



# Diversity control for improving the analysis of consensus clustering



Milton Pividori<sup>a,b,\*</sup>, Georgina Stegmayer<sup>a</sup>, Diego H. Milone<sup>a</sup>

<sup>a</sup> Research Institute for Signals, Systems and Computational Intelligence, *sinc(i)*, UNL-CONICET, Department of Informatics, FICH, Universidad Nacional del Litoral, Ciudad Universitaria CC 217, Ruta Nac. No 168, km 472.4, Santa Fe 3000, Argentina

<sup>b</sup> Center of Research and Development of Information Systems Engineering, CIDISI, UTN-CONICET, Department of Information Systems Engineering, FRFS, Universidad Tecnológica Nacional, Lavaisse 610, Santa Fe 3000, Argentina

## ARTICLE INFO

### Article history:

Received 21 September 2015

Revised 18 March 2016

Accepted 22 April 2016

Available online 28 April 2016

### Keywords:

Cluster ensembles  
Consensus clustering  
Diversity analysis  
Diversity control  
Ensemble diversity

## ABSTRACT

Consensus clustering has emerged as a powerful technique for obtaining better clustering results, where a set of data partitions (ensemble) are generated, which are then combined to obtain a consolidated solution (consensus partition) that outperforms all of the members of the input set. The diversity of ensemble partitions has been found to be a key aspect for obtaining good results, but the conclusions of previous studies are contradictory. Therefore, ensemble diversity analysis is currently an important issue because there are no methods for smoothly changing the diversity of an ensemble, which makes it very difficult to study the impact of ensemble diversity on consensus results. Indeed, ensembles with similar diversity can have very different properties, thereby producing a consensus function with unpredictable behavior. In this study, we propose a novel method for increasing and decreasing the diversity of data partitions in a smooth manner by adjusting a single parameter, thereby achieving fine-grained control of ensemble diversity. The results obtained using well-known data sets indicate that the proposed method is effective for controlling the dissimilarity among ensemble members to obtain a consensus function with smooth behavior. This method is important for facilitating the analysis of the impact of ensemble diversity in consensus clustering.

© 2016 Published by Elsevier Inc.

## 1. Introduction

Clustering is fundamental for understanding the structure of data [45] and it has been used in a wide range of areas, including engineering, financial, biological science, and medical applications [25,29,34,40,44]. However, the correct choice of a clustering algorithm, or even setting its parameters, requires knowledge of the data set and the data distribution assumed by algorithms, since they can strongly affect the final results obtained [22]. Clustering algorithms have been developed to solve a wide range of different problems, but there is no universal method that can be applied to solve all. Thus, different

\* Corresponding author at: Research Institute for Signals, Systems and Computational Intelligence, *sinc(i)*, UNL-CONICET, Department of Informatics, FICH, Universidad Nacional del Litoral, Ciudad Universitaria CC 217, Ruta Nac. No 168, km 472.4, Santa Fe 3000, Argentina. Tel.: +54 342 4575233x 190; fax: +54 3424575224.

E-mail addresses: [mpividori@sinc.unl.edu.ar](mailto:mpividori@sinc.unl.edu.ar) (M. Pividori), [gstegmayer@sinc.unl.edu.ar](mailto:gstegmayer@sinc.unl.edu.ar) (G. Stegmayer), [dmilone@sinc.unl.edu.ar](mailto:dmilone@sinc.unl.edu.ar) (D.H. Milone).

URL: <http://www.sinc.unl.edu.ar> (M. Pividori)

but equally valid solutions can be obtained using various algorithms, which is one reason why clustering is considered to be an ill-posed problem among researchers [24,50].

In the past decade, consensus clustering (or cluster ensembles) has emerged as a powerful approach for mitigating the issues of conventional cluster analysis. First, a set of data partitions is generated, which is called an *ensemble*. Next, a *consensus function* combines the ensemble into a consolidated solution or *consensus partition*, which has greater overall accuracy [10,18,19,21,39,42,54,56]. Given the ill-posed nature of clustering, the accuracy is typically measured by comparing the final solution with a known reference partition, which is generally based on the class labels associated with the data set [20,30,41,46,50]. Although this reference partition might not be the only valid structure for the data, many studies have tried to determine how ensembles should be built, and which characteristics they should have to obtain high accuracy. In particular, among these characteristics, the level of disagreement between ensemble members, which is called the *ensemble diversity*, has been identified as a key factor in the cluster ensemble problem [7,17,20], and various diversity measures have been proposed [2,14,15,33,55].

Many strategies have been used to explore how diversity affects consensus performance [8,20], where they usually aim to generate a set of ensembles with different diversity, before observing the performance of the consensus function. One of the most common approaches involves generating the ensemble members by randomly varying a parameter [8,15,17,27], which can be the clustering algorithm itself [28,36], the number of clusters [12,49,57], or its initialization [26,37]. Instead of changing the clustering algorithm, a common method involves changing the data by randomly selecting sub-samples [11,32,38,48], using different features [23,37,43,51,52], employing random projections [8,37,39], or combining several methods together [47,53]. An alternative to the purely random approach generates the ensemble that maximizes a given criterion. First, a pool of partitions is created by using the strategies described above and a subset of this pool is then selected, which maximizes the objective function. For example, this greedy approach was used in [20], where a set of criteria were defined to obtain low, medium, and highly diverse ensembles.

These methods have been used widely to explore the dissimilarity within an ensemble, but the results obtained indicate that there is an important problem with current methods for ensemble diversity analysis. Indeed, previous studies provide opposing opinions regarding this issue, where some have suggested that more diverse ensembles are better for obtaining more accurate solutions [8,20], whereas others have proposed that moderate diversity is the preferred choice [15]. In addition to these contradictory results, high variability has been found not only among data sets but also when different ensemble generation strategies are employed. Moreover, plots of the accuracy as a function of diversity have shown that ensembles with similar diversity can differ greatly in their accuracy [15,20]. These confusing results show that current approaches can generate diversity but they cannot control it, and this limitation may lead to unpredictable outputs by the consensus method. This is an important issue and it must be addressed before any analysis of the impact of diversity on consensus clustering. This unpredictable behavior occurs because as one diversity measure is being observed, another properties of the ensemble are changing, thereby leading to erratic behavior by the consensus function. In addition, it is difficult to generate ensembles that are uniformly distributed in the diversity range under evaluation, which could lead to a biased analysis. Both of these reasons demonstrate the need to control the ensemble diversity in order to effectively analyze its impact on the consensus results.

Due to the importance of diversity in consensus clustering, the issues highlighted above motivated us to unveil a new problem in this area and to propose a novel method that allows fine-grained control of the ensemble diversity. To the best of our knowledge, no methods have been proposed previously for controlling disagreement among ensemble partitions. Our method extracts information from the ensemble structure and then uses it to make small changes, which decrease and increase the diversity among ensemble members in a smooth manner. The results that we obtained using six well-known data sets demonstrate that this method is effective for controlling the ensemble diversity, where the consensus function behaves in a smooth manner, thereby providing a novel approach for studying the impact of diversity on consensus clustering.

The remainder of this paper is organized as follows. In Section 2, we explain the problems with current methods and we define the steps in the diversity control method. Section 3 describes the data sets and performance measures used for testing. Section 4 presents the evaluation procedure and the results obtained. In Section 5, we summarize our conclusions as well as suggesting possible improvements and future research.

## 2. Novel method for controlling diversity

Current methods assume implicitly that ensembles with a particular level of diversity are comparable; thus, equal diversity values should represent similar ensembles, or at least similar inputs for the consensus function, which is expected to produce similar results. However, this might not be the case in practice. An example of such results is shown in Fig. 1, where the accuracy of the consensus partition [20] is plotted as a function of the pairwise ensemble diversity. Similar results can be found in [15]. Two problems are evident based on this plot. The first is the behavior of the output consensus accuracy ( $y$ -axis) when the pairwise diversity ( $x$ -axis) is around 0.21, where the diversity values are close to each other but many points differ greatly in their accuracy. A similar behavior can be observed around a diversity value of 0.28. Thus, ensembles with similar diversity can represent very different inputs for the consensus function. The second problem is that the diversity range is not always sampled uniformly; for example, there are far less ensembles with diversity values in [0.10, 0.20] and even none in [0.31, 0.38]. These two issues make the study of diversity a fairly difficult problem. As stated earlier,

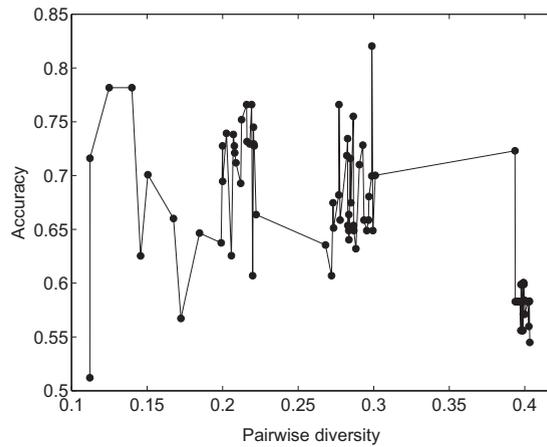


Fig. 1. Accuracy of the consensus partition as a function of the pairwise diversity for the Wine data set using the method proposed by Lam-On et al. [20].

Table 1

Notations used in the proposed method for diversity control.

Symbol	Description
$\pi$	Single data partition
$\Pi$	Ensemble with groups of partitions
$q$	Number of groups of partitions in $\Pi$
$\Pi_i$	Group of partitions (member of $\Pi$ )
$p_i$	Size of group $\Pi_i$
$\tilde{\pi}_i$	Representative partition for group $\Pi_i$
$\Pi^*$	Ensemble with controlled diversity
$\pi^*$	Final consensus partition
$\phi_i$	Clusterer
$m$	Number of clusterers used
$n$	Number of times a clusterer is run
$Y$	Normalized mutual information

a possible explanation for this behavior is that different properties of the ensemble may change while a single diversity measure is being observed. This makes it very difficult to analyze how diversity affects the consensus function, because it is not possible to make strong conclusions when the target variable (the accuracy in this case) exhibits erratic behavior. Furthermore, the analysis is made even more difficult due to the lack of uniformity in the sampling method for the range of diversity under examination. Thus, it is necessary to control the ensemble diversity in a smooth manner before it can be studied correctly.

In this section, we propose a diversity control method. Table 1 presents the notations used and the general process is depicted in Fig. 2. The first step generates an ensemble  $\Pi$  from the data. This ensemble is then divided into *groups of partitions*, which are created by clustering over the ensemble members. The ensemble with controlled diversity  $\Pi^*$ , which is produced by resampling from the original groups, is finally combined to obtain the consensus partition. The resampling step is shown in detail at the right of the figure. First, information about the ensemble structure is extracted, which is obtained by comparing each group with the others. Instead of a direct group comparison, *representative partitions*  $\tilde{\pi}$  are defined for each group. A weight is calculated for each group based on the similarity matrix between the representatives. Next, the distribution of these weights is adjusted using a nonlinear function and they are employed to resample from the groups of partitions in  $\Pi$ . Based on the adjusted distribution, the resampling process modifies the group sizes to change the ensemble diversity in a smooth manner. Thus, a new ensemble  $\Pi^*$  is created and its diversity is controlled by the distribution of the weights. Finally, a consensus function is used to derive the final partition  $\pi^*$  from  $\Pi^*$ . The following subsections explain these steps in detail.

### 2.1. Ensemble generation based on groups of partitions

Ensemble generation, which is the first step of the diversity control method, receives an input data set and produces an ensemble organized into groups of partitions. This step is depicted in detail in Fig. 3 and it comprises two phases:

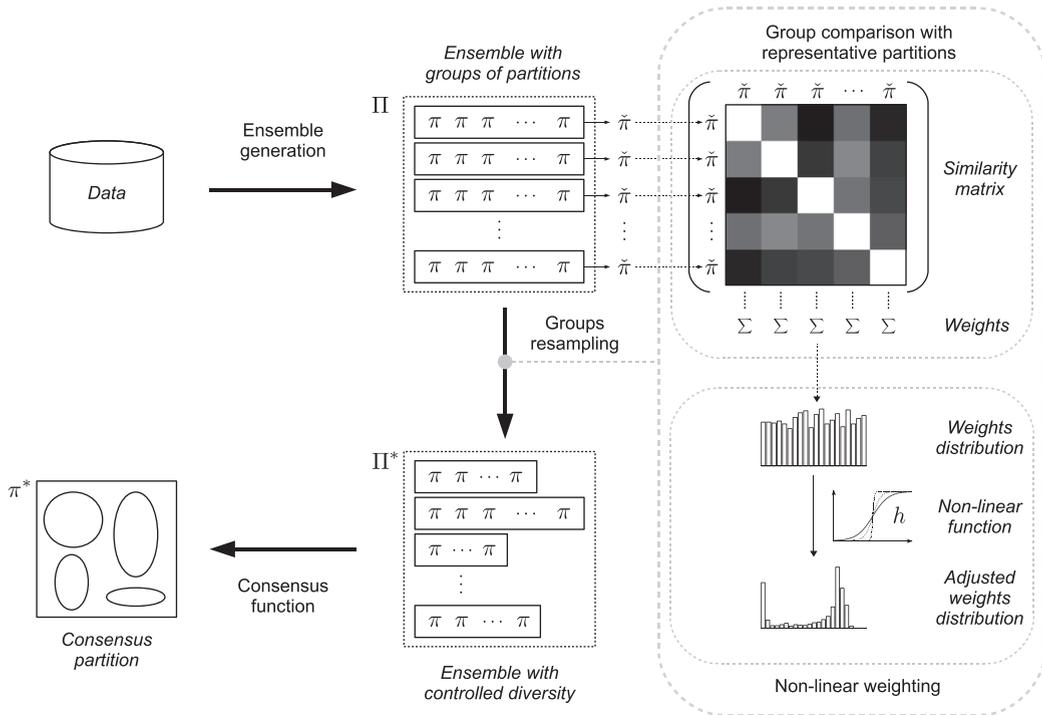


Fig. 2. The diversity control method: steps and intermediate outcomes.

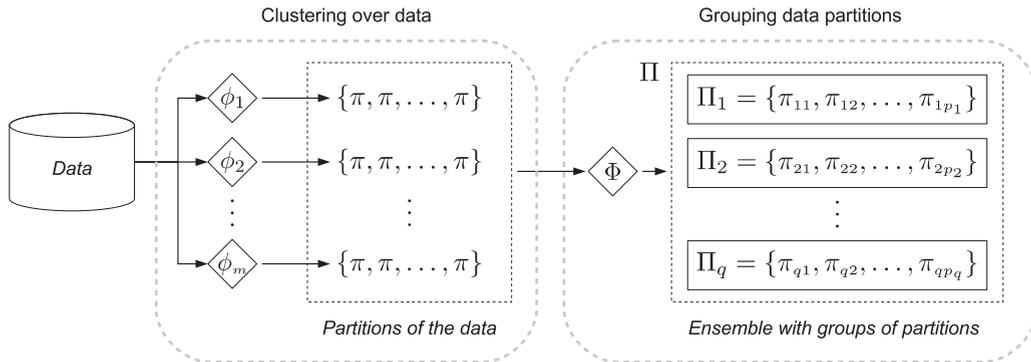


Fig. 3. Ensemble generation based on groups of partitions. The ensemble generation process comprises two phases: (1) clustering over data and (2) grouping data partitions.

(1) *clustering over data* where several partitions are obtained and (2) *grouping data partitions*, which identify different groups of partitions.

*Phase 1: Clustering over data.* Several methods can be used to create an ensemble. A common approach involves producing different partitions of the data using a single clustering algorithm and randomly varying some of its parameters [20,26]. In particular, *k*-means is a frequent choice and one of the following schemes for selecting the number of clusters is commonly employed: *k* is fixed and the cluster centers are randomly initialized [15,57]; or *k* is chosen randomly within an interval  $[k_{min}, k_{max}]$  [13,18,26]. Our proposed approach uses a combination of both. A diagram illustrating this clustering process is shown in the first part of Fig. 3. A clusterer  $\phi_i$  is defined for each *k* in the chosen interval, where  $\phi_i$  is run *n* times with random initializations, thereby producing a set of  $m \times n$  partitions. This process aims to obtain as many different partitions as possible. The ensembles generated in this manner have a known structure with sets of partitions, where each set is generated by the same clusterer only by varying the initialization.

*Phase 2: Grouping data partitions.* After producing the initial ensemble, the second phase of the ensemble generation step involves gathering information about its structure. This phase basically applies a clustering algorithm  $\Phi$  to the previously generated ensemble members, which obtains groups of partitions denoted as  $\Pi_i$ , with size  $p_i$ . Therefore, this phase does not

	$\check{\pi}_1$	$\check{\pi}_2$	$\check{\pi}_3$	$\check{\pi}_4$	$\check{\pi}_5$
$\check{\pi}_1$		0.71	0.80	0.75	0.69
$\check{\pi}_2$	0.71		0.79	0.75	0.63
$\check{\pi}_3$	0.80	0.79		0.79	0.70
$\check{\pi}_4$	0.75	0.75	0.79		0.68
$\check{\pi}_5$	0.69	0.63	0.70	0.68	
$\bar{r}_i$	0.74	0.72	0.77	0.74	0.68

**Fig. 4.** Similarity matrix for the representative partitions generated for the Wine data set. The last row represents the column average (discarding the main diagonal).

modify the initial ensemble and it simply regroups its members. Three grouping algorithms are proposed for this purpose, each of which produces groups of partitions with different properties. The first algorithm is the simplest and it groups the ensemble members according to their number of clusters. Thus, each group of partitions produced using this approach has the same  $k$  and this method is called ensemble grouping by  $k$  ( $Gk$ ). The second method, called full ensemble grouping (FG), involves applying a hierarchical agglomerative clustering algorithm to the ensemble members. This method uses the distance  $1 - \Upsilon(\pi_i, \pi_j)$ , where  $\Upsilon(\pi_i, \pi_j)$  is the normalized mutual information (NMI) [36] between partitions  $\pi_i$  and  $\pi_j$ . In contrast to  $Gk$ , FG allows to specify the number of groups  $q$  to be produced. The third algorithm, called  $FGk$ , is a combination of both, where it takes the groups formed by FG and then splits them according to  $Gk$ .

## 2.2. Group comparison with representative partitions

The previous step of the method produces an ensemble  $\Pi$  with groups of partitions  $\Pi_i$ . The next step involves comparing those groups to obtain information about the structure of  $\Pi$ , which can be achieved simply by performing a pairwise comparison between groups. However, given the structure of  $\Pi$ , it is possible to define a more efficient method for comparing groups. It should be noted that each group member shares some properties with the others, and thus each group of partitions can be considered as a cluster. Therefore, instead of comparing all of the group members, the similarity between two groups can be estimated by comparing their corresponding representative partitions. The representative partition  $\check{\pi}_i$  (see Fig. 2) is the single partition that best represents its group  $\Pi_i$ . A convenient definition of the representative that maximizes the information shared with group members is:

$$\check{\pi}_i = \arg \max_{\check{\pi}} \sum_{\pi \in \Pi_i} \Upsilon(\pi, \check{\pi}). \quad (1)$$

A consensus function can be used as a method for obtaining a partition  $\check{\pi}$ . Group members could certainly have different numbers of clusters (particularly when FG is employed), so it is necessary to decide how many clusters the representative partition should contain. This is achieved by obtaining a consensus partition for each  $k$  in  $\Pi_i$ , where that with the maximum agreement among group members is finally selected as the representative. For instance, if group  $\Pi_i$  contains many data partitions with  $k = 3, 5$  and  $7$ , then three consensus partitions are derived from  $\Pi_i$ , with  $3, 5$  and  $7$  clusters. Thus, the consensus partition that maximizes (1) is selected as the representative for group  $\Pi_i$ .

After obtaining all of the representative partitions, they are compared with each other to produce a similarity matrix

$$r_{ij} = \Upsilon(\check{\pi}_i, \check{\pi}_j). \quad (2)$$

This matrix contains information about the ensemble structure. The similarity of a group with respect to the other groups is estimated by  $\bar{r}_i = \frac{1}{q-1} \sum_{j \neq i} r_{ij}$ . An example of a similarity matrix is shown in Fig. 4, which compares five representatives. According to this similarity matrix, group  $\Pi_3$  is most similar to the rest of the ensemble and  $\Pi_5$  is most different. This information can be used to modify  $\Pi$  and obtain a new ensemble  $\Pi^*$  with controlled diversity.

## 2.3. Nonlinear group weighting and resampling

After gathering the information about the ensemble structure, the next stages produce a new ensemble  $\Pi^*$  (with controlled diversity) by adjusting the distribution of  $\bar{r}_i$  to obtain a weight  $w_i$  for each group and then resampling from the

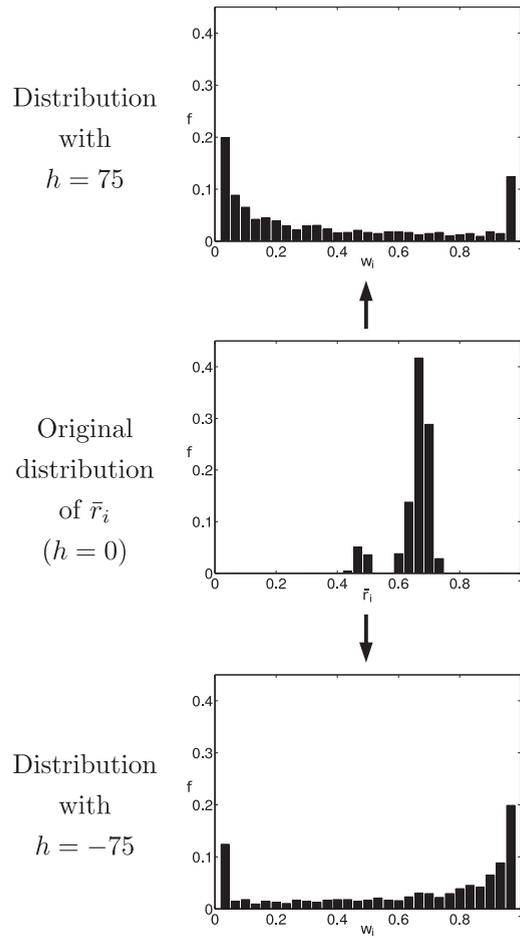


Fig. 5. Adjustment of the distribution of  $\bar{r}_i$  using a sigmoid function with two different values for parameter  $h$ .

groups in ensemble  $\Pi$ . If small increases in diversity are desired, it is intuitive to increase the size of the most different groups slightly, whereas the proportion of the most similar groups is reduced. By contrast, the opposite operation should decrease the diversity. Both mechanisms provide a way of controlling the diversity.

The similarity of a group with the rest of the ensemble,  $\bar{r}_i$ , is used to calculate a weight  $w_i$ . The mechanism employed to obtain fine-grained control over ensemble diversity involves applying a nonlinear function to  $\bar{r}_i$  values. This function is used to adjust the distribution of  $\bar{r}_i$  in a smooth manner, thereby allowing small increases or decreases in diversity. In the proposed method, we employ a sigmoid function and the computation of the group weights is given by

$$w_i(h) = \frac{p_i}{1 + e^{-h(\bar{r}_i - \bar{r})}}, \tag{3}$$

where  $p_i$  is the size of group  $\Pi_i$ ,  $\bar{r}$  is the mean of  $\bar{r}_i$ , and  $h$  controls the change in diversity. A new ensemble  $\Pi^*$  can be created for each  $h$  (with  $\Pi^* = \Pi$  for  $h = 0$ ). Thus, when  $h > 0$ , the method makes  $\Pi^*$  more diverse than  $\Pi$  because the most different groups in the ensemble receive higher weights. By contrast, when  $h < 0$ ,  $\Pi^*$  has lower diversity than  $\Pi$  because the most similar groups are favored. It is possible to obtain smooth increases (decreases) in the ensemble diversity by gradually increasing (decreasing)  $h$ .

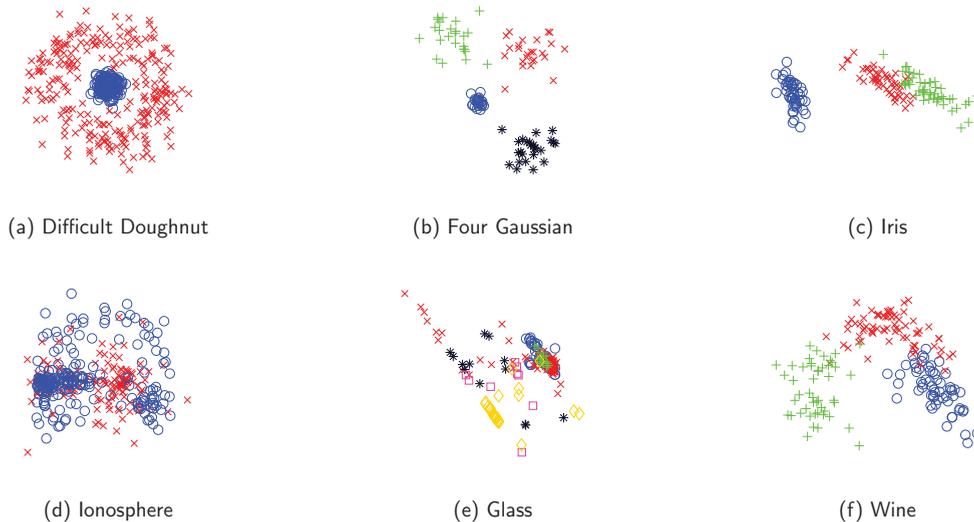
An example of the application of this method is shown in Fig. 5, which presents three histograms. The distribution of  $\bar{r}_i$  (middle) is modified by using two different values for  $h$  (top and bottom). The histogram in the middle shows that  $\bar{r}_i$  is greater than the mean for the majority of the groups. This explains why most of the groups are favored when the method tries to decrease the diversity (see the histogram at the bottom with a sigmoid using  $h = -75$ ). The opposite effect is obtained when the method attempts to increase the ensemble diversity ( $h = 75$ ).

After obtaining the weights  $w_i$ , the new groups in ensemble  $\Pi^*$  are produced by resampling the groups in  $\Pi$  with probability  $P_i = w_i / \sum_i w_i$ . For example, if we suppose that there are three groups, which obtain weights of  $w_1 = 0.8$ ,  $w_2 = 0.3$  and  $w_3 = 0.4$  for a particular value of  $h$ , and the size of ensemble  $\Pi^*$  is 100, then according to  $P_1$ ,  $P_2$  and  $P_3$ , 53 of its members will be drawn uniformly from group  $\Pi_1$ , 20 from  $\Pi_2$  and 27 from  $\Pi_3$ . When a slightly different value is used

**Table 2**

Descriptions of the artificial and real data sets used in the experiments.  $N$  is the number of data points,  $D$  the number of dimensions, and  $k$  the number of clusters.

Data set	$N$	$D$	$k$
Difficult Doughnut	500	12	2
Four Gaussian	100	12	4
Iris	150	4	3
Ionosphere	351	34	2
Glass	214	9	6
Wine	178	13	3



**Fig. 6.** Cluster shapes for the data sets employed. For artificial data sets (6a and 6b) the two noiseless dimensions were used to obtain the plots. For real data sets, principal components analysis projection was employed.

for  $h$  to calculate  $w_i$ , the amount of partitions drawn from each group will also change slightly, thereby obtaining smooth changes in diversity.

The final step in the overall method involves deriving a consensus partition from ensemble  $\Pi^*$ . In previous studies, several consensus functions have been introduced and different approaches have been used to combine a set of partitions. Among these consensus methods, some of the most popular are based on graph representations [5,9,31,36], and co-association matrices or evidence accumulation [12,13]. Any consensus approach can be employed during this step in the proposed method.

### 3. Materials and performance measures

In this section, we describe the data sets and performance measures used to test the proposed method. These measures were used to quantify the performance of the diversity control method in different steps.

#### 3.1. Data sets

Six well-known artificial and real data sets were employed (Table 2). Difficult Doughnut and Four Gaussian [26] were created artificially by generating two dimensions of data and then adding 10 dimensions of uniform noise. These two data sets had very different cluster shapes (see Fig. 6a and b), thereby representing different levels of difficulty for the clustering algorithm. The real data sets were obtained from the UCI repository [1]. As shown in Table 2 and Fig. 6, they contained different number of objects, dimensions, and classes, and their clusters had distinct shapes and degrees of overlapping.

#### 3.2. Performance measures

The diversity control method has different steps, so it can be evaluated at the end of each. We assessed its performance using different types of measures, which quantified three distinct aspects of the overall process. The first type comprised *ensemble measures* for quantifying the diversity of the ensembles generated. The next type comprised *clustering measures*,

which we used to observe the impact of the generated ensembles on the consensus partition from different perspectives. Finally, a smoothness measure was used to evaluate the fine-grained control over ensemble diversity, which assessed how the ensemble and clustering measures evolved with diversity. A higher level of smoothness was associated with better control over the ensemble diversity.

### 3.2.1. Ensemble measures

*Pairwise diversity.* Two pairwise measures were used to quantify the ensemble diversity. Given an ensemble  $\Pi^*$  of size  $P$ , the first measures the dissimilarity among the ensemble members [8], which can be defined as the mean of the ensemble members dissimilarities

$$D_{p1} = \frac{2}{P(P-1)} \sum_{i=1}^{P-1} \sum_{j=i+1}^P (1 - \Upsilon(\pi_i, \pi_j)). \quad (4)$$

The second measure,  $D_{p2}$ , quantifies the spread of the diversity [15], which is calculated as the standard deviation of the dissimilarities among the ensemble members.

*Non-pairwise diversity.* The average NMI (ANMI) [36] indicates the average amount of information shared between the consensus partition  $\pi^*$  and the members of its corresponding ensemble  $\Pi^*$ ,

$$\tilde{\Upsilon} = \frac{1}{P} \sum_{\forall \pi_i \in \Pi^*} \Upsilon(\pi^*, \pi_i). \quad (5)$$

ANMI can be considered as a level of representativeness of  $\pi^*$ . The complement of (5) can be used as an ensemble diversity measure [15,27], where  $D_{np1}$  is defined as the average of the differences between  $\pi^*$  and each member of  $\Pi^*$ . The standard deviation of these differences represents another measure, which is denoted as  $D_{np2}$ .

### 3.2.2. Clustering measures

These measures comprise external and internal criteria for validating consensus partitions [16,45]. The external criteria compare the consensus partition with a reference partition of the data, which is drawn independently. By contrast, internal criteria validate  $\pi^*$  using only the inherent information related to the data.

*Accuracy:* refers to the degree of agreement with a reference or true partition. The accuracy is calculated as

$$A = \Upsilon(\pi^*, \tilde{\pi}), \quad (6)$$

where  $\tilde{\pi}$  is the reference partition of the data, which is often based on the class labels (which were available for all of the data sets used in this study).

*Compactness:* measures the cluster homogeneity in a partition by calculating the intra-cluster variance. The global compactness for partition  $\pi^*$  is given by

$$C = \frac{1}{k} \sum_i^k C_i = \frac{1}{k} \sum_i^k \left( \frac{1}{|\Omega_i|} \sum_{\forall \mathbf{x}_j \in \Omega_i} \|\mathbf{x}_j - \boldsymbol{\mu}_i\|_2 \right), \quad (7)$$

where  $k$  is the number of clusters  $\Omega_i$  in  $\pi^*$ ,  $\mathbf{x}_j$  is a data vector, and  $\boldsymbol{\mu}_i$  is the centroid of the cluster. Values of  $C$  closer to 0 indicate more compact clusters.

*Separation:* quantifies the degree of separation between clusters in a partition. This measure is given by

$$S = \frac{2}{k(k-1)} \sum_{i=1}^{k-1} \sum_{j=i+1}^k \|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|_2. \quad (8)$$

Higher values of  $S$  indicate greater separation of the clusters in partition  $\pi^*$ .

*Davies–Bouldin index:* this measure combines compactness and separation [4], and it is defined as

$$B = \frac{1}{k} \sum_{i=1}^k \max_{j \neq i} \left( \frac{C_i + C_j}{\|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|_2} \right). \quad (9)$$

This index is a function of the ratio of the sum of the within-cluster scatter relative to the between-cluster separation. Therefore,  $B$  values close to 0 indicate that a partition  $\pi^*$  has compact and separated clusters.

*Dunn index:* this index combines the dissimilarity between clusters and their diameters [6], and it is given by

$$D = \frac{\min_{0 < m \neq n \leq k} \left\{ \min_{\substack{\forall \mathbf{x}_i \in \Omega_m \\ \forall \mathbf{x}_j \in \Omega_n}} \{\|\mathbf{x}_i - \mathbf{x}_j\|_2\} \right\}}{\max_{0 < m \leq k} \max_{\forall \mathbf{x}_i, \mathbf{x}_j \in \Omega_m} \{\|\mathbf{x}_i - \mathbf{x}_j\|_2\}}. \quad (10)$$

This index measures the relationship between the minimum inter-cluster distance and the maximum intra-cluster distance. If the partition contains well-separated clusters, the minimum distance among them is usually large and their maximum diameter is expected to be small. Therefore, larger values of  $D$  indicate better partitions.

### 3.2.3. Smoothness measure

Control over the diversity of an ensemble can be assessed by studying the evolution of ensemble and clustering measures, which involves observing the behavior of the inputs of the consensus function (ensembles  $\Pi^*$ ) and their corresponding outputs (consensus partitions  $\pi^*$ ). On the one hand, if a single diversity measure is being controlled, it is expected that other diversity measures will change in a smooth manner. On the other hand, current approaches assume that ensembles with similar diversity values are similar inputs for the consensus function, and thus they are expected to produce comparable consensus partitions. To verify whether this assumption actually holds, clustering measures can be calculated over the consensus partitions. If they change in a smooth manner, this indicates that ensembles with similar diversity are actually similar inputs for the consensus function.

The autocorrelation coefficient at lag 1 can be used as a smoothness measure. It is known that this coefficient can be employed to detect non-randomness [3], and this conforms with the intuitive concept of smoothness. From this viewpoint, the randomness in a sequence refers to the independence or unpredictability of one value relative to another. For example, for accuracy, this coefficient can be calculated as

$$\rho_1 = \frac{\sum_d (A_d - \bar{A})(A_{d+1} - \bar{A})}{\sum_d (A_d - \bar{A})^2}, \quad (11)$$

where  $A$  is a sequence of accuracies sorted by diversity  $d$  and  $\bar{A}$  is their mean. Any of the ensemble or clustering measures defined previously can be used instead of the accuracy. Thus, values of  $\rho_1$  closer to 1 indicate the smooth behavior of a sequence of values, so control over the diversity of the ensemble is better. By contrast,  $\rho_1$  near 0 indicates rough changes, thereby demonstrating almost no control of the diversity of an ensemble.

## 4. Results and discussion

We evaluated the proposed method in several different ways, as described in this section. First, we specified the experimental setup and determined the ability of the method to control the ensemble diversity for all of the data sets described in Section 3.1. After that, the evolution of the clustering measures was used to indicate the effects of the method on the performance of the consensus function. Finally, we calculated the autocorrelation at lag 1 ( $\rho_1$ ) for the ensemble and clustering measures.

### 4.1. Experimental setup

For phase 1 in the ensemble generation step (Section 2.1), the base clustering algorithm employed was  $k$ -means, with  $k_{min} = 2$  and  $k_{max} = 20$  ( $m = 19$ ), and 20 random initializations ( $n = 20$ ) for each  $k$ . The number of groups of partitions produced in phase 2 was 19 for both Gk and FG ( $q = 19$ ). The representative of each group and the consensus partition  $\pi^*$  obtained at the final step were derived using a supra-consensus function [36]. This supra-consensus function employed 10 different consensus methods and it selected the method that maximized the ANMI. These internal methods comprised three graph-based functions: Cluster-based similarity partitioning algorithm (CSPA), HyperGraph partitioning algorithm (HGPA) and Meta-Clustering Algorithm (MCLA) [36], as well as seven methods based on evidence accumulation [13]. The latter methods accumulated the results found by all of the ensemble members in a co-association matrix, before applying a hierarchical agglomerative clustering algorithm to derive a consensus partition. Seven types of distances were used as linkage criteria: nearest (single), furthest (complete), unweighted average, weighted average, unweighted center of mass, weighted center of mass, and inner squared (minimum variance). The entire experiment was repeated 50 times.

For further details and testing with more experimental setups, the full source code<sup>1</sup> is freely available for download. In addition, the diversity control algorithm can be tested rapidly using a *web-demo*<sup>2</sup>, which is an accessible web interface developed with [35].

### 4.2. Control of ensemble diversity

The diversity control results for all of the data sets are shown in Fig. 7, in which the pairwise ensemble diversity ( $D_{p1}$ ) is plotted as a function of the sigmoid parameter  $h$ . Each repetition is indicated by a different line style and color. A single ensemble  $\Pi^*$  was created for each  $h$  value by considering a wide interval  $[-100, 100]$  in order to observe how the method behaved with intermediate cases, but also with the extreme cases. The size of  $\Pi^*$  was 100 and the grouping algorithm was FG.

The results show that although different behaviors were observed for each data set, in all cases the method could increase and decrease the pairwise ensemble diversity by adjusting  $h$ . In addition, the method effectively produces a smooth change in  $D_{p1}$ , where the full diversity range was uniformly sampled. When the method was used to decrease or increase  $D_{p1}$ , each curve finally converged to a certain level, although each arrived at different lower and upper bounds. Saturation

<sup>1</sup> <https://sourceforge.net/projects/sourcesinc/files/divcontrol/0.10/>.

<sup>2</sup> <http://fich.unl.edu.ar/sinc/web-demo/divcontrol/>.

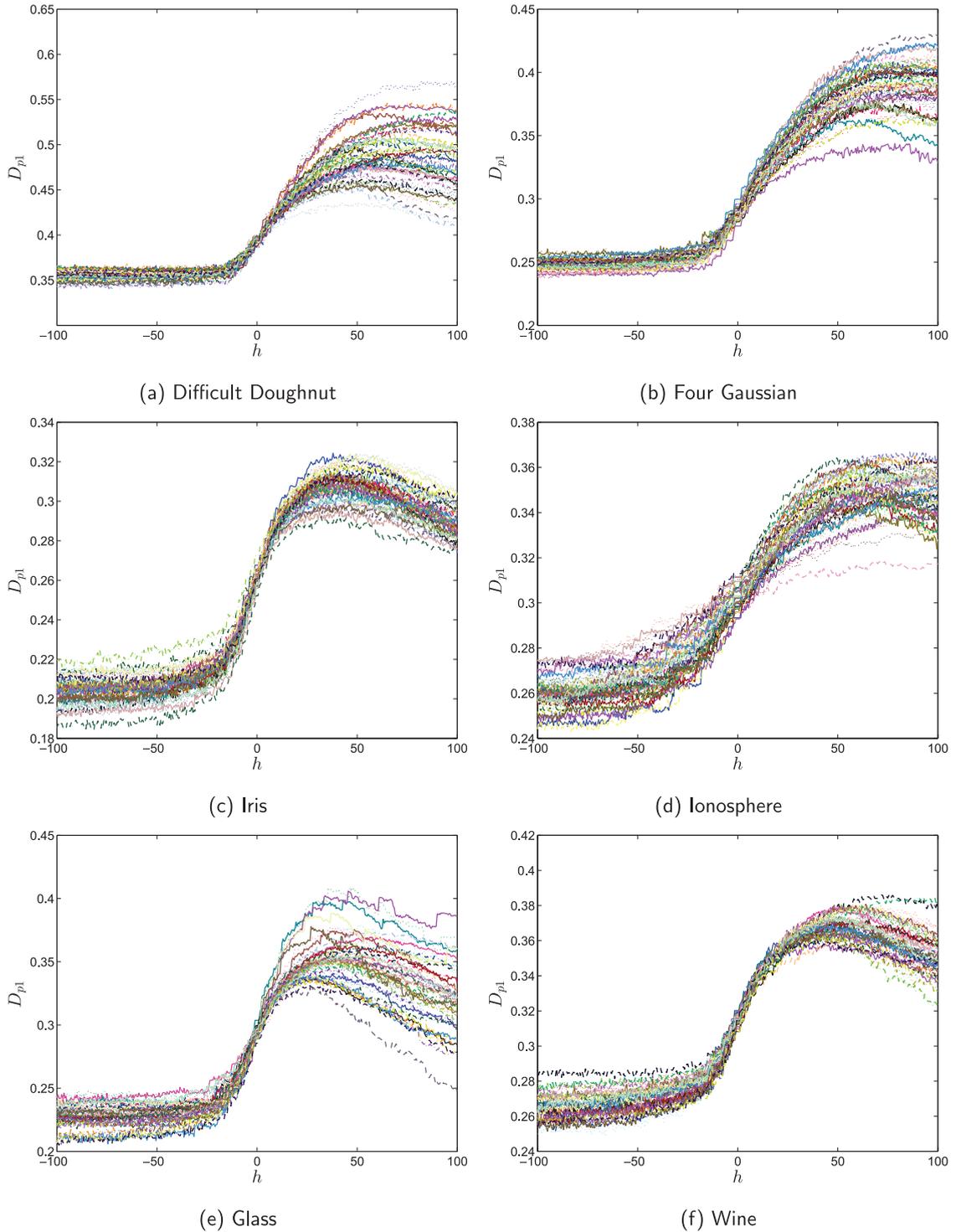


Fig. 7. Evolution of the control of diversity for six data sets.

occurred at both sides of the curve because when higher values were used for  $h$ , minor changes were observed in the sigmoid function, and thus there was almost no difference in the size of the groups among the newly created ensembles. In this extreme situation, the only source of change was the random selection of members from each group of partitions. Therefore, no changes were expected in the ensemble diversity at the highest values of  $h$ . Another interesting behavior was observed as the diversity increased ( $h > 0$ ), where a point of maximum diversity was reached and it then decreased to a

**Table 3**  
Smoothness of the ensemble measures.

$\rho_1$	Difficult Doughnut					Four Gaussian				
	RN	CB	Gk	FG	FGk	RN	CB	Gk	FG	FGk
$D_{p2}$	0.69	0.95	<b>0.98</b>	<u>0.98</u>	<u>0.98</u>	0.75	0.95	<u>0.98</u>	<u>0.98</u>	<b>0.98</b>
$D_{np1}$	0.66	0.71	<u>0.96</u>	<u>0.97</u>	<b>0.98</b>	0.15	0.74	<u>0.95</u>	<u>0.97</u>	<b>0.98</b>
$D_{np2}$	0.48	<u>0.95</u>	0.86	<b>0.97</b>	<u>0.92</u>	0.03	0.57	0.64	0.80	<b>0.89</b>
$\rho_1$	Iris					Ionosphere				
	RN	CB	Gk	FG	FGk	RN	CB	Gk	FG	FGk
$D_{p2}$	0.61	0.95	<u>0.98</u>	<u>0.98</u>	<b>0.98</b>	0.17	0.90	<u>0.97</u>	<b>0.98</b>	<u>0.98</u>
$D_{np1}$	0.07	0.70	<u>0.97</u>	<u>0.97</u>	<b>0.98</b>	0.65	0.62	<u>0.98</u>	<b>0.98</b>	<u>0.98</u>
$D_{np2}$	0.10	0.84	<u>0.96</u>	<u>0.97</u>	<b>0.97</b>	0.50	0.72	<u>0.97</u>	<u>0.97</u>	<b>0.98</b>
$\rho_1$	Glass					Wine				
	RN	CB	Gk	FG	FGk	RN	CB	Gk	FG	FGk
$D_{p2}$	0.51	0.95	0.97	0.97	<b>0.98</b>	0.56	0.93	<u>0.98</u>	<u>0.98</u>	<b>0.98</b>
$D_{np1}$	0.46	0.92	<u>0.94</u>	<u>0.96</u>	<b>0.97</b>	0.02	0.64	<u>0.96</u>	<u>0.96</u>	<b>0.96</b>
$D_{np2}$	0.02	0.86	<u>0.92</u>	0.88	<b>0.94</b>	0.17	0.73	<u>0.93</u>	<b>0.94</b>	<u>0.93</u>

certain level. This change occurred because high values of  $h$  made some  $w_i \approx 0$ , thereby producing empty groups, whereas others  $w_i$  were sufficiently large to introduce repeated partitions.

The results show clearly that the proposed method could decrease or increase the ensemble diversity in an effective manner. A wide range was employed for  $h$ , but in practice, it would be necessary to focus on a more useful range for  $h$  values in the vicinity of  $h = 0$ , where the diversity exhibits large changes.  $h$  can take continuous values, so it is easy to reduce the step size and explore this range, which we tested in the following experiments. The diversity could be controlled independently of the data characteristics by controlling the diversity for data sets with compact and well-separated clusters (e.g., Four Gaussian), as well as those with far more complex structures (such as Ionosphere and Glass). In addition, it should be noted that the full diversity range was uniformly sampled. This results in a fine-grained method to control the level of disagreement among ensemble members in a smooth manner.

#### 4.3. Performance evaluation

Next, we calculated the classical performance measures described in Section 3.2 for the final consensus partition, which allowed us to observe how these quantities evolved for ensembles with controlled diversity. The results are shown in Fig. 8 for the six measures calculated based on the Wine data set<sup>3</sup>. The average of all the repetitions is plotted for each  $h$  as well as the corresponding confidence intervals in gray ( $\alpha = 0.05$ ).

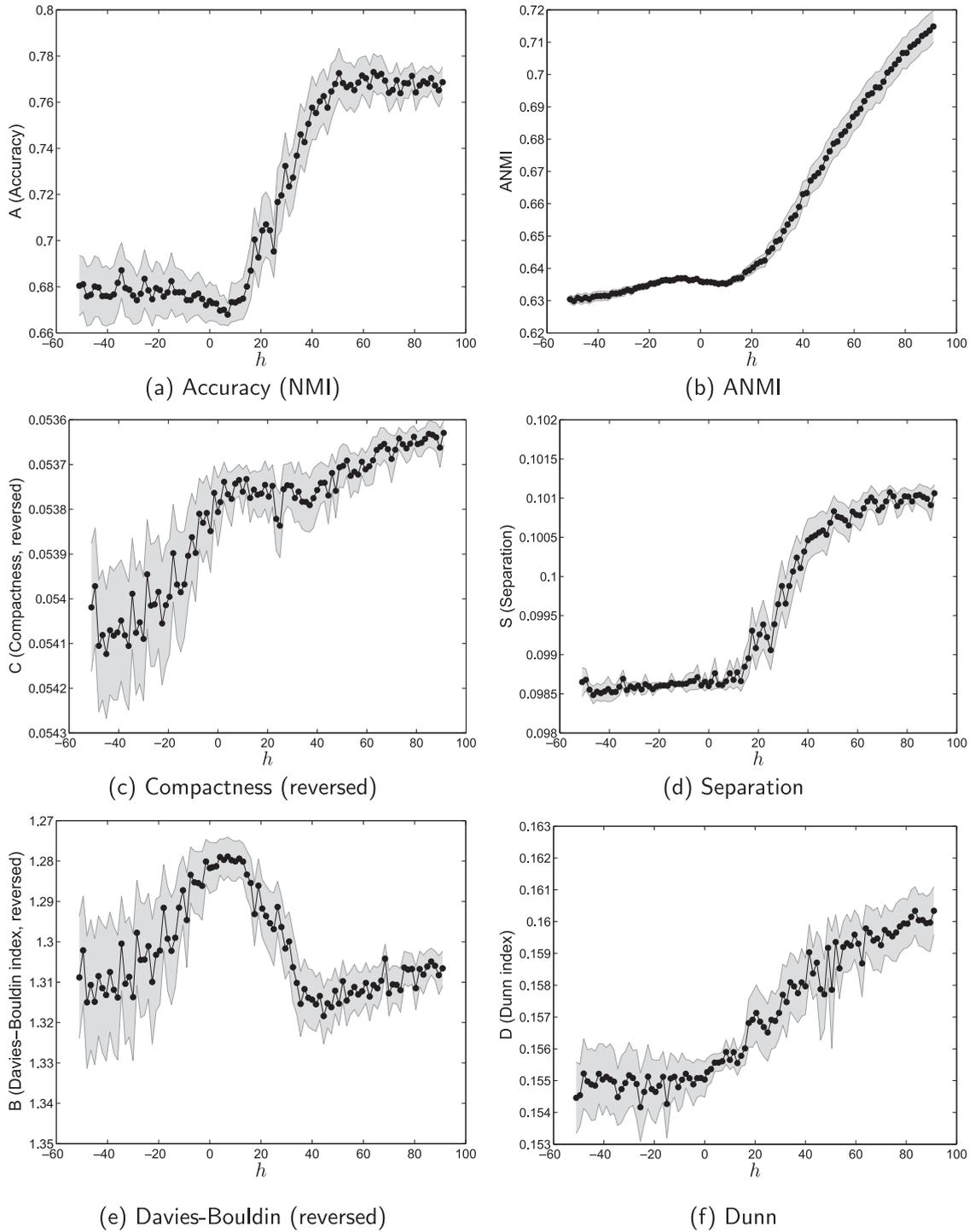
In addition to the control of ensemble diversity, as demonstrated earlier, the results showed that the proposed method could also induce smooth changes in the performance measures. In most cases, the quantities tended to improve with higher values of  $h$  (more diverse ensembles). Most of them did not appear to be affected greatly, but the accuracy and ANMI exhibited relatively high increases. Thus, the partitions  $\pi^*$  appeared to have almost equally compact and separated clusters for all values of  $h$ , but they definitely represented different partitions of the data. Therefore, the proposed method could change the final partition properties in a smooth manner by controlling the diversity. The ability to produce smooth changes in the performance measures is a very important feature of the proposed method and future studies could explore the influence of the ensemble on the consensus quality.

#### 4.4. Smoothness of the performance measures

The smoothness ( $\rho_1$ ) was determined for the ensemble and clustering measures using different approaches for ensemble generation. Two existing methods were compared with our proposed approach: a random generation method (RN) described in [8,15,17,27], which randomly creates a set of ensembles with different diversities; and the method proposed by Lam-On et al. [20], which we refer to as category-based (CB), where this method generates a set of ensembles that belong to “low,” “medium,” and “high” diversity categories. The proposed method for diversity control used three alternative algorithms to generate groups: Gk, FG and FGk. Only a useful range for  $h$  was employed in these tests, i.e., a range where the diversity exhibited large changes. Sequences of ensembles were generated for these five methods and sorted by ascending diversity. The average smoothness of the ensemble and clustering measures were determined for all of the repetitions.

The smoothness results for the ensemble measures are presented in Table 3, where the average  $\rho_1$  was calculated for three diversity measures:  $D_{p2}$ ,  $D_{np1}$  and  $D_{np2}$ . Bold numbers indicate the maximum  $\rho_1$  obtained for each data set and en-

<sup>3</sup> Full results for all of the data sets are available in the supplementary material.



**Fig. 8.** Six performance measures calculated for the consensus partitions obtained from the Wine data set. The compactness and Davies-Bouldin index are presented in reverse, so the values at the top are better for all of the measures.

semble measure. The differences between underlined and non-underlined values are statistically significant ( $\alpha = 0.05$ ). For all of the data sets, we found that RN always obtained the lowest scores for  $\rho_1$ . Thus, the ensemble diversity values for  $D_{p2}$ ,  $D_{np1}$  and  $D_{np2}$  changed roughly compared with  $D_{p1}$ . By contrast, CB obtained higher  $\rho_1$  values, which indicates that it could generate ensembles that changed more smoothly than the random approach. This may be explained by the “greedy” nature of CB when creating the diversity categories because although it contains a random component (the first partition of the ensemble is selected randomly), this method selects the next partition incrementally according to the target diversity cate-

**Table 4**  
Smoothness of the clustering measures.

$\rho_1$	Difficult Doughnut					Four Gaussian				
	RN	CB	Gk	FG	FGk	RN	CB	Gk	FG	FGk
A	0.03	<u>0.91</u>	0.77	<b>0.95</b>	0.82	0.03	0.64	0.60	<u>0.76</u>	<b>0.79</b>
C	0.02	<u>0.88</u>	0.68	<b>0.93</b>	0.77	0.03	0.47	0.55	0.66	<b>0.75</b>
S	0.02	<u>0.92</u>	0.77	<b>0.95</b>	0.82	0.04	0.47	<u>0.60</u>	<u>0.72</u>	<b>0.77</b>
B	0.02	0.89	0.80	<b>0.95</b>	0.82	0.03	0.17	<u>0.60</u>	<u>0.71</u>	<b>0.78</b>
D	0.28	0.35	0.51	<b>0.73</b>	0.57	0.02	0.57	0.65	0.74	<b>0.80</b>
$\rho_1$	Iris					Ionosphere				
	RN	CB	Gk	FG	FGk	RN	CB	Gk	FG	FGk
A	0.04	0.39	0.61	<b>0.81</b>	0.71	0.06	<b>0.58</b>	0.32	0.43	0.41
C	0.01	<u>0.45</u>	0.29	<u>0.59</u>	<b>0.70</b>	0.19	0.18	0.34	<b>0.50</b>	0.31
S	0.01	0.41	0.48	<b>0.78</b>	<u>0.75</u>	0.15	<b>0.61</b>	0.57	0.46	0.55
B	0.07	0.16	0.52	<b>0.76</b>	<u>0.70</u>	0.23	0.50	<b>0.56</b>	0.45	0.52
D	0.07	0.44	0.67	<b>0.85</b>	<u>0.71</u>	0.05	0.27	<b>0.45</b>	0.33	0.27
$\rho_1$	Glass					Wine				
	RN	CB	Gk	FG	FGk	RN	CB	Gk	FG	FGk
A	0.04	0.32	<u>0.74</u>	<b>0.83</b>	<u>0.78</u>	0.05	0.40	0.56	<b>0.71</b>	0.62
C	0.03	0.28	<u>0.60</u>	<u>0.60</u>	<b>0.70</b>	0.03	<b>0.46</b>	0.28	0.37	0.39
S	0.11	0.57	<u>0.87</u>	<u>0.88</u>	<b>0.90</b>	0.03	0.55	0.72	<b>0.81</b>	0.72
B	0.11	0.42	<u>0.87</u>	<u>0.88</u>	<b>0.89</b>	0.03	0.35	0.28	<b>0.44</b>	0.38
D	0.04	0.57	<u>0.77</u>	<u>0.77</u>	<b>0.82</b>	0.04	0.33	0.37	<b>0.50</b>	0.39

gory. These diversity categories generally produce a set of similar ensembles, thereby obtaining smooth differences among them. However, this behavior has a side effect because the diversity categories frequently differ greatly from each other, so CB does not sample uniformly from the diversity range.

The diversity control method (with all three alternatives) obtained the highest  $\rho_1$  values for all of the data sets, with statistically significant differences compared with the other methods. This is because the ability to control diversity resulted in small changes among the ensembles, and thus *all* of the ensembles differed from each other in a smooth manner, whereas CB created cohesive but separate ensemble categories. Overall, all of the grouping algorithms obtained similar performance, but FGk produced the best results followed closely by FG. The higher performance of these algorithms compared with Gk was due to the cohesion between the generated groups of partitions. Indeed, Gk only groups by  $k$ , which might leave very different partitions within the same group.

Table 4 presents the changes in the outputs of the consensus function, where five clustering measures are shown: accuracy (A), compactness (C), separation (S), the Davies–Bouldin index (B), and the Dunn index (D). The trends in these results were similar to those obtained for the ensemble measures. The ensembles generated by RN allowed the consensus function to derive unpredictable partitions. Using this method, the smoothness scores for clustering measures were always close to 0, which indicates that the ensembles changed in a highly irregular manner according to the measures based on the output of the consensus function. Although CB was far superior to RN, the diversity control method obtained the best performance. In addition to producing ensembles that changed in a smooth manner from the viewpoint of diversity (Table 3), the diversity control method could also induce smooth changes in the consensus partitions.

For the diversity control method, a joint analysis of the smoothness for the ensemble and clustering measures showed that given a slight change in the input, the consensus function could produce an acceptable slight change in the output. This was the case for the Four Gaussian, Iris, and Glass data sets, but especially for Difficult Doughnut, where the proposed method appeared to even control the quality of the consensus partition.

These results suggest that current methods can generate ensembles with differences in diversity, but their internal structures might actually differ in an unpredictable manner. Thus, these methods are less useful and reliable in cases where it is necessary to analyze how diversity affects the consensus performance. By contrast, our proposed method can control a particular measure of ensemble diversity (as shown in Section 4.2), but it also guarantees that all aspects of the ensemble change in a smooth manner. This was demonstrated by the high smoothness scores obtained from the perspectives of the diversity and consensus function. Therefore, it can be stated that the proposed method is more suitable for generating ensembles that differ according to the fine-grained control of their diversity.

## 5. Conclusions

In this study, we identified and addressed a problem in existing methods for generating diverse ensembles. Thus, when these methods are used to create sets of clustering solutions with several levels of disagreement, ensembles with similar diversity may actually have very different properties, which leads to unpredictable behavior by the consensus function. As a

consequence, studying ensemble diversity is a difficult problem. By contrast, our proposed method can increase and decrease the dissimilarity among ensemble members in a smooth manner, thereby providing a novel approach for ensemble diversity analysis. Our method analyzes the structure of the ensemble and makes small changes to it, which allows the fine-grained control of diversity. The performance of our method was demonstrated experimentally by evaluating different aspects of the overall method. The proposed method could produce ensembles that changed not only in a smooth manner from the viewpoint of diversity, but also according to different quality measures based on the output of the consensus function.

The proposed diversity control method is an important step in the development of a more robust approach for effectively exploring how ensemble diversity affects the final quality of clusters. In future research, we will perform more in-depth investigations using different data sets, base clustering algorithms, consensus functions, and validation measures.

## Acknowledgments

This study was supported by the National Scientific and Technical Research Council [PIP 2013 117], National University of Litoral [CAI+D 2011 548], and National Agency for Science and Technology Promotion (ANPCyT) [PICT 2014 2627].

## Supplementary material

Supplementary material associated with this article can be found, in the online version, at [10.1016/j.ins.2016.04.027](https://doi.org/10.1016/j.ins.2016.04.027).

## References

- [1] M. Lichman, {UCI} Machine Learning Repository, University of California, Irvine, School of Information and Computer Sciences, 2013 <http://archive.ics.uci.edu/ml>.
- [2] J. Azimi, X. Fern, Adaptive cluster ensemble selection, in: Proceedings of the Twenty-first International Joint Conference on Artificial Intelligence (IJCAI'09), Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2009, pp. 992–997.
- [3] G. Box, G. Jenkins, G. Reinsel, *Time Series Analysis: Forecasting and Control*, Wiley Series in Probability and Statistics, Wiley, 2013.
- [4] D.L. Davies, D.W. Bouldin, A cluster separation measure, *IEEE Trans. Pattern Anal. Mach. Intell. PAMI-1* (2) (1979) 224–227, doi:[10.1109/TPAMI.1979.4766909](https://doi.org/10.1109/TPAMI.1979.4766909).
- [5] C. Domeniconi, M. Al-Razgan, Weighted cluster ensembles: Methods and analysis, *ACM Trans. Knowl. Discov. Data* 2 (4) (2009) 17:1–17:40, doi:[10.1145/1460797.1460800](https://doi.org/10.1145/1460797.1460800).
- [6] J.C. Dunn, Well separated clusters and optimal fuzzy-partitions, *J. Cybern.* 4 (1) (1974) 95–104.
- [7] X.Z. Fern, W. Lin, Cluster ensemble selection, *Stat. Anal. Data Min.* 1 (3) (2008) 128–141, doi:[10.1002/sam.v1:3](https://doi.org/10.1002/sam.v1:3).
- [8] X.Z. Fern, C.E. Brodley, Random projection for high dimensional data clustering: A cluster ensemble approach, in: Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), 2003, pp. 186–193.
- [9] X.Z. Fern, C.E. Brodley, Solving cluster ensemble problems by bipartite graph partitioning, in: Proceedings of the Twenty-first International Conference on Machine Learning (ICML '04), ACM, New York, NY, USA, 2004, p. 36, doi:[10.1145/1015330.1015414](https://doi.org/10.1145/1015330.1015414).
- [10] A. Fiori, A. Mignone, G. Rospo, Decoclu: Density consensus clustering approach for public transport data, *Inf. Sci.* 328 (2016) 378–388, doi:[10.1016/j.ins.2015.08.054](https://doi.org/10.1016/j.ins.2015.08.054).
- [11] B. Fischer, J. Buhmann, Bagging for path-based clustering, *IEEE Trans. Pattern Anal. Mach. Intell.* 25 (11) (2003) 1411–1415, doi:[10.1109/TPAMI.2003.1240115](https://doi.org/10.1109/TPAMI.2003.1240115).
- [12] A.L.N. Fred, A. Jain, Data clustering using evidence accumulation, in: Proceedings of the Sixteenth International Conference on Pattern Recognition, vol. 4, 2002, pp. 276–2804, doi:[10.1109/ICPR.2002.1047450](https://doi.org/10.1109/ICPR.2002.1047450).
- [13] A. Fred, A. Jain, Combining multiple clusterings using evidence accumulation, *IEEE Trans. Pattern Anal. Mach. Intell.* 27 (6) (2005) 835–850, doi:[10.1109/TPAMI.2005.113](https://doi.org/10.1109/TPAMI.2005.113).
- [14] F. Gullo, A. Tagarelli, S. Greco, Diversity-based weighting schemes for clustering ensembles, in: Proceedings of the 2009 SIAM International Conference on Data Mining, 2009, pp. 437–448, doi:[10.1137/1.9781611972795.38](https://doi.org/10.1137/1.9781611972795.38). chapter 37.
- [15] S.T. Hadjitodorov, L.I. Kuncheva, L.P. Todorova, Moderate diversity for better cluster ensembles, *Inf. Fus.* 7 (3) (2006) 264–275, doi:[10.1016/j.inffus.2005.01.008](https://doi.org/10.1016/j.inffus.2005.01.008).
- [16] M. Halkidi, Y. Batistakis, M. Vazirgiannis, On clustering validation techniques, *J. Intell. Inf. Syst.* 17 (2–3) (2001) 107–145, doi:[10.1023/A:1012801612483](https://doi.org/10.1023/A:1012801612483).
- [17] J. Hu, T. Li, H. Wang, H. Fujita, Hierarchical cluster ensemble model based on knowledge granulation, *Knowl. Based Syst.* 91 (2016) 179–188, doi:[10.1016/j.knsys.2015.10.006](https://doi.org/10.1016/j.knsys.2015.10.006).
- [18] D. Huang, J. Lai, C.-D. Wang, Ensemble clustering using factor graph, *Pattern Recognit.* 50 (2016) 131–142, doi:[10.1016/j.patcog.2015.08.015](https://doi.org/10.1016/j.patcog.2015.08.015).
- [19] X. Huang, X. Zheng, W. Yuan, F. Wang, S. Zhu, Enhanced clustering of biomedical documents using ensemble non-negative matrix factorization, *Inf. Sci.* 181 (11) (2011) 2293–2302, doi:[10.1016/j.ins.2011.01.029](https://doi.org/10.1016/j.ins.2011.01.029).
- [20] N. Iam-On, T. Boongoen, S. Garrett, C. Price, A link-based approach to the cluster ensemble problem, *IEEE Trans. Pattern Anal. Mach. Intell.* 33 (12) (2011) 2396–2409, doi:[10.1109/TPAMI.2011.84](https://doi.org/10.1109/TPAMI.2011.84).
- [21] N. Iam-On, T. Boongoen, S. Garrett, C. Price, A link-based cluster ensemble approach for categorical data clustering, *IEEE Trans. Knowl. Data Eng.* 24 (3) (2012) 413–425, doi:[10.1109/TKDE.2010.268](https://doi.org/10.1109/TKDE.2010.268).
- [22] A.K. Jain, Data clustering: 50 years beyond k-means, *Pattern Recognit. Lett.* 31 (8) (2010) 651–666, doi:[10.1016/j.patrec.2009.09.011](https://doi.org/10.1016/j.patrec.2009.09.011).
- [23] L. Jing, K. Tian, J.Z. Huang, Stratified feature sampling method for ensemble clustering of high dimensional data, *Pattern Recognit.* 48 (11) (2015) 3688–3702, doi:[10.1016/j.patcog.2015.05.006](https://doi.org/10.1016/j.patcog.2015.05.006).
- [24] J. Kleinberg, An impossibility theorem for clustering, in: *Neural Information Processing Systems*, MIT Press, 2002, pp. 446–453.
- [25] G. Kou, Y. Peng, G. Wang, Evaluation of clustering algorithms for financial risk analysis using {MCDM} methods, *Inf. Sci.* 275 (2014) 1–12, doi:[10.1016/j.ins.2014.02.137](https://doi.org/10.1016/j.ins.2014.02.137).
- [26] L.I. Kuncheva, D.P. Vetrov, Evaluation of stability of k-Means cluster ensembles with respect to random initialization, *IEEE Trans. Pattern Anal. Mach. Intell.* 28 (11) (2006) 1798–1808, doi:[10.1109/TPAMI.2006.226](https://doi.org/10.1109/TPAMI.2006.226).
- [27] L. Kuncheva, S. Hadjitodorov, Using diversity in cluster ensembles, in: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, vol. 2, 2004, pp. 1214–12192, doi:[10.1109/ICSMC.2004.1399790](https://doi.org/10.1109/ICSMC.2004.1399790).
- [28] M. Law, A. Topchy, A. Jain, Multi objective data clustering, in: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2004), vol. 2, 2004, pp. II-424–II-430, doi:[10.1109/CVPR.2004.1315194](https://doi.org/10.1109/CVPR.2004.1315194).
- [29] J. Lee, K.-H. Hsu, Gea: A goal-driven approach to discovering early aspects, *IEEE Trans. Softw. Eng.* 40 (6) (2014) 584–602, doi:[10.1109/TSE.2014.2322368](https://doi.org/10.1109/TSE.2014.2322368).
- [30] L. Limin, F. Xiaoping, A new selective clustering ensemble algorithm, in: Proceedings of the Ninth IEEE International Conference on e-Business Engineering (ICEBE), 2012, pp. 45–49, doi:[10.1109/ICEBE.2012.17](https://doi.org/10.1109/ICEBE.2012.17).

- [31] S. Mimaroglu, E. Erdil, Combining multiple clusterings using similarity graph, *Pattern Recognit.* 44 (3) (2011) 694–703, doi:[10.1016/j.patcog.2010.09.008](https://doi.org/10.1016/j.patcog.2010.09.008).
- [32] B. Minaei-Bidgoli, A.P. Topchy, W.F. Punch, in: *A Comparison of Resampling Methods for Clustering Ensembles*, IC-AI, 2004, pp. 939–945.
- [33] M. Naldi, A. Carvalho, R. Campello, Cluster ensemble selection based on relative validity indexes, *Data Min. Knowl. Discov.* 27 (2) (2013) 259–289, doi:[10.1007/s10618-012-0290-x](https://doi.org/10.1007/s10618-012-0290-x).
- [34] G. Stegmayer, D.H. Milone, L. Kamenetzky, M.G. López, F. Carrari, A biologically-inspired validity measure for comparison of clustering methods over metabolic datasets, *IEEE ACM Trans. Comput. Biol. Bioinf.* 9 (3) (2012) 706–716.
- [35] G. Stegmayer, M. Pividori, D.H. Milone, A very simple and fast way to access and validate algorithms in reproducible research, *Brief. Bioinf.* 17 (1) (2016) 180–183 <http://bib.oxfordjournals.org/content/early/2015/07/27/bib.bbv054.abstract>.
- [36] A. Strehl, J. Ghosh, C. Cardie, Cluster ensembles – A knowledge reuse framework for combining multiple partitions, *J. Mach. Learn. Res.* 3 (2002) 583–617.
- [37] A. Topchy, A. Jain, W. Punch, Combining multiple weak clusterings, in: *Proceedings of the Third IEEE International Conference on Data Mining, (ICDM 2003)*, 2003, pp. 331–338, doi:[10.1109/ICDM.2003.1250937](https://doi.org/10.1109/ICDM.2003.1250937).
- [38] A. Topchy, B. Minaei-Bidgoli, A. Jain, W. Punch, Adaptive clustering ensembles, in: *Proceedings of the Seventeenth International Conference on Pattern Recognition, (ICPR 2004)*, vol. 1, 2004, pp. 272–2751, doi:[10.1109/ICPR.2004.1334105](https://doi.org/10.1109/ICPR.2004.1334105).
- [39] A. Topchy, A. Jain, W. Punch, Clustering ensembles: Models of consensus and weak partitions, *IEEE Trans. Pattern Anal. Mach. Intell.* 27 (12) (2005) 1866–1881, doi:[10.1109/TPAMI.2005.237](https://doi.org/10.1109/TPAMI.2005.237).
- [40] R. Vidal, Subspace clustering, *IEEE Signal Process. Mag.* 28 (2) (2011) 52–68, doi:[10.1109/MSP.2010.939739](https://doi.org/10.1109/MSP.2010.939739).
- [41] F. Wang, C. Zhang, N. Lu, Boosting GMM and its two applications, in: *Multiple Classifier Systems*, in: *Lecture Notes in Computer Science*, vol. 3541, Springer Berlin Heidelberg, 2005, pp. 12–21, doi:[10.1007/11494683\\_2](https://doi.org/10.1007/11494683_2).
- [42] F. Wang, X. Wang, T. Li, Generalized cluster aggregation, in: *Proceedings of the Twenty-first International Joint Conference on Artificial Intelligence (IJCAI)*, 2009, pp. 1279–1284.
- [43] J. Wu, H. Liu, H. Xiong, J. Cao, J. Chen, K-means-based consensus clustering: A unified view, *IEEE Trans. Knowl. Data Eng.* 27 (1) (2015) 155–169, doi:[10.1109/TKDE.2014.2316512](https://doi.org/10.1109/TKDE.2014.2316512).
- [44] R. Xu, D. Wunsch, Clustering algorithms in biomedical research: A review, *IEEE Rev. Biomed. Eng.* 3 (2010) 120–154, doi:[10.1109/RBME.2010.2083647](https://doi.org/10.1109/RBME.2010.2083647).
- [45] R. Xu, D. Wunsch, *Clustering*, Wiley-IEEE Press, 2009.
- [46] Y.-M. Xu, C.-D. Wang, J.-H. Lai, Weighted multi-view clustering with feature selection, *Pattern Recognit.* 53 (2016) 25–35, doi:[10.1016/j.patcog.2015.12.007](https://doi.org/10.1016/j.patcog.2015.12.007).
- [47] F. Yang, X. Li, Q. Li, T. Li, Exploring the diversity in cluster ensemble generation: Random sampling and random projection, *Expert Syst. Appl.* 41 (10) (2014) 4844–4866, doi:[10.1016/j.eswa.2014.01.028](https://doi.org/10.1016/j.eswa.2014.01.028).
- [48] Y. Yang, J. Jiang, Hybrid sampling-based clustering ensemble with global and local constitutions, *IEEE Trans. Neural Netw. Learn. Syst.* 27 (5) (2016) 952–965, doi:[10.1109/TNNLS.2015.2430821](https://doi.org/10.1109/TNNLS.2015.2430821).
- [49] Y. Yang, K. Chen, Temporal data clustering via weighted clustering ensemble with different representations, *IEEE Trans. Knowl. Data Eng.* 23 (2) (2011) 307–320, doi:[10.1109/TKDE.2010.112](https://doi.org/10.1109/TKDE.2010.112).
- [50] J. Yi, T. Yang, R. Jin, A. Jain, M. Mahdavi, Robust ensemble clustering by matrix completion, in: *Proceedings of the Twelfth IEEE International Conference on Data Mining (ICDM)*, 2012, pp. 1176–1181, doi:[10.1109/ICDM.2012.123](https://doi.org/10.1109/ICDM.2012.123).
- [51] Z. Yu, P. Luo, J. You, H.S. Wong, H. Leung, S. Wu, J. Zhang, G. Han, Incremental semi-supervised clustering ensemble for high dimensional data clustering, *IEEE Trans. Knowl. Data Eng.* 28 (3) (2016) 701–714, doi:[10.1109/TKDE.2015.2499200](https://doi.org/10.1109/TKDE.2015.2499200).
- [52] Z. Yu, H.-S. Wong, H. Wang, Graph-based consensus clustering for class discovery from gene expression data, *Bioinformatics* 23 (21) (2007) 2888–2896, doi:[10.1093/bioinformatics/btm463](https://doi.org/10.1093/bioinformatics/btm463).
- [53] Z. Yu, H.-S. Wong, J. You, G. Yu, G. Han, Hybrid cluster ensemble framework based on the random combination of data transformation operators, *Pattern Recognit.* 45 (5) (2012) 1826–1837.
- [54] Z. Yu, J. You, H.-S. Wong, G. Han, From cluster ensemble to structure ensemble, *Inf. Sci.* 198 (2012) 81–99, doi:[10.1016/j.ins.2012.02.019](https://doi.org/10.1016/j.ins.2012.02.019).
- [55] Z. Yu, L. Li, Y. Gao, J. You, J. Liu, H.-S. Wong, G. Han, Hybrid clustering solution selection strategy, *Pattern Recognit.* 47 (10) (2014) 3362–3375, doi:[10.1016/j.patcog.2014.04.005](https://doi.org/10.1016/j.patcog.2014.04.005).
- [56] Z. Yu, L. Li, H.-S. Wong, J. You, G. Han, Y. Gao, G. Yu, Probabilistic cluster structure ensemble, *Inf. Sci.* 267 (2014) 16–34, doi:[10.1016/j.ins.2014.01.030](https://doi.org/10.1016/j.ins.2014.01.030).
- [57] C. Zhong, X. Yue, Z. Zhang, J. Lei, A clustering ensemble: Two-level-refined co-association matrix with path-based transformation, *Pattern Recognit.* 48 (8) (2015) 2699–2709, doi:[10.1016/j.patcog.2015.02.014](https://doi.org/10.1016/j.patcog.2015.02.014).