

# A density-based cluster validity approach using multi-representatives

Maria Halkidi \*, Michalis Vazirgiannis

*Department of Informatics, Athens University of Economics and Business, 76 Patision Street, Athens 104 34, Greece*

Received 12 July 2006; received in revised form 27 July 2007

Available online 12 January 2008

Communicated by T.K. Ho

## Abstract

Although the goal of clustering is intuitively compelling and its notion arises in many fields, it is difficult to define a unified approach to address the clustering problem and thus diverse clustering algorithms abound in the research community. These algorithms, under different clustering assumptions, often lead to qualitatively different results. As a consequence the results of clustering algorithms (i.e., data set partitionings) need to be evaluated as regards their validity based on widely accepted criteria.

In this paper a cluster validity index, *CDbw*, is proposed which assesses the *compactness* and *separation* of clusters defined by a clustering algorithm. The cluster validity index, given a data set and a set of clustering algorithms, enables (i) the selection of the input parameter values that lead an algorithm to the best possible partitioning of the data set, and (ii) the selection of the algorithm that provides the best partitioning of the data set. *CDbw* handles efficiently arbitrarily shaped clusters by representing each cluster with a number of points rather than by a single representative point. A full implementation and experimental results confirm the reliability of the validity index showing also that its performance compares favourably to that of several others.

© 2008 Elsevier B.V. All rights reserved.

**Keywords:** Cluster validity; Clustering; Quality assessment; Unsupervised learning

## 1. Introduction

Since clustering is an *unsupervised learning procedure* and there is no a priori knowledge of data distribution in the underlying set, the significance of the clusters defined for a data set needs to be validated. Given a data set and a clustering algorithm running on it with different input parameter values, we obtain different partitionings of the data set into clusters. Then we need to select among the defined partitionings which one best fits the concerned data set. This, the *cluster validity problem*, is generally accepted as a cornerstone issue of the clustering process.

However, the notion of “good” clustering is strictly related to the application domain and its specific requirements. Nevertheless it is generally accepted that the answer to the validity of the clustering results has to be sought in

measures of *separation among the clusters* and *cohesion within clusters*. These are widely known as objective cluster validity criteria. To define these measures and evaluate clusters we have to take into account specific aspects of their definition.

In this work we tackle the cluster validity problem based on the *density properties* of clusters. This implies that we measure the *compactness* and *separation* of clusters evaluating the density distribution within and between clusters. We define and evaluate a new validity index, *CDbw* (composed density between and within clusters) and a methodology that given a data set,  $S$ , and a set of algorithms  $A = \{\text{alg}_i\}$  enables (i) finding the set of input parameter values that lead each  $\text{alg}_i$  to the best possible clustering results, and (ii) taking into account the results of (i), finding  $\text{alg}_i$  that returns the best partitioning of  $S$  among those defined by the considered algorithms.

There are cases that a clustering algorithm finds the correct number of clusters but partitions the data set in a

\* Corresponding author. Fax: +30 2108203513.

E-mail addresses: [mhalk@aub.gr](mailto:mhalk@aub.gr), [mvazirg@aub.gr](mailto:mvazirg@aub.gr) (M. Halkidi).

wrong way (i.e., it fails to discover the real clusters into which the data can be organized). Assuming different clustering algorithms' sessions on a given data set (i.e., the application of the clustering algorithm to a data set using specific values for its input parameters), a set of different partitionings but all containing the correct number of clusters (that is, the number of clusters that present in the underlying data) is defined. *CDbw* enables finding the best partitioning of a data set among the aforementioned ones. Moreover, it adjusts well to non-spherical cluster geometries, contrary to the validity indices proposed in the literature (an overview is presented in (Halkidi et al., 2001)). It achieves this by considering multiple representative points per cluster. The cluster validity index is fully implemented and experiments prove its efficiency for various data sets and clustering algorithms.

We note, here, that the cluster validity approaches can be considered to be a tool for assisting the user with the clustering process and they cannot be expected to solve all the problems related to the unsupervised learning. *CDbw* aims to assist with the evaluation of clustering results based on the criteria of the clusters' *well-separation* and *cohesion* and the selection of the clustering that best approximates the clusters into which the given data can be organized. The users can exploit the results of the cluster validity and based on their requirements they could select the clustering that is suitable for their application domain. We note that our method finds the best partitioning among those that have been defined by the selected algorithm. Hence if the clustering algorithm does not manage to find the actual partitioning of a dataset then the cluster validity approach, of course, is not able to find the partitioning either. However it can be used to select among the defined clusterings, the one that mostly approximates the real clusters according to the requirement of well-separated and cohesive clusters.

The rest of the paper is organized as follows. Section 2 motivates the definition of a new validity index and discusses some background information. Then, in Section 3 we present the fundamental concepts of our approach discussing also in detail the proposed cluster validity index. In Section 4 we describe an experimental study of our approach while we present its comparison to other cluster validity indices. In the sequel, Section 5 reviews cluster

validity related concepts and some cluster validity criteria related to our work. Finally, we conclude in Section 6 by briefly presenting our contributions and indicating directions for further research.

## 2. Preliminaries and motivation of the cluster validity approach

The validity assessment of clustering results is a complex problem and it depends on the application domain. In the sequel, we motivate the aspects of assessing the validity of clustering results, using examples. As Fig. 1 depicts, a clustering algorithm (here, the *K*-Means (Berry and Linoff, 1996) algorithm is used) with different input parameter values (for brevity, further referred to as *ipvs*) results in different clusterings. The data set is falsely partitioned in most of the cases. Only a specific set of *ipvs* (in this case when number of clusters = 3) lead to the actual partitioning of the data set. If there is no prior knowledge about the data structure, it is difficult to find the best *ipvs* for a given algorithm.

Cross-algorithm comparison takes place in the example of Fig. 6 where different clustering algorithms (*K*-Means (Berry and Linoff, 1996), CURE (Guha et al., 1998), DBSCAN (Ester et al., 1996), a clustering algorithm provided by the CLUTO toolkit (Karypis, 2003) are used to partition the DS2 data set. The algorithms ran under specific set of *ipvs* to define a partitioning of DS2 into four clusters. As we can observe in Fig. 6a–c, *K*-Means, CURE (with *ipvs*  $a = 0.3$ ,  $r = 10$ ) and the CLUTO algorithm partition DS2 wrongly into four clusters. On the other hand, DBSCAN (see Fig. 6d) with suitable *ipvs* gives better results since it partitions the data set discovering its real four clusters. A similar example is presented in Fig. 7.

Following up the examples discussed above, each algorithm provides a partitioning of a dataset but does not deal with the validity of the clustering results. They aim to find the best possible partitioning for the given *ipvs* but there is no indication that the defined clusters are the ones that best fit data.

Visual perception of the clusters structure enables a profound assessment of the partitioning validity. However, in case of large high-dimensional data sets (e.g., more than three dimensions), effective visualization can be cumbersome.

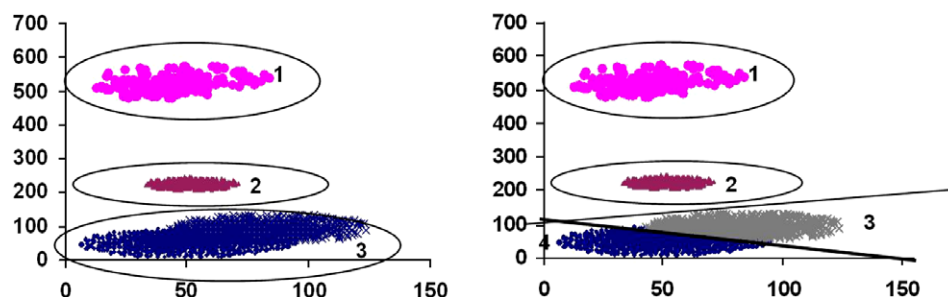


Fig. 1. The different partitionings defined by *K*-Means when it runs with different input parameter values.

some. Moreover the perception of clusters based on visualization is a difficult task for humans not accustomed to higher dimensional spaces. What is then needed is a visual-aids-free assessment of some objective criterion, indicating the validity of clusterings defined by a clustering algorithm. This should be applicable to a potentially high-dimensional data set and handle efficiently arbitrarily shaped clusters (i.e., clusters of non-spherical geometry).

The fundamental criteria for clustering algorithms include *compactness* and *separation* of clusters. However, the clustering algorithms aim at satisfying these criteria based on initial assumptions (e.g., initial locations of the cluster centers) or input parameter values (e.g., the number of clusters, minimum diameter or number of points in a cluster). What is missing is an approach that satisfies a global optimization of the clustering criteria, comparing the different clusterings defined for a data set.

Another issue of concern is the *geometry* of the clusters that has been treated in several algorithms recently (Ester et al., 1996; Sheikholeslami et al., 1998). The problem is that when a cluster's geometry is deviating from the hyper-spherical shape, the majority of clustering algorithms has problems to identify the correct clusters. Even in cases that an algorithm achieves to handle arbitrarily shaped clusters, it is based on specific assumptions.

The above observations motivate the definition of a cluster validity index, *CDBw*, taking into account (a) the *density distribution between and within clusters* to assess the compactness and separation of the defined clusters, (b) the *changes of the density distribution* within clusters to assess the clusters' cohesion, and (c) the requirements for handling *awkward cluster geometries*.

The cluster's geometry issue is addressed in *CDBw* by considering multiple representative points for each cluster defined by an algorithm. This approach improves geometry-related efficiency compared to other related ones (a survey of cluster validity approaches is presented in (Halkidi et al., 2001)) that consider a single representative point per cluster.

Below we introduce the terms and concepts that will be used throughout the paper.

Assuming that  $S$  is a data set presenting clustering tendency (i.e., one can identify sparse and dense areas in the data space) and there is a partitioning  $C$  of  $S$  that represents its dense areas as distinct partitions (i.e., the underlying clusters in  $S$ ), we call  $C$  *actual partitioning* of  $S$ . In other words, the actual partitioning is used in the context of this paper to represent the partitions that corresponds to the clusters that are expected to be identified in the underlying data. The results of a clustering algorithm  $A$  applied to  $S$  comprise a partitioning of  $S$  into a set of clusters that is called *clustering* of  $S$ . If for each cluster  $C_i$  there is a partition  $P_j$  of the actual partitioning such that  $C_i = P_j$  (i.e., contain the same data objects) then we claim that the algorithm discovered the *real* clusters or the *actual partitioning*.

There are cases that an algorithm  $A$  applied to  $S$  with different ipvs, results in different clusterings none of which

resembles the actual partitioning. Among these clusterings, the one that is most similar to (approximates with high accuracy) the actual partitioning is further called *best partitioning*<sup>1</sup> of  $S$  by  $A$ . The best partitioning refers to the best possible partitioning of  $S$  among those defined by the clustering approaches. As it will be further discussed, it is important to discover the ipvs for  $A$  applied to  $S$  that result in the best partitioning.

Also the term “*correct number of clusters*” is used to refer to the number of clusters in the actual partitioning of a data set while the number of clusters in case of best partitioning is further called “*best number of clusters*”.

### 3. A cluster validity approach based on density

In this section, we formalize our cluster validity index putting emphasis on the geometric aspects of clusters and exploiting the density notion of clusters as well. It is a relative validity index since it aims to compare different clusterings defined for a given data set and select the one that best fits the data (i.e., best partitioning). Its definition is based on a set of representative points per cluster and the measures of (i) *clusters' cohesion* (in terms of relative intra-cluster density), and (ii) *clusters' separation* (in terms of distance and inter-cluster density).

#### 3.1. Cluster representative points definition

Let  $D = \{V_1, \dots, V_c\}$  be a partitioning of a data set  $S$  into  $c$  clusters where  $V_i$  is the set of representative points of the cluster  $C_i$ , such that  $V_i = \{v_{i1}, \dots, v_{ir}\}$   $r$  = number of representatives per cluster and  $v_{ij}$  is the  $j$ th representative of the cluster  $C_i$ . Each cluster is represented by a set of  $r$  points that are generated by selecting well-scattered points within this cluster. These  $r$  points achieve to capture the geometry of the respective cluster.

The contribution of the proposed approach is the use of multi-representatives in cluster validity so as to capture the shape of the clusters in the clustering evaluation process and not a method to define representative points. Hence we select to use one of the widely used approaches, which is based on the “furthest-first” technique (Hochbaum and Shmoys, 1985), to define the representative points of the clusters. We note that other approaches for finding the clusters' representative can be used as well. According to this approach, in the first iteration the point farthest from the center of the cluster under concern is chosen as the first representative point. In each subsequent iteration a point from the cluster is chosen that is farthest from the previously chosen representative points. Thus the function results in a set of points that represent the geometry of the cluster periphery (boundaries of the cluster).

<sup>1</sup> In the context of this paper the terms “partitioning” and “clustering” are interchangeable.

The extended analysis on the procedure for selecting the representatives is out of the scope of this paper. We note that the number of clusters depends on the nature of data and can be either user-defined or selected based on statistical properties of data. In this paper, we empirically determine the value of  $r$ .

**Definition 2.1.** *Closest representative points.* Let  $V_i$  and  $V_j$  be the set of representatives of the clusters  $C_i$  and  $C_j$ , respectively. A representative point of  $C_i$ , let  $v_{ik}$ , is considered to be the *closest representative* in  $C_i$  of the representative  $v_{jl}$  of the cluster  $C_j$ , further referred to as  $\text{closest\_rep}^i(v_{jl})$ , if  $v_{ik}$  is the representative point of  $C_i$  with the minimum distance from  $v_{jl}$ , i.e.,  $d(v_{jl}, v_{ik}) = \min_{v_{ix} \in V_i} \{d(v_{jl}, v_{ix})\}$ , where  $d$  is the Euclidean distance. The set of closest representatives of  $C_j$  with respect to  $C_i$  is defined as follows:  $CR_j^i = \{(v_{ik}, v_{jl}) | v_{jl} = \text{closest\_rep}^i(v_{ik})\}$ .

**Definition 2.2.** *Respective closest representative points.* The set of respective representative points of the clusters  $C_i$  and  $C_j$  is defined as the set of mutual closest representatives of the clusters under concern, i.e.,  $\text{RCR}_{ij} = \{(v_{ik}, v_{jl}) | v_{ik} = \text{closest\_rep}^i(v_{jl}) \text{ and } v_{jl} = \text{closest\_rep}^j(v_{ik})\}$ .

In other words, the  $\text{RCR}_{ij}$  set is defined as the intersection of the closest representative of  $C_i$  with respect to  $C_j$  and the closest representative of  $C_j$  with respect to  $C_i$ , i.e.,  $\text{RCR}_{ij} = CR_j^i \cap CR_i^j$ .

### 3.2. Clusters' separation in terms of density

In this paper we evaluate the separation of the defined clusters based on the density distribution in the area between the clusters. Here, the term “area between clusters” implies the area between the respective closest representatives of the clusters. Considering that representative points efficiently capture the shape and extent of the clusters, the density in the area between closest points of clusters is an indication of how close the clusters are.

**Definition 3.** *Density between clusters.* It measures the number of points distributed in the area between the respective clusters. Let  $\text{clos\_rep}_{ij}^p = (v_{ik}, v_{jl})$  be the  $p$ th pair of respective closest representative points of clusters  $C_i$  and  $C_j$ , i.e.,  $\text{clos\_rep}_{ij}^p \in \text{RCR}_{ij}$ , and  $u_{ij}^p$  the middle point of the line segment defined by the  $p$ th pair  $\text{clos\_rep}_{ij}^p$  (see Fig. 2). The density between the clusters  $C_i$  and  $C_j$  is defined as follows:

$$\text{Dens}(C_i, C_j) = \frac{1}{|\text{RCR}_{ij}|} \times \sum_{i=1}^{|\text{RCR}_{ij}|} \left( \frac{d(\text{clos\_rep}_{ij}^p)}{2 \cdot \text{stdev}} \cdot \text{cardinality}(u_{ij}^p) \right) \quad (1)$$

where  $d(\text{clos\_rep}_{ij}^p)$  is the Euclidean distance between the pair of points defined by  $\text{clos\_rep}_{ij}^p \in \text{RCR}_{ij}$ ,  $|\text{RCR}_{ij}|$  presents the cardinality of the set  $\text{RCR}_{ij}$  and the term “stdev”

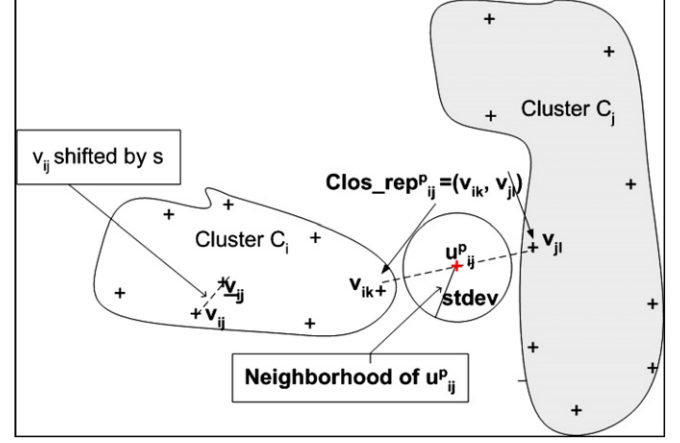


Fig. 2. Inter-cluster density definition.

is the average standard deviation of the considered clusters. We note that the term *density* between clusters is used as equivalent to the term *cardinality* between the clusters, which is defined in Eq. 2

$$\text{cardinality}(u_{ij}^p) = \frac{\sum_{l=1}^{n_i+n_j} f(x_l, u_{ij}^p)}{n_i + n_j} \quad (2)$$

where  $x_l$  corresponds to the data points of the clusters under concern (i.e.,  $x_l \in C_i \cup C_j$ ), while  $n_i$  and  $n_j$  are the number of points that belong to clusters  $C_i$  and  $C_j$ , respectively.

Specifically,  $\text{cardinality}(u_{ij}^p)$  represents the average number of points in  $C_i$  and  $C_j$  that belong to the neighborhood of  $u_{ij}^p$ . To define the neighborhood of a data point, the scattering of data points on each dimension is considered to be an important factor. In other words, if the scattering of data is large and the neighborhood of points is small then there might be no points included in the neighborhood of any of the data points. On the other hand, if the scattering is small and the neighborhood is large, then the entire data set might be in the neighborhood of all the data points. Selecting different neighborhoods for different data sets can reasonably solve this problem. The standard deviation can be used to approximately represent the scatter of data points. Therefore the neighborhood of a point can be considered to be a function (e.g., average, min, max) of standard deviation on each data dimensions. In this work, the goal is to evaluate the density in different areas within and between the defined clusters. Thus we select to define the neighborhood of a data point,  $u_{ij}^p$ , as the hyper-sphere centered at  $u_{ij}^p$  (see Fig. 2) with radius the average standard deviation of the considered clusters,  $\text{stdev}$ . Other definitions of points' neighborhood can also be used. However a study regarding the definition of the point's neighborhood is beyond the scope of this work.

Based on the above discussion the function  $f(x, u_{ij})$  is defined as

$$f(x, u_{ij}) = \begin{cases} 1, & \text{if } d(x, u_{ij}) < \text{stdev and } x \neq u_{ij} \\ 0, & \text{otherwise} \end{cases} \quad (3)$$



where stdev is the standard deviation of the considered clusters.

A point belongs to the neighborhood of  $u_{ij}^p$  if its distance from  $u_{ij}^p$  is smaller than the average standard deviation of the clusters (i.e.,  $d(x, u_{ij}^p) < \text{stdev}$ ). On the other hand, the actual area between clusters, whose density we are interested in estimating, is defined to be the area between the respective closest representative points (see Fig. 2) and its size is defined to be  $d(\text{clos\_rep}_{ij}^p)$ . Since the term  $\text{cardinality}(u_{ij}^p)$  represents the number of points distributed in the area whose size is defined by the standard deviation of the considered clusters (i.e., the hyper-sphere with diameter  $2 \cdot \text{stdev}$ ), without loss of generality, the “actual” number of points between the clusters is defined to be the  $d(\text{clos\_rep}_{ij}^p)/(2 \cdot \text{stdev})$  percentage of points belonging to the neighborhood of  $u_{ij}^p$  (i.e.,  $\text{cardinality}(u_{ij}^p)$ ). The above justifies the definition of density (cardinality) between the  $p$ th pair of the respective representatives of clusters  $C_i$  and  $C_j$  as  $\frac{d(\text{clos\_rep}_{ij}^p)}{2 \cdot \text{stdev}} \cdot \text{cardinality}(u_{ij}^p)$ .

**Definition 4.** *Inter-cluster density.* Let  $\mathbf{C} = \{C_i | i = 1, \dots, c\}$  be a partitioning of a data set into  $c$  clusters,  $c > 1$ . The inter-cluster density measures for each cluster  $C_i \in \mathbf{C}$ , the maximum density between  $C_i$  and the other clusters in  $\mathbf{C}$ . More specifically, it is defined by Eq. 4

$$\text{Inter\_dens}(\mathbf{C}) = \frac{1}{c} \sum_{i=1}^c \max_{j=1, \dots, c, j \neq i} \{\text{Dens}(C_i, C_j)\} \quad (4)$$

where  $c > 1, c \neq n$ .

**Definition 5.** *Clusters’ separation (Sep).* It measures the separation of clusters taking into account the inter-cluster density with respect to the distance between clusters. A good partitioning is characterized by long distances between clusters’ representatives and low density between them (i.e., well-separated clusters). Then, the clusters’ separation is defined by the equation (Eq. 5)

$$\text{Sep}(\mathbf{C}) = \frac{\frac{1}{c} \sum_{i=1}^c \min_{j=1, \dots, c, j \neq i} \{\text{Dist}(C_i, C_j)\}}{1 + \text{Inter\_dens}(\mathbf{C})}, c > 1, c \neq n \quad (5)$$

where  $\text{Dist}(C_i, C_j) = \frac{1}{|\text{RCR}_{ij}|} \sum_{i=1}^{|\text{RCR}_{ij}|} d(\text{clos\_rep}_{ij}^p)$  and  $|\text{RCR}_{ij}|$  is the cardinality of the set  $\text{RCR}_{ij}$  as defined earlier.

According to the definitions above, *Inter\_dens* assesses the maximum number of points distributed in the area between the clusters under concern. This is an indication of how close the clusters are. Without loss of generality, we assume that the maximum inter-cluster density is detected between a cluster and its closest one, since the closest the clusters are the more probable is to find areas of high density between them. Also the area between

clusters is measured in terms of the distance between their respective closest representatives. Then,  $\text{Sep}(\mathbf{C})$  is perceived to measure the respective number of data points per unit of space between the closest clusters, i.e., the *relative density between clusters*.

### 3.3. Clusters’ compactness in terms of density

We previously introduced the concept of multiple representative points. They are initially generated by selecting well-scattered points in the cluster that represent well the geometric features of the cluster. We exploit these points to assess the separation of the clusters as it is discussed above. Besides the cluster’s separation, we also take into account the cluster’s compactness and cohesion. This implies that clusters should not only be well separated but also dense. *Cluster’s compactness* is a measure of cluster’s inherent quality, which increases when the clusters are characterized by high internal density.

The center of a cluster (the mean of the data points in the cluster) is not necessarily a point within the cluster but it is perceived as the most central point of cluster space around which the data points belonging to the cluster are distributed. Moreover cluster center can be considered as a reference point toward which the cluster representatives can be gradually moved in order to get instances of initial representatives at different areas in cluster space. Measuring the density in the neighborhood of these representatives, the density distribution within a cluster can be estimated. Thus without loss of generality the cluster center can be considered as a good approximation of the cluster space core.

Let  $v_{ij}$ , further called *shrunk representative*, correspond to the  $j$ th representative point of the cluster  $C_i$ ,  $v_{ij}$ , shrunk (shifted) towards the center of the cluster by a shrinking factor  $s \in [0, 1]$  (see Fig. 2). Thus the  $k$ th dimension of  $v_{ij}$  can be defined as  $v_{ij}^k = v_{ij}^k + s \cdot (C_i^k \cdot \text{center} - v_{ij}^k)$ , where  $C_i \cdot \text{center}$  is the center of cluster  $C_i$ . The shrinking factor,  $s$ , is user-defined to control the compactness of clusters in the validity checking process according to the application needs. A high value of  $s$  shrinks the representatives closer to the cluster center and thus it favours more compact clusters. On the other hand, a small value of  $s$  shrinks more slowly the representatives and the validity checking process favours elongated clusters. Selecting a suitable shrinking factor we can manage to have instances of representatives within the cluster.

To eliminate the influence of  $s$  to the cluster validity results, the density within clusters is estimated for different values of  $s$ . More specifically, the value of  $s$  is increasing so that the representative points are gradually shrunk and the respective values of density are calculated at these shrunk points. The average value of a cluster’s intra-cluster density, as calculated for the different values of  $s$ , is considered to be the density within the considered clusters. It is evident that we are able to get a better view of density

distribution within a cluster, calculating the density at different areas of the cluster.

**Definition 6.** *Relative intra-cluster density* measures the relative density within clusters with respect to (wrt.) a shrinking factor  $s$ . This implies the number of points that belong to the neighborhood of the representative points of the defined clusters shrunk by  $s$ , let  $v_{ij}$ , (i.e., points belong to the hyper-sphere centered at  $v_{ij}$  with stdev radius). Then the *relative intra-cluster density* with respect to the factor  $s$  is defined as follows:

$$\text{Intra\_dens}(\mathbf{C}, s) = \frac{\text{Dens\_cl}(\mathbf{C}, s)}{c \cdot \text{stdev}}, \quad c > 1$$

$$\text{where } \text{Dens\_cl}(\mathbf{C}, s) = \frac{1}{r} \sum_{i=1}^c \sum_{j=1}^r \text{cardinality}(v_{ij}) \quad (6)$$

The cardinality of a point  $v_{ij}$  is defined as  $\text{cardinality}(v_{ij}) = \sum_{l=1}^{n_i} f(x_l, v_{ij}) / n_i$ , where  $n_i$  is the number of the points,  $x_l$ , that belong to the cluster  $C_i$ , i.e.,  $x_l \in C_i \subseteq S$  and the function  $f$  is defined as in Eq. 3. It represents the proportion of points in cluster  $C_i$  that belong to the neighborhood of a representative  $v_{ij} \forall j$  (i.e., the representatives of  $C_i$  shrunk by a factor  $s$ ). The neighborhood of a data point,  $v_{ij}$ , is defined to be a hyper-sphere centered at  $v_{ij}$  with radius the average standard deviation of the considered clusters, stdev.

**Definition 7.** The *compactness* of a clustering  $\mathbf{C}$  in terms of density is defined by the equation

$$\text{Compactness}(\mathbf{C}) = \sum_s \text{Intra\_dens}(\mathbf{C}, s) / n_s \quad (7)$$

where  $n_s$  denotes the number of different values considered for the factor,  $s$ , based on which the density at different areas within clusters is calculated. Usually, we consider that the values of the shrinking factor,  $s$ , is gradually increasing in  $[0.1, 0.8]$  (the cases that  $s = 0$ , and  $s \geq 0.9$  refer to the trivial case that the representative points correspond to the boundaries and the center of cluster, respectively).

Then considering that the representative points are shrunk by a factor  $0.1 \leq s \leq 0.8$ , and  $s_i = s_{i-1} + 0.1$ , we get from Eq. 7:  $\text{Compactness}(\mathbf{C}) = \sum_{s \in [0.1, 0.8]} \text{Intra\_dens}(\mathbf{C}, s) / 8$ . In other words, the term *Compactness* ( $\mathbf{C}$ ) corresponds to the average density within a set of clusters,  $\mathbf{C}$ , defined for a data set.

### 3.4. Assessing the quality of a data clustering

In the previous sections (Sections 3.2 and 3.3) we introduce some measures based on which the *compactness* and *separation* of clusters are evaluated. However, none of these measures could lead to a reliable evaluation of clusters' validity if they are taken into account separately.

Thus the requirement for a global measure that assesses the quality of a dataset clustering in terms of its validity arises.

#### 3.4.1. Clusters' cohesion

Besides the compactness of clusters, another requirement of clusters' quality is that the changes of density distribution within clusters should be significantly small. This implies that not only the average density within clusters (as measured by *Compactness*) has to be high but also the density changes as we move within the clusters should to be small. The above requirements are strictly related to the evaluation of clusters' cohesion, i.e., the density-connectivity of objects belonging to the same clusters.

**Definition 8.** *Intra-density changes*, measure the changes of density within clusters. It is given by the equation

$$\text{Intra\_change}(\mathbf{C}) = \frac{\sum_{i=1, \dots, n_s} |\text{Intra\_dens}(\mathbf{C}, s_i) - \text{Intra\_dens}(\mathbf{C}, s_{i-1})|}{(n_s - 1)} \quad (8)$$

where  $n_s$  is the number of different values that the factor  $s$  takes. Significant changes to the intra-cluster density indicate that there are areas of high density that followed by areas of low density and vice versa.

**Definition 9.** *Cohesion* measures the density within clusters with respect to the density changes observed within them. It is defined as follows:

$$\text{Cohesion}(\mathbf{C}) = \frac{\text{Compactness}(\mathbf{C})}{1 + \text{Intra\_change}(\mathbf{C})} \quad (9)$$

#### 3.4.2. Separation wrt compactness

The best partitioning (see Section 2 for a definition of the term) requires maximal compactness (i.e., intra-cluster density) in such a way that the clusters are well separated and vice versa. This implies that *compactness* and *separation* are closely related measures of clusters' quality. Furthermore there are cases that the clusters' separation tends to be meaningless with regard to the clusters' quality, if it is considered independently of the clusters' compactness. Then, it is evident that we need a measure that assists with evaluating the separation of clusters in conjunction with their compactness.

**Definition 10.** *SC (Separation wrt Compactness)* evaluates the clusters' separation with respect to their compactness

$$\text{SC}(\mathbf{C}) = \text{Sep}(\mathbf{C}) \cdot \text{Compactness}(\mathbf{C}) \quad (10)$$

In other words, considering a data set and its clustering  $\mathbf{C}$ ,  $\text{SC}$  is defined as the product of the density between clusters ( $\text{Sep}(\mathbf{C})$ ) and the density within the clusters defined in  $\mathbf{C}$  ( $\text{Compactness}(\mathbf{C})$ ).

### 3.5. CDbw definition

A reliable cluster validity index has to correspond to all the requirements of “good” clustering. This implies that it has to evaluate the cohesion of clusters as well as the separation of clusters in conjunction with their compactness. These requirements motivate the definition of the *validity index CDbw*. It is based on the terms defined in the equations Eq. 9 and Eq. 10 and is given by the following equation:

$$CDbw(C) = Cohesion(C) \cdot SC(C), \quad c > 1 \quad (11)$$

The above definitions refer to the case that a data set possesses clustering tendency, i.e., the data vectors can be grouped into at least two clusters. The validity index is not defined for  $c = 1$ .

#### 3.5.1. Discussion

The definition of *CDbw* (Eq. 11) indicates that all the criteria of “good” clustering (i.e., *cohesion* of clusters, *compactness* and *separation*) are taken into account, enabling reliable evaluation of clustering results. A clustering with compact and well-separated clusters and low variation of the density distribution within clusters results in high values for both *CDbw* terms (i.e., Cohesion and Separation wrt. Compactness). Moreover, *CDbw* exhibits no monotonous trends with respect to the number of clusters. Thus in the plot of *CDbw* versus the number of clusters, we seek the maximum value of *CDbw* which corresponds to the best partitioning of a given data set. The absence of a clear local maximum in the plot is an indication that the data set possesses no clustering structure.

In the trivial case that each point is considered to be a separate cluster, i.e.,  $c = n$ , the standard deviation of the clusters is 0. Then Eq. 1 and Eq. 6 cannot be defined when  $c = n$ . However, this is not a serious problem. In real-world cases, if the data can be organized into compact and well-separated clusters (i.e., the data set possesses a clustering structure), its best partitioning will correspond to a set of clusters whose number ranges between 2 and  $n - 1$ .

Nevertheless, considering the semantics of the terms *Intra\_dens* and *Inter\_dens*, which in the trivial case ( $c = n$ ), cannot be defined based on Eqs. 1 and 6, we proceed with the following statement:

*In the trivial case that each point is a separate cluster, i.e.  $c = n$ , the standard deviation of clusters is 0. Then*

[–] According to Eq. 2 the term  $\text{cardinality}(u_{ij}^p)$  is zero for any pair of the defined clusters. This implies that the density between clusters is also zero, i.e.,

$$\begin{aligned} \text{Dens}(C_i, C_j) &= 0 \forall i, j \in [1, n] \Rightarrow \text{Inter\_dens}(C) \\ &= 0, \quad \text{where } C = \{C_i | i = 1, \dots, n\} \end{aligned}$$

[–] The intra-cluster density measures the average density in the neighborhood of the clusters’ shrunk representatives. In case that  $c = n$ , there is only one point that

belong to a cluster which is also considered to be the cluster’s representative. According to Eq. 3,  $\forall x, x \neq v_{ij}$  and  $d(x, v_{ij}) = \text{stddev}_i = 0 \Rightarrow f(x, v_{ij}) = 0$ . Therefore the density within clusters is

$$\begin{aligned} \text{Dens\_cl}(C, s) &= \text{Dens\_cl}(C)/n = 0, \forall s \text{ and } C = \{C_i | i \\ &= 1, \dots, n\} \end{aligned}$$

Then, without loss of generality, we claim that  $\text{Intra\_dens}(C, s) = 0 \forall s$ , when  $c = n$ . As a consequence, we get from Eq. 7 that  $\text{Compactness}(C) = 0$ , when  $c = n$ .

Hence, based on Eq. 11,  $CDbw(C) = 0$  when  $c = n$ .

### 3.6. Time complexity

The complexity of the cluster validity index *CDbw*, is based on the complexity of the terms *Cohesion* and *Separation* wrt. *Compactness* as defined in the equations Eqs. 9 and 10, respectively. Let  $d$  be the number of attributes (data set dimension);  $c$  be the number of clusters;  $n$  be the number of the data points and  $r$  be the number of a cluster’s representatives. Then the complexity of selecting the closest representative points of  $c$  clusters is  $O(dc^2r^2)$ . Based on their definitions, the computational complexity of *SC* depends on the complexity of clusters’ compactness (*Compactness*) and separation (*Sep*) that is  $O(ncrd)$  and  $O(ndc^2)$ , respectively. Then the complexity of *SC* is  $O(ndr^2c^2)$ . Furthermore, based on Eq. 9, the computational complexity of clusters’ cohesion (*Cohesion*) is  $O(ncrd)$ . Then, we conclude that *CDbw* complexity is  $O(ndr^2c^2)$ . Usually,  $c, d, r \ll n$ , therefore the complexity of the validity index for a specific clustering is  $O(n)$ . The complexity of the whole cluster validity procedure which aims to find the best partitioning of a data set,  $S$ , among a set of  $k$  different partitionings defined by a clustering algorithm, will be  $O(kn)$ . In the context of this paper, the clustering process is not considered as part of the cluster validity process. Given that  $k$  is significantly smaller than  $n$  (number of data points), the complexity of the clustering validity process will be  $O(n)$ .

## 4. Experimental evaluation

In our experimental evaluation we focus mainly on evaluating the effectiveness of our approach to select the partitioning that best fits data among those defined for the concerned data set. We present results using different datasets and algorithms. The experimental section concludes with a comparison of *CDbw* to some of the most important validity indices proposed in the literature.

### 4.1. Data sets and methodology

We conducted experiments using synthetic data sets in order to evaluate the performance of the proposed cluster validity approach in various cases of the data sets’ structure. In all cases we evaluate the effectiveness of our

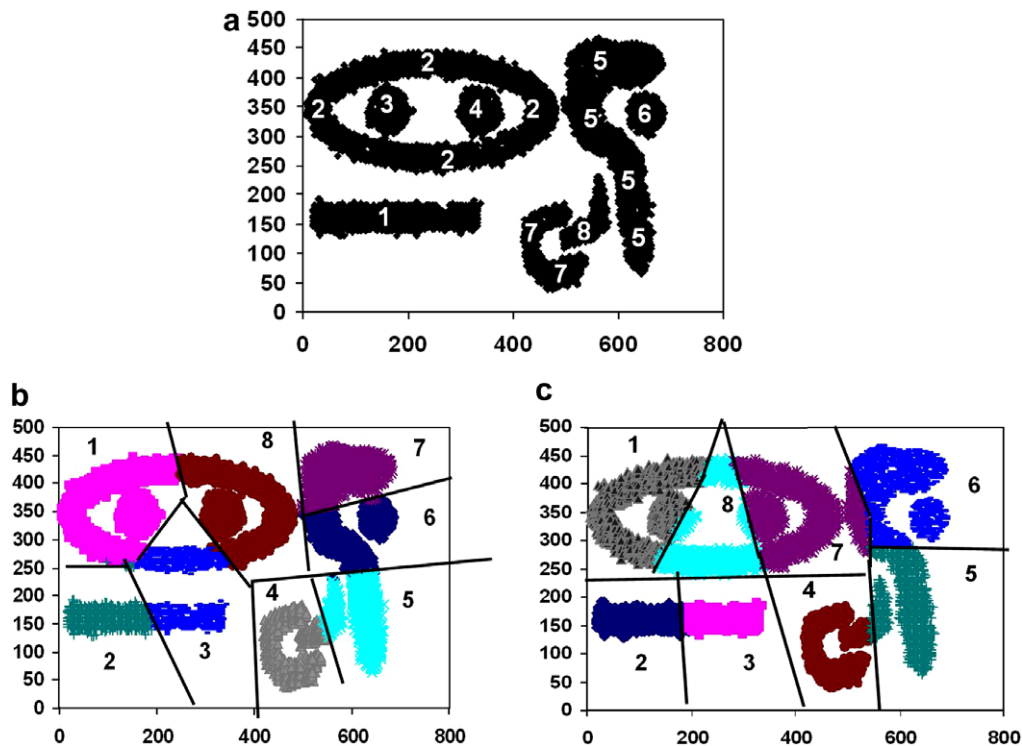


Fig. 3. Synthetic data sets: (a) DS1 and partitioning of DS1 using CLUTO, (b) *K*-Means, and (c) CURE.

approach with respect to a pre-specified clustering that has been defined for the datasets used in experiments. The labelling of the synthetic datasets is defined at the stage of their generation while in case of real datasets an expert of domain has given the correct labelling of the data. This labelling represents the actual partitioning of the considered data. To measure the accuracy of the clustering selected of *CDBw* in relation to the actual partitioning of the considered data, we have used pair-wise *F*-measure (Van Rijsbergen, 1979). Its definition is based on the traditional information measures (precision, recall), adapted for evaluating clustering by considering same-cluster pairs.

We experimented with various data sets containing different numbers of clusters of various shapes. Due to space constraints we report only results for data sets containing less than 10 clusters. As regards the data dimensionality, it ranges between 2 and 120 dimensions. The rest of our results with data sets of higher dimensions and data sets containing a larger number of clusters are qualitatively similar to those discussed below, thus they are omitted for brevity. Also, we have ignored the presence of noise in data so that the experimental study is independent of the efficiency of the clustering algorithms to handle noise.

#### 4.1.1. Determining the number of representatives $r$

To determine the value of  $r$  that we use for experiments we experimented with a representative sample of the datasets used for the experimental evaluation of the proposed approach. We varied the number of representatives,  $r$ , in the range from 1 (equivalent to the case where only the

cluster center is used for representation) to 30. We observed that *CDBw* fail to select the best partitioning when  $r \leq 5$  while there were no significant changes to the efficiency of *CDBw* when  $r \geq 10$ . Generally, we conclude that in the context of our experimental study a number of representatives around 10 ( $r \geq 10$ ) achieves to capture to a satisfactory degree the geometry of clusters. Thus *CDBw* gives reliable results with regard to the selection of the best partitioning of a data set.

#### 4.2. Selecting the best partitioning defined by a clustering algorithm

Though any clustering algorithm can be used, for each dataset we select to report the experimental results using the algorithm that achieves to find at least one partitioning of the data under concern which approximates their actual partitioning with high accuracy. This assists with having a most accurate evaluation of the effectiveness of *CDBw* as regards its ability to select the best partitioning of a dataset.

The initial set of experiments refers to a 2-dimensional data set, further referred to as DS1, 8 clusters (see Fig. 3a). It is a synthetic data set generated based on the data set used in (Karypis et al., 1999). We used a clustering algorithm provided by the CLUTO toolkit<sup>2</sup> to

<sup>2</sup> The results of this experiment was obtained by running the 'vcluster' program with parameters *clmethod* = graph, *sim* = dist, *agglofrom* = 30 and the number of clusters ranging between 3 and 10.



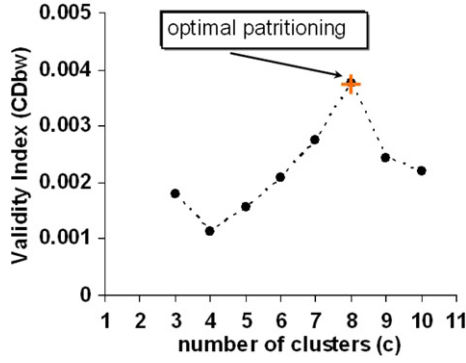


Fig. 4.  $Cdbw$  as a function of number of clusters for DS1 (CLUTO).

discover the clustering of DS1 with the number of clusters ranging in (Dave, 1996; Halkidi et al., 2000). For each of the partitionings obtained the  $Cdbw$  value is computed and the respective graph of the validity index values vs. the number of clusters is created (see Fig. 4). Based on this graph we observe that  $Cdbw$  reaches its maximum at the partitioning of the eight clusters. Then it is proposed as the best partitioning of DS1. We note that the selected set of partitions also corresponds to the real clusters of DS1 (see Fig. 3a).

#### 4.2.1. Multi-dimensional data sets

The validity of clustering results (i.e., that the set has been well partitioned) can be visually verified only in 2D or 3D cases. In higher dimensions it is difficult to verify the resulting clusters. The proposed validity index tackles this problem giving an indication of the best clustering without visualization of the data set. We have experimented with various data sets but due to lack of space, we select to report the behaviour of  $Cdbw$  using a representative dataset of 120 attributes containing four clusters (further referred to as Nd\_Set). We ran the CURE algorithm (with  $\alpha = 0.3$ ,  $r = 10$ ) on the data repeatedly, with the number of clusters  $c$  in the range 2–10. For each value of  $c$ , we obtained a partitioning of the data and calculated the respective value of  $Cdbw$ . The plot of  $Cdbw$  vs. the number of clusters (corresponding to the different partitionings of the data set) is depicted in Fig. 5. We observe that

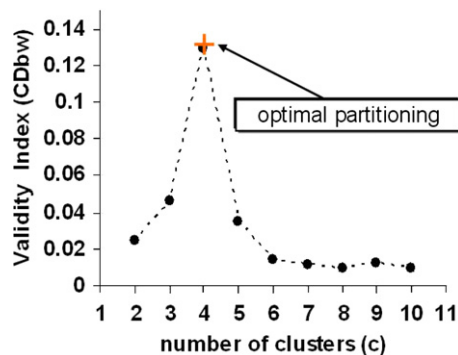


Fig. 5. Nd\_Set  $Cdbw$  vs the number of clusters for a 120-dimensional data set.

$Cdbw$  takes its maximum value when  $c = 4$ . Thus the partitioning of four clusters, as defined by CURE is proposed as the best partitioning of Nd\_Set.

Evaluating the accuracy of the selected partitioning in relation to the pre-specified clustering of Nd\_Set, we get that it perfectly approximates the actual partitioning of the data set (specifically,  $F$ -measure = 1).

#### 4.3. Selecting clustering algorithm

In previous sections, we performed experiments with different ipvs for clustering algorithms. In the sequel given a data set and a set of clustering algorithms, we show that  $Cdbw$  enables the selection of the best partitioning among those defined by different clustering algorithms. Thus, the clustering algorithm that finds the best partitioning of a data set can be selected.

We assume a data set  $S$  on which we ran different clustering algorithms  $A = \{A_i\}_{i=1}^m$  using for each of them the best ipvs,  $IP_{best}^i$ , as defined by  $Cdbw$ . Let  $\{C(A_i(IP_{best}^i))\}_{i=1}^m$  be the clusterings of  $S$  resulting from the execution of the aforementioned algorithms with  $IP_{best}^i$  used as their ipvs. It is noteworthy that the values of  $Cdbw$  are comparable for different clustering algorithms since the definition of the validity index only depends on the partitioning and not on the algorithm itself. Then, we find the algorithm that results in the best partitioning of  $S$  by solving  $\max_{A_i \in A} \{Cdbw(C_{best}^i(A_i(IP_{best}^i)))\}$ . In other words, for each of the clustering algorithms  $A_i$  the best partitioning of  $S$ ,  $C_{best}^i = C(A_i(IP_{best}^i))$ , is selected. Then  $C_{best} = \{C_{best}^i, i = 1, \dots, m\}$  is the set of best partitionings defined by the considered algorithms. The partitioning with the highest  $Cdbw$  value in  $C_{best}$  is selected as the overall best partitioning and the respective algorithm  $A_k$  running with the  $IP_{best}^k$  set of ipvs is considered to be the most appropriate algorithm for  $S$ . In the sequel, we present the results of the experimental study we carried out using four widely used algorithms, one from each of the popular clustering algorithm categories:  $K$ -Means (partitional), DBSCAN (density-based), CURE (hierarchical) and an algorithm provided by the CLUTO toolkit which is based on the two-phase clustering approach of the CHAMELEON (Karypis, et al., 1999) algorithm.

The goal of the experimental results discussed below is not to evaluate the algorithms themselves and make inferences about their performance. Our approach is applied to the results of any clustering algorithm and achieves to select the partitioning that best fits the concerned data. Then the clustering algorithms in this experimental study are only considered to be the tools that assist with the definition of the clustering results to which our cluster validity approach is applied. Thus, in this work it is not important the algorithm and the values of its input parameters that we use but the clusterings that have been defined for the given data set.

A data set containing clusters with non-standard geometries that we used was DS1 (see Fig. 3). Specifically, we

consider the clusterings defined by the algorithms mentioned above while their *ipvs* are depicted in Table 1a. In case of DS1, *CDbw* takes its maximum value for the partitioning of eight clusters as defined by the CLUTO algorithm. We note that these are the real eight clusters presented in the DS1 as Fig. 3a also depicts. On the other hand, DBSCAN running with the set of *ipvs*, *Eps* = 13, *MinPts* = 35, discovers a set of eight clusters that approximates the real ones but it considers a part of the cluster 8 (in Fig. 3a) as noise. This observation justifies the slight decrease of the *CDbw* value in relation to its values when the real eight clusters are defined. Also the partitionings of DS1 into eight clusters as defined by *K*-Means and CURE are depicted in Fig. 3b and c, respectively. It is obvious that all algorithms except the CLUTO algorithm fail to partition it properly even in case that the correct number of clusters (i.e.,  $c = 8$ ) is considered.

Another example showing that an algorithm could cluster a data set finding the correct number of clusters but the wrong partitions is presented in Fig. 6. *CDbw* is able to evaluate the results of different clustering algorithms and select the best partitioning among those defined by the aforementioned algorithms, i.e., to select the best algorithm for a data set. According to Table 1b, in case of DS2 (see Fig. 6), *CDbw* takes its maximum value for the partitioning of four clusters as defined by DBSCAN. Fig. 6d presents the partitioning of DS2 into four clusters as defined by DBSCAN while its clustering into four clusters defined by *K*-Means, CURE and the CLUTO algorithm, are presented in Fig. 6a–c, respectively. It is

obvious that *K*-Means, CURE (with  $a = 0.3$ ,  $r = 10$ ) and the CLUTO algorithm (*clmethod* = graph, *sim* = dist, *agglofrom* = 30) failed to partition DS2 properly, even in case that the correct number of clusters (i.e.,  $c = 4$ ) is defined. This is verified by the *F*-measure values presented in Table 2. The partitioning that is selected by *CDbw* as the best one results in the highest value of *F*-measure, i.e., approximates the actual partitioning with the highest accuracy.

Similarly, we experimented with a data set containing ring shaped clusters. As Fig. 7 depicts the real clusters in DS3 are three. However, the majority of clustering algorithms fail to partition it into a right way. Fig. 7a and b present the partitioning of DS3 into three clusters as defined by *K*-Means and CURE, respectively. Moreover, Fig. 7c presents the partitioning of DS3 into three clusters as defined by DBSCAN and the CLUTO clustering algorithm. We observe that DBSCAN and the CLUTO algorithm are the only algorithms that achieve to discover the real clusters of DS3. This is also verified by our cluster validity approach. Specifically, Table 1(c) presents the values of *CDbw* for the clusterings defined by the aforementioned clustering algorithms. We observe that *CDbw* takes its maximum value for the clustering of three clusters as defined by DBSCAN and CLUTO, which also corresponds to the actual partitioning of DS3. It is also interesting that the values of *CDbw* corresponding to the best partitioning defined by DBSCAN and CLUTO are the same (0.32). This justifies our claim that *CDbw* values do not depend on the algorithms.

Table 1  
Best partitioning found by *CDbw* for different clustering algorithms

No clusters	<i>K</i> -Means		DBSCAN		CURE $r = 10$ , $a = 0.3$		CLUTO ( <i>clmethod</i> = graph, <i>sim</i> = dist, <i>agglofrom</i> = 30)	
	<i>ipvs</i>	<i>CDbw</i> value	<i>ipvs</i>	<i>CDbw</i> value	<i>ipvs</i>	<i>CDbw</i> value	<i>ipvs</i>	<i>CDbw</i> value
<i>(a) DS1</i>								
9	9	6.134E–4	–	–	9	8.491E–4	9	0.0024
8	8	0.0011	<i>Eps</i> = 13, <i>MinPts</i> = 35	0.00366	8	9.849E–4	<b>8</b>	<b>0.0037</b>
7	7	5.015E–4	<i>Eps</i> = 12, <i>MinPts</i> = 17	0.0027	7	0.00113	7	0.0028
6	6	7.765E–4	<i>Eps</i> = 13, <i>MinPts</i> = 15	0.0023	6	4.7152E–4	6	0.00209
5	5	5.604E–4	<i>Eps</i> = 14, <i>MinPts</i> = 15	0.0016	5	4.378E–4	5	0.00155
4	4	7.613E–4	<i>Eps</i> = 15, <i>MinPts</i> = 15	0.0011	4	4.8914E–4	4	0.00113
3	3	6.143E–4	<i>Eps</i> = 16, <i>MinPts</i> = 15	0.0018	3	6.5291E–4	3	0.00179
2	2	8.078E–4	<i>Eps</i> = 22, <i>MinPts</i> = 15	0.0012	2	1.1606E–4	2	–
<i>(b) DS2</i>								
6	6	0.0542	–	–	6	0.01234	6	0.06737
5	5	0.0440	–	–	5	0.0616	5	0.08407
4	4	0.0307	<b><i>Eps</i> = 1, <i>MinPts</i> = 4</b>	<b>0.1057</b>	4	0.0272	<b>4</b>	<b>0.1026</b>
3	3	0.0175	<i>Eps</i> = 2, <i>MinPts</i> = 15	2.89E–06	3	0.0229	3	0.0518
2	2	0.0494	<i>Eps</i> = 2, <i>MinPts</i> = 10	0.0749	2	0.0408	2	0.0749
<i>(c) DS3</i>								
6	6	0.01036	–	–	6	0.01149	6	0.0237
5	5	0.02257	–	–	5	0.02768	5	0.0263
4	4	0.02009	–	–	4	0.02432	4	0.0264
3	3	0.01993	<b><i>Eps</i> = 2, <i>MinPts</i> = 4</b>	<b>0.032</b>	3	0.02004	<b>3</b>	<b>0.032</b>
2	2	0.02743	<i>Eps</i> = 10, <i>MinPts</i> = 4	0.02743	2	0.02743	2	

*ipvs*: input parameters' value.

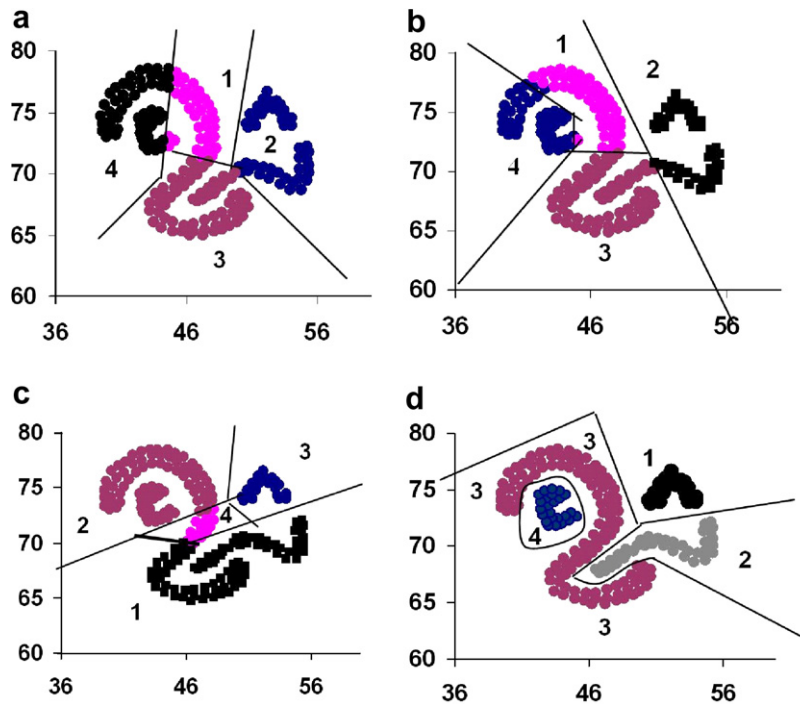


Fig. 6. Partitioning of DS2 into four clusters as defined by (a) K-Means, (b) CURE, (c) the CLUTO algorithm and (d) DBSCAN.

Table 2

Accuracy of the clusterings presented in Fig. 6 with respect to the expected partitioning of DS2

Algorithm	F-Measure
DBSCAN	1
CLUTO	0.5235
CURE	0.5037
KMEANS	0.4842

#### 4.4. Comparison to other cluster validity indices

In this Section we compare *Cdbw* to four of the most important validity indices proposed in the literature<sup>3</sup>, such as *RS-RMSSTD* (Sharma, 1996), *DB* (Theodoridis and Koutroubas, 1999), *SD* (Halkidi et al., 2000) and *S\_Dbw* (Halkidi and Vazirgiannis, 2001). The definition of the above indices is presented in (Halkidi et al., 2001).

*RMSSTD* and *RS* are representative examples of statistical validity indices and are jointly taken into account indicating the best number of clusters. The best partitioning of a data set corresponds to the number of clusters for which a significant local change in values of *RS* and *RMSSTD* occurs. As regards *DB*, *SD* and *S\_Dbw* the clustering for which the validity indices take its minimum value is selected as the best partitioning.

<sup>3</sup> Though Calinski and Harabasz (1974) is proposed as one of the best indices in 13 we exclude it from this study since it is predecessor of the four considered validity indices (*RS*, *RMSSDT*, *DB*, *SD* and *S\_Dbw*). Moreover its definition is based on terms that are similar to these used by the most recent indices *RMSSDT* and *RS* (see Jain et al., 1999).

Table 3 summarizes the results of the validity indices (*RS*, *RMSSDT*, *DB*, *SD*, *S\_Dbw* and *Cdbw*), for different clusterings of the aforementioned data sets as defined by a clustering algorithm (*K*-Means, CURE or DBSCAN). The comparison of validity indices refers to the same clustering results for each of the considered data sets. Also, we assume that there is at least a partitioning of the data sets among the evaluated ones that corresponds to their real clusters. As regards *Nd\_Set*, *SD* proposes three clusters as its best partitioning, while *S\_Dbw* selects the partitioning of eight clusters. On the other hand, *Cdbw*, *RMSSDT* & *RS* and *DB* propose four clusters, which corresponds to the number of clusters in the actual partitioning of *Nd\_Set*.

In case of DS1, DS2 and DS3 (containing arbitrarily shaped clusters), we consider the results of DBSCAN and the CLUTO algorithm since they achieve to handle efficiently arbitrarily shaped clusters. *Cdbw* selects the partitioning that contains the real eight clusters as the best one for DS1 whereas *RMSSDT* & *RS*, *DB*, *SD* and *S\_Dbw* fail, selecting the clusterings of three, four and ten clusters, respectively. Similarly, *Cdbw* finds the real four clusters as the best partitioning for DS2 (see Fig. 6d), on the contrary to *RS* & *RMSSTD*, *S\_Dbw* and *DB*, which propose three clusters as the best partitioning and *SD* that selects the partitioning of two clusters. Moreover, only *Cdbw* proposes the partitioning of DS3 into three clusters while all the others select the partitioning of two clusters as its best one. Based on the above observations we conclude that *Cdbw* achieves to find the clustering that best fits a data set, while other validity indices fail in

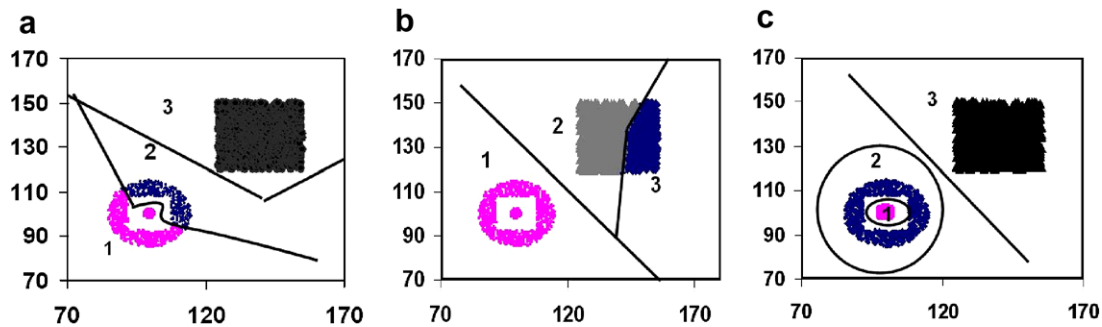


Fig. 7. Partitioning of DS3 into three clusters as defined by different clustering algorithms. (a) K-Means, (b) CURE and (c) DBSCAN, CLUTO.

Table 3

Best partitioning proposed by validity indices compared with  $CDbw^*$

Actual partitioning	DS1	DS2	DS3	Nd_Set
	8	4	3	4
RS, RMSSTD	3	3	2	4
DB	4	3	2	4
SD	4	2	2	2
S_Dbw	10	3	2	8
CDbw	8	4	3	4

some cases, especially when the data sets contain arbitrarily shaped clusters.

## 5. Related work

Since clustering algorithms discover clusters, which are not known a-priori, the final partitioning of a data set requires some sort of evaluation in most applications (Razaee et al., 1998). Requirements for the evaluation of clustering results are well known in the research community and a number of efforts have been made especially in the area of pattern recognition. However, the issue of cluster validity is rather under-addressed in the area of databases and data mining applications, even though recognized as important. There are three approaches to investigate cluster validity (Theodoridis and Koutroubas, 1999). The first is based on *external criteria*. This implies that we evaluate the results of a clustering algorithm based on a pre-specified structure, which is imposed on a data set and reflects our intuition about the clustering structure of the data set. The second approach is based on *internal criteria*, meaning that the results of a clustering algorithm are evaluated in terms of quantities that involve the vectors of the data set themselves (e.g., proximity matrix). The third approach is based on *relative criteria*. Here the basic idea is to choose the best clustering among a set of different clusterings defined according a pre-specified criterion. A number of validity indices appear in the literature for each of the above approaches (Theodoridis and Koutroubas, 1999). A cluster validity index for crisp clustering that is proposed in (Dunn, 1974), attempts to identify “compact and well-separated clusters”. Other validity indices for

crisp clustering have been proposed in (Dave, 1996 and Halkidi and Vazirgiannis, 2001). The implementation of most of these indices is computationally expensive, especially when the number of clusters and number of objects in the data set grows a lot (Smyth, 1996). In (Milligan and Cooper, 1985) an evaluation study of 30 validity indices proposed in the literature is presented. The results of this study rate the indices Caliski and Harabasz (1974), Je(2)/Je(1) (1984), C-index (1976), Gamma and Beale among the six best indices. However, it is noted that although the results concerning these methods are encouraging they are likely to be data dependent, i.e., the characteristics of data can affect their performance in an unpredictable way. Thus there is no guarantee that they will be best for real data set. An overview of fuzzy cluster validity indices is presented in (Theodoridis and Koutroubas, 1999). Some initial efforts in this field is the *partition coefficient* (1974) and the *classification entropy* (1984) proposed by Bezdek. Also Maulik and Bandyopadhyay (2002) proposed a fuzzy clustering validity index that is defined based on (i) the ratio of the whole dataset variance and the fuzzy variance of defined clusters and (ii) the maximum separation between two clusters over all possible pairs of clusters. A cluster validity index for estimating the validity of the Fuzzy C-Means results is presented in (Kim et al., 2003). The proposed validity index measures the degree of overlap between clusters estimating the inter-cluster proximity between fuzzy clusters. Thus this index only evaluates the separation of clusters while it ignores their compactness as criterion of clusters validity. Other fuzzy validity indices are proposed in (Gath and Geva, 1989; Xie and Beni, 1991). In general terms aforementioned indices due to their definition tend to favour convex clusters where the statistical measures such as variance and distances between points can give a good indication of clusters’ compactness and separation.

A practical clustering algorithm based on Monte Carlo cross-validation is proposed in (Smyth, 1996). This approach differs significantly from the one we propose. While we evaluate clusterings based on widely recognized validity criteria of clustering, the evaluation approach proposed in (Smyth, 1996) is based on density functions considered for the data set. Thus, it uses concepts related to probabilistic models in order to estimate the number of



clusters, better fitting a data set, while we use concepts directly related to the data.

Abul et al. (2003) propose three methods for evaluating the validity of clustering results. The first method validates the clustering results based on supervised classifiers. The idea on which this method is based is that if the labels generated by a clustering algorithm are valid then they can be used to build an accurate classifier. The rationale behind the second method is that if a clustering is valid then each of its subsets should be valid as well. The third method is similar to the second one, evaluating each cluster separately in terms of its stability and compactness.

A cluster validity approach that is based on resampling and the evaluation of clustering solutions stability is introduced in (Levine and Domany, 2001). The authors introduce the concept of figure of merit, which reflects the stability of cluster partition against resampling. Criteria related to the data distribution in clusters and the validity of clusters in terms of their compactness and separation are not used. Specifically, the figure of merit measures the extent to which the clustering assignment obtained from the resamples agrees with that of the full sample. Among different clustering solution the one which is more robust according to the figure of merit is considered to be the best solution.

Another cluster validation approach based on stability is proposed in (Lange et al., 2004). The authors introduce a stability measure that aims to quantify the reproducibility of clustering results on a second sample of data. According to this approach, we split the data set into two halves  $X$ ,  $X'$  and a clustering algorithm  $A_k$  is applied to both. The clustering of  $X$  defined by  $A_k$  (that is  $A_k(X)$ ) is used to train a classifier  $\phi$ . Then the dissimilarity of two solutions  $A_k(X)$  and  $\phi(X')$  is calculated in terms of stability measure. The stability measure is estimated for different number of clusters. The number of clusters that corresponds to the smallest estimate of stability measure is chosen as the preferred partitioning of the given data.

A validity approach for selecting the best parameter of kernel functions used in the context of Support Vector Clustering (SVC), is presented in (Chiang and Wang, 2004). The proposed measure is defined based on widely used cluster validity criteria regarding clusters' compactness and separation. The compactness of clusters is evaluated based on the overall distance between the data points inside the clusters while the separation is defined as the minimum distance between the support vectors between clusters.

The proposed approach aims to introduce some new criteria in the cluster validity so that we efficiently handle arbitrarily shaped clusters. It exploits measures that assess the density distribution in areas between and within the defined clusters. We also introduce the idea of evaluating the density changes within clusters in order to estimate the clusters' cohesion. The cohesion concept used in this work seems very close to the path-based clustering criteria proposed by Fred and Leita (2003). However these crite-

ria are based on distances and measure the dissimilarity increments between clusters contrary to our cohesion criterion that estimates density changes within clusters.

## 6. Conclusions

In this paper, we defined a new validity index,  $CDbw$ , and a methodology for finding the clustering among those defined by an algorithm or different clustering algorithms that best fits data.  $CDbw$  adjusts well to non-spherical and skewed cluster geometries, contrary to the validity indices proposed in the literature. It achieves this by considering multi-representative points per cluster. The proposed cluster validity index is fully implemented and experiments prove its efficiency for various data sets and algorithms.

The proposed approach is defined in the context of crisp clustering corresponding to the results of the majority of clustering algorithms. One of our future work directions will be the definition of the cluster validity index so that both structural and fuzzy aspects of the data distribution are taken into account. Furthermore, we plan an extension of this effort to be directed towards an integrated algorithm for cluster analysis putting emphasis on the geometric features of clusters, using sets of representative points, or even multidimensional curves. Another interesting research direction that we consider for our future work is to adapt user constraints in the clustering and cluster validity process so as to overcome the possible limitation of the current approach in relation to the specific requirements of the application domains.

## Acknowledgements

We wish to thank C. Rodopoulos and C. Amanatidis for the implementation of CURE algorithm. Also we are grateful to Drs. Joerg Sander and Eui-Hong (Sam) Han for providing information and the source code for DBSCAN and CURE algorithms, respectively. The work of Dr. M. Halkidi was partially funded by the Marie Curie Outgoing Int. Fellowship (MOIF-CT-2004-509920) from EU Commission.

## References

- Abul, O., Lo, A., Alhajj, R., Polat F., Barker, K., 2003. Cluster validity analysis using subsampling. In: Proc. IEEE Int. Conf. Systems Man Cybernet., Washington DC, October 2003.
- Berry, M., Linoff, G., 1996. Data Mining Techniques For marketing Sales and Customer Support. John Wiley & Sons, Inc.
- Calinski, R.B., Harabasz, J., 1974. A dendrite method for cluster analysis. Commun. Stat. 3, 1–27.
- Chiang, Jen-Chieh, Wang, Jeen-Shing, 2004. A validity-guided support vector clustering algorithm for identification of best cluster configuration. In: Proc. IEEE Int. Conf. Systems Man Cybernet., Washington DC, October 2004.
- Dave, R.N., 1996. Validating fuzzy partitions obtained through c-shells clustering. Pattern Recognition Lett. 17, 613–623.

- Dunn, J.C., 1974. Well-separated clusters and optimal fuzzy partitions. *J. Cybern.* 4, 95–104.
- Ester, M., Kriegel, H.-P., Sander, J., Xu, X., 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. 2nd Int. Conf. Know. Disc. Data Min.*, Portland, OR, pp. 226–231.
- Fred, A., Leitaio, J., 2003. A new cluster isolation criterion based on dissimilarity increments. *IEEE TPAMI* 25 (8).
- Gath, I., Geva, B., 1989. Unsupervised optimal fuzzy clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* 11 (7), July.
- Guha, S., Rastogi, R., Shim, K., 1998. CURE: An efficient clustering algorithm for large databases. In *Proc. ACM SIGMOD Int. Conf. Manage. Data*, June 2–4, Seattle, Washington, USA.
- Halkidi, M., Vazirgiannis, M., 2001. Clustering validity assessment: Finding the optimal partitioning of a data set. In: *Proc. ICDM Conf.*, California, USA, November 2001.
- Halkidi, M., Vazirgiannis, M., Batistakis, Y., 2000. Quality scheme assessment in the clustering process. In: *Proc. PKDD Conference*, Lyon, France, 2000.
- Halkidi, M., Batistakis, Y., Vazirgiannis, M., 2001. On clustering validation techniques. *J. Intell. Inform. Syst.* 17 (2/3), 107–145.
- Hochbaum, D., Shmoys, D.B., 1985. A best possible heuristic for the  $k$ -center problem. *Math. Oper. Res.* 10, 180–184.
- Karypis, G., 2003. CLUTO: A clustering Toolkit. Release 2.1.1. <<http://www-users.cs.umn.edu/~karypis/cluto/>>.
- Karypis, George, Han, Eui-Hong (Sam), Kumar, Vipin, 1999. CHAMELEON: A hierarchical clustering algorithm using dynamic modeling. *IEEE Comput.*
- Kim, W., Lee, K.H., Lee, D., 2003. Fuzzy cluster validation index based on inter-cluster proximity. *Pattern Recognition Lett.* 24, 2561–2574.
- Jain, A.K., Murty, M.N., Flynn, P.J., 1999. Data clustering: A review. *ACM Comput. Surv.* 31 (3).
- Lange, T., Roth, V., Braun, M., Buhmann, J., 2004. Stability-based validation of clustering solutions. *Neural Comput.* 16, 1299–1323.
- Levine, E., Domany, E., 2001. Resampling method for unsupervised estimation of cluster validity. *Neural Comput.* 13 (11), 2573–2593.
- Maulik, U., Bandyopadhyay, S., 2002. Performance evaluation of some clustering algorithms and validity indices. *IEEE Trans. Pattern Anal. Mach. Intell.* 24 (12), 1650–1654.
- Milligan, G.W., Cooper, M.C., 1985. An examination of procedures for determining the number of clusters in a data set. *Psychometrika* 50, 159–179.
- Razaei, R., Lelieveldt, B., Reiber, J., 1998. A new cluster validity index for the fuzzy c-mean. *Pattern Recognition Lett.* 19, 237–246.
- Sharma, S.C., 1996. *Applied Multivariate Techniques*. John Wiley & Sons.
- Sheikholeslami, C., Chatterjee, S., Zhang, A., 1998. WaveCluster: A multiresolution clustering approach for very large spatial database. In: *Proc. 24th VLDB Conf.*, New York, USA.
- Smyth, P., 1996. Clustering using Monte Carlo cross-validation. In: *Proc. KDD Conf.*, pp. 126–133.
- Theodoridis, S., Koutroubas, K., 1999. *Pattern Recognition*. Academic Press.
- Van Rijsbergen, C.J., 1979. *Information Retrieval*, second ed. Butterworth, London.
- Xie, X., Beni, G., 1991. A validity measure for fuzzy clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* 13 (4).