Partition Selection Approach for Hierarchical Clustering Based on Clustering Ensemble

Sandro Vega-Pons and José Ruiz-Shulcloper

Advanced Technologies Application Center (CENATAV), Havana, Cuba

Abstract. Hierarchical clustering algorithms are widely used in many fields of investigation. They provide a hierarchy of partitions of the same dataset. However, in many practical problems, the selection of a *representative* level (partition) in the hierarchy is needed. The classical approach to do so is by using a cluster validity index to select the *best* partition according to the criterion imposed by this index. In this paper, we present a new approach based on the clustering ensemble philosophy. The *representative* level is defined here as the consensus partition in the hierarchy. In the consensus computation process, we take into account the similarity between partitions and information from the evaluation of partitions with different cluster validity indexes. An experimental comparison on several datasets shows the superiority of the proposed approach with respect to the classical approach.

Keywords: Hierarchical clustering, partition selection, clustering ensemble, cluster validity index.

1 Introduction

Clustering algorithms can be divided into *Partitional* and *Hierarchical* [1]. Partitional clustering algorithms create a partition of the data by grouping the objects in clusters according to their (dis)similarity values. On the other hand, hierarchical clustering algorithms build a hierarchy of nested partitions of a dataset. This hierarchy is usually associated to a *dendrogram*, which can be cut at different levels to obtain the different partitions in the hierarchy (see Fig. 1).

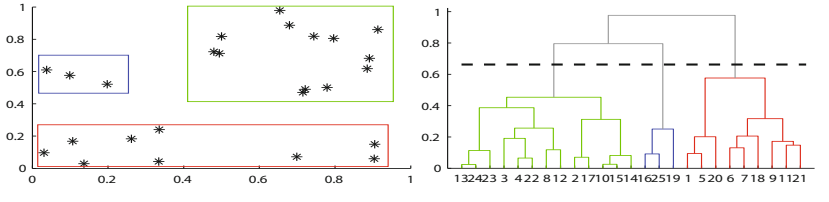

Fig. 1. A dataset of 25 2D points and the dendrogram produced by the Average-Link algorithm. The broken line cutting the dendrogram (right) produces a partition of the objects with 3 clusters (left).

I. Bloch and R.M. Cesar, Jr. (Eds.): CIARP 2010, LNCS 6419, pp. 525–532, 2010.

Hierarchy of partitions can offer more information about the structure of the objects in the dataset. With a hierarchy, the group of objects can be seen at different levels; from the bottom level where each object forms an independent cluster (singleton clusters) to the top level with only 1 cluster containing all the objects. However, working with the entire hierarchy is commonly very complex. Thus, in many practical problems, the selection of a *representative* partition of the hierarchy is needed.

In the traditional approach, the *representative* partition is obtained by using cluster validity indexes (CVI). Every partition in the hierarchy is evaluated by a CVI (used as *stopping rule*) and the partition with better results is selected. Many CVIs have been used with this purpose, e.g., in [2] 30 CVIs are presented and experimentally evaluated. Nowadays, classical CVIs like *Calinski-Harabasz* (CH) index, *Hartigan* (HA) index and the *Dunn* index [3] together with the *Highest-Lifetime*(HL) index [4] are some of the most used. However, new indexes still appear every year in the literature, e.g., the COP index [5].

The main drawback of the CVI (stopping rule) based approach to determine the *representative* level in a hierarchy is that there is no CVI capable of working *correctly* for all datasets and for all clustering algorithms. In other words, every CVI implicitly or explicitly evaluates a partition, according to a particular property given by the mathematical definition of the index. These properties are usually related to *compactness*, *separability* or *connectivity* among clusters. If the property measured by the index is *consistent* with the used clustering algorithm and the particular dataset, the index could contribute with valuable information, but if this is not the case, the results could be very different from the expected ones. Due to this limitation of CVIs, Everitt *et al.* [6] said that it is advisable not to depend on a single CVI for selecting the *representative* partition, but to synthesize the results of several indexes.

In this paper, we propose a new approach for the selection of the *representative* partition in a hierarchy based on the clustering ensemble philosophy. We call it *Partition Selection based on Cluster Ensemble* (PSCE) approach. With PSCE, we define the *representative* partition in a hierarchy taking into account the evaluation of several CVIs, as well as the similarity measures between partitions in the hierarchy. This way, we select as a result the partition in the hierarchy that better represents the common characteristics in the hierarchy.

In Section 2, the proposed approach is formally presented. In Section 3, experimental results by using different datasets and hierarchical clustering algorithms, as well as the comparison with the classical approach are shown. Finally, in Section 4, we present the conclusions of this research.

2 Partition Selection Based on Cluster Ensemble

Clustering ensemble methods combine partitions of the same dataset in a final consensus clustering. Formally, we denote $X = \{x_1, x_2, \ldots, x_n\}$ the original set of objects, where each x_i is a tuple of some $\alpha-$dimensional feature space $\mathbb{G}^\alpha$ for all $i = 1, \ldots, n$. $\mathbb{P} = \{P_1, P_2, \ldots, P_m\}$ is a clustering ensemble, where each

$P_j = \{C_1^j, C_2^j, \ldots, C_{d_j}^j\}$ is a partition of the set of objects X with d_j clusters, for all $j = 1, \ldots, m$. We also denote $\mathbb{P}_X$ the set of all possible partitions of X and the consensus partition is represented by P^*. The consensus partition P^* is usually defined through the *median partition* problem:

$$P^* = \arg\max_{P \in \mathbb{P}_X} \sum_{i=1}^{m} \Gamma(P, P_i) \tag{1}$$

where Γ is a similarity measure between partitions.

Our approach to determine the *representative* partition in a hierarchy is based on the philosophy of clustering ensembles. When a hierarchical clustering algorithm is applied to the set of objects X, a hierarchy of partitions is obtained. A hierarchy $\mathbb{H} = \{P_1, P_2, \ldots, P_m\}$ is a set of nested partitions of X, where $P_i \preceq P_j, \forall i < j$. $\preceq$ is the partial order relationship *nested in*, and $P \preceq P'$ if and only if, for all cluster $C' \in P'$ there are clusters $C_{i_1}, C_{i_2}, \ldots, C_{i_v} \in P$ such that $C' = \bigcup_{j=1}^{v} C_{i_j}$. It is easy to see that $\mathbb{H} \subset \mathbb{P}_X$. Thus, we define the *representative level in the hierarchy* as the partition that better summarizes the information in the hierarchy $\mathbb{H}$ taking into account two parameters. First, the evaluation of several CVIs to all partitions in the hierarchy. Second, the similarity values between each pair of partitions in the hierarchy. Formally, the *representative* partition $\hat{P}$ in the hierarchy $\mathbb{H}$ is defined as:

$$\hat{P} = \arg\max_{P \in \mathbb{H}} \sum_{i=1}^{m} (\mathcal{E}(P_i) \cdot \Gamma(P, P_i)) \tag{2}$$

where $\mathcal{E}(P_i)$ is an evaluation of each partition $P_i \in \mathbb{H}$ and Γ a similarity measure between partitions. This evaluation can be used to give more importance to partitions that hold some desired properties. Notice that unlike the original median partition problem (1), our *best* partition $\hat{P}$ is one of the partitions in the clustering ensemble as it is shown in (2) ($\hat{P} \in \mathbb{H}$). Thus, this problem is easier than the original median partition problem, since the search space here ($\mathbb{H}$) is much more smaller than the search space ($\mathbb{P}_X$) in (1).

Among the different clustering ensemble methods, we based our approach on the *Weighted Partition Consensus via Kernels* (WPCK) [7] method. This method satisfies the following properties that are convenient for the partition selection problem:

- It is possible to compute a weight value for each partition, taking into account the evaluation of several cluster validity indexes. The weight value ω_i assigned to P_i can be used as the value $\mathcal{E}(P_i)$ in equation (2).
- It is very easy to restrict the search to partitions in $\mathbb{H}$. In fact, this is an intermediate step of WPCK. In WPCK, firstly, the *best* partition in the cluster ensemble is computed and this solution is improved afterwards by searching in the whole search space $\mathbb{P}_X$.
- The algorithm is theoretically well grounded and has a low computational cost, $\mathcal{O}(n \cdot m \cdot rMax)$[1].

[1] n is the number of objects, m the number of partitions and $rMax$ is a maximum number of iteration for the algorithm.

Therefore, the steps of the proposed algorithm to find the *representative* partition in the hierarchy (PSCE) are the following:

Subhierarchy selection
We extract from the hierarchy $\mathbb{H}$ a subset of partitions. Every partition in the hierarchy has a different number of clusters. Consequently, we will select a subset of partitions that has a number of clusters in a *reasonable* range. This range is a parameter of the algorithm, e.g., $[2, 10]$, $[2, 30]$, $[2, \sqrt{n}]$ could be used. We denote $\mathbb{H}_{[q,t]}$ the *subhierarchy* of $\mathbb{H}$, where P_q is the top level, P_t is the bottom level, and every partition $P_s \in \mathbb{H}$ with s clusters belongs to $\mathbb{H}_{[q,t]}$ if and only if $p \leq s \leq t$. For simplicity, we denote $v = t - q + 1$ the number of partitions in $\mathbb{H}_{[q,t]}$. The complete hierarchy $\mathbb{H}$ could be used, i.e., selecting the range $[1, n]$ ($v = n$). However, smaller ranges are recommended in order to decrease the computational cost.

Evaluation of each partition
We obtain the evaluation value of each partition through the application of several cluster validity indexes. Firstly, a set of internal CVIs $\mathbb{I} = \{I_1, I_2, \ldots, I_r\}$ is defined. We use this set of indexes to evaluate the behavior of each partition with respect to a set of different properties, where each index evaluates a particular property. These properties can be related with one or more of the following concepts: *compactness*, *separability*, *connectivity*, *symmetry*, etc. The property measured by each index is given by its mathematical expression. Formally, we define a *hierarchy index* as a function $I : \mathbb{H}_{[q,t]} \to [0, 1]$, where $I(P)$ is the evaluation of the partition P by the index I. It is assumed that the highest values represent better fulfilment of the index. Traditional internal CVIs such as CH, HA and HL can be easily transformed to satisfy the *hierarchy index* definition. This way, the evaluation of each partition $\mathcal{E}(P_i)$ in (2) is computed by:

$$\mathcal{E}(P_i) = \frac{1}{r} \cdot \sum_{j=1}^{r} (1 - |I_j(P_i) - M_j|) \tag{3}$$

where $M_j = \max_{P_i \in \mathbb{H}_{[q,t]}} I_j(P_i)$. Thus, $\mathcal{E}(P_i)$ is computed as a measure of how close to the maximum value, the evaluation of each index in $\mathbb{I}$ in the partition P_i is. As we obtain a weight value for each partition, for simplicity, we denote $\omega_i = \mathcal{E}(P_i)$.

Similarity measure between partitions
Besides the evaluation measure $\mathcal{E}$, we need a similarity measure Γ between partitions in order to solve the problem (2). We use the similarity measure $\hat{k}$ ($\Gamma = \hat{k}$) between partitions proposed in [7], which is formally defined as $\hat{k} : \mathbb{P}_X \times \mathbb{P}_X \to [0, 1]$ such that:

$$\hat{k}(P_i, P_j) = \frac{k(P_i, P_j)}{\sqrt{k(P_i, P_i) \cdot k(P_j, P_j)}}, \quad \text{where} \quad k(P_i, P_j) = \sum_{S \subseteq X} \delta_S^{P_i} \delta_S^{P_j} \tag{4}$$

$$\text{and} \quad \delta_S^{P_a} = \begin{cases} \frac{|S|}{|C|}, \text{ if } \exists C \in P_a, S \subseteq C; \\ 0, \quad \text{otherwise.} \end{cases}$$

This similarity measure is a positive semi-definite kernel [7].

In this method, the general problem (2) can be transformed by using the results of the above steps in the following way:

$$\hat{P} = \arg \max_{P \in \mathbb{H}_{[q,t]}} \sum_{i=1}^{v} \left(\omega_i \cdot \hat{k}(P, P_i) \right) \tag{5}$$

Obtaining the *representative* partition

As $\hat{k}$ is a kernel function, there is a mapping from $\mathbb{P}_X$ into a Hilbert space $\mathcal{H}$, $\phi : \mathbb{P}_X \rightarrow \mathcal{H}$, such that $\hat{k}(P_i, P_j) = \langle \phi(P_i), \phi(P_j) \rangle_{\mathcal{H}}$. A similar problem to (5) (*kernel consensus problem*) is presented in [7], the only difference is that the search space is $\mathbb{P}_X$ instead of $\mathbb{H}_{[q,t]}$. The kernel property of $\hat{k}$ allows mapping the *kernel consensus problem* in an equivalent problem in $\mathcal{H}$ that can be easily solved. Let ψ be the solution in the Hilbert space $\mathcal{H}$, in order to solve the *kernel consensus problem* would be necessary to find P^* such that $\psi = \phi(P^*)$, i.e., finding the pre-image of the solution [8]. However, in our case, we are solving problem (5) where the search space is $\mathbb{H}_{[q,t]}$. Thus, we need to find the partition $P \in \mathbb{H}_{[q,t]}$ such that $\phi(P)$ is closest to ψ. Formally, the *representative* partition is defined as:

$$\hat{P} = \arg \min_{P \in \mathbb{H}_{[q,t]}} \|\phi(P) - \psi\|_{\mathcal{H}}^2$$

where

$$\|\phi(P) - \psi\|_{\mathcal{H}}^2 = \tilde{k}(P, P) - 2 \sum_{i=1}^{v} \omega_i \tilde{k}(P, P_i) + \sum_{i=1}^{v} \sum_{j=1}^{v} \omega_i \omega_j \tilde{k}(P_i, P_j) \tag{6}$$

Therefore, we can find the *representative* partition by computing the distance of each partition in the hierarchy $\mathbb{H}_{[q,t]}$ to the theoretical solution ψ using equation (6), and selecting the partition closer to ψ.

2.1 Computational Complexity Analysis

The computation of all weight values for all partitions is $\mathcal{O}(v \cdot r \cdot f(\mathbb{I}))$, where v is the number of partitions in $\mathbb{H}_{[q,t]}$, r is the number of *hierarchy indexes* and $f(\mathbb{I})$ is the computational cost of the most computationally expensive *hierarchy index*. In practice, r is a small number, hence, we can consider $\mathcal{O}(v \cdot f(\mathbb{I}))$ the computational complexity of the weight assigning mechanism. Given the weight values, it is needed to compute equation (6) for each partition in $\mathbb{H}_{[q,t]}$. The last term in equation (6) does not depend on the particular partition analyzed and can be computed only one time in $\mathcal{O}(v^2 \cdot n)$, where n is the number of objects. This is because the computational complexity of the similarity measure $\hat{k}$ is $\mathcal{O}(n)$ (see [7]). Once this last value is obtained, equation (6) can be solved in $\mathcal{O}(v \cdot n)$ for one partition, and for the v partitions in $\mathbb{H}_{[q,t]}$ can be computed in $\mathcal{O}(v^2 \cdot n)$. Thus, the complete computation of equation (6) for all partitions in $\mathbb{H}_{[q,t]}$ is $\mathcal{O}(v^2 \cdot n)$. Finally, the

global computational complexity of the selection of the *representative* partition is $\mathcal{O}(v \cdot f(\mathbb{I})) + \mathcal{O}(v^2 \cdot n)$. With a proper selection of the *hierarchical indexes* $\mathbb{I}$ and the subhierarchy $\mathbb{H}_{[q,t]}$, this computational cost will be lower than $O(n^2)$, which is the common complexity of the hierarchical clustering algorithms. However, in the worst case (v close to n) the algorithm complexity becomes $\mathcal{O}(n^3)$, thereby the importance of a proper selection of the subhierarchy and the hierarchy indexes.

3 Experimental Results

We used 8 datasets in our experiments (see Table 1), 5 from the UCI Machine Learning Repository [9] and the other 3 are 2D synthetical datasets. For all these datasets the ideal data partition (*ground-truth*) is available. Therefore, in the experiments, we compared the obtained results with the *ground-truth* of each dataset. We used the Normalized Mutual Information (NMI) [10] measure to evaluate the algorithm results. This is a very used similarity measure between partitions that evaluates the resulting partition by measuring the information shared between the result and the *ground-truth*.

Table 1. Overview of datasets

Name	**Inst-per-classes**	**2D synthetic datasets**		
Cassini	120-60-120			
Half-Rings	100-100			
Smiley	33-33-50-84			
Wine	59-41-78			
Opt-Digits	10-11-11-11-12-5-8-12-9-11			
Iris	50-50-50			
Glass	70-76-17-13-9-29			
Ionosphere	126-225	Cassini	Half-Rings	Smiley

In each experiment, hierarchies are obtained by using 3 well-known hierarchical clustering algorithms: *Single-Link* (SL), *Complete-Link* (CL) and *Average-Link* (AL) [1]. For each dataset, we compare the results obtained by the proposed *Partition Selection based on Cluster Ensemble* (PSCE) approach and the stopping rule approach with the following indexes: *Highest-Lifetime*(HL), *Calinski-Harabasz* (CH) and *Hartigan* (HA). In Table 2, they are denoted SR-HL, SR-CH and SR-HA respectively. We also present for each algorithm the *Nearest to Ground-Truth* (NGT) value, which is computed by evaluating all the partitions in the hierarchy with respect to the *ground-truth* using the NMI measure and taking the highest value. Notice that NGT values depend on the quality of the hierarchies. Besides, the results of SR-HL, SR-CH, SR-HA and PSCE are upper bounded by the NGT value of each hierarchy. In all cases, we used the subhierarchy $\mathbb{H}_{[2,35]}$ composed by the partitions with s clusters, with $2 \leq s \leq 35$. For all generated hierarchy, the NGT value was obtained in a partition of the subhierarchy $\mathbb{H}_{[2,35]}$. Hence, the range $[2, 35]$ is appropriated for these experiments and allows decreasing the computational cost of the algorithms.

We used 5 *hierarchy indexes* in the evaluation of partitions step of our approach: Variance, Connectivity, HL, CH and HA. The first two are very simple indexes [7].

Table 2. Comparison of SR-HL, SR-CH, SR-HA and PSCE methods for the selection of the representative partition in a hierarchy. The hierarchies were generated by the application of the SL, CL, and AL hierarchical clustering algorithms on the 8 datasets. The results were evaluated by using the NMI measure. In each case, the best results are highlighted. The NGT value is also presented for each hierarchy. Each cell in the most right column (AVE) is the average value of its entire row.

Alg	Method	Cassini	Half-R	Smiley	Wine	Opt-D	Iris	Glass	Ionosp	AVE
SL	SR-HL	**0.941**	**0.720**	0.846	**0.102**	0.706	**0.733**	**0.154**	**0.076**	0.534
	SR-CH	**0.941**	0.488	**0.863**	0.092	0.250	**0.733**	**0.154**	0.008	0.441
	SR-HA	**0.941**	0.488	**0.863**	0.092	0.250	0.545	**0.154**	**0.076**	0.426
	PSCE	**0.941**	**0.720**	0.853	**0.102**	**0.798**	0.720	**0.154**	**0.076**	**0.545**
	NGT	0.970	0.961	1.0	0.502	0.801	0.733	0.394	0.129	0.686
CL	SR-HL	0.657	0.197	0.712	**0.790**	0.789	**0.756**	0.446	0.143	0.561
	SR-CH	0.551	0.353	0.291	0.665	0.250	**0.756**	0.442	0.037	0.418
	SR-HA	0.522	0.353	0.646	0.709	0.723	**0.756**	0.442	0.037	0.523
	PSCE	**0.743**	**0.393**	**0.820**	0.709	**0.805**	**0.756**	**0.516**	**0.160**	**0.612**
	NGT	0.792	0.442	0.865	0.798	0.825	0.756	0.590	0.193	0.657
AL	SR-HL	**0.779**	0.066	**0.766**	0.693	0.730	0.643	0.452	0.082	0.526
	SR-CH	**0.779**	0.347	0.685	**0.775**	0.250	**0.685**	0.452	0.082	0.506
	SR-HA	0.513	0.347	0.623	**0.775**	0.712	0.643	0.452	0.082	0.518
	PSCE	**0.779**	**0.433**	0.728	**0.775**	**0.814**	0.661	**0.454**	**0.083**	**0.590**
	NGT	0.792	0.474	0.883	0.775	0.843	0.783	0.501	0.169	0.652

Variance is a way of measuring the compactness of the clusters in a partition. Connectivity evaluates the degree of connectedness of clusters in a partition, by measuring how many neighbors of each object belong to the same cluster as the object. The other 3 indexes are the same used independently in the stopping rule approach. However, in this case, all of them were normalized to the range $[0, 1]$. We do not report the results of Variance and Connectivity used as stopping rules because the results were very bad. The simplicity of these indexes does not allow them to play as a stopping rule with a certain degree of accuracy. However, in the PSCE approach they can be very useful, since each index evaluates the partitions from a different perspective and all these points of view are combined to obtain the final result.

In Table 2, the experimental results are summarized. From the last column of this table it can be seen that PSCE has the best average performance in all cases. In the Single-Link (SL) hierarchies, SR-HL and PSCE work very similar. However, in the Complete-Link (CL), and Average-Link (AL) hierarchies, the PSCE approach clearly outperforms the other techniques. The results in this table corroborate the capability of the PSCE approach to work well in different circumstances, i.e., different clustering algorithms and different datasets.

From Table 2 it can also be seen that the *Nearest to Ground-Truth* NGT value is almost never reached. This fact ratifies that a single index cannot work correctly for all datasets in the case of the stopping rule approach. Besides, this means that a better and more complete set of hierarchy indexes could be used in order to improve even more the results of the PSCE approach.

4 Conclusions

In this paper, we have presented a new approach for the selection of a representative partition in a hierarchy, based on the philosophy of clustering ensembles. In

this approach, the evaluation of the partitions in the hierarchy by using different cluster validity indexes is considered in order to obtain the final result. Hence, different criteria about the quality of the partitions in the hierarchy are combined to compute the representative level. Besides, the similarity values between partitions are also taken into account in this process. Consequently, the representative partition is theoretically well defined as a weighted consensus among the partitions in the hierarchy. The main drawback of the traditional (stopping rule) approach is that if the characteristics of the used index are not in correspondence with the dataset and with the algorithm applied to generate the hierarchy, the results will not be satisfactory. The proposed approach is more robust to the change of datasets and clustering algorithms, due to the consensus definition of the representative partition and the possibility of combining the information from different cluster validity indexes. Experimental results, obtained by using different clustering algorithms and different datasets, corroborate this last assertion. On the other hand, the proposed approach is computationally more expensive than the traditional approach. However, a proper selection of the subhierarchy and hierarchy indexes used by the algorithm could decrease the computational complexity to be comparable with the classical approach.

Recently, the idea of searching for the representative partition of a hierarchy, not only in the explicit levels of the hierarchy, but in an extended partition set was proposed [5]. As future work, we will generalize our current approach to this extended partition set, where better results could be found.

References

[1] Jain, A.K., Murty, M., Flynn, P.: Data clustering: A review. ACM Computing Surveys (CSUR) 31(3), 264–323 (1999)
[2] Milligan, G.W., Cooper, M.C.: An examination of procedures for determing the number of clusters in a data set. Psychometrika 50(2), 159–179 (1985)
[3] Xu, R., Wunsch, D.C.: Clustering. IEEE Press Series on Computational Intelligence. John Wiley & Sons, Chichester (2009)
[4] Fred, A.L.N., Jain, A.K.: Combining multiple clustering using evidence accumulation. IEEE Trans. on Pat. Analysis and Mach. Intelligence 27, 835–850 (2005)
[5] Gurrutxaga, I., Albisua, I., Arbelaitz, O., Martín, J., Muguerza, J., Pérez, J., Perona, I.: Sep/cop: An efficient method to find the best partition in hierarchical clustering based on a new cluster validity index. Pattern Recognition 43(10), 3364–3373 (2010)
[6] Everitt, B., Landau, S., Leese, M.: Cluster analysis, 4th edn. Arnold, London (2001)
[7] Vega-Pons, S., Correa-Morris, J., Ruiz-Shulcloper, J.: Weighted partition consensus via kernels. Pattern Recognition 43(8), 2712–2724 (2010)
[8] Bakir, G., Weston, J., Scholkopf, B.: Learning to find pre-images. In: Thrun, S., Saul, L. (eds.) Advances in Neural Information Processing Systems (NIPS 2003), vol. 16, pp. 449–456. MIT Press, Cambridge (2004)
[9] Frank, A., Asuncion, A.: UCI machine learning repository. University of California, Irvine (2010), `http://archive.ics.uci.edu/ml`
[10] Strehl, A., Ghosh, J.: Cluster ensembles: a knowledge reuse framework for combining multiple partitions. J. Mach. Learn. Res. 3, 583–617 (2002)