters is suggested regarding to the value of dataset simplicity.

Definition 2: Partition stability. As it is mentioned the stable cluster is one which is repeatedly appeared in different partitionings over a dataset, it seems to be rational that the stability of a whole partition is defined by weighted averaging the stability values over its clusters. Therefore, we suggest the stability (P_i) as Equation (3):

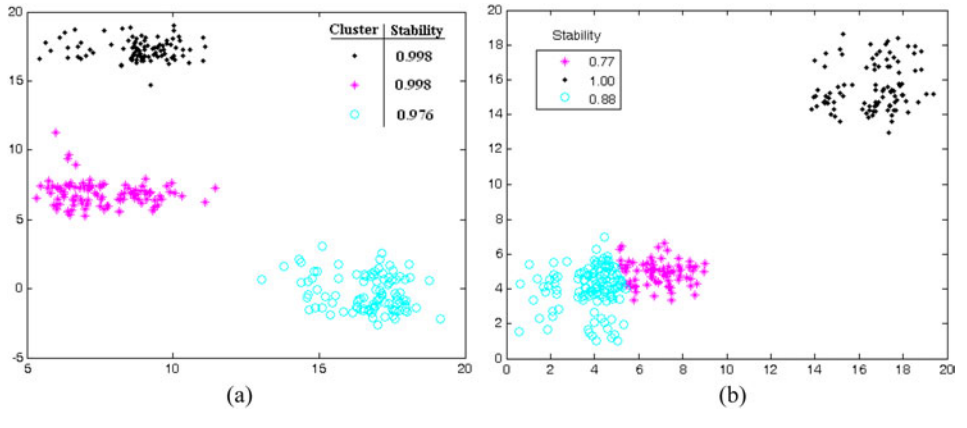


Figure 4. The stability of the clusters in two artificial datasets. (a) The 'well-separated' data has stable clusters with high stability value. (b) The diamond (◆) cluster is stable, whereas the circle (o) and star (∗) ones have lower stability.

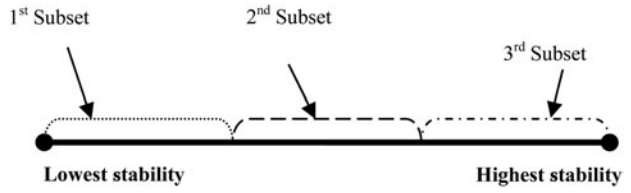


Figure 5. Subsets of primary clusters in sorted space of clusters regarding to their cluster stability values.

$$\text{stability}(P) = \frac{1}{N}\sum_{i=1}^{k} |C_i|\,\text{stability}(C_i), \tag{3}$$

where N, $|C_i|$ and k are size of data, number of patterns in cluster C_i and the number of clusters in partition P, respectively.

There is a fact that clustering results over simple datasets, like well-separated dataset, generally get high stability value. It is the case that most of the times the clusters can be simply discovered by base clustering algorithms. Logically, the high stable partitions derived from applying multiple clustering algorithms means that the dataset is easy and well separated. So, it can be rationally true that: 'the more stable partitions obtained by different clustering algorithms, the simpler dataset'.

Definition 3: Dataset simplicity. We suggest dataset simplicity to be the average of partition stabilities obtained from applying different base clustering algorithms over the dataset. Therefore, we define Equation 4 as a simplicity measure for a dataset D:

$$\text{simplicity}(D) = \frac{1}{B}\sum_{i=1}^{B} \text{stability}(P_i), \tag{4}$$

where B is the number of partitionings derived from base clusterings and P_i is ith partition.

Similar to what is done in partitioning the primary clusters into threefold subsets, here the datasets are classified based on their simplicity value in three categories: hard, medium and easy. Datasets with the simplicity under 0.5, in the interval [0.5–0.55] and upper 0.55 are, respectively, assigned to hard, medium and easy. These intervals are tuned empirically. Also, our experiments confirm that classifying based on these intervals works well; however, it cannot be claimed that this particular classification of the datasets is the optimal configuration.

Now the goal is to find the best subset of the base clusters for any specific dataset; i.e. we explore for a function f that maps a dataset D to an index i, indicating that the subset i (in Figure 5) is the best choice for the dataset D. This function is mathematically presented here:

$$\begin{gathered} f(D) \\ \text{simplicity}(D) \rightarrow \{1, 2, 3\} \end{gathered} \tag{5}$$

The examination configurations to solve Equation (5) (finding f) are as follows. At first, 15 artificial datasets (belong to the class of hard datasets) are created. Then, subsets 1, 2 and 3 (as shown in Figure 5), respectively, are used for constructing the ensemble.

After that, the performance of the final clusters is evaluated. This examination reveals that which subset of clusters is better to be used for hard datasets. This is done for both medium and easy datasets, in the same way.

3.4 Combining selected clusters

In this section, the selected clusters are combined and final clusters are extracted from them.

There are some combinational methods for aggregating the primary partitionings. One of the most popular approaches is the co-association-based combination (Alizadeh et al., in press; Fred & Jain, 2005; Minaei-Bidgoli et al., 2011). The EAC method accumulates the co-occurrence information between datapoints. Each entry of the co-association matrix is calculated from Equation (6):

$$C(i,j) = \frac{n_{i,j}}{m_{i,j}}, \tag{6}$$

where $n_{i,j}$ is the number of times the pair of objects with indices i and j was grouped together in a cluster, over the m clusterings, and $m_{i,j}$ is the number of resampled datasets where the pair of objects were simultaneously present, $0 \leq m_{i,j} \leq B$.

Since only a fraction of clusters are selected in our method, the common EAC method cannot completely disclose the pairwise similarity for erecting the co-association matrix. So, we modify Equation (6) in such a way that it can be applied for the case that only selected clusters participate in the final combination. In other words, the EAC method to construct the co-association matrix works well to present the knowledge of all primary partitions; not suitable for extracting knowledge from incomplete partitions. We modify Equation (6) and suggest Equation (7) alternatively to better disclose the inter-pattern relationships and call our method Extended Evidence Accumulation Clustering (EEAC). Each entry of the co-association matrix in EEAC is computed based on the definition shown in Equation (7):

$$C(i,j) = \frac{n_{i,j}}{\max(n_i, n_j)}, \tag{7}$$

where n_i and n_j are the number of times the ith and jth objects are presented in the selected clusters. Also, $n_{i,j}$ counts the number of selected clusters which are shared by objects with indices i and j.

The core of the EAC technique is mapping of the partitions into the co-association matrix, C. This corresponds to a nonlinear transformation of the original feature space into a new representation, summarised in the similarity matrix, C, induced by inter-pattern relationships available in the clustering ensemble. We can now apply any clustering algorithm over this new similarity matrix in order to extract a consistent data partition (Fred & Jain, 2005). Here, we emphasise the neighbourhood relationship and apply the single link, average-link and complete-link methods to the matrix C.

To strengthen the experimental results and have a comparison between different combination methods and also investigation of the effect of them over the final results, we apply the original EAC (Fred & Jain, 2002, 2005), CSPA, HGPA and MCLA methods (Strehl & Ghosh, 2002).

3.5 Post-labelling of missing samples

Because of choosing different k in the base partitionings, even the most stable cluster is highly likely divided into two or more unstable clusters. So, the base partitionings are diverse enough to present a good coverage of clusters with different stabilities to be chosen. It means that the proposed algorithm has been fed with a great option including different clusters with different properties and stability values. So, the probability of being missed any sample in the retained clusters is very low.

Although missing samples occurs rarely, the method does not guaranty that all the observations are included in at least one of the retained clusters. So a post-processing is inevitable to label the missing samples. The post-processing is assigning each missing sample to the nearest cluster centre in the feature space by using the Euclidian distance.

4. Empirical studies

4.1 Datasets

The proposed method is examined over 10 different UCI standard datasets. Datasets are tried to be diverse enough in their number of true classes, features and samples. A large variety of used datasets can validate better obtained results. Brief information about the used datasets is presented in Table 1. More information about any dataset is available in Newman et al. (1998).

Since this method tries to find a relation between the simplicity of a dataset and special subset of primary clusters, 15 artificial datasets are generated for each class of datasets. Each artificial dataset includes 300 datapoints distributed in three classes in 2D space. To provide these datasets, first three seed points as cluster centres are created. Then, the other datapoints are distributed around them by randomly choosing of uniform, normal or k distribution (Redding, 1999). Generating data in 1D causes clusters to be linear and in 2D creates circular clusters.

The simplicity value of any artificial dataset is sensitive to the parameters of distribution chosen for generating datapoints. Therefore, different datasets for each special class of simplicity are obtained by tuning the distribution's parameters. Figures 6–8 demonstrate these datasets.

4.2 Experiments

All experiments are reported over means of 10 independent runs of the algorithms. The final accuracy of the clustering algorithms which is one of the performance measurements is

Table 1. Characteristics of standard datasets.

		Class	Features	Samples
1	Breast-cancer*	2	9	683
2	Iris	3	4	150
3	Bupa*	2	6	345
4	SAHeart*	2	9	462
5	Ionosphere	2	34	351
6	Glass*	6	9	214
7	Halfrings	2	2	400
8	Galaxy*	7	4	323
9	Yeast*	10	8	1484
10	Wine*	3	13	178

* The datasets that are normalised in such a way that each feature of them has the mean and the variance equal to 0 and 1, respectively.

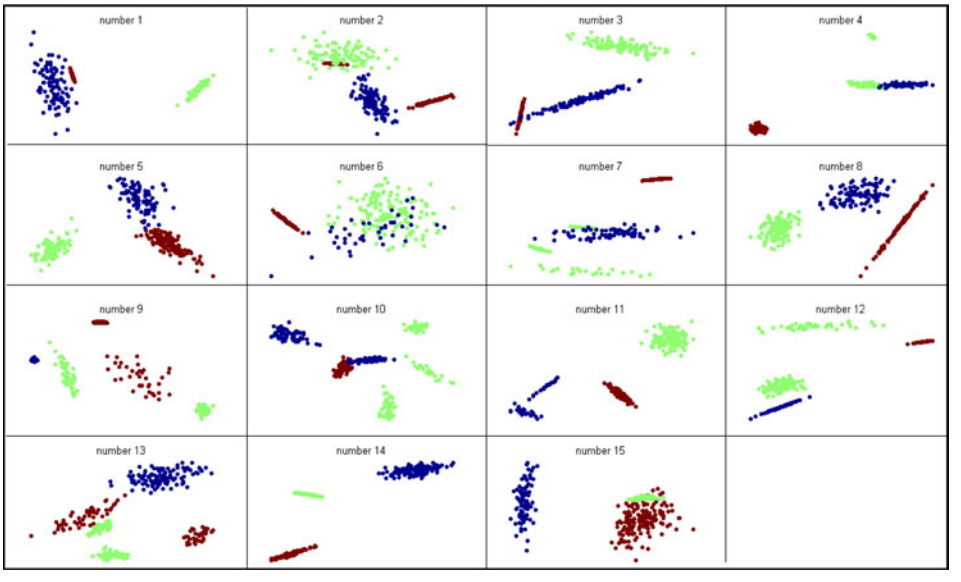


Figure 6. Artificial datasets from easy class.

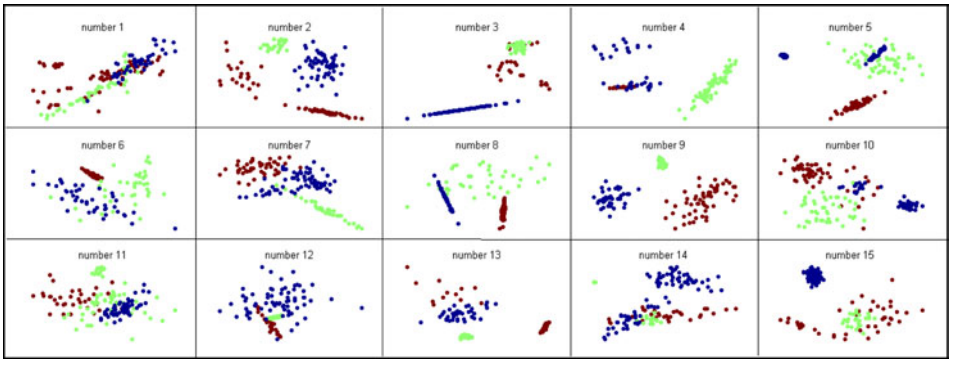


Figure 7. Artificial datasets from medium class.

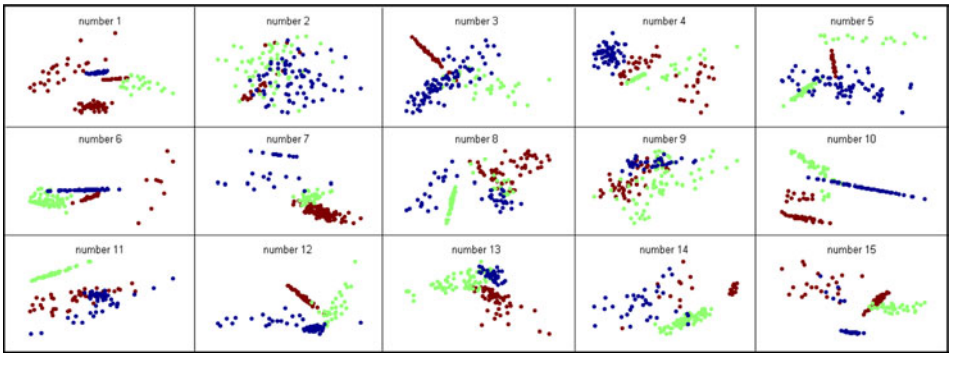


Figure 8. Artificial datasets from hard class.

evaluated by re-labelling between obtained clusters and the ground truth labels and then counting the percentage of the true classified samples. The proposed method is implemented in MATLAB (ver. 7.1).

Since it has been shown in Kuncheva, Hadjitodorov, and Todorova (2006) that k-means is one of the best options as the base algorithm to generate an ensemble, it is used 120 times to create primary partitions. To enforce diversity in the results, the k-means uses 50% sampling of dataset by changing of parameter k, from k to $k + 3$. Furthermore, one of the linkage algorithms is applied to extract final clusters from co-association matrix.

In Table 2, (a)–(c) show the obtained results of easy, medium and hard datasets, respectively, over three subsets of primary clusters. The average results over 15 different datasets are reported in the last rows. All the results of these tables are obtained using single linkage as the consensus function over co-association matrix. Obtained results show that in all cases one can use the third subset of primary clusters (the most stable cluster) for all kind of datasets; however, the results are very close to each other in the case of easy datasets. In other words, the simplicity of dataset is independent of performance of clustering ensemble using a special subset of primary results. So, the most stable clusters are usually more efficient to be used for the ensemble. Therefore, this method suggests thresholding over stability value of base clusters to participate in the ensemble. Hereafter, this method which employs that 33% of the highest stable clusters for combination is considered as the proposed method.

4.3 Results and discussion

Accuracy, F-measure and NMI are three of the most appropriate measures to evaluate the consensus partition. We evaluate and report the results based on these three measures. All of them are computed during a comparison between the consensus clustering results and the true labels in the original dataset. So it is obvious that the true labels of instances are not used during the clustering procedure and they are just used for evaluation of the final results. Figure 9 comprises the proposed method with full ensemble using accuracy measure. Numbers 1–10 are the numbers of datasets according to Table 1. It shows that although only 33% of the base clusters are available in the ensemble; the performance of the proposed method is comparable to the full ensemble. In other words, we can keep the performance of clustering by decreasing the size of ensemble.

As it is shown in Figure 9, the results of consensus partition based on the stable subset of clusters can outperform over or equal to the one based on all clusters. This is true in both aspects of computational cost and quality of results. The computational cost of construction of the co-association matrix for k clusters in a dataset with n records is $k \times n^2$. Order of this section is dominant section in the cluster ensemble algorithms, where the size of reference set, $r \lll k$. Since we used 33% of the primary clusters in the ensemble, the computational cost of proposed clustering ensemble algorithm is decreased by 66%.

Furthermore, the results are evaluated by using F-measure as another justification in Table 3. The average of F-measure over all 10 datasets is reported in the latest column. This table also confirms the results of Figure 9 which we can reduce the size of ensemble without loss of performance in terms of F-measure.

In the following, the average F-measure, NMI and accuracy over all 10 datasets are reported, respectively, in (a)–(c) columns of Table 4, sorted decreasingly. The average rank of proposed methods versus corresponding full methods is reported in the last row. In all three cases, the average rank obtained by proposed methods outperforms the other methods.

Table 2. Experiments by accuracy measure over (a) easy datasets, (b) medium datasets and (c) hard datasets.

(a)					(b)					(c)				
# of dataset	Simplicity of dataset	Accuracy by Subset 1	Accuracy by Subset 2	Accuracy by Subset 3	# of dataset	Simplicity of dataset	Accuracy by Subset 1	Accuracy by Subset 2	Accuracy by Subset 3	# of dataset	Simplicity of dataset	Accuracy by Subset 1	Accuracy by Subset 2	Accuracy by Subset 3
1	0.56	94.70	97.67	96.47	1	0.48	54.13	49.60	49.73	1	0.36	50.93	60.47	55.40
2	0.61	61.53	81.27	58.67	2	0.56	59.80	55.47	65.73	2	0.51	54.73	62.80	56.40
3	0.59	88.83	95.27	92.67	3	0.51	87.73	86.47	79.40	3	0.50	70.87	77.67	74.13
4	0.60	69.13	73.00	85.27	4	0.51	55.07	68.60	64.27	4	0.49	56.07	64.07	66.07
5	0.56	88.53	98.00	98.33	5	0.55	74.73	90.00	90.73	5	0.49	46.67	77.07	77.27
6	0.68	81.60	87.33	87.33	6	0.53	79.67	67.73	79.73	6	0.44	72.47	63.20	68.80
7	0.53	77.90	68.57	76.00	7	0.55	57.13	91.93	81.27	7	0.40	41.13	63.73	63.93
8	0.58	90.40	83.03	95.57	8	0.54	72.07	89.20	89.33	8	0.47	57.53	54.33	47.93
9	0.63	72.67	42.33	50.23	9	0.57	100.00	93.87	100.00	9	0.48	54.13	49.60	49.73
10	0.62	60.30	54.67	68.33	10	0.54	66.33	75.20	92.67	10	0.55	58.53	72.00	72.00
11	0.59	81.33	84.20	63.00	11	0.40	41.13	63.73	63.93	11	0.48	68.93	63.73	82.80
12	0.58	76.07	73.73	84.70	12	0.44	72.47	63.20	68.80	12	0.51	46.33	75.87	68.00
13	0.59	77.43	74.43	72.00	13	0.57	92.00	92.00	92.00	13	0.52	68.60	83.07	88.00
14	0.58	84.67	84.67	84.00	14	0.53	59.53	64.00	64.33	14	0.52	84.73	89.47	89.67
15	0.53	89.33	88.90	85.30	15	0.53	66.67	63.33	80.67	15	0.36	50.93	60.47	55.40
Average	0.59	79.63	79.14	79.86	Average	0.52	69.23	74.29	77.51	Average	0.48	59.40	68.36	68.58

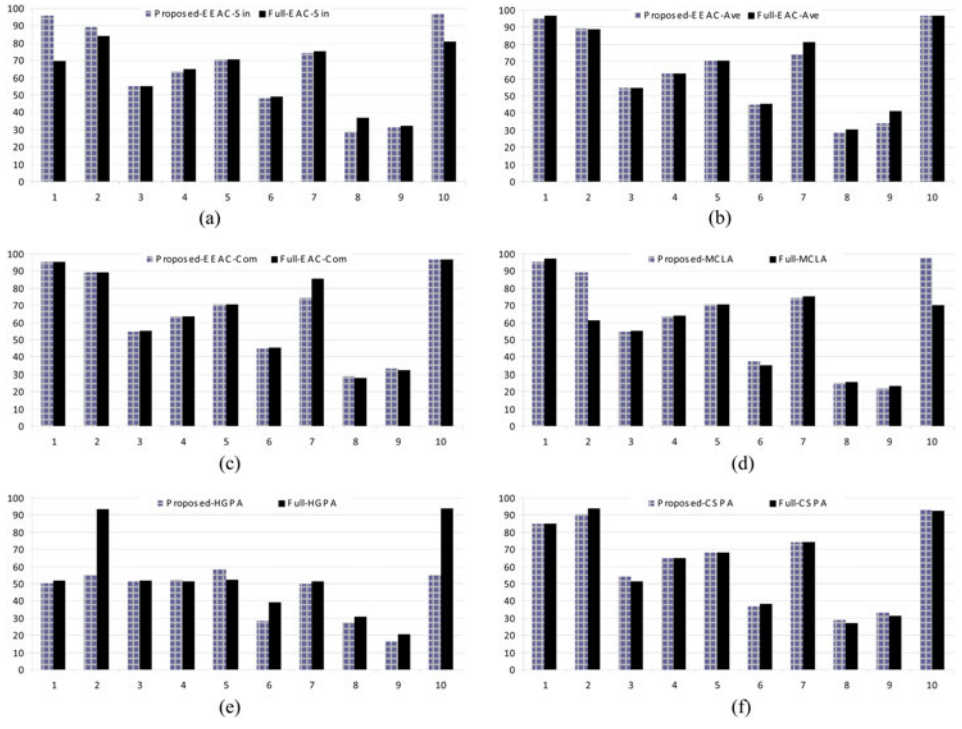


Figure 9. Classification accuracy results by different consensus functions. (a)–(c) The results applying hierarchical clustering algorithm as consensus function over the co-association matrix. (d)–(f) The results applying the consensus functions directly over the selected clusters.

Generally the NMI measure can be applied to determine the robustness and stability of a cluster or partition. So, here it is used as a qualification metric of consensus partition. Table 4(b) shows that the robustness of consensus partition in the proposed methods can be increased. This is valid for both *F*-measure and accuracy according to (a) and (c) columns in Table 4.

One can infer from this table that the ensemble size can be reduced not only for reduction of time complexity but also for improvement of performance of consensus partition in terms of *F*-measure, NMI and accuracy. This can be considered that if one lets robust clusters be used in the final ensemble, this will uncover the structure of data better than using all of them, provided that there is at least one cluster as the representative of each real cluster of a dataset in the final ensemble. This is why there are a great number of primary clusters that are not only non-representative of a real cluster in a dataset but also they blemish the final results. In other words, they are not real or valid clusters in the dataset. As a result, they are filtered by the proposed procedure and consequently the performance will be improved.

Figure 10 depicts the accuracies of the proposed method by choosing the 10%, 20%, ..., 100% of the most stable clusters by applying different consensus functions as final aggregators. This figure shows accuracy of the final results that may decrease while the ensembles size increases. In addition, if both of them increase, the improvement is not considerable.

Table 3. Experimental results by *F*-measure.

	Data sets										
Methods	Breast-cancer* 1	Iris 2	Bupa* 3	SAHeart* 4	Ionosphere 5	Glass* 6	Halfrings 7	Galaxy* 8	Yeast* 9	Wine* 10	Average 10 data
Proposed-MCLA	0.95	0.90	0.65	0.66	0.70	0.45	0.78	0.40	0.34	0.98	0.68
Full-MCLA	0.97	0.75	0.63	0.66	0.70	0.36	0.79	0.34	0.22	0.78	0.62
Proposed-HGPA	0.50	0.60	0.51	0.52	0.59	0.32	0.50	0.31	0.19	0.58	0.46
Full-HGPA	0.52	0.94	0.52	0.51	0.53	0.44	0.52	0.33	0.27	0.94	0.55
Proposed-CSPA	0.86	0.90	0.55	0.66	0.69	0.43	0.78	0.32	0.45	0.93	0.66
Full-CSPA	0.87	0.94	0.51	0.66	0.69	0.44	0.78	0.30	0.44	0.93	0.66
Proposed-EEAC-Ave	0.95	0.90	0.65	0.66	0.70	0.58	0.78	0.43	0.48	0.97	0.71
Full-EAC-Ave	0.96	0.90	0.66	0.65	0.70	0.58	0.82	0.33	0.53	0.97	0.71
Proposed-EEAC-Sin	0.95	0.90	0.66	0.67	0.70	0.66	0.78	0.44	0.49	0.97	0.72
Full-EAC-Sin	0.90	0.86	0.69	0.80	0.70	0.76	0.93	0.56	0.70	0.91	0.78
Proposed-EEAC-Com	0.95	0.90	0.65	0.66	0.70	0.58	0.78	0.37	0.52	0.97	0.71
Full-EAC-Com	0.95	0.90	0.64	0.67	0.70	0.58	0.84	0.32	0.44	0.97	0.70

* Note that some of datasets which are marked with star (*) in Table 1 are normalized in such a way that each feature of them has the mean and the variance equal to 0 and 1 respectively.

Table 4. The average of three clustering evaluation measures.

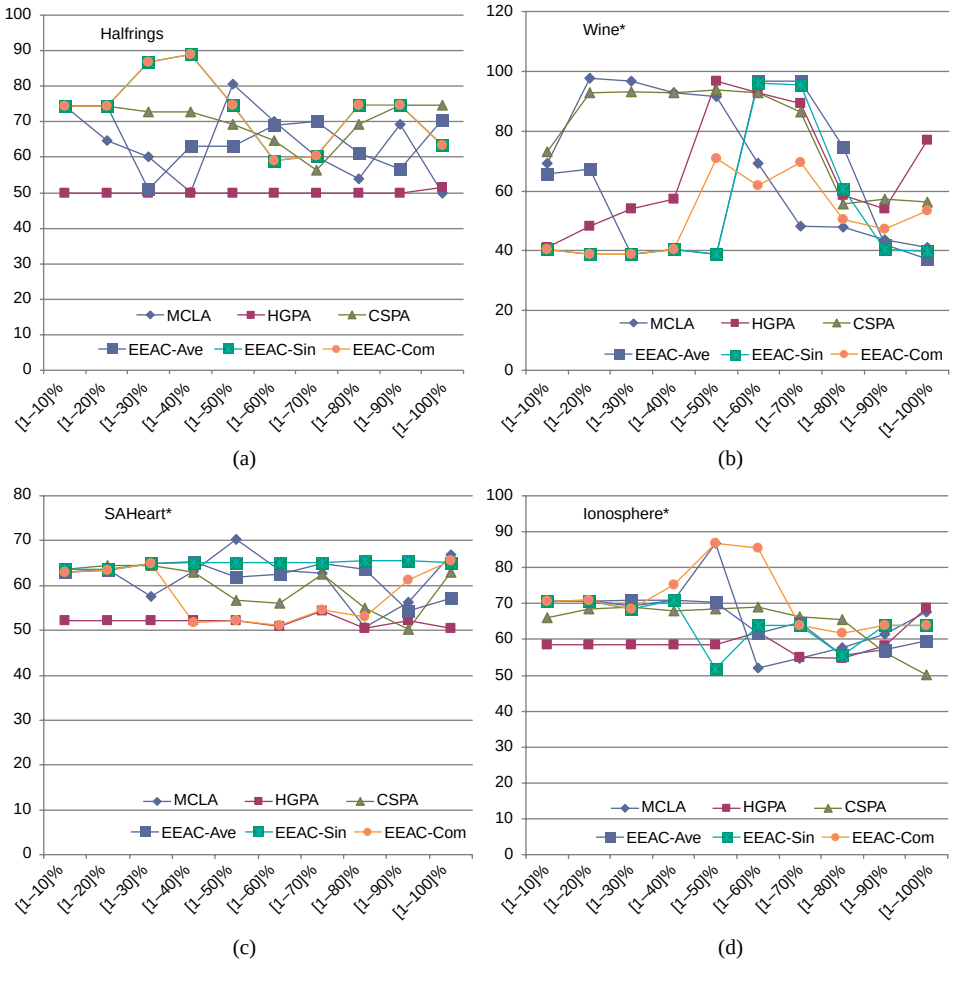
(a)			(b)			(c)		
Methods	**FM**	**Rank**	**Methods**	**NMI**	**Rank**	**Methods**	**ACC**	**Rank**
Full-EAC-Sin	0.78	0	Full-EAC-Ave	0.42	0	Full-EAC-Ave	0.67	0
Proposed-EEAC-Sin	0.72	1	Full-EAC-Com	0.40	1	Full-EAC-Com	0.66	1
Proposed-EEAC-Ave	0.71	2	Proposed-EEAC-Com	0.39	2	Proposed-EEAC-Sin	0.65	2
Full-EAC-Ave	0.71	3	Proposed-EEAC-Ave	0.38	3	Proposed-EEAC-Ave	0.65	3
Proposed-EEAC-Com	0.71	4	Proposed-EEAC-Sin	0.37	4	Proposed-EEAC-Com	0.65	4
Full-EAC-Com	0.70	5	Proposed-CSPA	0.36	5	Proposed-MCLA	0.63	5
Proposed-MCLA	0.68	6	Full-CSPA	0.35	6	Proposed-CSPA	0.63	6
Proposed-CSPA	0.66	7	Proposed-MCLA	0.33	7	Full-CSPA	0.63	7
Full-CSPA	0.66	8	Full-EAC-Sin	0.28	8	Full-EAC-Sin	0.62	8
Full-MCLA	0.62	9	Full-HGPA	0.26	9	Full-MCLA	0.58	9
Full-HGPA	0.55	10	Full-MCLA	0.25	10	Full-HGPA	0.54	10
Proposed-HGPA	0.46	11	Proposed-HGPA	0.09	11	Proposed-HGPA	0.45	11
Average Rank	4.8	6.2	Average Rank	5.3	5.7	Average Rank	5.2	5.8

Note: The green colour stands for clustering based on proposed cluster selection while the red colour stands for clustering based on full clusters. All values are average of applying the methods along with all 10 datasets.

5. Conclusion

In this paper, a new combinational clustering algorithm is proposed. Instead of using all available clusters in the generated ensemble, this method uses a subset of them. This paper also reveals which kind of primary clusters are more efficient to be used for combination. In addition, it is shown that the size of ensemble can be reduced while the performance of the clustering ensemble in terms of accuracy, NMI and *F*-measure is more or less constant or even increased. Experimental results over 10 UCI standard datasets confirm that the size of ensemble in the proposed method is reduced to one-third of the original one while the performance generally gets better in comparison with the full ensemble; however, it is expected to decrease.

Although the proposed method is presented as a general framework, the method is limited by some slow preliminary clustering algorithms like hypergraph algorithms (HGPA, CSPA and MCLA) and hierarchical clustering algorithms (single, average and complete linkages); i.e. the time complexity of the proposed method is dominated by the above-mentioned algorithms. Also,



(Continued)

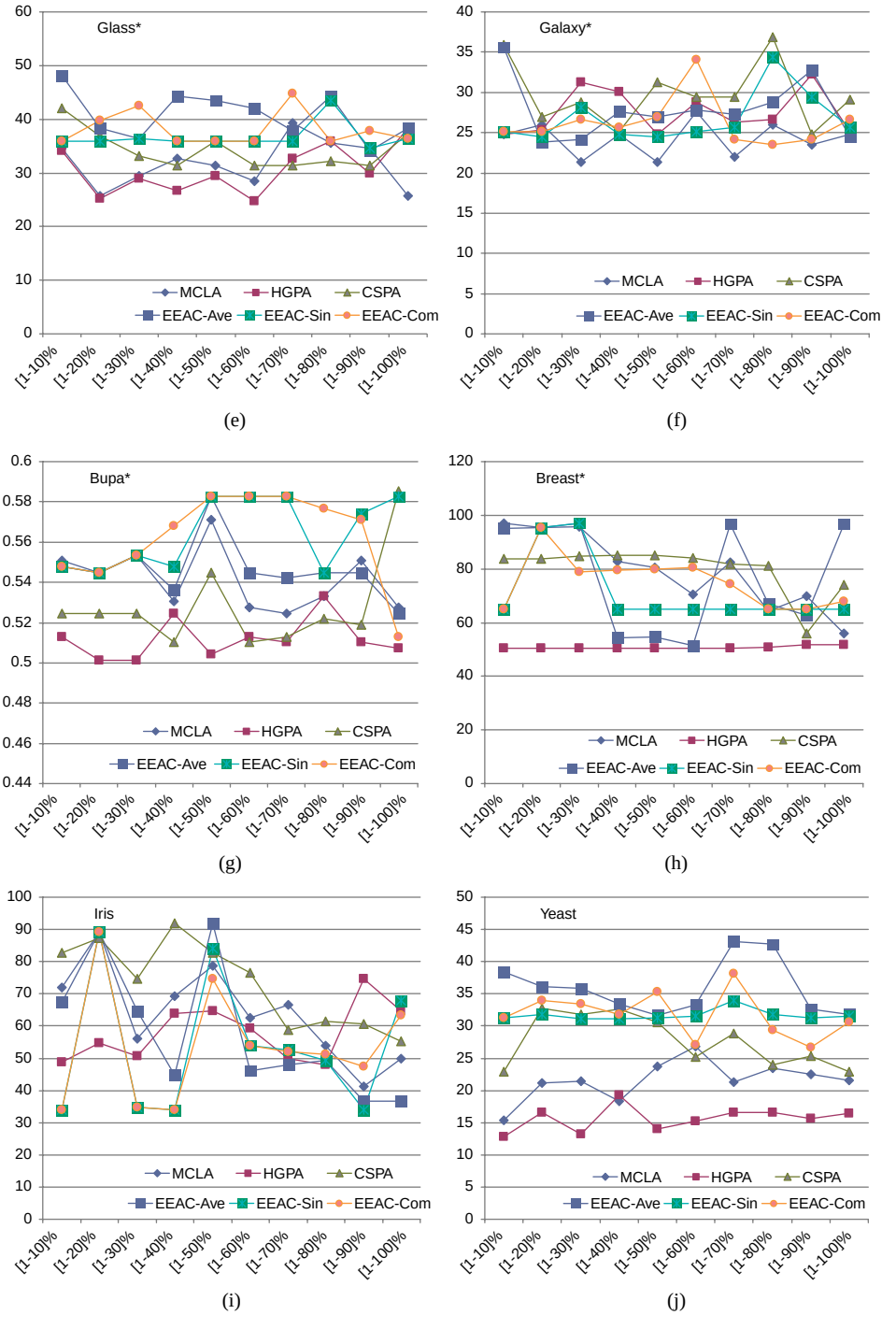


Figure 10. The effect of increasing the size of the ensemble with the most stable clusters applying different consensus functions over 10 different datasets.

handling the missing values in a dataset is an important problem in this field which can be discussed as a future work of this paper.

Acknowledgements

This work was supported, in part, by the Research Institute for ICT – ITRC grant program. We would like to appreciate the ITRC for this support.

Notes

1. Email: b_minaei@iust.ac.ir
2. Email: parvin@iust.ac.ir

References

Al-Razgan, M., & Domeniconi, C. (2006). Weighted clustering ensembles. *Proceedings of the SIAM international conference on data mining*, USA, (pp. 258–269).

Alizadeh, H., Minaei-Bidgoli, B., & Parvin, H. (2013). Optimizing fuzzy cluster ensemble in string representation. *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*, *27* (2), 1–22.

Alizadeh, H., Minaei-Bidgoli, B., & Parvin, H. (in press). Cluster ensemble selection based on a new cluster stability measure. *Intelligent Data Analysis*, *18*(3).

Alizadeh, H., Parvin, H., & Parvin, S. (2012). A framework for cluster ensemble based on a max metric as cluster evaluator. *IAENG International Journal of Computer Science, IJCS*, *39*(1), pp. 1–10.

Azimi, J., & Fern, X. (2009). Adaptive cluster ensemble selection. *Proceedings of the international joint conference on artificial intellegence, IJCAI*, USA, pp. 992–997.

Baumgartner, R., Somorjai, R., Summers, R., Richter, W., Ryner, L., & Jarmasz, M. (2000). Resampling as a cluster validation technique in fMRI. *Journal of Magnetic Resonance Imaging*, *11*, 228–231.

Ben-Hur, A., Elisseeff, A., & Guyon, I. (2002). A stability based method for discovering structure in clustered data. *Pasific symposium on biocomputing*, Hawaii, *Vol. 7*, (pp. 6–17).

Breckenridge, J. (1989). Replicating cluster analysis: Method, consistency and validity. *Multivariate Behavioral Research*, *24*(2), pp. 147–161.

Caruana, R., Niculescu-Mizil, A., Crew, G., & Ksikes, A. (2004). Ensemble selection from libraries of models. *Proceedings of the 21st international conference on machine learning*, Canada, p. 18.

Christou, I. T. (2011). Coordination of cluster ensembles via exact methods. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, *33*(2), 279–293.

Domeniconi, C., & Al-Razgan, M. (2009). Weighted cluster ensembles: Methods and analysis. *ACM Transactions on Knowledge Discovery from Data*, *2*(4), pp. 1–40.

Dudoit, S., & Fridlyand, J. (2003). Bagging to improve the accuracy of a clustering procedure. *Bioinformatics*, *19*(9), 1090–1099.

Estivill-Castro, V., & Yang, J. (2003). Cluster validity using support vector machines, DaWaK 2003, LNCS 2737 (pp. 244–256).

Faceli, K., de Carvalho, A. C., & de Souto, M. C. (2007). Multi-objective clustering ensemble. *International Journal of Hybrid Intelligent Systems*, *4*(3), 145–156.

Fern, X. Z., & Brodley, C. E. (2004). Solving cluster ensemble problems by bipartite graph partitioning. *International conference on machine learning*, Canada, p. 36.

Fern, X., & Lin, W. (2008). Cluster ensemble selection. *Statistical Analysis and Data Mining*, *1*(3), 128–141.

Fred, A. (2001). Finding consistent clusters in data partitions. In J. Kittler & F. Roli (Eds.), *Proceedings of 2nd international workshop multiple classifier systems*, UK, (pp. 309–318).

Fred, A., & Jain, A. K. (2002). Data clustering using evidence accumulation. *Proceedings of the 16th IAPR international conference on pattern recognition, ICPR02* , Quebec (pp. 276–280).

Fred, A., & Jain, A. K. (2005). Combining multiple clusterings using evidence accumulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(6), 835–850.

Fred, A., & Jain, A. K. (2006). Learning pairwise similarity for data clustering. *Proceedings of the 18th international conference on pattern recognition*, Hong Kong, pp. 925–928, *(ICPR'06)*.

Fred, A., & Lourenco, A. (2008). Cluster ensemble methods: From single clusterings to combined solutions. *Studies in Computational Intelligence (SCI)*, *126*, 3–30.

Fridlyand, J., & Dudoit, S. (2001). *Applications of resampling methods to estimate the number of clusters and to improve the accuracy of a clustering method*. Stat. Berkeley Tech Report No. 600.

Gionis, A., Mannila, H., & Tsaparas, P. (2007). Clustering aggregation. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *1*(1), p. 4.

Hadjitodorov, S. T., Kuncheva, L. I., & Todorova, L. P. (2006). Moderate diversity for better cluster ensembles. *Information Fusion*, *7*(3), 264–275.

Jain, A. K. (2009). Data clustering: 50 years beyond *k*-means. *Pattern Recognition Letters*, *31*(8), 651–666.

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice Hall, USA.

Kaufman, L., & Rosseeuw, P. J. (1990). *Finding groups in data: An introduction to cluster analysis*. John Wiley & Sons, Inc.

Kuncheva, L. I. (2005). *Combining pattern classifiers, methods and algorithms*. New York: Wiley.

Kuncheva, L. I., Hadjitodorov, S. T., & Todorova, L. P. (2006). Experimental comparison of cluster ensemble methods. *Proceedings of FUSION*, Italy, pp. 1–7.

Lange, T., Braun, M. L., Roth, V., & Buhmann, J. M. (2002). Stability-based model selection. *Advances in neural information processing systems*. Canada, pp. 617–624.

Lange, T., Roth, V., Braun, M. L., & Buhmann, J. M. (2004). Stability based validation of clustering solutions. *Neural Computation*, *16*(6), 1299–1323.

Law, M. H. C., Topchy, A. P., & Jain, A. K. (2004). Multiobjective data clustering. *Proceedings of IEEE conference on computer vision and pattern recognition*, USA, *Vol. 2*, (pp. 424–430).

Levine, E., & Domany, E. (2001). Resampling method for unsupervised estimation of cluster validity. *Neural Computation*, *13*, 2573–2593.

Luxburg, U. V., & Ben-David, S. (2005). Towards a statistical theory of clustering. *Pascal workshop on statistics and optimization of clustering*, UK, p. 10.

Meila, M. (2003). Comparing clusterings by the variation of information. *Proceedings of computational learning theory*, USA, 173–187.

Minaei-Bidgoli, B., Parvin, H., Alinejad, R. H., Alizadeh, H., & Punch, W. F. (2011). Effects of resampling method and adaptation on clustering ensemble efficacy. *Artificial Intelligence Review*, pp. 1–22.

Minaei-Bidgoli, B., Topchy, A., & Punch, W. F. (2004). Ensembles of partitions via data resampling. *Proceedings of the international conference on information technology, ITCC 04*, USA, pp. 188–192.

Mok, P. Y., Huang, H. Q., Kwok, Y. L., & Au, J. S. (2012). A robust adaptive clustering analysis method for automatic identification of clusters. *Pattern Recognition*, *45*, 3017–3033.

Newman, C. B. D. J., Hettich, S., & Merz, C. (1998). *UCI repository of machine learning databases*, http://www.ics.uci.edu/~mlearn/MLSummary.html

Parvin, H., Minaei-Bidgoli, B., & Alizadeh, H. (2011). A new clustering algorithm with the convergence proof, *KES 2011, Part I: LNAI 6881* (pp. 21–31).

Punera, K., & Ghosh, J. (2008). Consensus-based ensembles of soft clusterings. *Applied Artificial Intelligence*, *22*(7), 780–810.

Rakhlin, A., & Caponnetto, A. (2007). Stability of *k*-means clustering. *Advances in neural information processing systems*. Cambridge, MA: MIT Press, *19*: 1121.

Redding, N. J. (1999). *Estimating the parameters of the K distribution in the intensity domain*. Report DSTO-TR-0839, DSTO Electronics and Surveillance Laboratory, South Australia (pp. 60).

Roth, V., Braun, M. L., Lange, T., & Buhmann, J. M. (2002). Stability-based model order selection in clustering with applications to gene expression data, *ICANN 2002, LNCS 2415* (pp. 607–612).

Roth, V., & Lange, T. (2003). Feature selection in clustering problems. *Advances in neural information processing systems*, NIPS03, Canada, p. 16.

Roth, V., Lange, T., Braun, M., & Buhmann, J. (2002). A resampling approach to cluster validation. *International conference on computational statistics, COMPSTAT 30*, Germany, pp. 123–128.

Shamir, O., & Tishby, N. (2008). Cluster stability for finite samples. *Advances in Neural Information Processing Systems*, *20*, 1297–1304.

Shamir, O., & Tishby, N. (2007). Cluster stability for finite samples. *Twenty-first annual conference on neural information processing systems (NIPS07)*, Canada, pp. 1297–1304.

Singh, V., Mukherjee, L., Peng, J., & Xu, J. (2010). Ensemble clustering using semidefinite programming with applications. *Machine Learning*, *79*, 177–200.

Strehl, A., & Ghosh, J. (2002). Cluster ensembles – A knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, *3*, 583–617.

Tan, P. N., Steinbach, M., & Kumar, V. (2005). *Introduction to data mining* (1st ed.): Addison-Wesley Longman Publishing Co. Inc, Boston, MA, USA.

Topchy, A., Jain, A., & Punch, W. (2004). A mixture model for clustering ensembles. *SIAM international conference on data mining*, USA (pp. 379–390).

Topchy, A., Jain, A. K., & Punch, W. F. (2005). Clustering ensembles: Models of consensus and weak partitions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(12), 1866–1881.

Vega-Ponz, S., & Ruiz-Shulcloper, J. (2011). A survey of clusering ensemble algorithms. *International Journal of Pattern Recognition and Artificial Intelligence*, *25*(3), 337–372.

Wang, H., Shan, H., & Banerjee, A. (2009). Bayesian cluster ensembles. *SIAM international conference on data mining*, USA (pp. 211–222).

Wang, P, Domeniconi, C, & Laskey, K (2010). Nonparametric Bayesian clustering ensembles. *Machine learning and knowledge discovery in databases*, Spain, LNCS, 6323.