Data Min Knowl Disc (2013) 27:259–289
DOI 10.1007/s10618-012-0290-x

Cluster ensemble selection based on relative validity indexes

M. C. Naldi · A. C. P. L. F. Carvalho · R. J. G. B. Campello

Received: 28 July 2011 / Accepted: 1 September 2012 / Published online: 18 September 2012

Abstract Cluster ensemble aims at producing high quality data partitions by combining a set of different partitions produced from the same data. Diversity and quality are claimed to be critical for the selection of the partitions to be combined. To enhance these characteristics, methods can be applied to evaluate and select a subset of the partitions that provide ensemble results similar or better than those based on the full set of partitions. Previous studies have shown that this selection can significantly improve the quality of the final partitions. For such, an appropriate evaluation of the candidate partitions to be combined must be performed. In this work, several methods to evaluate and select partitions are investigated, most of them based on relative clustering validity indexes. These indexes select the partitions with the highest quality to participate in the ensemble. However, each relative index can be more suitable for particular data conformations. Thus, distinct relative indexes are combined to create a final evaluation that tends to be robust to changes in the application scenario, as the majority of the combined indexes may compensate the poor performance of some individual indexes. We also investigate the impact of the diversity among partitions used for the ensemble. A comparative evaluation of results obtained from an extensive collection of experiments involving state-of-the-art methods and statistical tests is presented. Based on the

Responsible editor: Charu Aggarwal.

M. C. Naldi (✉)
Federal University of Viçosa-UFV, Post Box 22, Rio Paranaíba, MG, CEP 38.810-000, Brazil
e-mail: murilocn@ufv.br

A. C. P. L. F. Carvalho · R. J. G. B. Campello
University of São Paulo-USP, Post Box 668, São Carlos, SP, CEP 13560-970, Brazil
e-mail: andre@icmc.usp.br

R. J. G. B. Campello
e-mail: campello@icmc.usp.br

obtained results, a practical design approach is proposed to support cluster ensemble selection. This approach was successfully applied to real public domain data sets.

Keywords Cluster ensemble selection · Combination · Relative validity indexes · Evaluation · Diversity

1 Introduction

Data clustering is a fundamental conceptual problem in data mining. Clustering algorithms aim at partitioning a data set by looking for a finite collection of clusters according to similarities between its objects. Such clusters are supposed to describe the underlying structure (if any) of the data. There are several clustering algorithms reported in the literature. It is well known that different clustering algorithms, or the same algorithm configured with different parameter values, may produce different partitions. However, a single consensus partition is expected in many applications. Cluster ensembles can combine different partitions into a single consensus one. By doing so, they can improve the quality of the final clusters obtained, being robust to distinct scenarios of application, including those involving noise and outliers (Strehl and Ghosh 2002; Topchy et al. 2004; Fred and Jain 2005; Ayad and Kamel 2008). They have also been frequently employed in applications that require the use of existing knowledge about the data set (Bollacker and Ghosh 1998) and in distributed clustering as well (Tumer and Agogino 2008).

A typical cluster ensemble technique initially produces a large set of base clustering solutions and then combines these solutions into a consensus clustering solution. Most of the ensemble techniques combine clustering solutions resulting from exclusive partitional clustering algorithms, i.e., exclusive (so-called non-overlapping) partitions (Jain and Dubes 1988). In order to formally define exclusive partitions, consider a data set $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, composed of n input vectors $\mathbf{x}_j$, each of which is described by a attributes or features. An exclusive partition is a collection $\pi = \{C_1, C_2, \ldots, C_k\}$ of k clusters or subsets C_i in which $C_1 \cup C_2 \cup \ldots \cup C_k = X$, $C_i \neq \emptyset$, and $C_i \cap C_l = \emptyset$ for $i \neq l$. Partitions generated for the sole purpose of being combined are named *base partitions* and the methods used to combine them into an ensemble are known as *consensus functions*. There are many consensus functions in the clustering literature. Some of them are based on co-association among base partitions (Greene et al. 2004; Fred and Jain 2005), whereas others employ graph partitioners (Strehl and Ghosh 2002; Fern and Brodley 2004) or cumulative voting (Tumer and Agogino 2008; Ayad and Kamel 2008).

According to many authors (e.g. Greene et al. 2004; Kuncheva and Hadjitodorov 2004; Hadjitodorov et al. 2006; Handl and Knowles 2007; Fern and Lin 2008), the diversity and the quality of the set of base partitions are critical characteristics for a successful ensemble. Different approaches have been adopted to produce base partitions with these characteristics. In (Strehl and Ghosh 2002; Weingessel et al. 2003), each base partition is obtained by running one out of a set of different clustering algorithms on the same data set. These ensembles are named heterogeneous ensembles. Another approach, named homogeneous ensemble, uses different runs of the same

clustering algorithm, but from distinct initializations and parameter values (Topchy et al. 2004; Kasturi and Acharya 2004; Fred and Jain 2005; Hadjitodorov et al. 2006). The production of different base partitions using data sampling (Monti et al. 2003; Dudoit and Fridlyand 2003), distinct subsets of attributes (Strehl and Ghosh 2002), and noise addition (Dimitriadou et al. 1999) have also been reported.

Although diversity between base partitions is considered to be a relevant issue in cluster ensemble, only a few works have investigated the impact of diversity on the quality of the consensus partition (Kuncheva and Hadjitodorov 2004; Hadjitodorov et al. 2006; Hadjitodorov and Kuncheva 2007). Fewer works have shed some light on how the quality of the base partitions affects the consensus partition (Fern and Lin 2008). Recently, new techniques have been developed to improve clustering ensembles by selecting a subset of the base partitions using diversity and quality (Fern and Lin 2008; Azimi and Fern 2009). These techniques are known as *cluster ensemble selection* (CES) and their main purpose is to form an ensemble with a subset of the base partitions, here named *selected set*, which performs equally to or better than ensembles of the full set of base partitions, also known as *full ensembles*. Recent results show that CES may achieve significant performance improvements when compared to full ensembles (Fern and Lin 2008; Azimi and Fern 2009). Although these results seem promising, CES has so far been investigated with respect to no more than a couple of consensus functions, which may not be enough to represent the variety of consensus functions proposed in the literature (Strehl and Ghosh 2002; Fred and Jain 2005; Kuncheva et al. 2006; Tumer and Agogino 2008). Moreover, these works investigated fewer than a dozen data sets, which may be insufficient to access the full potential of this approach.

Relative clustering validity (or validation) indexes have been successfully used during decades to evaluate the relative quality of partitions (Milligan and Cooper 1985; Halkidi et al. 2001; Vendramin et al. 2009, 2010). We believe that such indexes can be used to evaluate base partitions in the CES process, which, to the best of our knowledge, has not been tried before. Once the evaluations have been performed, it is possible to select the most appropriate base partitions to construct the ensemble. However, validation indexes are each endowed with particular features that may cause an index to outperform others in specific classes of problems (Milligan and Cooper 1985; Vendramin et al. 2009, 2010). Therefore, it may be difficult for the user to choose a specific index among a large variety of possibilities. One possible solution is to combine the evaluations resulting from a set of relative validation indexes into a single evaluation. The main rationale behind the *combination of relative indexes* (CRI) is to obtain an evaluation robust to changes in the application scenario, as the majority of the combined indexes may counterbalance the poor performance of individual indexes. In the context of CES, CRI has an additional important property: it may lead to more diverse selected partitions.

In this paper, we investigate the use of relative validation indexes in CES, applying them to five well known consensus functions: the evidence accumulation algorithms EAC-SL and EAC-AL by Fred and Jain (2005), and the graph-based algorithms CSPA, HGPA, and MCLA by Strehl and Ghosh (2002). A collection of relative validation indexes is used to evaluate the clustering results for 491 data sets (484 artificial and 7 real). The evaluations obtained from all indexes are combined using 3 different

combination methods. We show that relative validation indexes can be successfully used in CES, either combined or individually, through a comparative evaluation of results involving a well known CES algorithm, an extensive collection of experiments, and statistical tests. Additionally, a method to measure diversity in the selected set is used to investigate the impact of diversity on the ensemble. Based on these results, it is possible to indicate in which scenarios the use of CES is more effective than the full ensemble and in which scenarios it is not. Moreover, for cases in which no previous information about the application scenario is available, we propose a practical method to compare and select ensemble partitions using relative validation indexes. This method is successfully applied to real public domain data sets.

This paper is organized as follows. In Sect. 2, we describe previous works using CES methods found in the literature. In Sect. 3, we review the relative clustering validity indexes that will be used in the proposed CES methods. Section 4 presents the CES methods and a methodology for practical applications proposed in this paper. In Sect. 5, the CES methods are experimentally investigated. Finally, the main conclusions are summarized in Sect. 6.

2 Related work

Several works in the clustering literature suggest that diversity and quality of the base partitions are crucial for a successful cluster ensemble (Kuncheva and Hadjitodorov 2004; Hadjitodorov et al. 2006; Fern and Lin 2008; Azimi and Fern 2009). Kuncheva and Hadjitodorov (2004) proposed the overproduction of clusters in base partitions as a method to enhance diversity in ensembles. Based on experimental results, the authors observed that diverse ensembles tend to be more accurate than non-diverse ones. Later, Hadjitodorov et al. (2006) observed that the quality of the ensemble did not seem to grow monotonically by increasing the diversity among base partitions and, based on this observation, they suggested the use of ensembles involving a set of base partitions with moderate diversity.

In the supervised learning paradigm, it has been shown that the selection of a subset of classifiers based on their quality and diversity may result in a performance similar to or better than that obtained when the whole set of classifiers is used (Margineantu and Dietterich 1997; Caruana et al. 2006). Inspired by those results, Fern and Lin (2008) proposed the combination of partition quality and diversity for CES, using the *Sum of the Normalized Mutual Information* (SNMI) measure introduced in (Strehl and Ghosh 2002). In particular, given a set Π of r partitions denoted by $\Pi = \{\pi_1, \pi_2, \ldots, \pi_r\}$, the SNMI measure between a partition π and the set Π is denoted as:

$$SNMI(\pi, \Pi) = \sum_{i=1}^{r} NMI(\pi, \pi_i) \tag{1}$$

where $NMI(\pi, \pi_i)$ is the *Normalized Mutual Information* (NMI) between partition π and the ith partition in the set Π, which is computed by considering partitions π and π_i as two random variables and the corresponding cluster labels for the data objects as observed values of these variables (Strehl and Ghosh 2002). Intuitively, a partition

π maximizing SNMI maximizes the information it shares with all the partitions in Π and can be considered to capture its general trend.

Fern and Lin (2008) suggested that the quality of a base partition is proportional to its SNMI with respect to the set of base partitions and that the diversity of the selected set is inversely proportional to the SNMI between each of its members and the other members of this set. The most successful method proposed by Fern and Lin (2008) is named *Cluster And Select* (CAS), which combines quality and diversity by first grouping base partitions according to their similarities and then selecting for the ensemble the partitions with the highest SNMI within each cluster obtained. Results obtained with CAS indicate that explicitly considering both quality and diversity in CES may result in statistically significant SNMI improvements over full ensembles.

Recently, Azimi and Fern (2009) investigated the use of CES to avoid consensus partitions excessively different from the base partitions they result from. According to the authors, such consensus partitions represent unstable solutions, which can, however, be improved by means of a proper selection of base partitions. Particularly, Azimi and Fern proposed to first select the base partitions most similar to the consensus partition obtained from the full ensemble and, then, generate a new consensus partition from the selected set. They showed that this procedure can result in partitions with enhanced SNMI.

It is worth noticing that the use of SNMI to measure quality in CES needs caution. SNMI reflects the amount of information a consensus partition captures from a given set of base partitions (Strehl and Ghosh 2002). Therefore, the more similar the base partitions are to the consensus partition, the higher is the SNMI value. Thus, maximum SNMI value is reached when the base partitions and the consensus partition are identical. As a result, a CES method that maximizes SNMI may exhibit a tendency to build selected sets with similar or identical base partitions, which is not a desirable characteristic for cluster ensembles, as there is no point in combing such base partitions.[1] Moreover, desirable characteristics for partitions, such as clusters external isolation and internal cohesion (i.e., compaction and separation) (Halkidi et al. 2001), are not considered by the SNMI measure.

Finally, it is also worth noticing that, although the above-mentioned works did provide important contributions to the understanding of CES, their experimental results involved fewer than a dozen data sets and no more than a couple of consensus functions, which represent a small fraction of the consensus functions reported in the literature (Strehl and Ghosh 2002; Dimitriadou 2003; Kuncheva et al. 2006; Tumer and Agogino 2008).

3 Review of relative validity indexes

Many different relative clustering validity measures are very useful in practice as quantitative criteria for evaluating the quality of data partitions (Vendramin et al. 2010). This evaluation is based on desirable cluster properties, such as external isolation and

[1] A need to counterbalance this tendency is the reason Fern and Lin (2008) also use diversity among base partitions as an additional selection criterion.

internal cohesion (Halkidi et al. 2001). In contrast to external validity indexes, which measure the level of *agreement* between a pair of partitions (Jain and Dubes 1988), relative validity indexes are able to compare the *quality* of two or more partitions of a given data set in a relative way, using solely the data themselves.[2] Revising the rich literature on relative validation indexes is out of the scope of this paper. Instead, we chose six relative validation indexes to evaluate the quality of base partitions and enable the selection of the most promising ones for an ensemble. These indexes have linear asymptotic computational complexity in relation to the number of objects of the evaluated partition ($O(n)$), having minor impact on the overall computational cost of CES. Furthermore, they have shown to be capable of discriminating between high and low quality partitions in hundreds of experiments and data sets (Vendramin et al. 2009, 2010). In this section, these indexes are briefly described. For detailed reviews of these and others relative validation indexes, the reader may refer to (Milligan and Cooper 1985; Halkidi et al. 2001; Vendramin et al. 2009, 2010) and references therein.

3.1 Simplified Silhouette (SS)

A well-known index that is based on geometrical considerations about compactness and separation of clusters is the Silhouette Width Criterion (Rousseeuw 1987). However, the original index depends on the computation of distances between all objects. This computation can be simplified by using distances between objects and cluster centroids, originating the index called Simplified Silhouette (Hruschka et al. 2004a). In order to define this index, let us consider that the jth object of the data set, $\mathbf{x}_j$, belongs to a given cluster $C_p \in \{C_1, \cdots, C_k\}$, where k is the number of clusters in a given partition. Next, let the dissimilarity between the jth object and the centroid of its cluster C_p be denoted by $a_{p,j}$. Also, let $b_{p,j}$ be the dissimilarity between the jth object and the centroid of its closest neighboring cluster. Then, the simplified silhouette of the individual object $\mathbf{x}_j$ is defined as[3]:

$$s_{\mathbf{x}_j} = \frac{b_{p,j} - a_{p,j}}{\max\{a_{p,j}, b_{p,j}\}} \tag{2}$$

where the denominator is just a normalization term. The higher $s_{\mathbf{x}_j}$, the better the assignment of $\mathbf{x}_j$ to cluster C_p. If C_p is a singleton, i.e., if it is constituted uniquely by $\mathbf{x}_j$, then it is assumed by convention that $s_{\mathbf{x}_j} = 0$ (Kaufman and Rousseeuw 1990). This prevents the SS index, defined as the average of $s_{\mathbf{x}_j}$ over $j = 1, 2, \cdots, n$, i.e.

$$\text{SS} = \frac{1}{n} \sum_{j=1}^{n} s_{\mathbf{x}_j} \tag{3}$$

[2] Notice, therefore, that *relative index* here refers to *internal* validation measures which are also relative (Jain and Dubes 1988).

[3] Since the Simplified Silhouette is based on cluster centroids, in principle it can be applied to data sets described by numerical attributes only. For data sets with categorical attributes, either centroids must be replaced with medoids (cluster representatives) or the original Silhouette Width Criterion must be used.

to elect the trivial solution $k = n$ (with each object of the data set forming a cluster on its own) as the best one. Clearly, the best partition is expected to be selected when SS is maximized, which implies minimizing the intra-group distance ($a_{p,j}$) while maximizing the inter-group distance ($b_{p,j}$).

3.2 Alternative Simplified Silhouette (ASS)

A variant of the simplified silhouette criterion can be obtained by replacing Eq. (2) with the following alternative definition of the silhouette of an individual object (Hruschka et al. 2004b):

$$s_{\mathbf{x}_j} = \frac{b_{p,j}}{a_{p,j} + \epsilon} \tag{4}$$

where ϵ is a small constant (e.g. 10^{-6} for normalized data) used to avoid division by zero when $a_{p,j} = 0$. Note that the rationale behind Eq. (4) is the same as that of (2), in the sense that both favor larger values of $b_{p,j}$ and lower values of $a_{p,j}$. The difference lies in how they favor, linearly in (2) and non-linearly in (4).

3.3 Calinski–Harabasz (VRC)

The Variance Ratio Criterion (Calinski and Harabasz 1974) evaluates the quality of a data partition as:

$$\text{VRC} = \frac{\text{trace}(\mathbf{B})}{\text{trace}(\mathbf{W})} \times \frac{n-k}{k-1} \tag{5}$$

where $\mathbf{W}$ and $\mathbf{B}$ are the $a \times a$ within-group and between-group dispersion matrices[4], respectively, defined as:

$$\mathbf{W} = \sum_{l=1}^{k} \mathbf{W}_l \tag{6}$$

$$\mathbf{W}_l = \sum_{\mathbf{x}_i \in C_l} (\mathbf{x}_i - \overline{\mathbf{x}}_l)(\mathbf{x}_i - \overline{\mathbf{x}}_l)^T \tag{7}$$

$$\mathbf{B} = \sum_{l=1}^{k} n_l (\overline{\mathbf{x}}_l - \overline{\mathbf{x}})(\overline{\mathbf{x}}_l - \overline{\mathbf{x}})^T \tag{8}$$

where n_l is the number of objects assigned to the lth cluster (C_l), $\overline{\mathbf{x}}_l$ is the a-dimensional vector of sample means within that cluster (cluster centroid) and $\overline{\mathbf{x}}$ is the a-dimensional vector of overall sample means (data centroid or grand mean of the data). As such, the within-group and between-group dispersion matrices sum up to the scatter matrix of the data set, i.e., $\mathbf{T} = \mathbf{W} + \mathbf{B}$, where $\mathbf{T} = \sum_{i=1}^{n} (\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^T$. The trace of matrix

[4] Recall that a is the number of attributes that describe the data objects.

W is the sum of the within-cluster variances (its diagonal elements). Analogously, the trace of **B** is the sum of the between-cluster variances. As a consequence, compact and separated clusters are expected to have small trace(**W**) values and large trace(**B**) values. Hence, the better the data partition the greater the value of the ratio between trace(**B**) and trace(**W**). The normalization term $(n-k)/(k-1)$ prevents this ratio to increase monotonically with the number of clusters, thus making VRC an optimization (maximization) criterion with respect to k.

3.4 PBM

Another criterion, named PBM (Pakhira et al. 2004), is also based on the within-group and between-group distances:

$$\text{PBM} = \left(\frac{1}{k}\frac{E_1}{E_K}D_K\right)^2 \tag{9}$$

where E_1 is a constant that denotes the sum of distances between the objects and the grand mean of the data, i.e. $E_1 = \sum_{i=1}^{n} ||\mathbf{x}_i - \overline{\mathbf{x}}||$, $E_K = \sum_{l=1}^{k} \sum_{\mathbf{x}_i \in C_l} ||\mathbf{x}_i - \overline{\mathbf{x}}_l||$ represents the sum of within-group distances, and $D_K = \max_{l,m=1,\cdots,k} ||\overline{\mathbf{x}}_l - \overline{\mathbf{x}}_m||$ is the maximum distance between group centroids. According to this equation, the best partition should be indicated when PBM is maximized, which implies maximizing D_K while minimizing E_K.

3.5 Davies–Bouldin (DB)

The Davies–Bouldin index (Davies and Bouldin 1979) is somewhat related to VRC, since it is also based on a ratio involving within-group and between-group distances. Specifically, the index evaluates the quality of a given data partition as follows:

$$\text{DB} = \frac{1}{k}\sum_{l=1}^{k} D_l \tag{10}$$

where $D_l = max_{l \neq m}\{D_{l,m}\}$. Term $D_{l,m}$ is the within-to-between cluster spread for the lth and mth clusters, given by $D_{l,m} = (\overline{d}_l + \overline{d}_m)/d_{l,m}$, where $\overline{d}_l$ and $\overline{d}_m$ are the average within-group distances for the lth and the mth clusters, respectively, and $d_{l,m}$ is the inter-group distance between these clusters. These distances are defined as $\overline{d}_l = (1/n_l)\sum_{\mathbf{x}_i \in C_l} ||\mathbf{x}_i - \overline{\mathbf{x}}_l||$ and $d_{l,m} = ||\overline{\mathbf{x}}_l - \overline{\mathbf{x}}_m||$, where $\|\cdot\|$ is a norm (e.g. Euclidean).

Term D_l represents the worst case within-to-between cluster spread involving the lth cluster. Minimizing D_l for all clusters clearly minimizes the Davies–Bouldin index. Hence, good partitions, composed of compact and separated clusters, are distinguished by small values of DB in (10).

3.6 Dunn

The Dunn index (Dunn 1974) is another validity criterion based on geometrical measures of cluster compactness and separation. It is defined as:

$$\mathrm{DN} = \min_{\substack{p,q \in \{1,\ldots,k\} \\ p \neq q}} \left\{ \frac{\delta_{C_p,C_q}}{\max\limits_{l \in \{1,\ldots,k\}} \Delta_{C_l}} \right\} \tag{11}$$

where Δ_{C_l} is the *diameter* of the lth cluster and δ_{C_p,C_q} is the *set distance* between clusters C_p and C_q. The original definitions of *diameter* and *set distance* in (11) were generalized in (Bezdek and Pal 1998), giving rise to 17 variants of the original Dunn index. One of theses variants is used here in this paper, in which the set distance across clusters C_p and C_q is defined as $\delta_{C_p,C_q} = ||\overline{\mathbf{x}}_p - \overline{\mathbf{x}}_q||$, whereas the diameter Δ_{C_l} of a given cluster l is calculated by $\Delta_{C_l} = \frac{2}{n_l} \sum_{\mathbf{x}_i \in C_l} ||\mathbf{x}_i - \overline{\mathbf{x}}_l||$. Note that the definitions of Δ_{C_l} and δ_{C_p,C_q} are directly related to the concepts of within-group and between-group distances, respectively. Bearing this in mind, it is easy to verify that partitions composed of compact and separated clusters are distinguished by large values of DN in (11).

4 Cluster Ensemble Selection (CES)

In this work, we investigate the use of validation indexes in CES. More specifically, the six relative validation indexes presented in Sect. 3 were chosen to evaluate the quality of the obtained base partitions and to enable the selection of the finest base partitions for the ensemble.

4.1 Single index selection (SIS)

This method is named *Single Index Selection* (SIS) and is described by Algorithm 1, where Π_b is the original (full) collection of candidate base partitions, s is the number of base partitions to be selected for the ensemble, *index* is the chosen relative validation index, Π_s is the resulting selected set, and $|\cdot|$ stands for cardinality.

Algorithm 1 Single Index Selection (SIS)

Require: A collection of candidate base partitions, Π_b, the number of base partitions to be selected, s ($s < |\Pi_b|$), and a relative validation index, *index*;

1: **repeat**
2: $t \leftarrow \arg \max_{m \in \{1,\ldots,|\Pi_b|\}} f(m) = index(\pi_m)$;
3: insert π_t into Π_s;
4: remove π_t from Π_b;
5: **until** $|\Pi_s| = s$;
6: **return** Π_s;

If the relative validation index distinguishes better partitions by smaller values (e.g. Davies–Bouldin), i.e., if it is a minimization (rather than a maximization) index, then the max operator (Step 2 of Algorithm 1) should be replaced by min.

Provided that *index* demands $O(I)$ time to be computed, the asymptotic complexity of Algorithm 1 is $O(|\Pi_b|\,(I + s))$, but it can also be implemented to run in $O(|\Pi_b|\,(I + \log|\Pi_b|))$ time by using a sorting procedure.

The single index selection will be experimentally evaluated in Sect. 5 for each relative validation index reviewed in Sect. 3. In Sects. 4.2 and 4.4, we propose to combine these indexes to evaluate and select partitions.

4.2 Combination of relative indexes (CRI)

Each validation index has its own peculiarities, being more (or less) adequate for specific data conformations. The choice of a particular index from the large variety of existing indexes is not an easy task, especially because, in most practical cases, there is no information about their expected performances in the application scenario in hand. In fact, a few papers have assessed the performance of relative validation indexes and these studies have involved a particular class of data sets (Milligan and Cooper 1985; Vendramin et al. 2009, 2010). Moreover, evaluations based on a single validation index in CES may reduce the diversity of the ensemble members, since the best evaluated partitions are likely to have similar characteristics. A possible alternative to overcome these drawbacks is to average the individual evaluations provided by a committee of relative validation indexes (CRI, as defined in Sect. 1), expecting that a good performance of most indexes will compensate the weak performance of some. In this context, it is expected that the evaluations made by a CRI tend to be more robust to variations in the application scenarios than the use of a single index. Furthermore, when adopted in CES, the CRI strategy may result in selected partitions with higher diversity than those obtained by using a single validation index, as each member of the CRI evaluates the partitions in a different way.

In order to define the CRI methods proposed in this paper, let us consider a function $rank$ that returns the rank of partition π_m among the base partitions in the set Π_b when evaluated by the uth index of the combination, referred to here as $index_u$. For example, if π_m is a base partition from Π_b and it is the best partition according to the uth index, then $rank(index_u, \pi_m, \Pi_b) = 1$. If π_m is the second best partition, then $rank(index_u, \pi_m, \Pi_b) = 2$ and so on. Moreover, let us consider a function $diversity$, which returns the mean pairwise dissimilarity between a partition π_i and a set of partitions Π, calculated as follows:

$$diversity(\pi_i, \Pi) = 1 - \sum_{\pi_j \in \Pi, j \neq i} \frac{s(\pi_i, \pi_j)}{|\Pi| - 1} \tag{12}$$

where $|\cdot|$ stands for set cardinality and $s(\pi_i, \pi_j)$ is a given measure of similarity between partitions π_i and π_j. In this work, the Jaccard external index (Jaccard 1908) is used to calculate the similarity between partitions. This very simple and intuitive index compares two partitions (π_i and π_j) of the same data set as:

$$s(\pi_i, \pi_j) = \frac{a}{a+b+c} \tag{13}$$

where

- a: Number of pairs of data objects belonging to the same cluster in π_i and to the same cluster in π_j.
- b: Number of pairs of data objects belonging to the same cluster in π_i and to different clusters in π_j.
- c: Number of pairs of data objects belonging to different clusters in π_i and to the same cluster in π_j.

The Jaccard index has been chosen for being well-known and for resulting in similarities with values in the interval [0, 1], which makes it easier their interpretation and conversion to dissimilarities. However, other external indexes—e.g. the Adjusted Rand (Hubert and Arabie 1985) or NMI indexes (Strehl and Ghosh 2002)—can also be used in Eq. (12). It is important to note that, although the Jaccard index is usually associated with external information about the data set (so-called "ground truth" or "ideal partition"), no information external to the data set and the set of base partitions is used by any method proposed in this paper.

We propose three CRI methods for CES. The first one, named *Sum of Ranks* (SR), builds a ranking for each validation index based on the evaluation of all base partitions. The sum of the individual rankings is then calculated for each base partition (over the different indexes) and the base partitions with the lowest sums are selected for the ensemble. Algorithm 2 summarizes the SR method, where s is the number of base partitions to be selected for the ensemble, v is the number of validation indexes combined, Π_s is the resulting selected set, and $|\cdot|$ stands for cardinality.

Algorithm 2 Sum of Ranks (SR)

Require: A collection of candidate base partitions, Π_b, the number of base partitions to be selected, s ($s < |\Pi_b|$), and a collection of relative validation indexes, $index_i$ ($i = 1, \ldots, v$);

1: **repeat**
2: $t \leftarrow \arg \min_{m \in \{1,\ldots,|\Pi_b|\}} f(m) = \sum_{i=1}^{v} rank(index_i, \pi_m, \Pi_b)$;
3: insert π_t into Π_s;
4: remove π_t from Π_b;
5: **until** $|\Pi_s| = s$;
6: **return** Π_s;

Provided that the most computationally demanding index runs in $O(I_{max})$ time and that the rankings of the base partitions with respect to the different indexes are precomputed, Algorithm 2 can be implemented to run in $O(v\,(|\Pi_b| I_{max} + |\Pi_b| \log |\Pi_b|))$ time based on multiple sorting procedures.

Like the SR method, the second method builds a ranking of the base partitions for each validation index. However, the rankings are not summed. Instead, the best ranked partitions, according to each index, are selected. This method is named *Best Rank Position* (BRP) and is presented in Algorithm 3. Its complexity is the same as that of Algorithm 2, i.e., $O(v\,(|\Pi_b| I_{max} + |\Pi_b| \log |\Pi_b|))$.

Algorithm 3 Best Rank Position (BRP)

Require: A collection of candidate base partitions, Π_b, the number of base partitions to be selected, s ($s < |\Pi_b|$), and a collection of relative validation indexes, $index_i$ ($i = 1, \ldots, v$);
1: $i \leftarrow 0$;
2: **repeat**
3: $i \leftarrow i + 1$;
4: $t \leftarrow \arg \min_{m \in \{1,\ldots,|\Pi_b|\}} f(m) = rank(index_i, \pi_m, \Pi_b)$;
5: insert π_t into Π_s;
6: remove π_t from Π_b;
7: $i \leftarrow i \bmod v$;
8: **until** $|\Pi_s| = s$;
9: **return** Π_s;

The third method is named *Sum of Ranks with Diversity* (SRD). Although the evaluation of partitions and their rankings are calculated precisely as in the SR method, the sum of ranks for each partition is weighted by the diversity between the partition and the set of base partitions (computed according to Eq. (12)). Next, partitions are selected for the ensemble in ascending order of weighted rank sums. The objective of this CRI is to enhance the diversity of the ensemble, in comparison to the SR method. The method is summarized in Algorithm 4.[5]

Algorithm 4 Sum of Ranks with Diversity (SRD)

Require: A collection of candidate base partitions, Π_b, the number of base partitions to be selected, s ($s < |\Pi_b|$), and a collection of relative validation indexes, $index_i$ ($i = 1, \ldots, v$);
1: **repeat**
2: $t \leftarrow \arg \min_{m \in \{1,\ldots,|\Pi_b|\}} f(m) = (1 - diversity(\pi_m, \Pi_b)) * \sum_{i=1}^{v} rank(index_i, \pi_m, \Pi_b)$;
3: insert π_t into Π_s;
4: remove π_t from Π_b;
5: **until** $|\Pi_s| = s$;
6: **return** Π_s;

In terms of complexity, the difference with respect to SR in Algorithm 2 is that SRD in Algorithm 4 needs to compute *diversity* for every partition. Provided that the external index used by function *diversity* runs in $O(E)$ time (for a single pair of partitions) and given that computing *diversity* for all base partitions according to Eq. (12) demands the external index to be computed for all pairs of partitions, the complexity of Algorithm 4 can be written as $O(v\,(|\Pi_b| I_{max} + |\Pi_b| \log |\Pi_b|) + E|\Pi_b|^2)$. In this paper we have used the Jaccard index in (13), for which E is $O(n^2)$. If computing time is a concern, however, a linear time external index (e.g. NMI) is recommended.

As a summary of the proposed CES methods, it follows that the rationale behind both SR and SRD (Algorithms 2 and 4) is to select those partitions with the best *average* ranks computed according to multiple validity criteria; the difference is that the

[5] Notice that smaller weight values are assigned to more diverse partitions because lower ranks are associated with better quality partitions and, accordingly, the selected partitions should be those that minimize the weighted sum of ranks.

latter uses an weighted average that favors more diverse partitions. Differently, BRP (Algorithm 3) does not take averages into account. Instead, it selects those partitions that are top ranked according to each criterion individually.

4.3 Diversity selection

Based on the methods proposed in (Hadjitodorov et al. 2006; Fern and Lin 2008; Azimi and Fern 2009), we also investigate the explicit use of diversity in CES. In (Hadjitodorov et al. 2006), the diversity of a set of partitions is given by the mean pairwise dissimilarity between them, calculated using the Adjusted Rand (AR) index. A similar methodology is adopted in (Fern and Lin 2008; Azimi and Fern 2009) using the SNMI measure. In the present work, the Jaccard external index is adopted for the reasons already explained in Sect. 4.2.

In order to evaluate the explicit use of diversity in CES, a method is proposed here that begins by selecting the base partition with the highest diversity with respect to the original set of partitions, measured according to Eq. (12). Next, it iteratively selects the base partition with the highest diversity regarding the set of base partitions selected in previous iterations (selected set), until the desired number of partitions is obtained. This CES method will be referred to here as *Diversity* and is summarized in Algorithm 5.

Algorithm 5 *Diversity*

Require: A collection of candidate base partitions, Π_b, and the number of base partitions to be selected, s ($s < |\Pi_b|$);

$t \leftarrow \arg \max_{m \in \{1,\ldots,|\Pi_b|\}} f(m) = diversity(\pi_m, \Pi_b)$;
insert π_t into Π_s;
remove π_t from Π_b;
repeat
  $t \leftarrow \arg \max_{m \in \{1,\ldots,|\Pi_b|\}} f(m) = diversity(\pi_m, \Pi_s)$;
  insert π_t into Π_s;
  remove π_t from Π_b;
until $|\Pi_s| = s$;
return Π_s;

Algorithm 5 can be implemented in $O(E\,(|\Pi_b|^2 + s\Pi_b))$, where E is the complexity of the external index used by function *diversity*. For the experiments in this paper we have used the Jaccard index given by (13), for which E is $O(n^2)$, but if computing time is a concern, then the NMI or some other linear time external index should be used instead.

4.4 The Best Validated Consensus Partition (BVCP) method

The experiments to be reported in Sect. 5.4 reveal a large variety of results obtained by different CES methods, data sets, and consensus functions. An important question

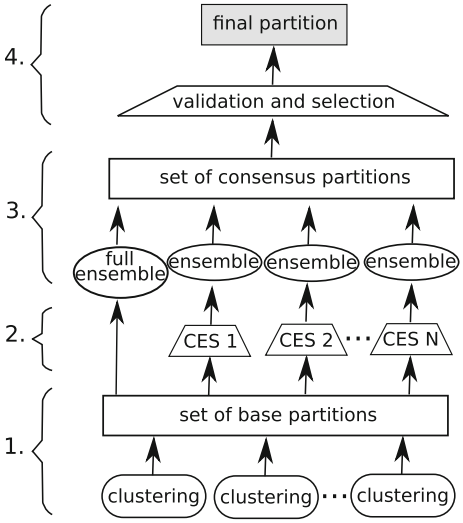

Fig. 1 Main steps of the BVCP method

arises: "How can good ensemble results be achieved in practical applications, where little or no information is available about the underlying clustering structure contained in the data (if any) or about the performance of the different methods (partition selection and consensus functions) when applied to such a sort of data?". It is not trivial (if possible) to design a methodology capable of producing the best results given such a variety of alternatives. However, we believe that it is possible to develop a method that is likely to produce good solutions.

We propose here the use of both the full ensemble and the (multiple) CES methods to generate candidate consensus partitions when there is no information to guide the choice of a particular method (like the performance of the CES methods when applied to the different consensus functions for the particular kind of data in hand). In this case, the final partition can be selected among the set of candidate consensus partitions through an evaluation carried out using a relative validation index or a combination of indexes (CRI). This method, named Best Validated Consensus Partition (BVCP), is composed of four main steps, illustrated in Fig. 1 and detailed in the following.

1. *Generation of base partitions:* generates base partitions by using one of the several methods presented in the literature (Kuncheva et al. 2006; Hadjitodorov et al. 2006) (see Sect. 5.2 for an example).
2. *CES application:* applies different CES methods, resulting in different selected subsets of base partitions. Since in most practical cases there is no information about the expected performance of a single relative validation index for the data in hand, the use of a CRI method (Sect. 4.2) may be recommended.
3. *Generation of consensus partitions:* combines the subset of base partitions resulting from each CES method into a candidate consensus partition, using a given consensus function. An additional consensus partition is derived from the full ensemble.

4. *Validation and selection:* validates each consensus partition produced in the previous step using relative validation index(es). As in practical clustering applications, the expected performance of the validation indexes is usually unknown for the data set in hand, so the use of a CRI may also be recommended in this step. The partition with the best (combined or individual) validation value, according to the index(es) adopted, is selected and considered as the BVCP final partition.

5 Experimental evaluation

In order to assess the performance of the proposed CES methods, experiments were carried out with artificial and real data sets. These data sets are described in Sect. 5.1. The generation of the base partitions is addressed in Sect. 5.2 and the consensus functions considered in this study are discussed in Sect. 5.3. In Sect. 5.4, we investigate the use of diversity and relative validation indexes—both individually (SIS) and combined (CRI)—in CES, when applied to collections of artificial data sets. The results of the different methods are compared against themselves and against those provided by the method CAS, proposed by Fern and Lin (2008). Experiments involving real data sets and the meta-ensemble method BVCP described in Sect. 4.4 are presented in Sect. 5.5.

5.1 Data sets

Initially, the methods proposed in this paper are experimentally investigated using three major collections of artificial data sets selected from the literature, each of which contains tens of artificially generated data sets. The first major artificial collection was generated by Vendramin et al. (2010), closely following the artificial data generator used in the classic study by Milligan and Cooper (1985). The spatial dispositions of the objects within clusters follow (mildly truncated) multivariate normal distributions, in such a way that the resulting structure could be considered to consist of natural clusters that exhibit the properties of external isolation and internal cohesion. Other features of these data sets are:

- Each data set has $n = 500$ objects;
- Overlap of cluster boundaries is permitted in all but the first dimension of the attributes space.
- Varied numbers of clusters, k, where $k \in \{2, 4, 6, 12, 14, 16\}$;
- Different numbers of attributes, a, where $a \in \{2, 3, 4, 22, 23, 24\}$;
- Three methods for distributing objects among clusters:
  - All objects are distributed as equally as possible among the clusters;
  - One cluster must contain 10 % of the data objects and the remaining objects are equally distributed among the other clusters;
  - If $k \in \{2, 4, 6\}$, one of the clusters must contain 60 % of the objects. Otherwise, one cluster must contain 20 % of the objects. The remaining objects are equally distributed among the other clusters;

– For each combination of k, a, and cluster balance, three sampling replications were generated, thus producing a total of 324 data sets.

For additional details about these data sets, please refer to (Vendramin et al. 2010). From now on, this collection will be referred as *Artificial1*.

Two major collections of artificial data sets analyzed in (Handl and Knowles 2007) are also used. The first one, named here *Artificial2*, contains clusters generated from multivariate normal distributions with variances higher than those in the collection *Artificial1*. The second one, referred to here as *Artificial3*, is composed of data sets generated by a high dimensional ellipsoidal cluster generator developed by Handl and Knowles (2007). The main features of the data sets collections *Artificial2* and *Artificial3* are:

– Both have overlapped clusters;
– Varied numbers of clusters, k, where $k \in \{4, 10, 20, 40\}$;
– The number of objects in each cluster is randomly drawn from the interval [50, 500] for data sets where $k \in \{4, 10\}$. For data sets where $k \in \{20, 40\}$, this number is drawn uniformly from the interval [10, 100].
– Data objects in *Artificial2* are described by either $a = 2$ or $a = 10$ attributes, while objects in *Artificial3* are described by either $a = 50$ or $a = 100$ attributes;
– Ten sampling replications were generated for each combination of k and a, producing a total of 80 data sets variations for each data collection.

In addition to the characteristics previously described, the three major artificial data collections exhibit different clustering complexity levels. The level of complexity can be estimated by the similarities between the obtained partitions and the known clusters (ground truth). The higher these similarities, the simpler is the clustering problem for the corresponding data set. To assess the complexity of the artificial data sets, the similarities among the base partitions (obtained as described in Sect. 5.2) and the known clusters were calculated by the Jaccard index. The mean Jaccard index values calculated for the best base partitions are 0.96, 0.83, and 0.44 for the data collections *Artificial1*, *Artificial2*, and *Artificial3*, respectively. As the Jaccard index returns values in the interval [0, 1], the obtained results suggest that *Artificial1* is the simplest data collection for clustering, whereas *Artificial3* is the most complex collection.

Experiments involving real data sets will be reported in Sect. 5.5. Most of these sets have been used in related works (Hadjitodorov et al. 2006; Fern and Lin 2008; Azimi and Fern 2009) and, for this reason, they were also considered in the experiments presented here. Four out of the seven real data sets were obtained from the UC Irvine (Asuncion and Newman 2007) repository: Iris (Fisher 1936), Wine (Aeberhard et al. 1992), Breast Cancer Wisconsin (without the 16 objects with missing attributes) (Mangasarian and Wolberg 1990), and Synthetic Control Chart Time Series (Alcock and Manolopoulos 1999). The fifth data set is composed of yeast gene-expression data, obtained from (Yeung et al. 2003), and will be named Yeast here. The last two are documents (text) data sets, formed with collections of articles from international journals. The first collection, named Articles, is formed by 253 articles related to politics, DNA research, weather, food and mobile computing (Naldi et al. 2011). The

Table 1 Main features of the real data sets used in the experiments

Name	#Objects (n)	#Attributes (a)	#Classes (k)	Minor class	Major class
Iris	150	4	3	50	50
Wine	178	13	3	48	71
Breast	683	9	2	239	444
Chart	600	60	6	100	100
Yeast	205	20	4	14	93
Articles	253	4636	5	42	62
cbrilpirivson	945	1431	5	101	276

second collection is composed solely of articles from computing related journals, being less heterogeneous and more difficult to cluster than the first collection. This collection is called cbrilpirivson (Paulovich et al. 2008), as their article subjects are **c**ase-**b**ased **r**easoning, **i**nductive **l**ogic **p**rogramming, **i**nformation **r**etrieval, **i**nformation **v**isualization and **son**ification. Table 1 displays the main features of these real data sets.

The true classes in the real data sets are not always an ideal goal for clustering algorithms, as they may not be compact and well separated. The Iris data set, for instance, has two classes with a high degree of overlapping (virginica, versicolor), while the third class is well separated from the others. However, as we have no knowledge about the level of correspondence between the labeled classes and the natural clusters in these data sets, the true classes will be considered as the ground truth for the experiments presented in this study.

5.2 Generation of base partitions

The base partitions were generated from the artificial collections and the real data sets described in Sect. 5.1. Several sets of base partitions were produced by repeatedly running the k-means algorithm with different parameter values, a method frequently used in the cluster ensemble literature (Fred and Jain 2005; Kuncheva et al. 2006; Hadjitodorov et al. 2006; Fern and Lin 2008; Azimi and Fern 2009). For each data set, p runs of k-means were performed, with k ranging from 2 to $\sqrt{n}$, where n is the number of objects. This strategy was adopted to provide, for each data set, a number of base partitions proportional to the number of objects.

It is important to select the value of p with caution. While low values of p may not generate the diversity required for some types of ensembles, high values may require too much memory for the storage of the base partitions. In this paper, we selected values so as to create large sets of base partitions without exceeding the amount of memory available. Specifically, we selected $p = 10$ for the experiments with the *Artificial1* collection and real data sets and $p = 5$ for the experiments with the *Artificial2* and *Artificial3* data sets.

The proposed CES methods were used to select the best partitions for the ensembles. Each of these methods was applied to select subsets containing 10, 25, 50, and 75 % of all base partitions produced by k-means for each data set.

5.3 Consensus functions

Once selected, the base partitions are combined into consensus partitions with k clusters using different consensus functions, where k is the known number of clusters or classes in the corresponding data set. Two types of consensus functions stand out among the best known and will be used in our experiments: the evidence accumulation based consensus functions (Fred and Jain 2005) and the graph based consensus functions (Strehl and Ghosh 2002). The first type constructs an $n \times n$ matrix formed by the similarities between each pair of objects, similarities being computed based on the number of clusters these objects share in the base partitions (Kuncheva 2004). This procedure is known as evidence accumulation (Fred and Jain 2005). Next, a clustering algorithm generates the consensus partition using the co-association similarities previously computed. Hierarchical clustering algorithms are frequently used for this task, specially the average-link and single-link algorithms (Jain and Dubes 1988). These consensus functions are named Evidence Accumulation Clustering with Average-Link (EAC-AL) and Evidence Accumulation Clustering with Single-Link (EAC-SL) (Fred and Jain 2005). Both will be used in our experiments. These functions have asymptotic computational complexity of $O(n^2 (\log n + |\Pi|))$, where n is the number of data objects and $|\Pi|$ is the cardinality of the set of base partitions used for the ensemble. If a full ensemble is constructed, then $|\Pi|$ is the number of available base partitions ($|\Pi| = |\Pi_b|$). Instead, if a CES method precedes the consensus function, then $|\Pi|$ is the cardinality of the selected set ($|\Pi| = |\Pi_s| < |\Pi_b|$).

The second type of consensus functions considered here represents the data set objects and the clusters contained in the base partitions as nodes in a graph, in which their relationships correspond to edges/hyper-edges (Strehl and Ghosh 2002). The consensus partition is obtained by using a graph/hyper-graph partition technique (Karypis and Kumar 1999). The Cluster-based Similarity Partitioning Algorithm (CSPA), the Hyper-Graph Partitioning Algorithm (HGPA), and the Meta-CLustering Algorithm (MCLA) are graph based consensus functions proposed in (Strehl and Ghosh 2002) and frequently used in the literature (Kuncheva et al. 2006; Fern and Lin 2008). For this reason, they will be used in our experiments. Their asymptotic computational complexities are estimated as $O(n^2\, k\, |\Pi|)$, $O(n\, k^2\, |\Pi|^2)$, and $O(n\, k\, |\Pi|^2)$, respectively, where n is the number of data objects, k is the number of clusters in the consensus partition, and $|\Pi|$ is the cardinality of the set of base partitions used for the ensemble (i.e., $|\Pi| = |\Pi_s|$ if CES is performed or $|\Pi| = |\Pi_b|$ in case of a full ensemble).

It is important to note that the CES methods *Diversity* (Sect. 4.3) and SRD (Sect. 4.2) are based on pairwise comparisons of partitions, which in turn are based on external validity indexes. This may cause the asymptotic computational complexities of these methods to be equivalent or exceed the complexities of the consensus functions considered in the present study, especially if an external index based on pairwise comparisons of data objects is adopted. The CAS method for CES (Fern and Lin 2008), which will

Table 2 CES methods evaluated in the experiments

SIS methods
Simplified Silhouette (SS)
Alternative Simplified Silhouette (ASS)
Variance Ratio Criterion (VRC)
PBM
Davies–Bouldin (DB)
Dunn (DN)
CRI methods
Sum of Ranks (SR)
Best Rank Position (BRP)
Sum of Ranks with Diversity (SRD)
Other methods
Davies–Bouldin (DB)
Diversity
CAS

also be considered in our experiments, is highly computationally demanding as well, especially for large sets of base partitions.[6] Therefore, in order to justify the adoption of *Diversity*, SRD, and CAS, the quality of their results should be superior to that obtained by the full ensemble (i.e., when using all available base partitions, without any selection). In contrast, the other CES methods based on relative indexes (SIS, SR, and BRP) have complexities that are nearly linear with respect to $|\Pi_b|$ and linear with respect to n (if indexes such as those reviewed in Sect. 3 are used). Hence, their complexities are lower than those of the consensus functions considered in this study, possibly justifying their application even if their results are equivalent to those provided by the full ensemble (as they reduce the size of the collection of base partitions to be combined and, therefore, the computational burden of the consensus function).

5.4 Experiments with artificial data sets

In this study, the six relative validation indexes presented in Sect. 3 were employed for the evaluation of base partitions, individually (SIS methods—Sect. 4.1) and combined (CRI methods—Sect. 4.2). Additionally, the *Diversity* method, described in Sect. 4.3, and the CAS method, proposed by Fern and Lin (2008), were also employed and assessed. In summary, the CES methods investigated here are listed in Table 2.

For a feasible comparison of the quality of the CES methods investigated, the similarity between the resulting consensus partitions and the known clusters (ground truth) is measured using the Jaccard index. The mean values of the Jaccard index computed over the data sets of each major collection are presented as curves in Figs. 2, 3, 4, 5, and 6, where the ordinate axis represents the Jaccard values and the abscissa axis represents the proportion of selected base partitions[7]. Three sub-figures are presented in

[6] CAS has asymptotic computational complexity estimated as $O(n\,k_{max}^2\,|\Pi_b|^2 + |\Pi_b|^3)$, considering the NMI calculations and the use of spectral clustering (Ng et al. 2002), where k_{max} is the maximum number of clusters in the base partitions.

[7] Proportion equal to unit means the full ensemble (no selection).

each figure, one for each major artificial collection of data sets (*Artificial1*, *Artificial2*, and *Artificial3*). In order to make the analysis of the results easier, only the best and the mean Jaccard index values resulting from the different CES methods based on single relative validation indexes (SIS) are presented in these figures. A more detailed comparison focusing on these particular methods is provided in Sect. 5.4.6. The first line on top of the legends indicates the average (over the data sets) of the best Jaccard index value obtained among all the CES methods based on a single relative validation index (first column of Table 2), referred to here as *Best of the Relative Indexes* (BRI)[8]. The second line of the legends indicates the average (over the data sets) of the mean Jaccard index value computed with respect to all the CES methods based on a single relative index (first column of Table 2), referred to here as *Mean of Relative Indexes* (MRI). The three subsequent lines of the legends refer to the CES methods based on the proposed combinations of relative indexes (CRI): SR, BRP, and SRD described in Sect. 4.2. The last two lines refer to the *Diversity* (Sect. 4.3) and the CAS methods (Fern and Lin 2008), respectively.

Once the Jaccard index values were calculated for all consensus partitions, two hypothesis tests were applied to the areas under the curve (AUCs) computed from these values. The first test is the well known ANalysis Of VAriance (ANOVA) (Walpole et al. 2006), frequently adopted to compare samples from multiple origins. However, ANOVA assumes that the compared samples are drawn from populations with normal distributions and similar variances (Demšar 2006). As these requirements are not ensured here, we also applied the (non-parametric) Friedman test (Hollander and Wolfe 1999). When the null hypothesis was rejected for both tests, indicating that there is statistical evidence to support that the compared means are different, a post-hoc multiple comparison procedure (Hochberg and Tamhane 1987) was applied using Matlab® to find which differences did exhibit statistical significance. To maintain the actual level of statistical confidence in 95 %, a Bonferroni adjustment (Dunn 1961) was applied to the critical values from the t distribution before applying the procedure, to compensate for multiple comparisons. In the legends of Figs. 2, 3, 4, 5, and 6, the CES method with the highest mean AUC value is followed by the symbol (•) and the methods for which the mean AUC exhibits no statistically significant differences with respect to this highest value are followed by the symbol (○). Next, we present the experimental results for the different consensus functions.

5.4.1 Results for the EAC-AL consensus function

This section presents the experimental results for the EAC-AL consensus function. Figure 2a shows that the use of CES based on relative validation indexes improved the results obtained by the EAC-AL (Evidence Accumulation Clustering with Average-Link) consensus function (Fred and Jain 2005) for the *Artificial1* data sets. As the base partitions of these data sets resemble the known clusters, the validation indexes were able to efficiently select partitions with good quality among those

[8] It is worth remarking that BRI is only used here as a basis for comparison. Actually, this method is not realizable in practice because, in real application scenarios, we do not have the ground truth (known ideal partition) to determine the best single relative index for a given data set.

obtained by the clustering procedure. The CRI methods seem to be a good choice for these data sets, as SR, BRP, and SRD provided mean values statistically equivalent to the best result obtained by using validation indexes individually (BRI). These methods also presented better results than *Diversity* and CAS. However, the same cannot be said with respect to the *Artificial2* and *Artificial3* collections, probably because the base partitions generated from these data sets do not match very closely the known clusters, i.e., they convey little evidence of the expected structures. In these cases, a larger number of such (lower quality) base partitions may be required to enhance the quality of the ensemble through evidence accumulation (Fred and Jain 2005). This can be observed for most CES in Fig. 2b, c. These figures also suggest that enhancing the diversity among base partitions, which is achieved by the methods *Diversity* and CAS, is more effective when the overall quality of the base partitions is lower (more clearly for the *Artificial3* collection, when these methods outperformed the full ensemble).

5.4.2 Results for the EAC-SL consensus function

For the *Artificial1* data sets, the EAC-SL (Evidence Accumulation Clustering with Single-Link) consensus function (Fred and Jain 2005) stands out in comparison to the other consensus functions investigated in this study, resulting in the known clusters (unitary Jaccard index value) for most experiments. The same performance was not observed for the other (more complex) data sets, as illustrated in Fig. 3b, c.

Regarding the CES methods, Fig. 3 suggests that, when the overall quality of the base partitions is lower (which happens mostly for the *Artificial2* and *Artificial3* collections), larger amounts of base partitions tend to improve the ensemble. In addition, when a limited proportion of base partitions is selected, the diversity of the selected subset (explicitly considered by the methods *Diversity* and CAS) becomes important.

5.4.3 Results for the CSPA consensus function

According to Fig. 4a, b, results generated with CSPA (Cluster-based Similarity Partitioning Algorithm) consensus function (Strehl and Ghosh 2002) show differences smaller than 0.04 in the Jaccard values for the different CES methods when applied to

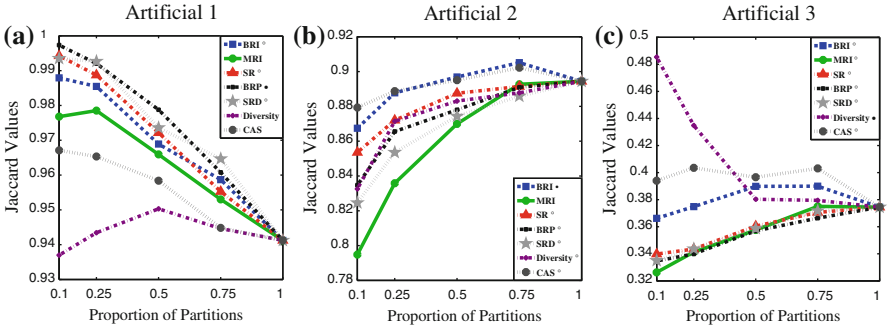

Fig. 2 Mean Jaccard index values for the EAC-AL consensus function

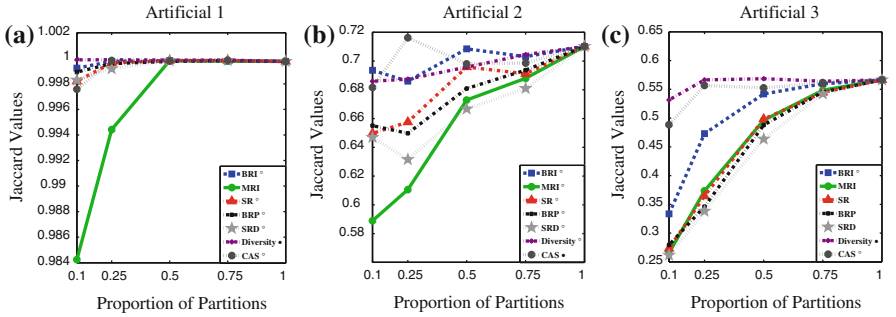

Fig. 3 Mean Jaccard index values for the EAC-SL consensus function

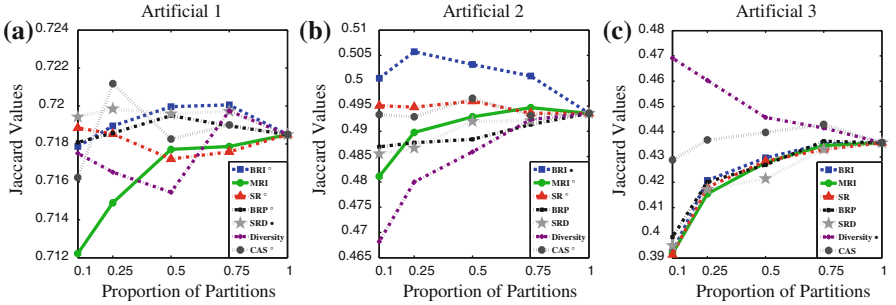

Fig. 4 Mean Jaccard index values for the CSPA consensus function

the *Artificial1* and *Artificial2* collections. In such cases, SR and the methods based on relative indexes individually are advantageous for having lower computational complexities than *Diversity*, SRD, CAS, and the full ensemble (as discussed in Sect. 5.3). Moreover, they achieved results statistically equivalent to the best ones obtained for these collections of data sets, according to the statistical tests performed. For the *Artificial3* dataset, the *Diversity* method achieved the best results. The behavior of the CES methods when applied to this collection of data sets is similar to that showed in Fig. 2c for the EAC-AL consensus function.

5.4.4 Results for the HGPA consensus function

According to Fig. 5, the mean quality of the consensus partitions using the HGPA (Hyper-Graph Partitioning Algorithm) consensus function (Strehl and Ghosh 2002) becomes stable after a monotonic increase, no matter the collection of data sets considered, which suggests that this consensus function exhibits low sensitivity to the mean quality of the selected partitions (which decreases with the complexity of the data set collections, being higher for *Artificial1* and lower for *Artificial3*). This behavior also evidences the superiority of the full ensemble when used with this consensus function. When a limited proportion of base partitions is selected for the ensemble, however, the results in Fig. 5 suggest that diversity is important to reduce the loss of accuracy.

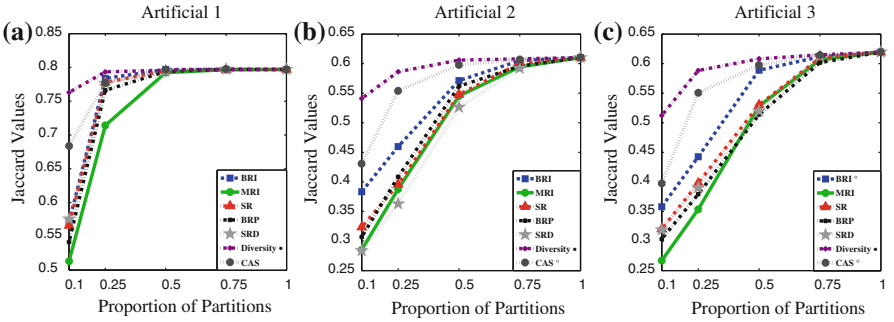

Fig. 5 Mean Jaccard index values for the HGPA consensus function

5.4.5 Results for the MCLA consensus function

Among all consensus functions investigated in this work, the MCLA (Meta-CLustering Algorithm) consensus function (Strehl and Ghosh 2002) was the most improved by the CES methods. This is illustrated by the descending curves in Fig. 6a, b. Even for the *Artificial3* collection, some CES methods based on single or combined relative validation indexes produced consensus partitions not statistically different from the full ensemble when at least 25 % of the base partitions were selected. Furthermore, the results from the SR selection method were very close to the best result based on single relative validation indexes (BRI)[9].

When used with MCLA, the CAS and *Diversity* methods produced results with quality inferior or similar to those obtained by the full ensemble for most data sets and proportions of partitions selected. For this reason, and due to the high computational cost of these methods, their use in conjunction with the MCLA consensus function may not be recommended. Additionally, the MCLA consensus function was more affected by the quality of the selected base partitions than by their diversity. The sensitivity to the quality of the base partitions becomes clear by comparing the range of Jaccard values in Fig. 6a–c.

5.4.6 Comparison of relative indexes

Figures 2, 3, 4, 5, and 6 allow the comparison of CES methods based on CRI, *Diversity*, and the best and average results from CES methods based on a single validation index (BRI and MRI, respectively). One should notice, however, that the values of BRI and MRI do not reveal by themselves the individual performances of the CES methods based on a single index. Table 3 shows such individual performances by presenting the mean and standard deviation of the AUCs computed over the 484 artificial data sets from the three major collections *Artificial1*, *Artificial2*, and *Artificial3*. The previously adopted hypothesis tests were also applied to these results. For each consensus

[9] Recall that BRI is not realizable in practice, as the best relative index is unknown when the ground truth is not available.

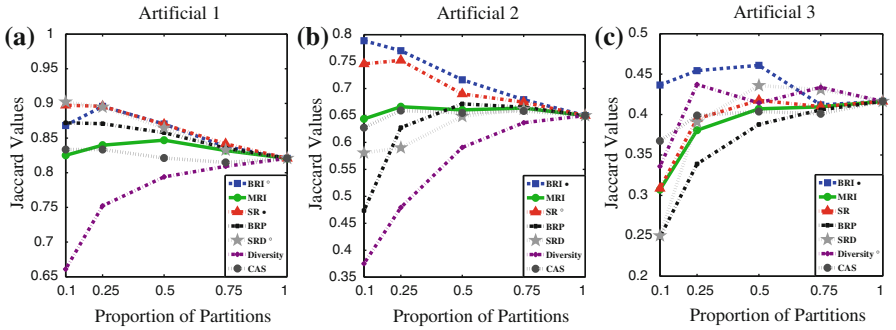

Fig. 6 Mean Jaccard index values for the MCLA consensus function

Table 3 Mean and standard deviation of the AUCs of Jaccard index values for each CES based on a single relative validation index (Single Index Selection (SIS)—Algorithm 1)

	SS	ASS	VRC	PBM	DN	DB
EAC-AL	0.764 (0.22)	**0.766 (0.22)**	**0.772 (0.21)**	0.752 (0.23)	0.7558 (0.22)	0.761 (0.22)
EAC-SL	**0.777 (0.24)**	0.776 (0.24)	**0.783 (0.24)**	0.754 (0.25)	0.7541 (0.24)	0.772 (0.24)
CSPA	**0.569 (0.21)**	**0.570 (0.21)**	**0.571 (0.21)**	0.567 (0.22)	0.5615 (0.21)	0.568 (0.21)
HGPA	**0.620 (0.19)**	**0.625 (0.20)**	**0.628 (0.19)**	0.590 (0.22)	0.5591 (0.18)	0.603 (0.18)
MCLA	**0.683 (0.22)**	**0.688 (0.21)**	0.673 (0.22)	0.6516 (0.24)	0.590 (0.22)	0.674 (0.22)

function, the best values and the values without significant statistical difference with respect to the best value are highlighted in bold.

The values in Table 3 evidence that the SS, ASS, and VRC indexes provided the best mean AUCs for the artificial data sets. As these data sets are mainly composed by clusters with multivariate normal distributions, such results suggest that SS, ASS, and VRC are suitable for this class of data. However, it is not always possible to know in advance the performance of a validation index of interest for specific data sets. For these cases, the use of CRI methods can be advantageous, as their results were superior to the mean results based on a single relative index (MRI) in most of the experiments. In particular, the SR method obtained results similar, sometimes without statistically significant difference, to the best results based on a single relative index (BRI), as it can be observed in Figs. 2, 3, 4, 5, and 6.

5.4.7 Summary of the results with artificial data sets

The results presented in Sect. 5.4 endorse the idea that CES methods based on relative validation indexes can obtain consensus partitions with quality higher than or equivalent to that of the full ensemble if the set of selected base partitions have altogether enough evidence to characterize the underlying structure of the data set. On the one hand, when base partitions are close to this structure, such as for the *Artificial1* data sets, subsets of base partitions selected by means of CES methods according to partition quality (measured by relative validation indexes) showed some tendency to

produce ensembles better than or comparable to the full ensemble. The reduction of the number of base partitions to be processed by the consensus functions (from $|\Pi_b|$ to $|\Pi_s|$) can also lead to computational savings. In these cases, the evaluation with CRI seems to be advantageous if there is no prior knowledge about the performance of the validation indexes of interest in the application scenario in hand, as the CRI methods showed to be more robust to variations in the application scenarios than the method based on individual indexes (SIS). Which method performed best depends on the consensus function and the data collection used. In general, the SR method stands out among the CRI, for the quality of its consensus partitions and for having nearly linear computational complexity in relation to the number of base partitions and number of objects. On the other hand, in cases where base partitions have little resemblance to the structure of interest, such as the *Artificial3* data sets, the full ensemble accumulates more evidence about the data. As a result, it tends to provide better solutions. In these cases, the CES methods based on diversity (*Diversity* and CAS) also achieved good results, possibly better than the full ensemble, since they incorporate complementary information about the data set in the consensus functions. However, these methods are based on comparisons between objects or clusters for all base partitions, resulting in a high computational cost. This cost may be similar or exceed the cost of the investigated consensus functions (presented in Sect. 5), favoring the use of the full ensemble.

Some consensus functions seem to be more sensitive to the selection of higher quality base partitions than others. For example, the MCLA function obtained better results when associated with CES methods based on relative validation indexes than the other functions investigated. In contrast, the results obtained with the HGPA function seem to be much more sensitive to the number (rather than the quality or diversity) of base partitions selected, since the quality of the resulting consensus partitions seems to grow monotonically as a function of this quantity. As a consequence, the HGPA function achieved the best results when using the full ensemble.

5.5 Experiments with real data sets

For the seven real data sets described in Sect. 5.1, experiments with the proposed CES methods, the full ensemble, and the BVCP method (Sect. 4.4) were performed. In this case, however, it is not possible to compare AUCs, as the full ensemble produces a single Jaccard value. Instead, the Jaccard index value calculated for the full ensemble is compared with the Jaccard index values of the CES and BVCP consensus partitions obtained from selected subsets composed of 10 % of the base partitions available. This percentage was chosen in order to significantly reduce the number of base partitions selected for the ensemble, thus emphasizing the effects of the CES approach when compared to the full ensemble. Moreover, for the sake of simplicity, we reduced the number of CES methods compared, which are: single index selection (SIS) with the simplified silhouette (SS); the CRI method SR; *Diversity*, and CAS. These methods were chosen for being among the most successful in the previous experiments (Sect. 5.4).

Table 4 Mean and standard deviation of Jaccard index values for EAC-AL ensembles

	Full ensemble	CAS	*Diversity*	SIS (SS)	CRI (SR)	BVCP
Iris	**0.6511 (0.02)**	0.6415 (0.03)	**0.6620 (0.07)**	**0.6455 (0.05)**	0.5877 (0.00)	**0.6707 (0.02)**
Wine	**0.8854 (0.03)**	**0.8751 (0.05)**	0.7348 (0.13)	0.7972 (0.07)	0.7975 (0.03)	0.8156 (0.03)
Breast	0.8526 (0.02)	**0.8661 (0.03)**	0.8470 (0.02)	**0.8719 (0.00)**	**0.8720 (0.00)**	**0.8716 (0.00)**
Chart	**0.5691 (0.03)**	**0.5466 (0.02)**	**0.5698 (0.03)**	**0.5472 (0.05)**	**0.5385 (0.04)**	**0.5354 (0.04)**
Yeast	**0.9737 (0.00)**	**0.9737 (0.00)**	**0.9577 (0.06)**	0.8564 (0.00)	0.8561 (0.00)	**0.9725 (0.00)**
Articles	**0.9826 (0.00)**	**0.9815 (0.00)**	0.9752 (0.01)	0.9775 (0.01)	**0.9803 (0.01)**	**0.9826 (0.00)**
cbrilpirivson	0.7145 (0.06)	0.6938 (0.07)	0.6616 (0.08)	**0.7471 (0.05)**	**0.7446 (0.05)**	**0.7697 (0.05)**

As discussed in Sect. 4.4, the BVCP method evaluates the consensus partitions produced by the full ensemble and the CES methods using relative validity indexes and, then, it selects the partition with the best validation value. We decided to use the SR method (Algorithm 2 with $s = 1$) during the fourth step of the algorithm (see Fig. 1). This choice was motivated by the following reasons: (i) unknown performance of the individual validation indexes for the data sets; (ii) better solutions obtained among the CRI methods for most experiments in Sect. 5.4; and (iii) linear computational complexity with respect to the number of objects (if indexes such as those reviewed in Sect. 3 are used) and nearly linear complexity with respect to the number of base partitions. Therefore, the BVCP final partition will be the consensus partition with the best sum of ranks resulting from the relative validation indexes adopted.

The mean and standard deviation of the Jaccard index values for the consensus partitions obtained with the full ensemble and CES methods are presented in tables 4-8 for the different consensus functions considered in this study. The same hypothesis tests used in Sect. 5.4 were adopted here. The best mean values and the values without statistically significant difference with respect to the best values are highlighted in bold.

The results from Table 4 indicate that the BVCP achieved results with no statistical difference to the best obtained with the EAC-AL consensus functions in 6 out of 7 data sets. For the Wine data set, although the consensus partitions produced by the BVCP method have mean Jaccard index values inferior to those obtained with the full ensemble and CAS, they are superior to those obtained by the CES methods based on single (SIS) or combined (CRI) relative validation indexes (SS and SR, respectively).

In general, the consensus partitions produced by the CES methods SIS (SS) and CRI (SR) were better than those obtained by the full ensemble, CAS, and *Diversity* when using with the EAC-SL consensus function. This superiority is more evident for the Wine and Breast data sets, as illustrated in Table 5. Notice that, regardless of the data set, the results of BVCP method exhibited no statistically significant differences with respect to the best results. This suggests that it is indeed possible to select high quality consensus partitions among those obtained by different CES methods and by the full ensemble, which is the basic idea behind BVCP.

As shown in Table 6, the full ensemble and the *Diversity* method achieved results equivalent to the best results obtained for most data sets (at least 5 out of 7), by using

Table 5 Mean and standard deviation of Jaccard index values for EAC-SL ensembles

	Full ensemble	CAS	*Diversity*	SIS (SS)	CRI (SR)	BVCP
Iris	**0.6679 (0.13)**	**0.5806 (0.10)**	0.5686 (0.06)	**0.6392 (0.05)**	**0.5883 (0.00)**	**0.6463 (0.08)**
Wine	0.5087 (0.12)	0.5900 (0.14)	0.4339 (0.12)	**0.7279 (0.13)**	**0.7340 (0.13)**	**0.6612 (0.13)**
Breast	0.5431 (0.00)	0.5933 (0.14)	0.5418 (0.00)	**0.8569 (0.08)**	**0.8725 (0.00)**	**0.8722 (0.00)**
Chart	**0.5590 (0.00)**	**0.5463 (0.02)**	**0.5633 (0.01)**	**0.5425 (0.05)**	**0.5567 (0.05)**	**0.5449 (0.01)**
Yeast	**0.9737 (0.00)**	**0.9737 (0.00)**	**0.9354 (0.07)**	0.8564 (0.00)	0.8561 (0.00)	**0.9729 (0.00)**
Articles	**0.9826 (0.00)**	**0.9815 (0.00)**	**0.9475 (0.09)**	**0.9658 (0.06)**	**0.9703 (0.06)**	**0.9820 (0.00)**
cbrilpirivson	0.2320 (0.00)	0.2751 (0.07)	0.2357 (0.02)	**0.3366 (0.10)**	0.2869 (0.09)	**0.3451 (0.10)**

Table 6 Mean and standard deviation of Jaccard index values for CSPA ensembles

	Full ensemble	CAS	*Diversity*	SIS (SS)	CRI (SR)	BVCP
Iris	**0.8159 (0.02)**	**0.8365 (0.04)**	**0.8003 (0.06)**	0.6391 (0.01)	0.6167 (0.03)	**0.8056 (0.06)**
Wine	**0.7554 (0.01)**	**0.7504 (0.01)**	**0.7517 (0.03)**	**0.7465 (0.01)**	**0.7479 (0.01)**	**0.7499 (0.01)**
Breast	0.6029 (0.00)	0.5948 (0.01)	**0.6044 (0.00)**	**0.6048 (0.00)**	**0.6051 (0.00)**	**0.6046 (0.00)**
Chart	**0.6184 (0.01)**	0.6044 (0.01)	**0.6280 (0.03)**	0.5153 (0.03)	0.5255 (0.03)	**0.6280 (0.03)**
Yeast	**0.4510 (0.00)**	**0.4508 (0.00)**	0.4325 (0.01)	**0.4508 (0.01)**	0.4432 (0.00)	**0.4498 (0.01)**
Articles	0.7258 (0.06)	0.7381 (0.04)	**0.7629 (0.06)**	0.7386 (0.05)	**0.7781 (0.04)**	**0.7880 (0.04)**
cbrilpirivson	**0.4891 (0.02)**	0.4695 (0.03)	**0.4738 (0.03)**	**0.4594 (0.03)**	0.4515 (0.03)	**0.4866 (0.02)**

Table 7 Mean and standard deviation of Jaccard index values for HGPA ensembles

	Full ensemble	CAS	*Diversity*	SIS (SS)	CRI (SR)	BVCP
Iris	**0.8349 (0.05)**	0.7007 (0.12)	**0.8010 (0.08)**	0.2793 (0.06)	0.3017 (0.11)	**0.8042 (0.08)**
Wine	**0.8962 (0.02)**	0.8263 (0.06)	**0.8503 (0.04)**	0.2863 (0.06)	0.2933 (0.06)	0.8260 (0.05)
Breast	**0.6693 (0.05)**	0.6287 (0.05)	**0.6299 (0.07)**	0.3536 (0.01)	0.3532 (0.00)	**0.6847 (0.03)**
Chart	**0.6185 (0.01)**	**0.6190 (0.03)**	**0.6274 (0.02)**	0.2763 (0.07)	0.2815 (0.08)	**0.6191 (0.02)**
Yeast	**0.4718 (0.01)**	0.4573 (0.01)	**0.4657 (0.01)**	0.1728 (0.00)	0.1703 (0.00)	**0.4674 (0.01)**
Articles	**0.9826 (0.00)**	0.8851 (0.11)	**0.9728 (0.02)**	0.2786 (0.19)	0.5148 (0.21)	**0.9820 (0.00)**
cbrilpirivson	**0.4219 (0.05)**	0.3547 (0.07)	**0.4797 (0.03)**	0.2756 (0.07)	0.2968 (0.07)	**0.4780 (0.03)**

the CSPA consensus function. Once again, the performance of the BVCP method exhibited no statistically significant differences in relation to the best results obtained for all data sets, which suggests that BVCP is a good method for applications in which no information about the data or about the expected behavior of the consensus function is available.

According to Table 7, the full ensemble is the best method for HGPA. In fact, in all experiments performed with the HGPA function for this study (artificial and real data), the best results were obtained using the full ensemble. It is worth mentioning that BVCP also produced results equivalent to the best results for 6 out of 7 data sets.

Table 8 Mean and standard deviation of Jaccard index values for MCLA ensembles

	Full ensemble	CAS	*Diversity*	SIS (SS)	CRI (SR)	BVCP
Iris	**0.8725 (0.02)**	**0.8198 (0.08)**	0.7962 (0.11)	0.6833 (0.00)	0.6701 (0.02)	0.6838 (0.00)
Wine	**0.8150 (0.02)**	**0.8287 (0.05)**	0.7949 (0.04)	0.8042 (0.02)	0.8001 (0.02)	**0.8140 (0.02)**
Breast	0.6196 (0.05)	0.6190 (0.05)	0.6083 (0.05)	**0.8786 (0.01)**	**0.8760 (0.00)**	**0.8749 (0.00)**
Chart	**0.5969 (0.01)**	0.5874 (0.03)	**0.6266 (0.03)**	0.4218 (0.05)	0.4028 (0.05)	**0.6205 (0.04)**
Yeast	0.5479 (0.07)	0.5836 (0.11)	0.4710 (0.04)	**0.8488 (0.04)**	**0.8402 (0.06)**	**0.8571 (0.01)**
Articles	**0.9826 (0.00)**	**0.9803 (0.01)**	**0.9750 (0.02)**	**0.9787 (0.01)**	**0.9804 (0.01)**	**0.9820 (0.00)**
cbrilpirivson	0.6842 (0.05)	0.6613 (0.09)	0.6002 (0.07)	**0.7230 (0.09)**	**0.7355 (0.06)**	**0.7718 (0.03)**

The Jaccard values in Table 8 show that the relative performances of the different methods varied significantly across the different data sets when using the MCLA consensus function. However, the BVCP method provided results equivalent to the best results in 6 out of 7 data sets, thus reinforcing the hypothesis that this method is a viable, possibly more robust choice in practical applications, where the ground truth is not available and the performance of the different methods cannot be compared with respect to the "right" solution.

6 Conclusions

This work proposed and compared the use of different relative clustering validation indexes in Clustering Ensemble Selection (CES), both individually (Single Index Selection—SIS) and combined (Combination of Relative Indexes—CRI). In particular, the CRI approach showed promising results for CES, as it produced results better than the average result obtained by the indexes individually in most data sets investigated.

Additionally, the analysis of the results obtained using 484 artificial and 7 real data sets indicated different behaviors of the consensus functions investigated, regarding the use of CES. The HGPA improved its results when the number of base partitions was increased, which disfavors the adoption of CES methods and favors the adoption of the full ensemble. Regarding the EAC-AL, EAC-SL, CSPA, and MCLA consensus functions, CES methods may result in consensus partitions equivalent to or better than the full ensemble if the selected partitions altogether have enough evidence of the underlying structure contained in the data set. This depends on the quality, diversity and number of selected base partitions. In particular, the MCLA consensus function seems to be favored by the quality and not by the cardinality or diversity of the selected set. If there is uncertainty regarding the quality and diversity of the base partitions or regarding the behavior of the consensus function, candidate consensus partitions produced by different CES methods and by the full ensemble as well can be evaluated and compared by means of relative validation indexes. For this scenario, we propose the BVCP method, which consists in selecting the best evaluated candidate consensus partition as the final partition. When applied to the 7 real data sets, the BVCP method generated results statistically equivalent to the best results obtained for, at least, 6

data sets, no matter the consensus function adopted. Such performance encourages the use of the BVCP method in practical applications, where the best method cannot be determined (and then selected), because a ground truth (known clustering solution) is not available.

Acknowledgements The authors acknowledge the Brazilian Research Agencies CNPq and FAPESP for financial support.

References

Aeberhard S, Coomans D, de Vel O (1992) Comparison of classifiers in high dimensional settings. Tech Rep 02, Department of Computer Science and Department of Mathematics and Statistics, James Cook University of North Queensland, Douglas

Alcock R, Manolopoulos Y (1999) Time-series similarity queries employing a feature-based approach. In: 7th hellenic conference on informatics, Ioannina, Greece, pp 27–29

Asuncion A, Newman D (2007) UCI machine learning repository. http://www.ics.uci.edu/~mlearn/MLRepository.html

Ayad HG, Kamel MS (2008) Cumulative voting consensus method for partitions with variable number of clusters. IEEE Trans Pattern Anal Mach Intell 30(1):160–173. doi:10.1109/TPAMI.2007.1138

Azimi J, Fern X (2009) Adaptive cluster ensemble selection. In: Twenty-first international joint conference on artificial intelligence (IJCAI-09), Pasadena, CA, USA, pp 992–997

Bezdek JC, Pal NR (1998) Some new indexes of cluster validity. IEEE Trans Syst Man Cybern 28(3):301–315

Bollacker KD, Ghosh J (1998) A supra-classifier architecture for scalable knowledge reuse. In: ICML '98: Proceedings of the fifteenth international conference on machine learning. Morgan Kaufmann Publishers Inc., San Francisco, pp 64–72

Calinski RB, Harabasz J (1974) A dendrite method for cluster analysis. Commun Stat 3:1–27

Caruana R, Munson A, Niculescu-Mizil A (2006) Getting the most out of ensemble selection. In: Proceedings of the 2006 sixth international conference on data mining, pp 828–833

Davies DL, Bouldin DW (1979) A cluster separation measure. IEEE Trans Pattern Anal Mach Intell 1:224–227

Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7:1–30

Dimitriadou E (2003) Explorative data analysis and applications. PhD thesis, Technische Universität Wien, Wien

Dimitriadou E, Weingessel A, Hornik K (1999) Fuzzy voting in clustering. In: Brewka G, Der R, Gottwald S, Schierwagen A (eds) Fuzzy-neuro systems. Leipziger Universittsverlag, Leipzig pp 63–74

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

Dunn JC (1974) Well separated clusters and optimal fuzzy partitions. J Cybern 4:95–104

Dunn OJ (1961) Multiple comparisons among means. J Am Stat Assoc 56(293):52–64. http://www.jstor.org/stable/2282330

Fern XZ, Brodley CE (2004) Solving cluster ensemble problems by bipartite graph partitioning. In: Proceedings of ICML'04, New York, NY, USA, p 36. doi:10.1145/1015330.1015414

Fern XZ, Lin W (2008) Cluster ensemble selection. J Stat Anal Data Min 1(3):128–141. doi:10.1002/sam.v1:3

Fisher R (1936) The use of multiple measurements in taxonomic problems. Ann Eugen 7(2):179–188

Fred ALN, Jain AK (2005) Combining multiple clusterings using evidence accumulation. IEEE Trans Pattern Anal Mach Intell 27(6):835–850

Greene D, Tsymbal A, Bolshakova N, Cunningham P (2004) Ensemble clustering in medical diagnostics. In: CBMS '04: proceedings of the 17th IEEE symposium on computer-based medical systems. IEEE Computer Society, Washington, DC, USA, pp 576–581. http://dx.doi.org/10.1109/CBMS.2004.40

Hadjitodorov ST, Kuncheva LI (2007) Selecting diversifying heuristics for cluster ensembles. In: 7th international workshop, pp 200–209

Hadjitodorov ST, Kuncheva LI, Todorova LP (2006) Moderate diversity for better cluster ensembles. Inf Fusion 7(3):264–275. doi:10.1016/j.inffus.2005.01.008

Halkidi M, Batistakis Y, Vazirgiannis M (2001) On clustering validation techniques. Intell Inf Syst J 17(2-3):107–145

Handl J, Knowles J (2007) An evolutionary approach to multiobjective clustering. IEEE Trans Evolution Comput 11(1):56–76

Hochberg Y, Tamhane AC (1987) Multiple comparison procedures. Wiley, New York

Hollander M, Wolfe DA (1999) Nonparametric statistical methods. Wiley-Interscience, New York

Hruschka ER, Campello RJGB, de Castro LN (2004a) Evolutionary algorithms for clustering gene-expression data. In: Proceedings of IEEE international conference on data mining, Brighton/England, pp 403–406

Hruschka ER, Campello RJGB, de Castro LN (2004b) Improving the efficiency of a clustering genetic algorithm. In: Advances in artificial intelligence—IBERAMIA 2004: 9th Ibero-American conference on AI, Puebla, Mexico, November 22–25. Proceedings. Lecture notes in computer science, vol 3315. Springer, New York, pp 861–870

Hubert LJ, Arabie P (1985) Comparing partitions. J Classif 2:193–218

Jaccard P (1908) Nouvelles recherches sur la distribution florale. Bull Soc Vandoise des Sci Nat 44:223–270

Jain A, Dubes R (1988) Algorithms for clustering data. Prentice Hall, Upper Saddle River

Karypis G, Kumar V (1999) A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM J Sci Comput 20(1):359–392. http://glaros.dtc.umn.edu/gkhome/metis/metis/overview

Kasturi J, Acharya R (2004) Clustering of diverse genomic data using information fusion. In: SAC '04: proceedings of the 2004 ACM symposium on applied computing. ACM, New York, pp 116–120. doi:10.1145/967900.967926

Kaufman L, Rousseeuw P (1990) Finding groups in data: an introduction to cluster analysis. Wiley series in probability and mathematical statistics. John Wiley & Sons, New York

Kuncheva LI (2004) Combining pattern classifiers. John Wiley & Sons, New York

Kuncheva L, Hadjitodorov S (2004) Using diversity in cluster ensembles. In: Systems, man and cybernetics, 2004 IEEE international conference on, vol 2, pp 1214–1219. doi:10.1109/ICSMC.2004.1399790

Kuncheva L, Hadjitodorov S, Todorova L (2006) Experimental comparison of cluster ensemble methods. In: Information fusion, 2006 9th international conference on, pp 1–7. doi:10.1109/ICIF.2006.301614

Mangasarian OL, Wolberg WH (1990) Cancer diagnosis via linear programming. SIAM News 23(5):1–18

Margineantu D, Dietterich T (1997) Pruning adaptive boosting. In: Proceedings of the 14th international conference on machine learning, pp 211–218

Milligan GW, Cooper MC (1985) An examination of procedures for determining the number of clusters in a data set. Psychometrika 50:159–179

Monti S, Tamayo P, Mesirov J, Golub T (2003) Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. Mach Learn 52(1-2):91–118

Naldi MC, Campello RJGB, Hruschka ER, Carvalho ACPLF (2011) Efficiency issues of evolutionary *k*-means. Appl Soft Comput 11(2): 1938–1952. doi:10.1016/j.asoc.2010.06.010

Ng AY, Jordan MI, Weiss Y (2002) On spectral clustering: analysis and an algorithm. In: Advances in neural information processing systems, vol 2, pp 849–856

Pakhira MK, Bandyopadhyay S, Maulik U (2004) Validity index for crisp and fuzzy clusters. Pattern Recog 37(3):487–501. doi:10.1016/j.patcog.2003.06.005, http://www.sciencedirect.com/science/article/B6V14-49YH94Y-3/2/399727cea74b53ae0b747d5f73922009

Paulovich FV, Nonato LG, Minghim R, Levkowitz H 2008) Least square projection: a fast high-precision multidimensional projection technique and its application to document mapping. IEEE Trans Visual Comput Graph 14:564–575. doi:10.1109/TVCG.2007.70443, www.lcad.icmc.usp.br/~paulovic/pex/repository/data.zip

Rousseeuw PJ (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J Computat Appl Math 20:53–65

Strehl A, Ghosh J (2002) Cluster ensembles: a knowledge reuse framework for combining multiple partitions. J Mach Learn Res 3:583–617

Topchy A, Jain A, Punch W (2004) A mixture model for clustering ensembles. In: Proceedings of the SIAM international conference on data mining (SDM'2004), Lake Buena Vista, Florida, USA, pp 331–338

Tumer K, Agogino AK (2008) Ensemble clustering with voting active clusters. Pattern RecognL ett 29(14):1947–1953. doi:10.1016/j.patrec.2008.06.011

Vendramin L, Campello RJGB, Hruschka ER (2009) On the comparison of relative clustering validity criteria. In: SIAM international conference on data mining, Sparks/USA, pp 733–744

Vendramin L, Campello RJGB, Hruschka ER (2010) Relative clustering validity criteria: a comparative overview. Stat Anal Data Min 3(4): 209–235. doi:10.1002/sam.10080

Walpole RE, Myers R, Myers SL (2006) Probability and statistics for engineers and scientists. Macmillan, New York

Weingessel A, Dimitriadou E, Hornik K (2003) An ensemble method for clustering. In: Distributed statistical computing (DSC'2003), Wien, Austria, pp 1–12

Yeung K, Medvedovic M, Bumgarner R (2003) Clustering gene-expression data with repeated measurements. Genome Biol 4(5):R34. doi:10.1186/gb-2003-4-5-r34, http://genomebiology.com/2003/4/5/R34