Received: 24 October 2020 | Revised: 31 January 2021 | Accepted: 11 May 2021

DOI: 10.1002/int.22521

RESEARCH ARTICLE

WILEY

New internal clustering validation measure for contiguous arbitrary-shape clusters

Juan Carlos Rojas-Thomas[1] | **Matilde Santos**[2]

[1]Informatics and Automatic Control, UNED, Madrid, Spain

[2]Institute of Knowledge Technology, University Complutense of Madrid, Madrid, Spain

Correspondence
Juan Carlos Rojas-Thomas, Informatics and Automatic Control, UNED, C/Juan del Rosal s/n, 28040-Madrid, Spain.
Email: correorojas@gmail.com

Abstract

In this study a new internal clustering validation index is proposed. It is based on a measure of the uniformity of the data in clusters. It uses the local density of each cluster, in particular, the normalized variability of the density within the clusters to find the ideal partition. The new validity measure allows it to capture the spatial pattern of the data and obtain the right number of clusters in an automatic way. This new approach, unlike the traditional one that usually identifies well-separated compact clouds, works with arbitrary-shape clusters that may be contiguous or even overlapped. The proposed clustering measure has been evaluated on nine artificial data sets, with different cluster distributions and an increasing number of classes, on three highly nonlinear data sets, and on 17 real data sets. It has been compared with nine well-known clustering validation indices with very satisfactory results. This proves that including density in the definition of clustering validation indices may be useful to identify the right partition of arbitrary-shape and different-size clusters.

KEYWORDS

arbitrary-shape clusters, clustering, density, internal validation index, real data sets, uniformity

1 | INTRODUCTION

The process of clustering consists of classifying in an unsupervised manner a set of data in groups. Clustering algorithms group objects that are similar to each other in clusters.[1] Different clustering algorithms have been developed and applied to different data sets of very varied fields (medical databases, images, etc.).

Although the different clustering algorithms have been proved very efficient in grouping data, one of the most important issues in clustering analysis is the evaluation of the performance of the algorithm.[2,3] This involves not only finding the right number of clusters but also the group partition that best fits the underlying structure of the data. This is the main objective of clustering validation.[4–6]

Clustering validity indices (CVIs) are defined to do so.[7] Among them, internal validation measures have the advantage of not requiring extra information about the data, nor do they require repetition of the clustering process. They only depend on some properties of the resulting clusters, such as the level of compactness, the separation between clusters, and the degree of roundness.[8]

Most of the internal indices are based on these two concepts, separation among clusters and the cluster compactness, although they differ in the way they measure them. Moreover, these indices usually assume that clusters form compact data clouds with a certain degree of separation. Nevertheless, for clusters not necessarily spherical these relative validity indices may fail.[9] Indeed, the existing CVIs are sensitive to clusters with arbitrary shapes, especially for high-dimensional data. The traditional validation measures are considerably dependent on the number of data objects in clusters, on cluster centers, and on average values and thus, do not work well for clusters with different densities and/or sizes.[10] Comparative studies of such indices show that there is no optimal CVI able to cope successfully with all the contexts.[11,12]

In this study, the clustering evaluation of data sets that may be not composed of spherical clusters is addressed. A new internal validation measure is proposed that is applied to contiguous clusters, that is, adjacent clusters with different shapes and sizes that form a single data cloud. This new index is based on a characteristic of the clusters called uniformity. It combines the spatial pattern of the data and the local density. That is, the Contiguous Density Region (CDR) index detects clusters that follow a uniform spatial pattern, and differentiates them based on their densities. The minimization of the CDR index estimates the optimal partition of the data among all the partitions generated by the clustering algorithm based on the structural characteristics of the data; therefore, it is a criterion to automatically select the number of clusters.

The performance of this new internal index has been compared with other traditional validation measures on different artificial and real data sets. Results prove that the spatial texture-based index performs better than the traditional approach that assumes well-separated compact clouds of points. It is able to deal with contiguous clusters of arbitrary shapes that can even be overlap.

The main goal of this study is to extend the internal evaluation of data clustering to cluster configurations in which the main discriminant is the spatial structure of the data. The utility of the proposal of new internal indices is that these CVIs can be applied to nonspherical clusters and it is motivated by the fact that real-world data sets have usually arbitrary shapes.[13] According to the authors' experience, in many supervised classification problems the structure of the data is much closer to such a configuration than the traditional one that considers well-separated compact clouds. Moreover, it is shown how including the concept of data density

within a cluster in the definition of the internal clustering validation indices helps handle different shape, density, and/or size contiguous clusters. This way we think we contribute to enriching the range of tools available to analyze the data.

The rest of the paper is organized as follows. Section 2 describes the state-of-the-art regarding the use of density in clustering. Section 3 presents the new contiguous region paradigm. The definition of the new index is proposed in Section 4. In Section 5, experimental results of the CVI for artificial and real data sets are shown and discussed. Finally, Section 6 summarizes the main conclusions and future works.

2 | RELATED WORK

The new proposed CVI uses the local density to generate partitions. Data density measure has been widely used in the definition of clustering algorithms[14] but not in the validation measures definition.

The algorithms based on this data characteristic are classified into two categories: density-based and grid-based clustering algorithms. While the first ones calculate the density in the local neighborhood of the data, the latter uses the whole gridding space to estimate the density of each cell, which is then used to obtain the clusters. Clusters are defined as dense regions separated by low-density regions. Both use the data density to detect well-separated data clouds, following the traditional approach, although the calculation of the density of the clusters allows these methods to detect clusters of nonspherical geometries.

A density-based algorithm needs to scan only once the original data set and can handle noise. The number of clusters is not required in advance.[15] These methods commonly define density in terms of a number of neighbors located in a restricted hypervolume centered on data. A baseline algorithm of this category is DBSCAN,[16] which uses as density the number of data contained within a hypersphere of radius fixed a priori. A new version of this algorithm, DBCLASD, that uses reverse nearest-neighbor counts as an estimate of observation density has been recently developed.[17] Other density-based approaches are presented in References [18,19]. In the latter, the DENCLUE algorithm identifies clusters by determining density-attractors and clusters of arbitrary shape and noise can easily be described by a simple density function.

On the other hand, the grid-based clustering algorithms use dense grid cells to form clusters.[20] A typical example is the GRIDCLUS algorithm,[21] which calculates the cell density by dividing the number of data contained in each block by its hypervolume. One of the problems of grid-based clustering algorithms is how to choose the grid size or density thresholds. Another difficulty is the exponential growth in the number of cells as the number of dimensions in the feature space increases. As a consequence, many of these algorithms are only efficient in spaces of low-dimensional characteristics or work with optimized partitions for each dimension of the space.[22] In Reference [23] a new clustering validity index (BCVI) is designed to better evaluate the quality of clustering results generated by the grid-*k*-means algorithm.

In spite of the extensive literature on density-based clustering algorithms, and on relative internal validation criteria, not much attention has been given to density-based clustering validation measures. In fact, it is very difficult to find studies about clustering validation for dense, sparse, and arbitrary-shaped clusters.[24]

Some works have developed internal indices for different-size or density arbitrary-shape clusters, but they are not based on the density neither is this concept included in the index definition.[9] For example, Žalik and Žalik[10] propose two validation indices for clusters that differ

in sizes and densities. They do not use the density to define the index. Instead, they use the compactness to define the first index, and the overlap measure of the clusters for the second index. Similarly, the index proposed in Reference [25] applies cluster compactness, defined as the average of all its diameters, and the intercluster distance. The paper by Rodríguez et al.[26] proposes an ensemble of different supervised classifiers to obtain the quality of partitions not having a hypersphere underlying structure. Lee et al.[11] propose a Support Vector Data Description (SVDD)-based CVI, in which the compactness of a cluster is measured in the kernel space, in an attempt to overcome the sensitivity of this measure for arbitrary shapes, subclusters, and noise in data. In a related paper, a relative validation index for density-based, arbitrarily shaped clusters, is proposed.[9] The index assesses clustering quality based on the relative density connection between pairs of objects. It uses a new distance measure to build a minimum spanning tree for each cluster, similarly to Reference [27], where two different versions of a new internal index for clustering validation based on graphs were proposed.

Other works take into account the spatial distribution of the data, such as in Reference [28], where authors propose a new internal cluster validity index based on the cluster center; the nearest-neighbor cluster is designed according to the geometric distribution of objects. The work presented in Reference [29] evaluates the performance of a set of internal clustering indices regarding a specific structural characteristic. Particularly, it deals with data sets whose clusters present asymmetry in their geometries. The work by Nerurkar et al.[30] focuses on the use of internal validation criteria, in particular the BetaCV and Dunn internal indices, as cost functions of swarm optimizers. In a similar way, Reséndiz, Castro, and Leal[31] propose the use of clustering validation indices and maximum entropy as cost functions to quantify the quality of automatic image segmentation. A recent paper by Xu et al.[18] proposes a unified validity index framework for complex data structure using hierarchical clustering.

Finally, we have studied in depth the closest recent papers that deal with internal CVIs that use the concept of density and analyzed the main differences with our proposal. Hu and Zhong[32] design a new robust density-involved distance measure. They then define an internal validity index based on this separation measure using minimum spanning trees. The index does not measure the separation of the clusters by their global densities but the similarity between two particular objects, which is the main difference with our approach. In fact, the artificial data sets used to test the index performance present very small adjacency between clusters, mostly defined by spherical and elliptical geometries. A new Local Cores-Based Cluster Validity (LCCV) index is proposed by Cheng et al.[33] Local cores, with local maximum density, are selected as representative points. A neighbor-based local density is defined and then the graph-based distance is used to evaluate the dissimilarity between local cores. The index has been shown to be effective for data sets that contain clusters with arbitrary shapes, but it does not apply to contiguous or overlapping clusters. In a more recent paper by the same authors,[34] they use local density peaks to represent the whole data set and define a shared neighbors-based distance between local density peaks to better measure the dissimilarity between objects on varied data. Similarly, Xie et al.[35] propose a density-core-based clustering validation index with a minimum spanning tree to solve the problem of noisy and arbitrary-shape clusters. In Reference [36], a new version of the Davies–Bouldin index uses a cylindrical distance to estimate the degree of separation between clusters. This distance definition captures the data density in a limited region of the space around straight line segments that connect the clusters centroids.

Therefore, to the best of our knowledge, density has been hardly included in the definition of internal clustering validation indices. That is why we propose to use this measure to define a new clustering validation measure to deal with different density and/or size contiguous clusters.

3 | NEW DENSITY-BASED APPROACH AND CDR DENSITY-BASED INTERNAL INDEX

Traditional clustering considers clusters as compact, well-separated clouds of data, with more or less complex geometries. For this reason, internal validation indices are mainly defined in terms of compactness and separability, and classified according to the way these two criteria are combined. However, Zahn[37] proposed the so-called density gradient problem. He worked with a family of graph–theory algorithms based on the minimal spanning tree that is capable of detecting different types of cluster structures in arbitrary data sets. This framework is shown in Figure 1 (left), where the clusters are represented as two adjacent homogeneous regions; the main difference between them lies in density rather than in distance. Zhong et al.[38] continued with this approach using an algorithm that generated the minimal spanning tree in two steps: first to detect separated clusters and then adjacent clusters.

Inspired by this density gradient problem, we have extended it and proposed a new clustering paradigm that allows us to define internal validation indices that are able to deal with noisy and arbitrary-shape clusters.

This new approach is called *Contiguous Region Paradigm*. It considers any n number of adjacent clusters. Moreover, it does not impose any restriction on the geometry of the whole data set, allowing the exploration of different spatial data configurations, including non-spherical ones. Besides, it is possible to consider different degrees of overlapping between neighboring clusters.

The clusters are considered as relatively homogeneous data clouds regarding their own density (local density), and they are adjacent to each other (Figure 1, right, three adjacent homogeneous clusters with different densities). The distance of the data to the nearest neighbors is usually used to calculate the density of the cluster. But in homogeneous clusters the distance is similar for all the data. That is why our proposal does not only try to identify the clusters based on the estimated local densities of the data, but also it mainly focuses on capturing the degree of homogeneity of these clouds, that is, the degree of uniformity of the distribution of the data in the feature space, understanding uniformity as a measure of the variability of the local density.

This property, uniformity, is the key factor to identify the clusters and to define the new validation index.

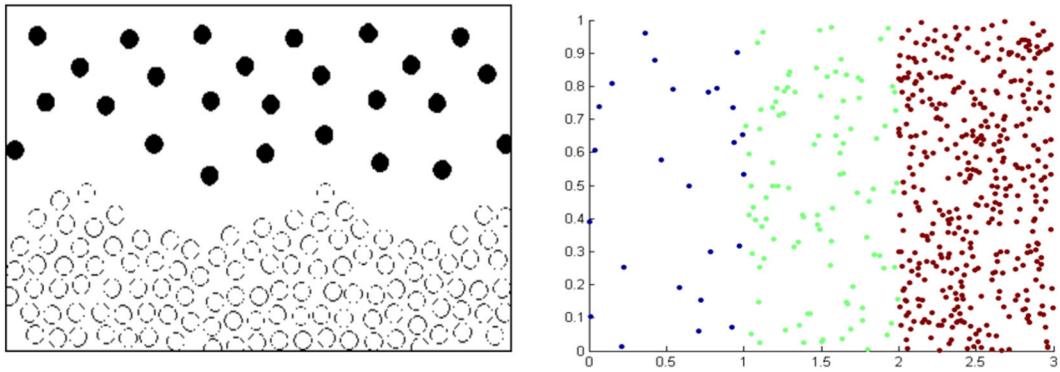

FIGURE 1 Density gradient problem (left) (Zahn, 1971)[37]; contiguous region paradigm with three adjacent clusters (right) [Color figure can be viewed at wileyonlinelibrary.com]

3.1 | New internal clustering validation index definition

The new clustering validation measure, called CDR index, identifies the partition that presents the largest difference between the clusters regarding their respective densities, while maintaining the degree of density variation within each cluster the lowest possible.

To measure it, once different partitions (with different number of clusters) have been obtaining by any clustering algorithm, we first obtain the absolute value of the variability of the local densities for each cluster. Then, these variability measures are normalized. Normalization is carried out by calculating the relative variability of the local densities with respect to the average value of each cluster local density. This feature of each cluster is the uniformity.

These normalized density variation values are sorted in increasing order in relation to the number of clusters of their respective partitions. Finally, the partition that shows the greatest relative improvement is selected.

The mathematical formulation of the index definition uses the following nomenclature: S is the input data set, with n_{tot} objects or points, x_i; g is the center of whole data set S; s is the number of dimensions of S; k is the number of clusters; C_i is the ith cluster, n_i is the number of data points in C_i, c_i is the center of cluster C_i, and $d(x, y) = ||x - y||^2$ is the distance between any two points x and y.

Definition 1. Local density.

The clusters to be analyzed form a single continuous cloud of data, so to establish a density threshold to distinguish dense from empty regions is no longer useful. It is necessary to calculate the density of each region.

The local density of an object, x_i, in the feature space S is defined as the distance to the nearest neighbor, x_j, that belongs to the same cluster, expressed as

$$local\_den(x_i)_{x_i \in C_k} = min\{d(x_i, x_j)\}, \quad \forall\, x_j \in C_k, i \neq j, \tag{1}$$

where $d()$ can represent any distance measure, and C_k is the kth cluster.

We have selected the Euclidean distance due to its extensive and proven use in the literature on clustering validation measures.[39] In addition, all the internal indices with which the proposal has been compared to use this distance measure as well. When the Euclidean distance is used, it is denoted as $d_e()$. Hence,

$$local\_den_e(x_i)_{x_i \in C_k} = min\{d_e(x_i, x_j)\}, \quad \forall\, x_j \in C_k, i \neq j. \tag{2}$$

Definition 2. Density of a cluster.

The average density of a cluster C_k is calculated as the average of all the local densities of the cluster, that is,

$$avg\_den(C_k) = \frac{\sum_{i=1}^{n_k} local\_den(x_i)}{n_k}, \quad \forall\, x_i \in C_k, \tag{3}$$

where n_k is the total number of data in cluster C_k.

Definition 3. Uniformity.

It measures the degree of local density variation within a cluster. Given a cluster C_k, its uniformity is given by

$$unif(C_k) = \begin{cases} \frac{\sum_{i=1}^{n_k} |local\_den(x_i) - avg\_den(C_k)|}{avg\_den(C_k)}, & x_i \in C_k, n_k > 1, \\ 0, & n_k = 1. \end{cases} \quad (4)$$

Thus, the uniformity of each cluster is calculated as the difference between the local density of the cluster and the average of all the local densities of the cluster.

The goal is to find the cluster that presents the smallest variability, that is, the most uniform spatial data distribution. Thus, the lower the value of (4), the greater the uniformity.

Definition 4. Contiguous Density Region (CDR) index.

Given a partition $P^k = \{C_1, ..., C_k\}$ of a data set S, with k clusters, the value of the *CDR* index is defined as

$$CDR(P^k) = \frac{\sum_{i=1}^{k} n_i * unif(C_i)}{n_{\text{tot}}}, \quad (5)$$

where n_i is the number of data in cluster C_i, and n_{tot} corresponds to the total number of data of data set S.

Equation (5) represents a weighted sum of the degree of uniformity of each individual cluster. The relative weight of each cluster depends on the percentage of data that belong to the cluster, out of the total number of data of the set. This way, it is guaranteed that the index favors partitions with significant clusters, and discards little relevant clusters, such as those formed by very few data or even a single data.

The goal of this index is to identify the cluster with the highest degree of uniformity regarding the density, so this index must be minimized (as it is usually the case for internal validation measures).

3.2 | CDR index calculation

Once these key terms have been defined, the minimum of the CDR index is calculated as follows.

1. Let $R = \{P^1, ..., P^m\}$ be the set of the different partitions generated by any clustering algorithm applied to the S data set. They are sorted according to the number of clusters each one has (from 1 to m clusters). The *CDR* index is calculated for each of these partitions.

$$CDR(R) = \{CDR(P^1), CDR(P^2), ..., CDR(P^m)\}. \quad (6)$$

Figure 2 shows an example of the values of the CDR index for 10 different partitions of a given data set. The horizontal axis represents the number of clusters of each partition, and the vertical axis represents the values of the CDR index. The searching of the local minimum is represented by the red arrow.

The process of finding the minimum index value starts with $CDR(P^2)$ (the smallest solution is the one with at least two clusters), and continues as long as $CDR(P^i) > CDR(P^{i+1})$, where i is the number of clusters of the partition, $1 < i \leq m$. When this condition is no longer met, that is, $CDR(P^i) \leq CDR(P^{i+1})$, the process stops and the first ith partitions are selected for later analysis (vertical red line in Figure 2, partitions with up to four clusters).

2. The next step is to measure the relative improvement between two consecutive partitions, k and $k-1$, through the following factor, CDR_f, defined as

$$CDR_f^k = \frac{CDR(P^k)}{CDR(P^{k-1})}, \quad k = 2, ..., m. \tag{7}$$

3. Finally, the jth partition that has the greatest improvement, that is, the lowest CDR_f^j value (3), is chosen.

$$j : CDR_f^j = \min_m \left\{ CDR_f^2, ..., CDR_f^m \right\}. \tag{8}$$

As it is possible to see in Figure 2, the partition with the smallest CDR value is not necessarily the one with the best CDR improvement factor. In this case, P^4 has the minimum CDR but P^3 is the one with the lowest CDR_f, that is, the best relative improvement of the index. So, P^3 would be selected as the best partition according to the CDR index.

4 | MATERIALS: ARTIFICIAL AND REAL DATA SETS

To test the internal validation measures, initially nine artificial data sets were generated. These synthetic data are used for the validation and design of experiments.[40] The artificial data sets consist of a single continuous cloud of data in a two-dimensional space which presents regions with different densities. This description allows a wide range of different spatial distributions of the data.

The method of generation of artificial data sets is as follows (Figure 3). Data of the clusters are uniformly distributed in a rectangular grid. The data are placed at the intersections of the lines of the grid. This way a cluster of data linearly spaced with certain density and rectangular shape is obtained. To generate clusters with nonsquare geometries, such as ring-shaped clusters, taking as a starting point the rectangular cluster, two parameters, the outer radius (larger) and the inner radius (smaller) are defined. A mask is applied to remove all the data that are not

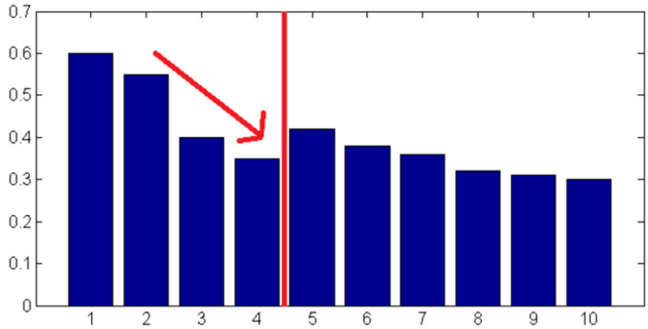

FIGURE 2 Searching for a local minimum of the CDR index values for 10 partitions [Color figure can be viewed at wileyonlinelibrary.com]

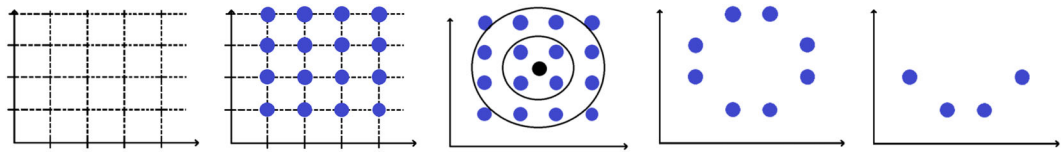

FIGURE 3 Example of generation of artificial data sets. From left to right: grid, rectangular cluster, ring-shape cluster, and arch-shape cluster [Color figure can be viewed at wileyonlinelibrary.com]

within the ring. To build clusters with arc geometry, a linear filter is applied to a ring cluster in such a way as to eliminate the data that are over or under a straight line parallel to any of the axes. The same procedure can be applied to obtain clusters with any desired shape. The density of each cluster may vary.

Figure 3 shows an example of how the artificial data sets are generated.

4.1 | Artificial data sets

For the generation of the different configurations of the data, according to the methodology described above, the following two criteria were adopted, where the term class refers to the correct number of clusters, that is, the target partition.

1. *Spatial distribution*: According to the geometrical distribution of the clusters, the next subcategories are defined:

i. *Linear:* Sets of clusters whose spatial distribution follows a straight line.
ii. *Nonlinear*: Sets of clusters whose spatial distribution is:

a. *A perpendicular arch*: A continuous region of clusters whose shape follows two perpendicular lines.
b. *A matrix*: A continuous region of clusters distributed along and across the space, resembling a matrix of clusters.

2. *Superposition*:

i. *Clusters that are not overlapped*: Adjacent clusters with well-separated edges.
ii. *Overlapped clusters*: Adjacent clusters with regions partially overlapped.

Combining these two criteria, nine different test sets have been built (Figure 4). In Figure 4, the points with the same color represent a cluster. As it is possible to see, the density of each cluster is different. The description of each data set of Figure 4, from top to bottom, left to right, is given below, where the number followed by *C* means the number of clusters.

(a) Linear 2C: two classes, horizontal linear configuration, same area, without overlapping, with 16 and 49 data points, respectively (Figure 4A).
(b) Linear 3C: three classes, horizontal linear configuration, same area, without overlapping, with 16, 49, and 100 data, respectively (Figure 4B).

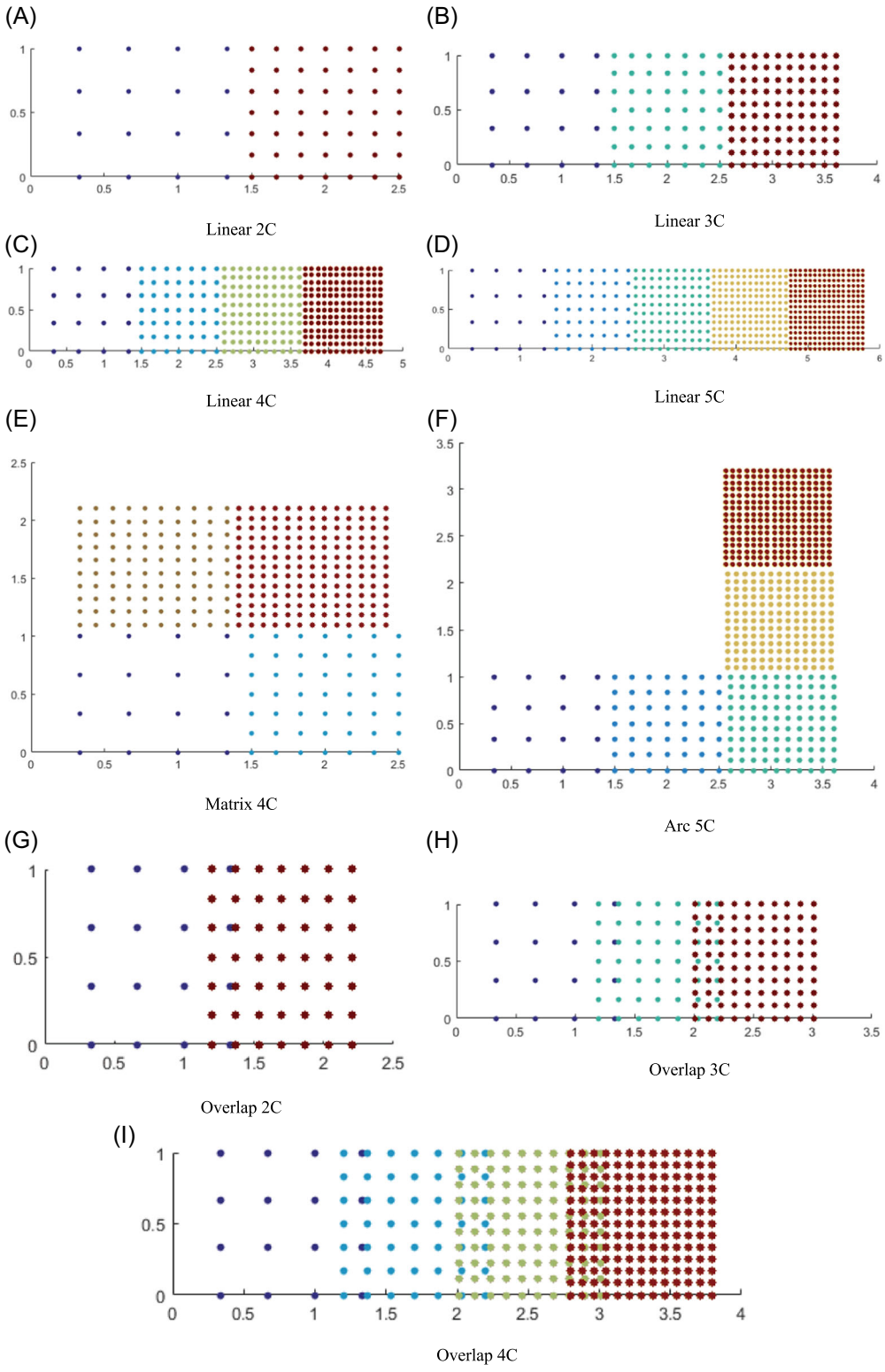

FIGURE 4 From top to bottom, left to right, artificial cluster configurations: (A) Linear 2C, (B) Linear 3C, (C) Linear 4C, (D) Linear 5C, (E) Matrix 4C, (F) Arc 5C, (G) Overlap 2C, (H) Overlap 3C, and (I) Overlap 4C [Color figure can be viewed at wileyonlinelibrary.com]

(c) Linear 4C: four classes, horizontal linear configuration, same area, without overlapping, with 16, 49, 100, and 169 data, respectively (Figure 4C).
(d) Linear 5C: five classes, horizontal linear configuration, same area, without overlapping, with 16, 49, 100, 169, and 256 data, respectively (Figure 4D).
(e) Matrix 4C: four classes, matrix configuration, same area, without overlapping, with 16, 49, 100, and 169 data, respectively (Figure 4E).
(f) Arc 5C: five classes with a perpendicular arch configuration, same quadrangular area, without overlapping, with 16, 49, 100, 169, and 256 data, respectively (Figure 4F).
(g) Overlap 2C: two classes with a horizontal linear configuration, same quadrangular area, overlapped, with 16 and 49 data, respectively (Figure 4G).
(h) Overlap 3C: three classes with a horizontal linear configuration, same quadrangular area, overlapped, with 16, 49, and 100 data, respectively (Figure 4H).
(i) Overlap 4C: four classes with a horizontal linear configuration, same quadrangular area, overlapped, with 16, 49, 100, and 169 data, respectively (Figure 4H).

4.2 | Artificial nonlinear data sets

Three different artificial data sets, taken from or inspired by the paper published by Cheng et al.,[33] have been also considered. They are highly nonlinear data sets, with different geometries and different data densities. Moreover, as it will be shown, these sets have been evaluated using two different clustering methods: *k*-means and hierarchical algorithms.

The description of these generated artificial data sets is the following.

(a) *Concentric rings*: One spherical cluster and two concentric ring-shaped clusters with 81, 112, and 180 data, respectively (Figure 5A).
(b) *Simple arches*: Three simple arc-shaped clusters with 90, 182, and 331 data points, respectively (Figure 5B).
(c) *Double arches*: Five alternating double-arch-shaped clusters, with 48, 84, 239, 466, and 572 data, respectively (Figure 5C).

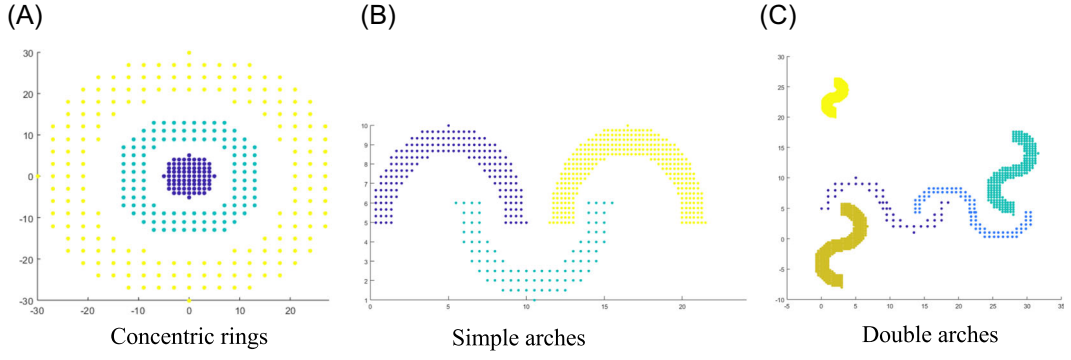

FIGURE 5 From left to right, artificial nonlinear cluster configurations: (A) concentric rings, (B) simple arches, and (C) double arches [Color figure can be viewed at wileyonlinelibrary.com]

4.3 | Real data sets

We have also worked with 17 *real data sets* extracted from the UCI Machine Learning Repository,[41] with different number of dimensions. These 17 widely used real data sets are: *Iris* (3 classes, 4 features, and 150 objects), *Breast Cancer Wisconsin* (Diagnostic) (2 classes, 30 features, and 569 objects), *Wine* (3 classes, 13 features, and 178 objects), *Vertebral Column* (3 classes, 6 features, and 310 objects), *Ecoli* (8 classes, 7 features, and 336 objects), *Haberman's Survival* (2 classes, 3 features, and 306 objects), *Breast Tissue* (6 classes, 9 features, and 106 objects), *Glass* (6 classes, 9 features, and 214 objects), *Seeds* (3 classes, 7 features, and 210 objects), *Spectf Heart* (2 classes, 44 features, and 80 objects), *Banknote Authentication* (2 classes, 4 features, and 1372 objects), *Connections Bench Sonar* (classes, 60 features, and 208 objects), *Fertility* (2 classes, 9 features, and 100 objects), *Parkinson* (2 classes, 22 features, and 195 objects), *Statlog Vehicle* (4 classes, 18 features, and 846 objects), *Yeast* (10 classes, 8 features, and 1484 objects) and finally, *Steel Plates* (7 classes, 27 features, and 1941 objects).

These sets are frequently used with supervised classification algorithms as they are labeled and the target number of clusters of each real data set is available.

5 | APPLICATION OF THE NEW CDR VALIDATION INDEX TO ARTIFICIAL AND REAL DATA SETS

In this study, the set of partitions $R = \{P^1, ..., P^m\}$ has been obtained with the k-means clustering algorithm, although any other clustering algorithm can be applied. Indeed, the hierarchical clustering algorithm with "single-linkage" has been also applied to the nonlinear data sets shown in Figure 5. The pseudocode of both algorithms can be found in Appendix A.

The k-means algorithm was selected due to its simplicity, its low computational cost in comparison to other clustering algorithms, and because it is the widest used in the internal index literature, and particularly, with the measures we are comparing the results.

The initialization technique of the k-means algorithm is k-means++, described in Reference [42]. Its goal is to guarantee that the initial seeds of the algorithm are as distant as possible, in such a way as to avoid dividing as much as possible genuine clusters. This procedure begins with a randomly chosen seed. For the remaining data the probability of being chosen as the next seed is calculated using the inverse of the distance to the original seed. This process is repeated until all the seeds have been selected.

For each data set that is going to be tested, up to 14 partitions are generated, starting with a minimum of two clusters up to a maximum of 15 clusters. The target partition is directly aggregated to the set of partitions to be evaluated by the indices replacing the one generated with the correct number of clusters. This makes the evaluation of the index capacity to detect adjacent clusters independent of the clustering algorithm. Previously, each data feature was scaled to the range of 0–100.

The new proposed index is going to be compared with other well-known validation measures. Particularly, the internal indices used in this study are: the Dunn index (Dunn), the Calinski–Harabasz index (CH), the Davies–Bouldin index (DB), the I index, the Xie and Beni index (XB), the Silohuette index (Sil), the SD index, the C index, and the CS index. Their definitions are presented in Appendix B; the detailed development of the expressions of the indices can be found in Reference [2].

All these indices are going to be compared on the same data sets. For each data set, the number of clusters selected by each internal index is registered, considering the 14 partitions previously generated.

The performance of the indices is compared by two criteria:

1. *Hits:* The number of times an index finds the correct number of clusters of the data sets.
2. *Average error*: The average of the absolute value of the difference between the number of clusters found by the index (prediction) and the target number of clusters of each data set (target). It is defined by

$$avg_{error} = \frac{\sum_{i=1}^{n} |target - prediction|}{n}, \tag{9}$$

where n is the total number of data sets analyzed.

5.1 | Experimental results with artificial data sets

The results of the experiments on the nine artificial data sets of Figure 4 are shown in Table 1. Each column represents the results of a specific internal index. The first one, CDR, is the one here proposed. The rest of the indices, as defined in,[2] are: Dunn (D), Calinski–Harabasz (CH), Davies–Bouldin (DB), the I index (I), Xie and Beni (XB), Silohuette (Sil), SD index, C index, and CS index.

The bolded values show the hits, that is, when the number of clusters predicted for the index matches the target number of clusters. The values underlined represent that the index has predicted a number of clusters that only differ in one from the correct one. The last two rows show the total number of hits (H) and the average error (E) of each index. As it can be

TABLE 1 Experimental results with artificial data sets

Set	Target	CDR	SD	XB	Sil	I	Dunn	CH	DB	C	CS
1	2	**2**	6	**2**	**2**	4	14	6	5	15	14
2	3	**3**	**3**	2	2	4	14	2	2	14	11
3	4	**4**	3	2	2	**4**	11	3	2	14	14
4	5	**5**	3	2	2	3	12	4	2	14	3
5	4	**4**	**4**	**4**	*3*	*3*	14	13	10	14	12
6	5	**5**	**5**	3	2	3	**5**	4	2	15	**5**
7	2	**2**	5	6	6	7	13	7	7	15	7
8	3	**3**	2	2	2	6	14	2	7	14	15
9	4	**4**	2	2	2	5	15	2	2	15	15
E	0.00	1.44	1.67	1.89	1.89	8.89	2.78	3.22	10.89	7.56	
H	**9**	**3**	**2**	**1**	**1**	**1**	**0**	**0**	**0**	**1**	

Abbreviations: C, C index; CDR, Contiguous Density Region; CH, Calinski–Harabasz; CS, CS index; Dunn, Dunn index; DB, Davies–Bouldin; I, the I index, SD, SD index; Sil, Silohuette; XB, Xie and Beni.

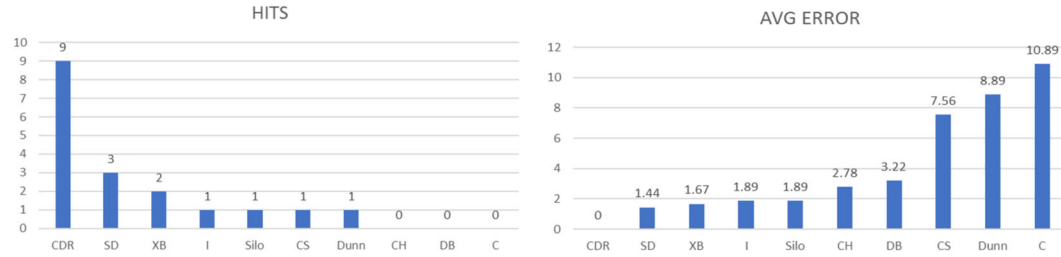

FIGURE 6 Average error rate (right); success rate (left) of the internal indices [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 2 Experimental results with nonlinear artificial data sets, *k*-means clustering algorithm

Set	Target	CDR	SD	XB	Sil	I	Dunn	CH	DB	C	CS
Rings	3	3	6	6	12	5	15	15	15	15	13
Arcs	3	3	2	2	2	6	3	15	14	15	14
Double arcs	5	5	3	3	3	3	2	15	3	3	3
E		0.00	2.00	2.00	4.00	2.33	5.00	11.33	8.33	8.67	7.67
H		3	0	0	0	0	0	0	0	0	0

Abbreviations: C, C index; CDR, Contiguous Density Region; CH, Calinski–Harabasz; CS, CS index; Dunn, Dunn index; DB, Davies–Bouldin; I, the I index, SD, SD index; Sil, Silohuette; XB, Xie and Beni.

seen, the new proposed CDR index shows a much better performance than the other indices regarding both criteria.

Indeed, whenever traditional indices perform well, the same happens with the new internal CDR index, as shown in Figure 6. However, the opposite is not always the case, which proves the greater capacity of generalization of the new approach. In this sense, this internal measure is able to capture the structure of the data even when they form clouds of data with a certain level of overlap.

5.2 | Comparison of different clustering algorithms with artificial nonlinear data sets

The three nonlinear data sets shown in Figure 5 were used to evaluate the internal indices applying both clustering algorithms, *k*-means and hierarchical. The procedure was the same described before. Partitions from 2 to 15 clusters were generated. Results are presented in Table 2, for the *k*-means algorithm, and Table 3 for the hierarchical clustering algorithm.

Figure 7 shows the average error obtained for each of the tested indices and the matches with the target number of clusters when the *k*-means algorithm is applied.

In Figure 8 the average error obtained for each of the tested indices and hits with the hierarchical clustering algorithm are presented.

As it is possible to see in Figures 7 and 8 and Tables 2 and 3, the new index finds the right partition, whatever the clustering algorithm applied, and results surpass the other measures.

TABLE 3 Experimental results with nonlinear artificial data sets, hierarchical clustering algorithm

Set	Target	CDR	SD	XB	Sil	I	Dunn	CH	DB	C	CS
Rings	3	3	4	2	2	5	2	15	2	2	15
Arcs	3	3	3	2	2	2	3	3	2	3	2
Double arcs	5	5	4	3	3	3	2	4	3	6	3
E		0.00	0.67	1.33	1.33	1.67	1.33	4.33	1.33	0.67	5.00
H		3	1	0	0	0	1	1	0	1	0

Abbreviations: C, C index; CDR, Contiguous Density Region; CH, Calinski–Harabasz; CS, CS index; Dunn, Dunn index; DB, Davies–Bouldin; I, the I index, SD, SD index; Sil, Silohuette; XB, Xie and Beni.

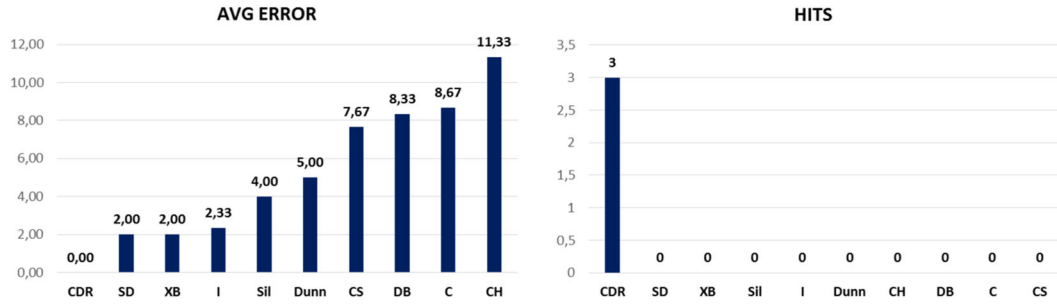

FIGURE 7 Average error rate (right); success rate (left) of the internal indices with the *k*-means clustering algorithm [Color figure can be viewed at wileyonlinelibrary.com]

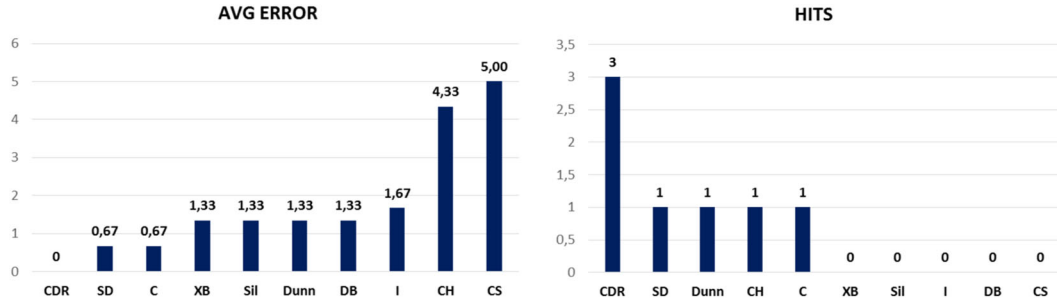

FIGURE 8 Average error rate (right); success rate (left) of the internal indices with the hierarchical clustering algorithm [Color figure can be viewed at wileyonlinelibrary.com]

5.3 | Experimental results with real data sets

The results of the experiments on the 17 real data sets are shown in Table 4. Again, the values are bolded when the result matches the target number of clusters, and underlined when the difference is only by one. The last two rows show the number of hits (*H*) and the average error (*E*). As it can be seen, the new proposed CDR index shows better performance than the other indices regarding both criteria.

TABLE 4 Experimental results with real data sets

Set	Target	CDR	SD	XB	Sil	I	D	CH	DB	C	CS
B. N.	2	**2**	5	4	15	3	**2**	5	15	15	15
Br. C.	2	**2**	3	**2**	**2**	**2**	3	**2**	**2**	15	11
Br. T.	6	2	3	2	2	2	2	2	2	11	11
V. Co.	3	*2*	5	2	2	10	15	2	13	15	13
C.B.S.	2	*3*	4	3	3	3	9	3	15	15	15
Ecoli	8	3	4	3	3	3	3	6	3	12	3
Fert.	2	**2**	12	12	15	4	4	4	12	15	12
Glass	6	2	*7*	2	2	2	8	2	15	14	15
H. S.	2	**2**	6	11	11	3	3	4	11	15	11
Iris	3	2	2	2	2	2	2	2	2	2	2
Park.	2	3	9	3	3	4	14	3	3	14	12
Seeds	3	4	2	2	2	2	11	2	2	12	2
Heart	2	**2**	15	15	13	4	12	4	13	14	12
Vehicle	4	**2**	2	2	2	2	10	2	2	15	10
Plates	7	**5**	10	2	2	2	2	2	10	12	13
Wine	3	**3**	**3**	2	2	2	12	2	**3**	12	**3**
Yeast	10	2	3	3	2	6	11	2	11	14	11
E		1.77	3.77	3.94	4.71	2.53	5.06	2.35	5.47	9.24	6.94
H		**6**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**

Abbreviations: C, C index; CDR, Contiguous Density Region; CH, Calinski–Harabasz; CS, CS index; D, Dunn; DB, Davies–Bouldin; I, the I index, SD, SD index; Sil, Silohuette; XB, Xie and Beni.

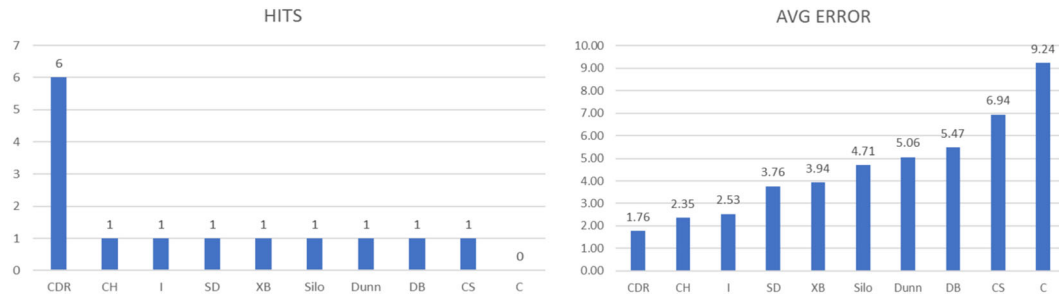

FIGURE 9 Average error rate (right); success rate (left) of the internal indices [Color figure can be viewed at wileyonlinelibrary.com]

Real data sets are more complex, data configurations are more varied, and also clusters are more different in shape and size. That is why in this case, as expected, the error is not zero in any case, neither with the CDR index (Figure 9). However, the number of hits of the proposed index is 6 out of 17, although in five of the wrong cases it was very close to the ideal value (±1).

This leads to the conclusion that in 11 out of 17 data sets (65% of the tests) the CDR index had a correct performance or very close to it, which is also confirmed by its low average error in comparison to the rest of the internal indices.

Analyzing the overall performance of the indices some conclusions can be drawn from Figures 6 to 9, where the results of the different indices are sorted according to the performance on artificial and real data sets.

The C index gave the worst results, no hits and large average error, both on artificial and real data sets. This may be due to the fact that this index is based on the sum of the Euclidean distances between all the pairs of data of the respective clusters. The total number of data pairs tends to decrease as the number of clusters increases. Since this index must be minimized, it always tends to favor partitions with a high number of clusters.

The CS index also gave bad results on both series of tests. This index calculates the diameter of the clusters as the average of the furthest neighbor. Therefore, the most distant data from the cluster centroid will have a strong influence on the final value. As this value appears in the numerator of the index to be minimized, again it tends to boots partitions with a high number of clusters.

The Dunn index occupies the last ranking positions as well. It calculates the cluster diameter (compactness) as the largest distance between all the data pairs of the cluster, and the distance between clusters (separation) as the smallest distance between a pair of data from both clusters. The diameter decreases faster than the intercluster distance while more clusters are added to the partition. Therefore, the ratio between those measures tends to find a higher number of clusters.

To summarize, the new proposed CDR index performs better than the other traditional internal validation measures it has been compared with. It is able to capture the local density of the data sets, showing a generalization capability when dealing with arbitrary-shape clusters.

In addition, these results also show that density could be used as a key element to differentiate clusters. Indeed, when this characteristic is included in an internal clustering validation index it helps measure the quality of the obtained partitions.

5.4 | Computational cost analysis

In this subsection, a comparative analysis of the performance in terms of computational time of the proposed index in relation to the other internal measures has been carried out. The CPU time used by each index in completing a series of experiments was obtained.

It has been evaluated on the artificial data set Linear 5C (Figure 4D), using the *k*-means clustering algorithm that is mostly used by the other indices. Partitions from 2 to 15 clusters were generated and used. For each index, the time required to select a partition as the optimal one was recorded. The results are presented in Table 5 and Figure 10. The values are ordered from lowest to highest CPU time (seconds).

It is possible to see that the proposed index requires more time than some of the other internal measures, but still not too high. It takes around 4 s with a PC Intel Core i5-3210M 2.50 GHz Processor with 8 GB RAM.

This may be due to the fact the computation of the distance among all points in a cluster is required. However, there are some proposals that address this problem to reduce the computational load, both from the algorithm optimization point of view and from its implementation.[43] Besides, the computational time is not critical in this application.

TABLE 5 Computational time for different indices

Index	Time
CH	0.2031
XB	0.3281
SD	0.3906
I	0.5625
DB	0.5781
Sil	1.2031
CS	2.2969
CDR	4.1719
C	4.5156
Dunn	7.1719

Abbreviations: C, C index; CDR, Contiguous Density Region; CH, Calinski–Harabasz; CS, CS index; Dunn, Dunn index; DB, Davies–Bouldin; I, the I index, SD, SD index; Sil, Silohuette; XB, Xie and Beni.

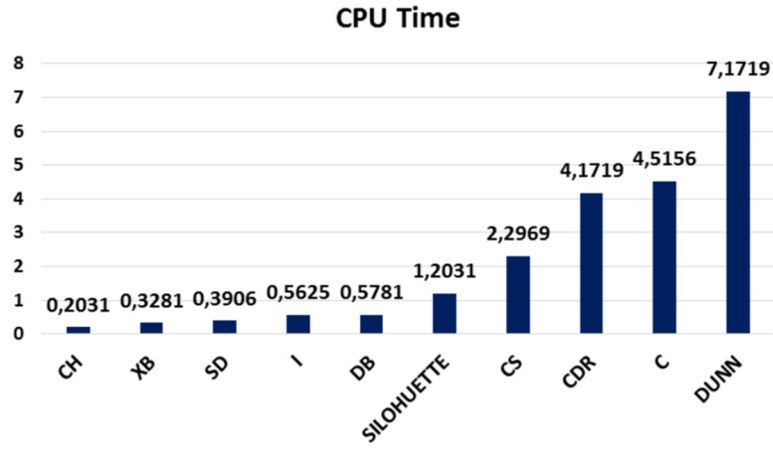

FIGURE 10 CPU time for each index, from the lowest (left) to the highest (right) [Color figure can be viewed at wileyonlinelibrary.com]

6 | CONCLUSIONS AND FUTURE WORKS

A new clustering validation measure that captures the spatial pattern of a data set has been defined. The CDR index is based on the uniformity, that has been defined by combining the local density of the cluster and the variation of this density.

The new internal validation CDR measure has been proved to work well with arbitrary-shape clusters that can be adjacent, and even overlapped. It has been applied to artificial and real data sets, with different sizes and shapes, and compared with the results obtained by other traditional validation indices.

In all the cases a better performance has been obtained, in terms of higher matches with the right number of classes of the target partition and smaller average error.

This proposal also proves that including the concept of density in the definition of the internal validation index helps measure the quality of a partition more accurately.

As future works, it would be interesting to define new internal evaluation indices that take into account other features of the real data. At least two new challenges can be addressed: first, the definition of new indices based on the proposed paradigm to recognize nonadjacent clusters that present similar density variability; second, to apply this paradigm to define new clustering algorithms. Moreover, it might be worthy to extend the index to sets with different data types or even to work with fuzzy clustering.

FUNDING INFORMATION

This study did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

DATA AVAILABILITY STATEMENT

Artificial sets are available on request. Real data sets are available in the cited public repository.

ORCID

Juan Carlos Rojas-Thomas http://orcid.org/0000-0002-3654-7900
Matilde Santos http://orcid.org/0000-0003-1993-8368

REFERENCES

1. Zhou R, Zhang Y, Feng S, Luktarhan N. A novel hierarchical clustering algorithm based on density peaks for complex datasets. *Complexity*. 2018;2018:1-8. https://doi.org/10.1155/2018/2032461
2. Arbelaitz O, Gurrutxaga I, Muguerza J, Pérez JM, Perona I. An extensive comparative study of cluster validity indices. *Pattern Recognition*. 2013;46(1):243-256.
3. Wu W, Xu Z, Kou G, Shi Y. Decision-making support for the evaluation of clustering algorithms based on MCDM. *Complexity*. 2020;2020:1-17. https://doi.org/10.1155/2020/9602526
4. Halkidi M, Batistakis Y, Vazirgiannis M. On clustering validation techniques. *J Intell Inf Syst*. 2001;17(2):107-145.
5. Kim M, Ramakrishna RS. New indices for cluster validity assessment. *Pattern Recognition Lett*. 2005;26(15):2353-2363.
6. Rodríguez-Ruiz J, Monroy R, Medina-Pérez MA, Loyola-González O, Cervantes B. Cluster validation in clustering-based one-class classification. *Expert Syst*. 2019;36(6):e12475.
7. Luna-Romera JM, Martínez-Ballesteros M, García-Gutiérrez J, Riquelme JC. External clustering validity index based on chi-squared statistical test. *Inf Sci*. 2019;487:1-17.
8. Deborah LJ, Baskaran R, Kannan A. A survey on internal validity measure for cluster validation. *Int J Comput Sci Eng*. 2010;1(2):85-102.
9. Moulavi D, Jaskowiak PA, Campello RJ, Zimek A, Sander J. Density-based clustering validation. In: *Proceedings of the 14th SIAM International Conference on Data Mining (SDM)*. Philadelphia, PA: Society for Industrial and Applied Mathematics; 2014:839-847.
10. Žalik KR, Žalik B. Validity index for clusters of different sizes and densities. *Pattern Recognition Lett*. 2011;32(2):221-234.
11. Lee SH, Jeong YS, Kim JY, Jeong MK. A new clustering validity index for arbitrary shape of clusters. *Pattern Recognition Lett*. 2018;112:263-269.
12. Yera A, Arbelaitz O, Jodra JL, Gurrutxaga I, Pérez JM, Muguerza J. Analysis of several decision fusion strategies for clustering validation. Strategy definition, experiments and validation. *Pattern Recognition Lett*. 2017;85:42-48.
13. Rojas-Thomas JC, Mora M, Santos M. Neural networks ensemble for automatic DNA microarray spot classification. *Neural Comput Appl*. 2019;31:2311-2327. https://doi.org/10.1007/s00521-017-3190-6
14. Jain AK. Data clustering: 50 years beyond K-means. *Pattern Recognition Lett*. 2010;31(8):651-666.

15. Xu D, Tian Y. A comprehensive survey of clustering algorithms. *Ann Data Sci.* 2015;2(2):165-193.
16. Ester M, Kriegel HP, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*. Portland, OR, USA: AAAI Press; 1996;96(34):226-231.
17. Bryant A, Cios K. RNN-DBSCAN: A density-based clustering algorithm using reverse nearest neighbor density estimates. *IEEE Trans Knowl Data Eng.* 2018;30(6):1109-1121.
18. Xu Q, Zhang Q, Liu J, Luo B. Efficient synthetical clustering validity indexes for hierarchical clustering. *Expert Syst Appl.* 2020;151:1-13. https://doi.org/10.1016/j.eswa.2020.113367
19. Hinneburg A, Gabriel HH. DENCLUE 2.0: Fast clustering based on kernel density estimation. In: Berthold MR, Shawe-Taylor J, Lavrac N, eds. *Advances in Intelligent Data Analysis VII. IDA 2007. Lecture Notes in Computer Science*. Berlin, Heidelberg: Springer; 2007:4723. https://doi.org/10.1007/978-3-540-74825-0_7
20. Ilango MR, Mohan V. A survey of grid based clustering algorithms. *Int J Eng Sci*. 2010;2(8):3441-3446.
21. Schikuta E. Grid-clustering: An efficient hierarchical clustering method for very large data sets. In: *Proceedings of the 13th International Conference on Pattern Recognition, IEEE Computer Society, Washington, DC, USA, ICPR '96*. IEEE; 1996;2:101-105.
22. Hinneburg A, Keim DA. Optimal grid-clustering: Towards breaking the curse of dimensionality in high-dimensional clustering. In: *International Conference on Very Large Data Bases*. Morgan Kaufmann Publishers; 1999:506-517.
23. Zhu E, Zhang Y, Wen P, Liu F. Fast and stable clustering analysis based on grid-mapping K-means algorithm and new clustering validity index. *Neurocomputing*. 2019;363:149-170.
24. Agrawal KP, Garg S, Patel P. Performance measures for densed and arbitrary shaped clusters. *Int J Comput Sci Commun.* 2015;6(2):338-350.
25. Chou CH, Su MC, Lai E. A new cluster validity measure and its application to image compression. *Pattern Anal Appl.* 2004;7(2):205-220.
26. Rodríguez J, Medina-Pérez MA, Gutierrez-Rodríguez AE, Monroy R, Terashima-Marín H. Cluster validation using an ensemble of supervised classifiers. *Knowl-Based Syst.* 2018;145:134-144.
27. Rojas-Thomas JC, Santos M, Mora M. New internal index for clustering validation based on graphs. *Expert Syst Appl.* 2017;86:334-349.
28. Zhou S, Xu Z. A novel internal validity index based on the cluster centre and the nearest neighbour cluster. *Appl Soft Comput.* 2018;71:78-88.
29. Rojas-Thomas JC, Santos M, Mora M, Duro N. Performance analysis of clustering internal validation indices with asymmetric clusters. *IEEE Lat Am Trans.* 2019;17(5):807-814.
30. Nerurkar P, Pavate A, Shah M, Jacob S. Performance of internal cluster validations measures for evolutionary clustering. In: Iyer B, Nalbalwar SL, Pathak NP, eds. *Computing, Communication and Signal Processing, Advances in Intelligent Systems and Computing*. Singapore: Springer; 2019:305-312. https://doi.org/10.1007/978-981-13-1513-8_32
31. Reséndiz JDH, Castro HMM, Leal ET. A comparative study of clustering validation indices and maximum entropy for sintonization of automatic segmentation techniques. *IEEE Lat Am Trans.* 2019;17(08):1229-1236.
32. Hu L, Zhong C. An internal validity index based on density-involved distance. *IEEE Access*. 2019;7:40038-40051.
33. Cheng D, Zhu Q, Huang J, Wu Q, Yang L. A novel cluster validity index based on local cores. *IEEE Trans Neural Networks Learn Syst.* 2018;30(4):985-999.
34. Cheng D, Zhu Q, Huang J, Wu Q, Lijun Y. Clustering with local density peaks-based minimum spanning tree. *IEEE Trans Knowl Data Eng.* 2021;33(2):374-387. https://doi.org/10.1109/TKDE.2019.2930056
35. Xie J, Xiong ZY, Dai QZ, Wang XX, Zhang YF. A new internal index based on density core for clustering validation. *Inf Sci.* 2020;506:346-365.
36. Thomas JCR, Peñas MS, Mora M. New version of Davies–Bouldin index for clustering validation based on cylindrical distance. In: *32nd International Conference of the Chilean Computer Science Society (SCCC)*. Temuco, Chile: IEEE; 2013:49-53. https://doi.org/10.1109/SCCC.2013.29
37. Zahn CT. Graph-theoretical methods for detecting and describing gestalt clusters. *IEEE Trans Comput*. 1971;100(1):68-86.

38. Zhong C, Miao D, Wang R. A graph-theoretical clustering method based on two rounds of minimum spanning trees. *Pattern Recognition*. 2010;43(3):752-766.
39. Naranjo R, Santos M, Garmendia L. A convolution-based distance measure for fuzzy singletons and its application in a pattern recognition problem. *Integr Comput-Aided Eng*. 2021;28(1):51-63.
40. León F, Rodríguez-Lozano FJ, Cubero-Fernández A, Palomares JM, Olivares J. SysGpr: Sistema de generación de señales sintéticas pseudo-realistas. *Rev Iberoam Autom Inf Ind*. 2019;16(3):369-379.
41. Asuncion A, Newman D. *UC Irvine Machine Learning Repository*. 2007. URL http://www.ics.uci.edu/~mlearn/MLRepository.html
42. Arthur D, Vassilvitskii S. K-means++: The advantages of careful seeding. In: *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. Tokyo: Society for Industrial and Applied Mathematics 2007 ACM-SIAM (SODA'07); 2007:1027-1035.
43. Garcia V, Debreuve E, Barlaud M. Fast k nearest neighbor search using GPU. In: *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops, Anchorage, AK, USA*. IEEE; 2008:1-6. https://doi.org/10.1109/CVPRW.2008.4563100

How to cite this article: Rojas-Thomas JC, Santos M. New internal clustering validation measure for contiguous arbitrary-shape clusters. *Int J Intell Syst*. 2021;36:5506-5529. https://doi.org/10.1002/int.22521

APPENDIX A

A.1 | *K*-means clustering algorithm pseudocode

K-Means clustering

1. Choose the number of clusters (*K*) and obtain the data points
2. Place the centroids *c_1*, *c_2*, ..., *c_k* randomly
3. Repeat Steps 4 and 5 until convergence or until the end of a fixed number of iterations
4. For each data point *x_i*:
   - find the nearest centroid (*c_1*, *c_2*, ..., *c_k*)
   - assign the point to that cluster
5. For each cluster $j = 1, ..., k$
   - new centroid = mean of all points assigned to that cluster
6. End

If the number of clusters is not known, the elbow method (based on the within-cluster distance to the centroid) can be applied to obtain it.

A.2 | Hierarchical clustering algorithm pseudocode

Hierarchical clustering

1. Begin with *n* clusters, each containing one object, number the clusters 1, ..., *n*.

(Continues)

2. Between-cluster distance $D(r, s)$ = between-object distance of the two objects in r and s, respectively, r, s = 1, 2, ..., n.
3. Let the square matrix $D = (D(r, s))$. If the objects are represented by quantitative vectors we can use Euclidean distance.
4. Find the most similar pair of clusters r and s, such that the distance, $D(r, s)$, is minimum among all the pairwise distances.
5. Merge r and s to a new cluster t and compute the between-cluster distance $D(t, k)$ for any existing cluster $k \neq r, s$.
6. Once the distances are obtained, delete the rows and columns corresponding to the old cluster r and s in the D matrix, because r and s do not exist anymore.
7. Add a new row and column in D corresponding to cluster t.
8. Repeat Step 4 a total of $n - 1$ times until there is only one cluster left.
9. End

APPENDIX B

B.1 | Internal indices definitions

In the table below the nine internal indices used for comparison purposes are defined. The definition is based on compactness and separation. The notation used in the formula of these measures is as follows.[29] D is the input data set, n is the number of points in D, g is the center of whole data set D, P is the number of dimensions of D, k is the number of clusters, C_i is the ith cluster, n_i is the number of data points in C_i, c_i is the center of cluster C_i, (C_i) is the variance vector of C_i, and $d(x, y)$ is the distance between points x and y. The next equation defines the total pairs of data within the clusters:

$$n_W = \sum_{C_i \in C} \binom{n_i}{2} = \sum_{i=1}^{k} \frac{n_i(n_i - 1)}{2}.$$

Index	Compactness	Separation	Definition
CH	$SSW = \sum_{i=1}^{M}\sum_{j=1}^{n_i} \| x_j - c_i \|^2$	$SSB = \sum_{i=1}^{M} n_i \, \|c_i - g\|^2$	$CH = \frac{SSB/\left(k-1\right)}{SSW/\left(N-k\right)}$
I	$E_i = \sum_{i=1}^{k}\sum_{j=1}^{N} \|x_i - c_i\|^2,$	$D_i = max_{i,j} \|c_i - c_j\|$	$I\left(k\right) = \left(\frac{1}{k} \times \frac{E_1}{E_i} \times D_i\right)^p$
	$E_1 = \sum_{1}^{N} \|x_i - g\|^2$		
Dunn	$diam\left(C_i\right) = max_{x,y \in C_i} d\left(x, y\right)$	$dist\left(C_i, C_j\right) = min_{x \in C_i, y \in C_j} d\left(x, y\right)$	$D = \frac{min\left(dist\left(C_i, C_j\right)\right)}{max\left(diam\left(C_i\right)\right)}$
DB	$disp(C_i) = \frac{1}{n_i} \sum_{x \in C_i} d(x, c_i)$	$dist\left(C_i, C_j\right) = d(c_i, c_j)$	$DB = \frac{1}{k} \sum_{i} max_{j \neq i} \frac{disp(C_i) + disp(C_j)}{dist\left(C_i, C_j\right)}$
SD	$Scat\left(k\right) = \frac{1}{k} \sum \frac{\| \sigma(c_i) \|}{\| \sigma(D) \|}$	$Dis\left(k\right) = \frac{max\, d(c_i, c_j)}{min\, d(c_i, c_j)} \sum (d(c_i, c_j))^{-1}$	$SD\left(k\right) = Dis\left(k_{max}\right) \bullet Scat\left(k\right) + Dis\left(k\right)$
XB	$Comp\left(C_i\right) = \sum_{x \in C_i} d^2(x, c_i)$	$dist\left(C_i, C_j\right) = d^2(c_i, c_j)$	$XB = \frac{\sum_{i=1}^{k} Comp(C_i)}{n \cdot min_{i,j} \cdot dist(C_i, C_j)}$
Sil	$a(x) = \frac{\sum_{y \in C_i,\, x \neq y} d(x,y)}{n_i - 1}$	$b(x) = min_{j \neq i} \frac{1}{nj} \sum_{y \in Cj} d\left(x, y\right)$	$S = \frac{1}{N}\left(\sum_{C_i \in C}\left(\sum_{x \in C_i} \frac{b\left(x\right) - a\left(x\right)}{max\left[b\left(x\right), a(x)\right]}\right)\right)$

(Continues)

Index	Compactness	Separation	Definition
C	$S(C) = \sum_{C_l \in C} \sum_{x_i x_j \in C_l} d_e(x_i, x_j)$	$S_{min}(C) = \sum min(n_W)_{x_i x_j \in D}\{d_e(x_i, x_j)\}$	$CI\left(C\right) = \frac{S\left(C\right) - S_{min}\left(C\right)}{S_{max}\left(C\right) - S_{min}\left(C\right)}$
	$S_{max}(C) = \sum max(n_W)_{x_i x_j \in D}\{d_e(x_i, x_j)\}$		
CS	$diam(C_k) = \frac{1}{\lvert C_k \rvert}$	$separation(C_k) = min_{C_l \in C, l \neq i}\{d(\bar{C}_k, \bar{C}_l)\}$	$CS\left(C\right) = \frac{\sum_{C_k \in C} diam(C_k)}{\sum_{C_k \in C} separation(C_k)}$
	$\sum_{x_i \in C_k} max_{x_j \in C_k}\{d_e(x_i, x_j)\}$		

Abbreviations: C, C index; CDR, Contiguous Density Region; CH, Calinski–Harabasz; CS, CS index; Dunn, Dunn index; DB, Davies–Bouldin; I, the I index; SD, SD index; Sil, Silohuette; XB, Xie and Beni.