Saitta, S., Raphael, B. and Smith, I.F.C. "A comprehensive validity index for clustering", Intelligent Data Analysis, vol. 12, no 6, 2008, pp. 529-548 http://content.iospress.com/articles/intelligent-data-analysis/ida00346

A Comprehensive Validity Index for Clustering

S. Saitta* B. Raphael† I.F.C. Smith‡

Abstract

Cluster validity indices are used for both estimating the quality of a clustering algorithm and for determining the correct number of clusters in data. Even though several indices exist in the literature, most of them are only relevant for data sets that contain at least two clusters. This paper introduces a new bounded index for cluster validity called the score function (SF), a double exponential expression that is based on a ratio of standard cluster parameters. Several artificial and real-life data sets are used to evaluate the performance of the score function. These data sets contain a range of features and patterns such as unbalanced, overlapped and noisy clusters. In addition, cases involving sub-clusters and perfect clusters are tested. The score function is tested against six previously proposed validity indices. In the case of hyper-spheroidal clusters, the index proposed in this paper is found to be always as good or better than these indices. In addition, it is shown to work well on multidimensional and noisy data sets. One of its advantages is the ability to handle single cluster case and sub-cluster hierarchies.

Keywords: clustering, validity index, number of clusters, K-means.

1 Introduction

One of the best known examples of unsupervised learning is clustering [23, 42, 46]. The goal of clustering is to group data points that are similar according to a chosen similarity metric (Euclidean distance is commonly used). Clustering can also be used in combination with other techniques such as genetic algorithms [30]. Clustering techniques have been applied in domains such as text mining [41], intrusion detection [35], DNA micro-arrays [15] and information exploration [21]. In these fields, as in many others, the number of clusters is usually not known in advance.

Clustering techniques that are proposed in the literature, although considerable [25], can be divided into four main categories [17]: partitional clustering (for example, K-means), hierarchical clustering (for example, BIRCH), density-based clustering (for example, DBSCAN) and grid-based clustering (for example, STING). Although the mixture of Gaussian approach can be mentioned, its computational complexity is too high to be used in practice. Clustering is known as a form of unsupervised learning, as well as numerical taxonomy and partitioning [43].

*Corresponding author: Graduated Research Assistant, Ecole Polytechnique Fédérale de Lausanne (EPFL), Station 18, Lausanne, Switzerland, Phone: ++41 (0)21 693 63 72, Fax: ++41 (0)21 693 47 48, Email: sandro.saitta@gmail.com

†Assistant Professor, National University of Singapore, 117566, Singapore. E-Mail: bdgbr@nus.edu.sg

‡Professor, Ecole Polytechnique Fédérale de Lausanne (EPFL), Station 18, Lausanne, Switzerland. E-Mail: ian.smith@epfl.ch.

One of the most popular techniques for clustering is K-means [23]. Reasons for the popularity of this technique include the absence of drawbacks of other types [17]. For example, hierarchical clustering has a higher complexity. Density-based clustering algorithms often require tuning non-intuitive parameters. Finally, density-based clustering algorithms do not always give clusters of good quality. Advantages of K-means include computational efficiency and easy interpretation of results. K-means is certainly the most widely used clustering algorithm in practice [1].
The drawbacks of K-means include, random choice of centroid locations at the start of the algorithm, treatment of variables as numbers and the unknown number of clusters k. The first can be handled through multiple runs. The paper by [22] contains a possible solution to the second through the use of a matching dissimilarity measure to handle categorical parameters. Concerning the third point, the number of clusters is an input parameter that is fixed *a priori* in the standard K-means algorithm. As many other data mining algorithms, K-means has reduced reliability when treating high-dimensional data because data sets are nearly always too sparse. This is due to the use of the Euclidean distance, that becomes meaningless in high-dimensional spaces [14]. A possible solution involves combining K-means with feature extraction methods such as principal component analysis (PCA) [9] and self-organizing maps (SOM) [44].
When performing clustering tasks, results should be treated with caution. Indeed, as noted in [24], clustering is a difficult subjective task. An impossibility theorem for clustering has even been proposed. In [29] it is shown that there is no clustering function that satisfies a set of three properties. However, this theorem can be relaxed for real-life usage of clustering algorithms. As written in [36], the two issues in clustering are i) determination of the number of clusters present in the data and ii) evaluating how good is the clustering itself. These two issues motivate research in the field of cluster validation. Validity indices are also useful for estimating the quality of clusters. An example is given in [12].
Other important challenges in clustering are fixing initial conditions [40] and treating high dimensional data sets [33]. Many cluster validation techniques are available [2, 13, 17, 18, 19]. This evaluation can be used to determine the most reliable number of clusters in a data set. Several indices have been proposed in the literature [2, 17, 27, 45, 47]. These indices were evaluated through plotting them to determine the number of clusters visually. Most of them have been compared with known results [5, 27]. Selected validity indices are briefly described below.
The Hubert statistic assesses how well the data fit a proposed crisp structure. The concept behind the Hubert statistic is the correlation measure. Since calculation of the original index is computationally expensive, a modified index was proposed. In the modified Hubert statistic [43], a *knee* on the plot indicates a possible value for the number of clusters. Finding this knee is somewhat subjective. The Dunn index [10] combines dissimilarity between clusters and their diameters to estimate the most reliable number of clusters. The Dunn index is computationally expensive ($O(n^2)$) and sensitive to noise [17]. An index based on a ratio of between and within scatter cluster matrices is proposed by [6]. The concepts of dispersion of a cluster and dissimilarity between clusters are used to compute the Davies-Bouldin index [8] which has recently been reported to be among the best [27]. The Silhouette index [26] uses average dissimilarity between points to show the structure of the data and consequently, its possible clusters. As stated in [3], the Silhouette index is only suitable for estimating the first choice or the best partition. The index proposed by [20] is based on average scattering for clusters and total separation between clusters. This index has to be tuned with a parameter that may vary the clustering results for small number of clusters. The Maulik-Bandyopadhyay index [36] is related to the Dunn index

and involves the tuning of a parameter. Finally, the Geometric index [32] has been developed for handling clusters of different densities and close clusters as well. A particular feature of this index is the use of the eigen-axes lengths as a way of measuring the intra-cluster distance.
All of these indices require the specification of at least two clusters. Although not often studied by the data mining community, the one cluster case is important as pointed out by [16] and is likely to happen in practice. More details about single cluster tests can be found in [16]. Several other validity indices exist in the literature [31, 32, 39]. Some are computationally expensive (i.e. more than $O(n)$) [17] while others are unable to discover the real number of clusters in all data sets [27]. This paper proposes a new validity index that helps overcome such limitations.
This article is organized as follows. Section 2 describes existing work in the domain of cluster validity indices. Section 3 proposes a new validity index and explains mathematical development behind its conception. Performance of the index is described in Section 4. Section 5 describes the known limitations of the proposed index. The last Section provides conclusions and directions for future work.

2 Related Work

Since it is not feasible to test every existing index, six validity indices that are suitable for hard partitional clustering are used to compare results with those of the new validity index. These indices serve as a basis for evaluating results from the proposed index on benchmark data sets. Notation for these indices have been adapted to provide a coherent basis. The metric used on the normalized data set is the Euclidean distance $d(x, y)$. The Euclidean distance is chosen since it is easily understood by non-specialists.
Dunn index: One of the oldest and most cited indices is proposed by [10]. The Dunn index (DU) identifies clusters which are well separated and compact. The goal is therefore to maximize the inter-cluster distance while minimizing the intra-cluster distance. The Dunn index for k clusters is defined by Equation 1:

$$DU_k = \min_{i=1,\ldots,k} \left\{ \min_{j=1+1,\ldots,k} \left(\frac{diss(c_i, c_j)}{\max_{m=1,\ldots,k} diam(c_m)} \right) \right\} \tag{1}$$

where $diss(c_i, c_j) = \min_{x \in c_i, y \in c_j} d(x, y)$ is the dissimilarity between clusters c_i and c_j and $diam(C) = \max_{x,y \in C} d(x, y)$ is the intra-cluster function (or diameter) of the cluster. If Dunn index is large, it means that compact and well separated clusters exist. Therefore, the maximum is observed for k equal to the most probable number of clusters in the data set.
Calinski-Harabasz index: This index [6] is based on a ratio of between cluster scatter matrix ($BCSM$) and within cluster scatter matrix ($WCSM$). The Calinski-Harabasz index (CH) is defined as follows:

$$CH_k = \frac{BCSM}{k-1} \cdot \frac{n-k}{WCSM} \tag{2}$$

where n is the total number of points and k the number of clusters. The $BCSM$ is based on the distance between clusters and is defined in Equation 3:

$$BCSM = \sum_{i=1}^{k} n_i \cdot d(z_i, z_{tot})^2 \tag{3}$$

where z_i is the center of cluster c_i and n_i, the number of points in c_i. The $WCSM$ is given in Equation 4:

$$WCSM = \sum_{i=1}^{k} \sum_{x \in c_i} d(x, z_i)^2 \tag{4}$$

where x is a data point belonging to cluster c_i. To obtain well separated and compact clusters, $BCSM$ is maximized and $WCSM$ minimized. Therefore, the maximum value for CH indicates a suitable partition for the data set.

Davies-Bouldin index: Similar to the Dunn index, Davies-Bouldin index [8] identifies clusters which are far from each other and compact. The Davies-Bouldin index (DB) is defined according to Equation 5:

$$DB_k = \frac{1}{k} \sum_{i=1}^{k} \max_{j=1,\ldots,k, i \neq j} \left\{ \frac{diam(c_i) + diam(c_j)}{d(z_i, z_j)} \right\} \tag{5}$$

where in this case, the diameter of a cluster is defined as in Equation 6:

$$diam(c_i) = \sqrt{\frac{1}{n_i} \sum_{x \in c_i} d(x, z_i)^2} \tag{6}$$

with n_i the number of points and z_i the centroid of cluster c_i. Since the objective is to obtain clusters with minimum intra-cluster distances, small values for DB are interesting. Therefore, this index is minimized when looking for the best number of clusters.

Silhouette index: The silhouette statistic [26] is another well known way of estimating the number of groups in a data set. The Silhouette index (SI) computes for each point a width depending on its membership in any cluster. This silhouette width is then an average over all observations. This leads to Equation 7:

$$SI_k = \frac{1}{n} \sum_{i=1}^{n} \frac{(b_i - a_i)}{\max(a_i, b_i)} \tag{7}$$

where n is the total number of points, a_i is the average distance between point i and all other points in its own cluster and b_i is the minimum of the average dissimilarities between i and points in other clusters. Finally, the partition with the highest SI is taken to be optimal.

Maulik-Bandyopadhyay index: A more recently developed index is named the I index [36]. For consistency with other indices it is renamed MB. This index, which is a combination of three terms, is given through Equation 8:

$$MB_k = \left(\frac{1}{k} \cdot \frac{E_1}{E_k} \cdot D_k \right)^p \tag{8}$$

where the intra-cluster distance is defined by $E_k = \sum_{i=1}^{k} \sum_{x \in c_i} d(x, z_i)$, E_1 being the value of E_k for $k = 1$ and the inter-cluster distance by $D_k = \max_{i,j=1}^{k} d(z_i, z_j)$. As before, z_i is the center of cluster c_i. The correct number of clusters is estimated by maximizing Equation 8. According to [36], p is chosen to be two.

Geometric index: The last index used for comparison is the Geometric index [32]. One of its advantages is its ability to accommodate data with clusters of different densities as well as clusters that overlap. The Geometric index (GE) is defined by Equation 9:

$$GE_k = \max_{1 \le r \le k} \frac{\left(2 \sum_{j=1}^{d} \sqrt{\lambda_{jr}}\right)^2}{\min_{1 \le q \le k, r \neq q} d(z_r, z_q)} \tag{9}$$

where d is the dimensionality of the data and λ_{jr} is the eigenvalue of the covariance matrix from the data. While the numerator is the squared eigen-axis length, the denominator represents the inter-cluster distance. The optimal solution is found by minimizing the index over the number of clusters.

3 A Bounded Validity Index

A typical goal of clustering is to maximize the inter-cluster distance (separability) while minimizing the intra-cluster distance (compactness). The index developed in this work - called a score function (SF) - is based on these two concepts. This section gives details related to the way the SF has been developed and the ideas that have lead to its development. The following definitions are used. Firstly, the Euclidean distance is used to measure to what degree two data points are separated. Secondly, the size of the i-th cluster, n_i, is given by the number of points it contains.

Two concepts used in the proposed index are the "between class distance" (*bcd*), representing the separability of clusters, and the "within class distance" (*wcd*) representing the compactness of clusters. Three approaches are commonly used to measure the distance between two clusters: single linkage, complete linkage and comparison of centroids. DU is based on single linkage and has a complexity of $O(n^2)$. Although SI does not fit well into these three categories, its computational complexity is the same as the first two. DB, MB and GE compare centroids. CH follows the third approach since the distances of centroids from the overall mean of the data are determined. The main advantage of using the distance from the overall mean of the data is that the minimum and maximum are not used when comparing centroids. The minimum and maximum are sensitive to outliers. In this work, the score function uses the third approach since the first two have high computational costs [17]. The *bcd* is given by Equation 10:

$$bcd = \frac{1}{nk} \sum_{i=1}^{k} d(z_i, z_{tot})^2 \cdot n_i \tag{10}$$

where n is the total number of data points, k is the number of clusters, z_i its centroid of the current cluster and z_{tot} the centroid of all the data points. The main quantity in the *bcd* is the distance *d()* between z_i and z_{tot}. As in the CH index, each distance is weighted by the cluster size n_i to limit the influence of outliers. This has the effect to reduce the sensitivity to noise. Like all other tested indices, n is used to avoid the sensitivity of *bcd* to the total number of points. Finally, the value of k in the denominator is used to penalize the addition of new clusters. Thus, *bcd* is reduced as k increases. In this way, the limit of one point per cluster is avoided. The *wcd* is given by Equation 11:

$$wcd = \frac{1}{k} \sum_{i=1}^{k} \sqrt{\frac{1}{n_i} \sum_{x \in c_i} d(x, z_i)^2} \tag{11}$$

Computing values for *wcd* involves determining the distance *d()* between each point and the centroid of its cluster. n_i is again used for taking into account the size of clusters. The mean is taken over the k clusters. A graphical representation of distances used in both Equation 10 and 11 can be found in Figure 1.
With Equations 10 and 11, *bcd* and *wcd* are independent of the number of data points. The main idea, as stated in the beginning of this section, is to maximize Equation 10 while minimizing Equation 11. Therefore, compact and well separated clusters are aimed. This can be done by maximizing the ratio of *bcd* and *wcd* as shown in Equation 12:

$$\frac{bcd}{wcd} \tag{12}$$

Equation 12 has two difficulties. The first difficulty occurs when the clusters are perfect. Here, Equation 11 is zero and the ratio of Equation 12 is indeterminate. Therefore, the ratio cannot be used in this form in the case of perfect clusters. The second difficulty occurs when there is only one cluster in the data. In this case, Equation 10 is zero and thus the ratio of Equation 12 is zero. This is not desirable since it means that the one cluster case is not comparable with other cases. A possible solution to these difficulties involves the use of the exponential notation. Consequently, the function given in Equation 13 is proposed:

$$\frac{e^{bcd}}{e^{wcd}} = e^{bcd-wcd} \tag{13}$$

A third difficulty is related to bounds. All other tested indices have no bounds. It is thus difficult to appreciate the results of such indices. Since the "distance" to either perfect clusters or no cluster at all is not known. The upper bound allows the examination of how close the current clusters are to the perfect cluster case. The bounds for Equation 13 are $]0, \infty[$. It is also desirable to avoid very large numbers for computational reasons. Again, exponential notation is used. Avoiding all of these difficulties leads to the formula for SF, defined by Equation 14:

$$SF = 1 - \frac{1}{e^{e^{bcd-wcd}}} \tag{14}$$

Thus, we seek to maximize Equation 14 to obtain the most reliable number of clusters. The score function is now bounded by]0,1[and deals with the perfect cluster case and the one cluster case. The strength of the SF depends on the fact that it is built on ideas from several indices. Since it is not based on minimum/maximum values, it is not influenced by outliers. The size of clusters is taken into account in both *bcd* and *wcd*. The comparison of centroids is used in the place of single or complete linkage. This avoids the computational complexity. The number of clusters k is used to penalize the addition of clusters. Finally, the exponential notation is used to both take care of the single and perfect cluster cases and to define bounds. As can be seen through Equations 10 and 11, computational complexity is linear. If n is the number of data points, then the proposed score function has a complexity of $O(n)$. Tests that have been conducted with benchmark problems indicate that this function provides good results. This is the subject of the next section.

4 Results

In this Section, the performance of selected clustering indices is compared. For this purpose, the standard K-means algorithm is used. K-means evolves k crisp and hyper-spheroidal clusters in order to minimize their intra-cluster distances, shown as the metric J in Equation 15:

$$J = \sum_{j=1}^{k} \sum_{x_i \in c_j} d(x_i, z_j)^2 \tag{15}$$

where k is the number of clusters, x_i the i-th data point and z_j the centroid of cluster c_j. The k starting centroids are chosen randomly among all data points. The data set is then partitioned according to the minimum squared distance. The cluster centers are iteratively updated by computing the mean of the points belonging to the clusters. The process of partitioning and updating is repeated until a stopping criterion is reached. This happens when either the cluster centers or the value of the metric J in Equation 15 do not significantly change over two consecutive iterations.
To control the randomness of K-means, it is launched $t = 20$ times from k_{min} to k_{max} clusters. The optimum - minimum or maximum, depending on the index - is chosen as the most suitable number of clusters. Indices for comparison have been chosen according to their performance and usage reported in the literature (see Section 1). Selected indices are Dunn (DU), Calinski-Harabasz (CH), Davies-Bouldin (DB), Silhouette (SI), Maulik-Bandyopadhyay (MB), and Geometric (GE). These are compared with the Score Function (SF). Subsection 4.1 shows the results according to the number of clusters identified for both artificial and real-life data sets. Subsection 4.2 studies the perfect cluster case. The special case of one cluster is outlined in subsection 4.3. Finally, the sub-cluster issue is presented in Section 4.4.

4.1 Number of clusters

In this subsection, there are two goals. The first goal is to test the score function on benchmark data sets. The second goal is to compare results between indices. k_{min} and k_{max} are taken to be respectively 2 and 10. If not explicitly stated, data sets used in this Section are composed of 1000 points in two dimensions.
Example 1: In the first data set, *Unbalanced*, three clusters of different compactness are present (see Figure 2a). Clusters of varying densities is an important issue [7]. Table 1 shows that, unlike other indices, Dunn is not able to correctly estimate the number of clusters (three). This is due to the definition of the Dunn index. The diameter, for example, can be affected by outliers since it is not based on a mean value.
Example 2: The second data set, *Overlapped*, consists of three clusters. Two of these clusters overlap (see Figure 2b). This data set is important since the ability to deal with overlapping clusters is one of the best ways to compare indices [4]. Table 2 shows the results for this data set. GE overestimates the number of clusters. A weakness of GE is to be based on the minimum distance between two clusters. This gives problems when dealing with overlapping clusters. DU, DB and SI identify the two overlapping clusters as one cluster. This is due to their dependence to a minimum or maximum value. This is not the case with CH, MB and SF which correctly estimate the three clusters.
Example 3: This data set, named *Noisy*, contains seven clusters with an additional noise. It can be seen in Figure 2c. It is rarely the case that clusters appear clearly in real situations.

The data are often noisy and some indices are sensitive to noise as pointed out in [17]. Table 3 contains the results for this specific data set. It can be seen that DU, CH, DB, SI and MB overestimate the correct number of clusters. Presence of noise is too strong for these indices to correctly estimate the number of clusters. Only GE and SF are able to determine the seven clusters.

Example 4: The following data set, named *Subcluster* contains five clusters, with two "pairs". It is visible in Figure 2d. It can happen in real-life that data sets contain clusters which are closely grouped together. Existing indices developed for hard clustering may not be able to deal with such situations. Table 4 presents the results for this data set. More details about sub-cluster hierarchies can be found in Section 4.4.

Example 5: The next data set, named *Wine*, is a real-life data set [37]. It is made of 178 points in 13 dimensions. *Wine* contains 3 clusters. Results of the seven indices are given in Table 5. Here, CH, DB, SI and SF are able to discover the three clusters. While MB underestimates the number of clusters, DU and GE over-estimate the correct value.

Example 6: In this last example, the *Cancer* data set is used [37]. It contains 569 points in 30 dimensions. *Cancer* is composed of 2 clusters and is a good example of a problem in a relatively high dimensional space. Results are presented in Table 6. Three indices, CH, SI and SF, are able to deal with these two clusters represented in 30 dimensional space. DU, DB, MB and GE are not able to catch the trend due to either the cluster shapes or the high dimensionality of the data.

Table 7 summarizes the results of the application of the seven indices to four artificial and two real-life data sets. SF is the only index performing well on all data sets. The closest index, in term of good results, is CH. This is due to the similarity of the two equations. Both CH and SF takes into account the number and size of clusters. Among all, CH and SF are the only two indices to be based on a comparison of cluster centroid (z_i) with overall centroid (z_{tot}).

In our experiments, SF correctly identified the number of clusters in all six data sets. The SF successfully processes the standard case with clusters of different size and compactness (*Unbalanced*), overlapped clusters (*Overlapped*), clusters with noise (*Noisy*), groups of clusters (*Subcluster*) and multidimensional data (*Wine* and *Cancer*).

To test the score function more completely, several other aspects are evaluated. For example, challenges such as perfect clusters and sub-clusters are important. The single cluster case has to be considered as well. Although not commonly studied in the literature, it may often happen in practice. Recent research by others that deal with clustering validity indices, have limited to cluster data from 2 to k_{max} clusters. Finally, a comparative study of all indices is done.

4.2 Perfect Clusters

The SF upper bound indicates the perfect cluster case; proximity to this bound (1.0) is a measure of closeness of data sets to perfect clusters. The next two data sets are used to test how the SF deals with perfect clusters. The data sets *Perfect3* and *Perfect5* are made of 1000 points in 2D and contain three and five clusters respectively which are nearly perfect (i.e. with a very high compactness).

The correct number of clusters is identified in both situations. An interesting observation is related to the maximum value for the SF. In the first case (0.854), the maximum is higher than in the second one (0.772). This is due to the dependence of the SF on the number of clusters k. This can be seen in Equations 10 and 11. More details of the influence of k can be found in

Section 5.1. Finally, the SF gives an idea of how good clusters are through the proximity of the value of the index to its upper bound of unity.

4.3 Single Cluster

Before attempting to identify a single cluster, the definition of a cluster should be clarified. Several definitions exist in the literature. A possible definition is given in [34]. It states that a cluster is considered to be "real" if it is significantly compact or isolated or both at the same time. Concepts of compactness and isolation are based on two parameters that define internal properties of a cluster. The main drawback of such definitions is that they are often too restrictive; few data sets satisfy such criteria. Another way of testing for the existence of a single cluster is the null hypothesis [11]. However, this test is usually carried on univariate data. An objective of the index, SF, is to accommodate the single cluster case. This case is not usually treated by other indices. In this subsection, k_{min} and k_{max} are taken to be respectively 1 and 8. Plot of SF with respect to the number of clusters provide indications related to how the single cluster case can be identified. Firstly, two situations may occur. Either the number of clusters is clearly located with a global maximum (Figure 3, left) or the SF has no clear global maximum (Figure 3, right).
Since in the first situation, the number of clusters is identifiable, the challenge lies in the second situation. In this case, there are two possibilities. They are: i) data forms a single cluster and ii) the correct number of clusters is higher than k_{max}.
In this paper, an empirical equation is proposed to distinguish between these two cases. For this purpose, three new data sets are introduced: *Single*, which contains 1000 points in 2D representing a single and spherical cluster, *SingleN* is the same cluster as *Single* plus added noise and *Single30* is a single cluster in a 30 dimensional space. It has been observed that in the single cluster cases, the value of the SF when $k = 2$, denoted as SF_2 is closer to the value for $k = 1$ (SF_1) than in other data sets. Therefore, the ratio between SF_1 and SF_2 is used as an indicator of single cluster as shown in Equation 16.

$$\frac{SF_1}{SF_2} \geq \epsilon \tag{16}$$

where SF_1 and SF_2 are respectively the value for SF when $k = 1$ and $k = 2$. Results of this indicator on artificial and real-life benchmark data sets are given in Table 8.
According to Table 8, it is empirically stated that the data set is likely to contain one cluster if Equation 16 is satisfied with $\epsilon \cong 0.6$. Only three data sets containing a single cluster satisfy the condition in Equation 16.

4.4 Sub-clusters

Another case is the sub-cluster situation. This occurs when existing clusters can be seen as a cluster hierarchy. If this information can be captured by the validity index, more information about the structure of the data can be given to the user. The data set *Subcluster* in Figure 2d is an example of this situation. The index SF is compared with the previously mentioned indices on this topic. Figure 4 shows the evolution of each validity index with respect to the number of clusters.
In Figure 4, MB is not able to find the correct number of clusters (neither the sub-clusters, nor the overall clusters). In the case of DU, only the overall three clusters are detected. The reason is

related to the distance measured between two clusters. Dunn uses the minimum between points in two different clusters c_i and c_j. This strategy is limited in the case of the *Subcluster* data set since clusters overlap. With SI, although the sub-cluster hierarchy is visible, the recommended number of clusters is three. Finally, the indices that are able to find five clusters and show a peak at three clusters are CH, DB, GE and SF.

4.5 Comparative Study

All of these indices are different. Distinguishing aspects are their definition, their optimization strategy (minimum/maximum), their complexity or their definition with specific numbers of clusters such as $k = 1$. An index may have an hyper-parameter to tune. This is the case of the MB index. The computational complexity is important. Although data sets tested in this article are small, other real-life examples may have tens or hundreds of thousands of points. In these cases, a validity index with a linear complexity is preferred over polynomial complexity. Since none of the other indices are bounded, the perfect cluster case is difficult to identify. When a value is obtained for a given index, it is usually difficult, or impossible, to know the proximity of the data set in relation to the perfect cluster situation. Since the single cluster case is usually not taken into consideration when developing indices, most of them are not defined for such a situation. This is the case for DU, DB, SI, MB and GE. All of these indices somehow involve the distance between two different clusters. In a single cluster case there is no such value. Although this problem does not appear for CH, the denominator of Equation 2 prevents the single cluster situation. Table 9 contains a summary of the important properties of the seven validity indices. Except for indices DB and GE, which have to be minimized, all indices have to be maximized on $k = 2..n$. Only SF can be maximized on $k = 1..n$ due to its definition. The standard computational complexity is O(n), with n being the number of points, except for DU and SI ($O(n^2)$). This is due to the way these two indices calculate the distance between clusters. MB is the only index with an hyper-parameter (p in Equation 9). This value is usually chosen to be two in the literature [28, 36]. Concerning the bounds, the SF is the only index that has a lower and upper bound. This is a strong advantage with regards to other indices since it increases the usefulness of the value. SF is also the only index to be defined for the single cluster case ($k = 1$). For all other indices, the number given in Table 9 refers to the Equation where $k = 1$ is an undefined issue. Finally, only CH, DB, GE and SF reveal sub-clusters in data.
To conclude, main drawbacks of the Dunn index are its computational load and its sensitivity to noise. It is useful for identifying clean clusters in data sets containing no more than hundreds of points. Although the Davies-Bouldin index gives good results for distinct groups, it is not designed to accommodate overlapping clusters. The Silhouette index is only suitable for estimating the first choice and therefore, it should not be applied to data sets with sub-clusters. The Maulik-Bandyopadhyay index has the particularity of being dependent on a user specified parameter. The Maulik-Bandyopadhyay and Geometric indices have been found to give bad results on multidimensional data sets. Although closely related to SF, CH has no upper bound and is not defined for $k = 1$.

5 Limitations

Since the SF depends on two exponentials, its evolution when the number of clusters is equal to the number of points requires specific study. In addition, the data sets presented so far

contain only hyper-spheroidal clusters. Additional tests with arbitrarily shaped clusters have been carried out. These issues are treated in the next subsections.

5.1 Score Function Evolution

In Section 3, the index SF has been adapted so that it is bounded. Therefore, the SF has a lower bound of zero (no cluster structure) and an upper bound of one (perfect clusters). The purpose of the study in this subsection is to investigate the behavior of the SF for a large number of clusters. More specifically, the limits of the SF when the number of clusters (k) tends to the number of points (n) is studied. When k tends to n, the *wcd* tends to zero (see Equation 11). This is the case when each point represents a single cluster. The evolution of *bcd* is described by Equation 17:

$$\lim_{k \to n} bcd = \frac{1}{n^2} \sum_{i=1}^{n} d(x, z_{tot})^2 \tag{17}$$

Equation 17 can be rewritten as a function of the standard deviation σ:

$$\lim_{k \to n} bcd = \frac{\frac{1}{n} \sum_{i=1}^{n} d(x, z_{tot})^2}{n} = \frac{\sigma^2}{n} \tag{18}$$

Consequently, the limit for SF when the $k \to n$ can be written as:

$$\lim_{k \to n} SF = 1 - \frac{1}{e^{e^{\sigma^2/n}}} \tag{19}$$

Two situations occur depending on the order of magnitude of σ^2 and n. They are presented in Equation 20:

$$\lim_{k \to n} SF = \begin{cases} 1 & for\ \sigma^2 \gg n \\ \sim 0.63 & for\ \sigma^2 \ll n \end{cases} \tag{20}$$

The second case is the most likely to happen when data is normalized. The evolution of the SF with both the *bcd* and the *wcd* is plotted with respect to the number of clusters. This number varies from $k_{min} = 1$ to $k_{max} = 30$. Results for the data set *Overlapped* are shown in Figure 5. Starting from zero (single cluster), the *bcd* has its maximum at $k = 2$ and decreases monotonically. The *wcd* starts with a high value and decreases monotonically as well. Concerning the SF, a maximum is observed at the correct number of clusters $k = 3$. The SF tends to 0.63 which is the limit found by Equation 20.

Figure 6 shows the results for the *Noisy* data set. After reaching a maximum for $k = 7$, the value of the SF stabilizes as predicted by Equation 20. The *wcd* decreases monotonically with a *knee* at $k = 7$. It is observed that the *bcd* closely follows the *wcd* starting at $k = 7$.

Finally, the case of a single cluster - *SingleN* - is studied (Figure 7). The *bcd* has a typical increase and then stabilizes. Instead of decreasing, the *wcd* grows from 1 to 3 clusters. This shows that k should not be increased. Thus, the SF has a minimum at $k = 3$ clusters and then grows slowly. This shows that in addition to validating Equation 16, the SF evolution indicates a single cluster presence in the data set.

Empirical tests have also been carried out. For a precise comparison of indices, the starting centroids are chosen to be the same in five runs. For each index, the best result over these five

runs is taken as the correct number of clusters. Seven data sets that contain 16, 25, 36, 49, 64, 81 and 100 clusters are used. Limits on k, k_{min} and k_{max}, are chosen to be, respectively, 2 and 110. Results are given in Table 10.
It is observed that all indices have difficulty finding the correct number of clusters for $k > 15$. This is due to the effect of the starting centroid locations. The probability of obtaining good centroid locations at the beginning - and therefore the correct number of clusters at the end - becomes smaller as the number of clusters increases [42]. This issue can be resolved for many situations using methodologies to find better starting centroid locations [38].
However, the higher the number of clusters, the less effective these methodologies become. To illustrate the dependency of K-means results to initial centroid locations, an additional test has been carried out. The data set containing 49 clusters (see Table 10) is used again. However, in this case, initial centroid locations are chosen so that each starting position is in a distinct cluster. Aside from DU and GE, all indices find the correct number of clusters. This thus shows that for high number of clusters, good results can be achieved only when starting centroids are correctly placed.

5.2 Arbitrarily Shaped Clusters

In the above subsections, data sets used to test the different indices contain hyper-spheroidal clusters. The purpose of this subsection is to study arbitrarily-shaped clusters. Three new data sets are introduced. *Rectangle* contains 1000 points in 2D representing five rectangular clusters. The data set *Nonconvex* is made of 284 regularly-spaced points in 2D. It contains three clusters, one of them is not convex. Finally, *Ellipsoidal* is a data set made of 3 ellipsoidal clusters (1000 points in 2D). These data sets are shown in Figure 8.
Regarding the *Rectangle* data set, all indices overestimate the correct number of clusters (5). Results for different indices are: DU (9), CH (10), DB (10), SI (7), MB (10), GE (10) and SF (10). While it is clear that the SF is not able to find the real number of clusters, other indices have the same difficulty. This is mainly due the size of the different clusters and their non-spheroidal shape. Moreover, as stated in [42], K-means is usually not reliable for non-spheroidal clusters.
Concerning the next data set, *Nonconvex*, the difficulty lies in the fact that one of the clusters is non-convex. In this case, the value of SF (4), although close, overestimates the correct number of clusters (3). The following indices are also close to the real number of clusters: DU (2), DB (4) and MB (4). This is not the case for CH (6), SI (6) and GE (10). In the case of non-convex clusters, another clustering algorithm than K-means is advised.
In the last data set, *Ellipsoidal*, the clusters are far from spherical in shape. All indices fail when estimating the number of clusters (3). All indices overestimate the real number of clusters: DU (9), CH (10), DB (10), SI (10), MB (10), GE (10) and SF (10). Since all indices involves the calculation of some diameter or variance of clusters, the process fail when applied to strongly ellipsoidal shaped clusters. Therefore, a limitation of the score function, as well as other tested indices using K-means, is their restriction to data sets containing hyper-spheroidal clusters.

6 Discussion and Conclusions

A variety of validity indices exist in the literature. However, most of them succeed only in certain situations. A new index for hard clustering called the score function (SF), is presented

and studied in depth in this paper. The index is based on an equation that computes the within and between class distances. It has been developed to accommodate special cases such as single cluster and perfect cluster cases.
The SF is able to estimate correctly the number of clusters in a variety of artificial and real-life data sets. In a data set involving unbalanced clusters, the SF is able to correctly estimate the number of clusters, which is not the case with the DU index. Concerning DU index, results confirm a previous study that found this index to be sensitive to noise. CH, MB and SF are the only three indices to succeed when confronted with overlapping clusters. The data set containing seven clusters with noise is correctly handled by GE and SF. However, GE is often found to overestimate the real number of clusters in most data sets. Finally, in the case of sub-cluster hierarchies, CH, DB, GE and SF are able to estimate the five clusters and overall three groups. In general, CH and SF give the best results.
More particularly, the SF is better or as good as six other validity indices (Silhouette, Dunn, Calinski-Harabasz, Davies-Bouldin, Maulik-Bandyopadhyay and Geometric) for the K-means algorithm on hyper-spheroidal clusters. It has been found that for arbitrarily-defined cluster shapes, the SF is usually not able to estimate the correct number of clusters. This is also the case for all other indices that were studied. In addition, the SF has been tested successfully on multidimensional real-life data sets. The proposed index can also accommodate perfect and single cluster cases. In order to identify the one cluster case, an empirical condition has been formulated. Finally, determining values for the index is computationally efficient.
Several extensions to the present work are in progress. A more detailed study of the sub-cluster case is an important part of this work. Applications to other clustering algorithms, such as stability-based clustering, are also under way.

Acknowledgments

This research is funded by the Swiss National Science Foundation through grant no 200020-109257. The authors recognize the two anonymous reviewers for their comments as well as Dr. Fleuret and Dr. Kripakaran for fruitful discussions.

References

[1] P. Berkhin. Survey of clustering data mining techniques. Technical report, Accrue Software, San Jose, CA, 2002.

[2] J.C. Bezdek and N.R. Pal. Some new indexes of cluster validity. *IEEE Transactions on Systems, Man and Cybernetics*, 28(3):301–315, 1998.

[3] N. Bolshakova and F. Azuaje. Cluster validation techniques for genome expression data. *Signal Processing*, 83(4):825–833, 2003.

[4] M. Bouguessa, S. Wang, and H. Sun. An objective approach to cluster validation. *Pattern Recognition Letters*, 27(13):1419–1430, 2006.

[5] M. Brun, C. Sima, J. Hua, J. Lowey, B. Carroll, E. Suh, and E.R. Dougherty. Model-based evaluation of clustering validation measures. *Pattern Recognition*, 40(3):807–824, March 2007.

[6] R.B. Calinski and J. Harabasz. A dendrite method for cluster analysis. *Communications in Statistics*, 3:1–27, 1974.

[7] C.H. Chou, M.C. Su, and E. Lai. A new cluster validity measure and its application to image compression. *Pattern Analysis Applications*, 7(2):205–220, 2004.

[8] D.L. Davies and W. Bouldin. A cluster separation measure. *IEEE PAMI*, 1:224–227, 1979.

[9] C. Ding and X. He. K-means clustering via principal component analysis. In *Proceedings of the* 21^{st} *International Conference on Machine Learning*, ACM International Conference Proceeding Series, page 29. ACM Press, 2004.

[10] J.C. Dunn. Well separated clusters and optimal fuzzy partitions. *Journal of Cybernetics*, 4:95–104, 1974.

[11] L. Engelman and J.A. Hartigan. Percentage points of a test for clusters. *Journal of the American Statistical Association*, 64:1647–1648, 1969.

[12] A.F. Famili, G. Liu, and Z. Liu. Evaluation and optimization of clustering in gene expression data analysis. *Bioinformatics*, 20(11):1535–1545, 2004.

[13] C. Fraley and A.E. Raftery. How many clusters? which clustering method? answers via model-based cluster analysis. *The Computer Journal*, 41(8):578–588, 1998.

[14] D. François. *High-dimensional data analysis: optimal metrics and feature selection.* PhD thesis, Université catholique de Louvain, Louvain-la-Neuve, Belgium, 2007.

[15] S. Garatti, S. Bittanti, D. Liberati, and A. Maffezzoli. An unsupervised clustering approach for leukaemia classification based on dna micro-arrays data. *Intelligent Data Analysis*, 11(2):175–188, 2007.

[16] A.D. Gordon. *Data science, classification and related methods (eds. Hayashi, C. and Yajima, K. and Bock H.H. and Ohsumi, N. and Tanaka, Y. and Baba, Y.)*, chapter Cluster Validation, pages 22–39. Springer, 1996.

[17] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. On clustering validation techniques. *Journal of Intelligent Information Systems*, 17(2-3):107–145, 2001.

[18] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. Cluster validity methods: part 1. *SIGMOD Rec.*, 31(2):40–45, 2002.

[19] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. Cluster validity methods: part 2. *SIGMOD Rec.*, 31(3):19–27, 2002.

[20] M. Halkidi, M. Vazirgiannis, and I. Batistakis. Quality scheme assessment in the clustering process. In *Proceedings of the Fourth European Conference on Principles of Data Mining and Knowledge Discovery*, volume 1910 of *LNCS*, pages 265–267. Springer-Verlag, 2000.

[21] M.A. Hearst. Clustering versus faceted categories for information exploration. *Communications ACM*, 49(4):59–61, 2006.

[22] Z. Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, 2(3):283–304, 1998.

[23] A. Jain and R. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.

[24] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[25] A.K. Jain, A. Topchy, M. Law, and J. Buhmann. Landscape of clustering algorithms. In *Proceedings of the 17th International Conference on Pattern Recognition*, pages 260–263, 2004.

[26] L. Kaufman and P.J. Rousseeuw. *Finding Groups in Data: an Introduction to Cluster Analysis*. John Wiley & Sons, 1990.

[27] M. Kim and R.S. Ramakrishna. New indices for cluster validity assessment. *Pattern Recognition Letters*, 26(15):2353–2363, 2005.

[28] M. Kim, H. Yoo, and R.S. Ramakrishna. Cluster validation for high-dimensional datasets. In *LNAI 3192*, pages 178–187. Springer-Verlag Berlin Heidelberg, 2004.

[29] J. Kleinberg. An impossibility theorem for clustering. In *16th conference on Neural Information Processing Systems*, 2002.

[30] E.E. Korkmaz, J. Du, R. Alhajj, and K. Barker. Combining advantages of new chromosome representation scheme and multi-objective genetic algorithms for better clustering. *Intelligent Data Analysis*, 10(2):163 – 182, 2006.

[31] R. Kothari and D. Pitts. On finding the number of clusters. *Pattern Recognition Letters*, 20(4):405–416, 1999.

[32] B.S.Y. Lam and H Yan. A new cluster validity index for data with merged clusters and different densities. In *2005 IEEE International Conference on Systems, Man and Cybernetics*, volume 1, pages 798– 803, 2005.

[33] T. Li, S. Zhu, and M. Ogihara. Algorithms for clustering high dimensional and distributed data. *Intelligent Data Analysis*, 7(4):305–326, 2003.

[34] R.F. Ling. On the theory and construction of k-clusters. *Computer Journal*, 15:326–332, 1972.

[35] Y.G. Liu, X.F. Liao, X.M. Li, and Z.F. Wu. A tabu clustering algorithm for intrusion detection. *Intelligent Data Analysis*, 8(4):325–344, 2004.

[36] U. Maulik and S. Bandyopadhyay. Performance evaluation of some clustering algorithms and validity indices. *IEEE Transactions Pattern Analysis Machine Intelligence*, 24(12):1650–1654, 2002.

[37] C.J. Merz and P.M. Murphy. *UCI* machine learning repository, 1996. http://www.ics.uci.edu/∼mlearn/MLSummary.html.

[38] J.M. Pe na, J.A. Lozano, and P. Larra naga. An empirical comparison of four initialization methods for the k-means algorithm. *Pattern Recognition Letters*, 20(10):1027–1040, 1999.

[39] D. Pelleg and A. Moore. X-means: Extending K-means with efficient estimation of the number of clusters. In *Proc. 17th International Conference on Machine Learning*, pages 727–734. Morgan Kaufmann, San Francisco, CA, 2000.

[40] S.A. Salem and A.K. Nandi. New assessment criteria for clustering algorithms. In *2005 IEEE Workshop on Machine Learning for Signal Processing*, pages 285–290, 2005.

[41] E. SanJuan and F. Ibekwe-SanJuan. Text mining without document context. *Inf. Process. Manage.*, 42(6):1532–1552, 2006.

[42] P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining*. Addison Wesley, 2006.

[43] S. Theodoridis and K. Koutroumbas. *Pattern Recognition*. Academic Press, 1999.

[44] J. Vesanto and E. Alhoniemi. Clustering of the selforganizing map. *IEEE Transactions on Neural Networks*, 11(3):586–600, 2000.

[45] S. Wu and T.W.S. Chow. Clustering of the self-organizing map using a clustering validity index based on inter-cluster and intra-cluster density. *Pattern recognition*, 37(2):175–188, 2004.

[46] R. Xu and D. Wunsch. Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, 16(3):645–678, 2005.

[47] X. Yang, A. Cao, and Q. Song. A new cluster validity for data clustering. *Neural Processing Letters*, 23(3):325–344, 2006.

List of Tables

k	2	3	4	5	6	7	8	9	10
DU	**0.056**	0.036	0.022	0.014	0.017	0.008	0.009	0.007	0.010
CH	950.6	**3453.3**	2725.0	2455.3	2111.1	2214.6	1961.3	2160.6	2107.9
DB	0.800	**0.457**	0.697	0.688	0.784	0.762	0.852	0.846	0.819
SI	0.682	**0.893**	0.819	0.716	0.714	0.728	0.593	0.521	0.565
MB	2.746	**7.600**	6.245	5.106	4.986	4.236	3.819	3.971	3.618
GE	3.257	**1.720**	1.842	1.876	1.939	1.931	2.043	2.107	2.212
SF	0.489	**0.648**	0.627	0.617	0.603	0.595	0.593	0.584	0.584

Table 1: Results of the seven validity indices on the *Unbalanced* data set (example 1). The best result on 20 runs is taken. The data set is shown in Figure 2a. Bold numbers show maximum values for all indices except DB and GE, where minimum values are desired. This indication is used for Tables 1-7. The correct number of clusters is $k = 3$.

k	2	3	4	5	6	7	8	9	10
DU	**0.091**	0.011	0.012	0.016	0.016	0.017	0.021	0.014	0.019
CH	1346.2	**2497.6**	2154.2	1996.0	1941.3	1887.3	1834.7	1772.8	1744.2
DB	**0.543**	0.562	0.653	0.809	0.784	0.766	0.767	0.753	0.731
SI	**0.779**	0.771	0.672	0.611	0.582	0.583	0.589	0.597	0.592
MB	4.426	**5.520**	4.646	3.800	3.208	2.827	2.592	2.374	2.061
GE	2.719	1.885	2.046	2.010	1.885	1.745	1.676	1.705	**1.625**
SF	0.577	**0.636**	0.612	0.593	0.588	0.582	0.579	0.577	0.576

Table 2: Results of the seven validity indices on the *Overlapped* data set (example 2). The data set is shown in Figure 2b. The correct number of clusters is $k = 3$.

k	2	3	4	5	6	7	8	9	10
DU	0.038	0.038	0.064	0.070	0.077	0.077	0.067	0.075	**0.081**
CH	769.3	1018.6	1476.1	1722.4	2174.7	2849.9	3136.7	3201.4	**3294.3**
DB	1.108	0.700	0.608	0.500	0.457	0.465	**0.440**	0.481	0.486
SI	0.580	0.636	0.740	0.768	0.803	0.829	0.843	0.852	**0.860**
MB	1.640	1.744	3.069	3.845	5.457	7.370	**7.937**	7.136	6.148
GE	4.215	2.717	1.860	1.613	1.079	**0.952**	1.191	1.534	1.507
SF	0.419	0.513	0.567	0.590	0.604	**0.612**	0.605	0.601	0.601

Table 3: Results of the seven validity indices on the *Noisy* data set (example 3). The data set is shown in Figure 2c. The correct number of clusters is $k = 7$.

k	2	3	4	5	6	7	8	9	10
DU	0.059	**0.069**	0.020	0.017	0.016	0.014	0.014	0.014	0.015
CH	979.3	2431.0	2647.7	**3774.1**	3351.0	3045.1	2833.7	2636.2	2550.7
DB	0.907	0.489	**0.467**	0.469	0.579	0.683	0.714	0.750	0.792
SI	0.657	**0.841**	0.821	0.810	0.735	0.729	0.677	0.635	0.661
MB	1.890	9.523	16.206	43.550	**54.825**	36.058	49.388	43.522	41.192
GE	3.793	1.235	1.147	**1.122**	1.510	1.435	1.525	1.570	1.658
SF	0.480	0.636	0.638	**0.641**	0.627	0.618	0.613	0.606	0.601

Table 4: Results of the seven validity indices on the *Subcluster* data set (example 4). The data set is shown in Figure 2d. The correct number of clusters is $k = 5$.

k	2	3	4	5	6	7	8	9	10
DU	0.160	0.232	0.232	0.210	0.190	0.235	0.212	**0.239**	0.234
CH	69.52	**70.94**	56.20	47.17	42.23	38.26	36.26	34.33	32.73
DB	1.505	**1.257**	1.501	1.481	1.402	1.421	1.307	1.423	1.425
SI	0.426	**0.451**	0.418	0.407	0.390	0.368	0.313	0.348	0.353
MB	**5.689**	5.391	3.546	3.445	2.682	2.008	1.893	1.733	1.380
GE	97.747	99.209	104.685	101.154	108.083	97.892	93.336	**86.958**	91.108
SF	0.269	**0.385**	0.314	0.324	0.253	0.240	0.231	0.233	0.242

Table 5: Results of the seven validity indices on the *Wine* data set (example 5). The data set is made of 178 points in a 13 dimension space. The correct number of clusters is $k = 3$.

k	2	3	4	5	6	7	8	9	10
DU	0.076	0.078	0.075	0.078	0.072	0.064	0.072	**0.079**	0.067
CH	**267.7**	197.1	159.0	140.4	128.8	118.6	109.7	103.3	98.1
DB	1.444	1.461	1.502	1.432	1.534	**1.391**	1.418	1.408	1.457
SI	**0.519**	0.492	0.441	0.427	0.279	0.257	0.259	0.244	0.228
MB	16.202	11.433	13.890	10.265	**26.346**	20.834	14.002	5.697	12.279
GE	2.599	2.497	2.426	2.558	2.946	2.546	2.231	2.273	**2.215**
SF	**0.657**	0.446	0.340	0.238	0.216	0.160	0.149	0.137	0.124

Table 6: Results of the seven validity indices on the *Cancer* data set (example 6). The data set is made by 569 points represented in 30 dimensions. The correct number of clusters is $k = 2$.

Data Sets	**DU**	**CH**	**DB**	**SI**	**MB**	**GE**	**SF**
Unbalanced	2(X)	3(O)	3(O)	3(O)	3(O)	3(O)	3(O)
Overlapped	2(X)	3(O)	2(X)	2(X)	3(O)	10(X)	3(O)
Noisy	10(X)	10(X)	8(X)	10(X)	8(X)	7(O)	7(O)
Subcluster	3(X)	5(O)	4(X)	3(X)	6(X)	5(O)	5(O)
Wine	9(X)	3(O)	3(O)	3(O)	2(X)	9(X)	3(O)
Cancer	9(X)	2(O)	7(X)	2(O)	6(X)	10(X)	2(O)

Table 7: Estimated number of clusters for six data sets and seven validity indices. Notation (O) and (X) respectively indicates when the correct number of clusters has been found or not.

Data sets	Indicator	Data sets	Indicator
Unbalanced	0.44	*SingleN*	**1.28**
Overlapped	0.37	*Single30*	**0.60**
Noisy	0.52	*Wine*	0.10
Subcluster	0.45	*Cancer*	0.01
Single	**0.61**		

Table 8: Results of the indicator (SF_1/SF_2) for nine benchmark data sets. Bold numbers indicate the single cluster cases.

	DU	**CH**	**DB**	**SI**	**MB**	**GE**	**SF**
On $k = 2..n$	max	max	min	max	max	min	max
Hyper-parameters	no	no	no	no	p in Equ. 9	no	no
Complexity	O(n^2)	O(n)	O(n)	O(n^2)	O(n)	O(n)	O(n)
Bounds	]0,∞[	]0,∞[	]0,∞[	]-∞,∞[	]0,∞[	]0,∞[	]0,1[
Single cluster	Equ. 1	Equ. 2	Equ. 5	Equ. 7	Equ. 8	Equ. 9	emp.
Sub-clusters	no	emp.	emp.	no	no	emp.	emp.

Table 9: Properties of the seven compared validity indices. The single cluster line refer to the Equation preventing from single cluster. The sub-clusters line shows *emp.* for indices that are shown empirically to find sub-clusters.

Indices	**16**	**25**	**36**	**49**	**64**	**81**	**100**
DU	11	18	28	NA	NA	NA	NA
CH	20	34	84	76	68	73	84
DB	15	35	36	38	59	63	84
SI	15	28	52	50	83	98	108
MB	110	110	110	70	83	98	103
GE	NA	NA	NA	NA	NA	NA	NA
SF	20	34	58	76	68	74	84

Table 10: Estimated number of clusters for seven data sets containing respectively 16, 25, 36, 49, 64, 81 and 100 clusters. For each validity index, the best value over 5 runs with fixed K-means starting centroid locations are given. NA stands for Not Available (for example due to infinite or divide by zero issues).

List of Figures

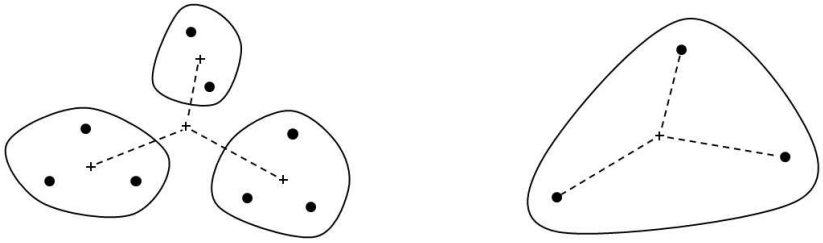

Figure 1: Graphical representations of *bcd* (left) and *wcd* (right).

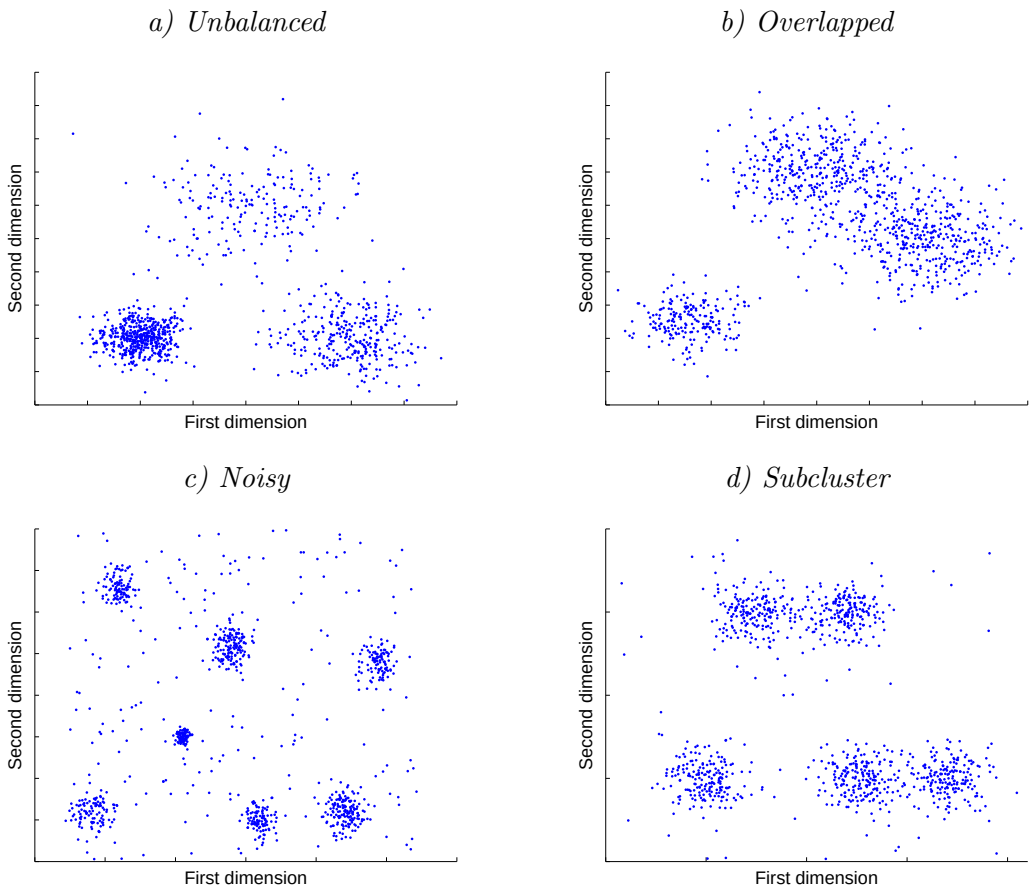

Figure 2: Four artificial data sets, *Unbalanced*, *Overlapped*, *Noisy* and *Subcluster*.

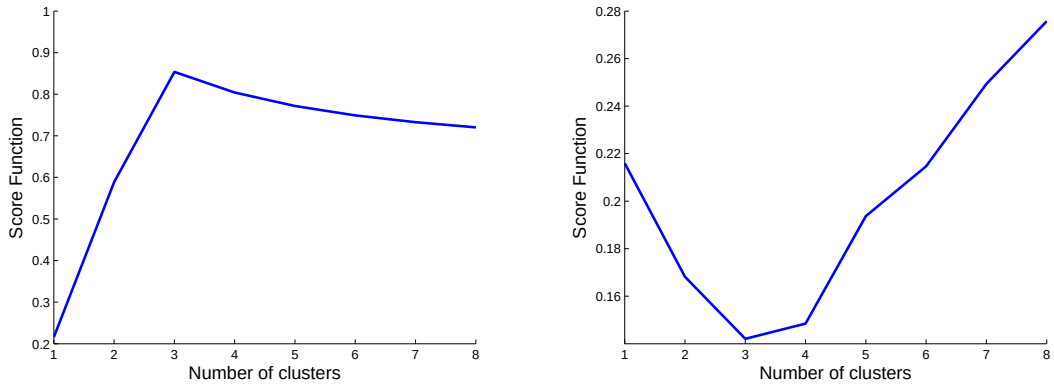

Figure 3: Difference of the SF trend with a data set containing three clusters (left) and single cluster (right).

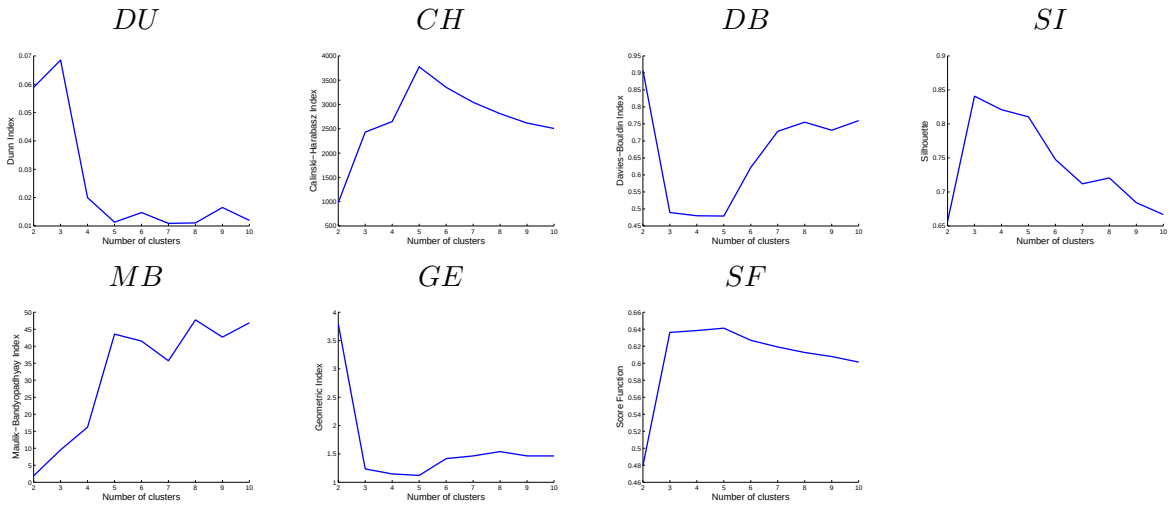

Figure 4: Comparison of DU, CH, DB, SI, MB, GE and SF for the sub-cluster case of Figure 2d. DB and GE must be minimized.

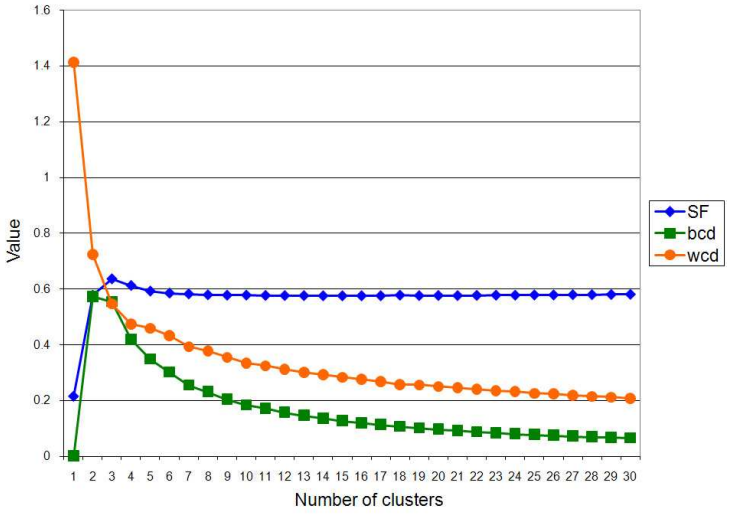

Figure 5: Evolutions of the SF and components *bcd* and *wcd* for the data set *Overlapped* from $k_{min} = 1$ to $k_{max} = 30$. For each number of cluster, the best over 20 runs is taken.

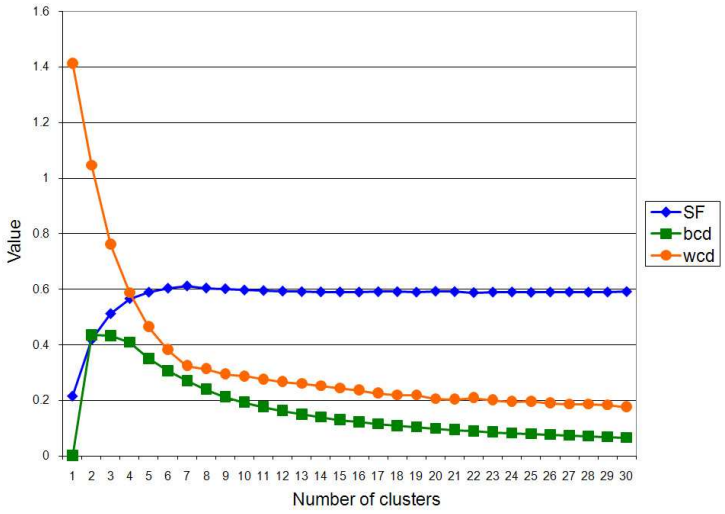

Figure 6: Evolutions of the SF and its main components *bcd* and *wcd* for the data set *Noisy* from $k_{min} = 1$ to $k_{max} = 30$. For each number of cluster, the best over 20 runs is taken.

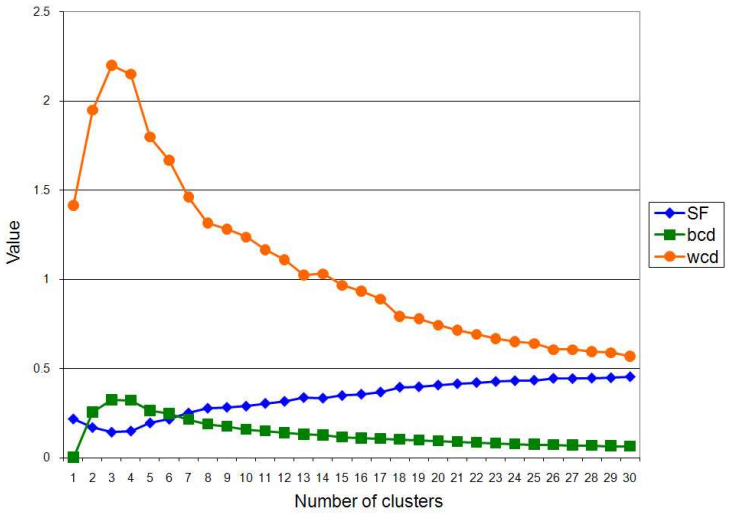

Figure 7: Evolutions of the SF and its main components *bcd* and *wcd* for the data set *SingleN* from $k_{min} = 1$ to $k_{max} = 30$. For each number of cluster, the best over 20 runs is taken.

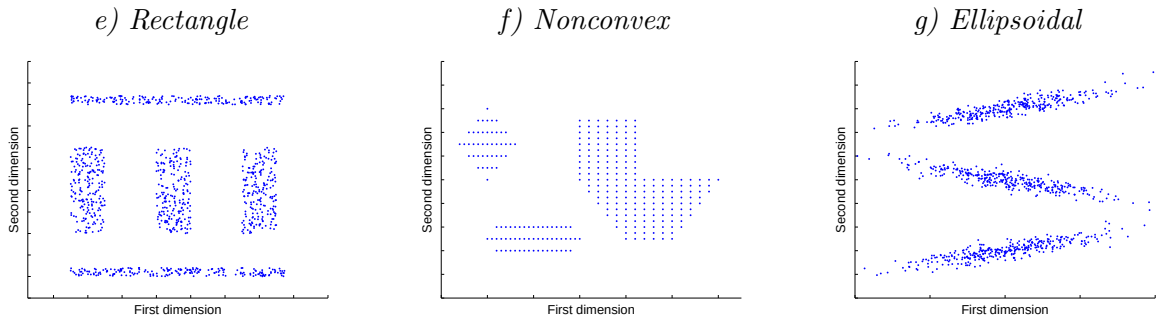

Figure 8: Three new artificial data sets. *Rectangle* and *Ellipsoidal* contain 1000 points in 2D while *Nonconvex* is made of 284 points in 2D.