# How many Clusters: A Validation Index for arbitrary shaped clusters.

Ariel E. Bayá, Pablo M. Granitto,

Abstract— Clustering validation indexes are intended to assess the goodness of clustering results. Many methods used to estimate the number of clusters rely on a validation index as a key element to find the correct answer. This paper presents a new validation index based on graph concepts, which has been designed to find arbitrary shaped clusters by exploiting the spatial layout of the patterns and their clustering label. This new clustering index is combined with a solid statistical detection framework, the Gap Statistic. The resulting method is able to find the right number of arbitrary shaped clusters in diverse situations, as we show with examples where this information is available. A comparison with several relevant validation methods is carried out using artificial and gene expression datasets. The results are very encouraging, showing that the underlying structure in the data can be more accurately detected with the new clustering index. Our gene expression data results also indicate that this new index is stable under perturbation of the input data.

Index Terms—Validation index, Clustering, Genomic Data.

# I. INTRODUCTION

: UNSUPERVISED data analysis is a valuable tool that helps the understanding of genomic data, one salient example are clustering methods. This technique has been widely used in the analysis of gene expression data and proteins [1]-[7] to help uncover the underlying structure in the data. Some authors [8]-[10] use the concept of "natural grouping" to describe the same idea, which refers to a group of patterns forming clusters described by a hidden definition, i.e. clusters formed by a rule unknown to the analyst. Clustering algorithms do not always uncover the natural groups and, as a general rule, results should be analysed afterwards to verify the quality of the clusters found. Handl et al. [11] and Halkidi et al. [12] discuss the use of a verifying step at the end of the clustering analysis to validate results. The validation of the results can be done by applying a similarity function that evaluates how well the clusters found by a clustering algorithm fit a definition of "natural groups". This definition is then central to a validation method since it will state what represents a meaningful cluster. At the end, this procedure will inform if a clustering result is good in the sense of a given clustering property, like connectivity, density or stability of the solution. Once the "natural groups" are defined it is possible to use the previous similarity function or clustering validation index (CVI) to find the most likely number of clusters for a given clustering algorithm.

: This work introduces a new validation measure that can detect arbitrary shaped clusters. In order to achieve this, the new

Manuscript received

CVI uses the minimum spanning tree (MST) of the data. The only assumption made about clusters, to assure their detection, is that the maximum first-neighbor distance should be smaller than clusters separation.

: The results presented here include a number of simulations that show the response of many well known validation methods when the clusters in the datasets deviate from the within sum of squares rule. For example, when clusters in the data do not have spherical shape or when a group of spherical clusters present a particular layout that do not follow the previous rule. From an application point of view one may need to validate solutions from a clustering task with a few examples in very high dimensional spaces. This is the case when grouping patients, tissues or subtypes of a disease, but not when grouping genes. In this kind of task it is possible that the shape of the clusters associated to the biological classes of the data are not globular in nature. A biased validation method only looking for a particular type of clusters, for instance spherical clusters, may not be able to point as valid a solution made by non globular clusters. The main contribution of this work is a new flexible clustering validation index based on a new distance measure that can find arbitrary shaped clusters.

: This paper is divided in 5 parts, Section 2 discusses previous relevant work done in clustering validation. This section also introduces the clustering validation methods used in Section 4. Section 3 presents the new Cluster Validation Index, named average intra-cluster distance (IC-av). This includes a brief analysis of how the new quality measure improves greatly the detection threshold of the Gap Statistic [13]. It also presents the limit point to detect clusters. Later in section 4, the new combination of validation index with the Gap Statistic is used on a number of artificial and real gene expression data. The results obtained by using the new validation index are compared with other existing methods and also with the golden rule, i.e. true number of clusters. The last part of this work, Section 5, presents some conclusions.

## II. RELATED WORK

: This section is not a comprehensive review of all the work done on clustering validation. The papers discussed in this section are intended to give a context to this work. Also the description of each method only provides the reader with general idea of the method, their main advantages and limitations.

: There are validation methods that include expert knowledge. Datta and Datta [14] and Stegmayer et al. [15], are examples of methods that include this kind of information in the validation process. The validation of the clustering process is based on the inclusion of biological information related to the problem. The use of this kind of information helps to define "natural groups" and thus it helps to find meaningful clusters in problems where that knowledge is valid. However, this type of methods can be too focused on the problem at hand, which can finally lead to an analysis that is only valid for a unique set of data. De

Ariel Baya (baya@cifasis-conicet.gov.ar) and Pablo Granitto (granitto@cifasis-conicet.gov.ar) are with the CIFASIS, French Argentine International Center for Information and Systems Sciences, UPCAM (France) / UNR-CONICET (Argentina), Bv 27 de Febrero 210 Bis, 2000 Rosario, República Argentina.

Villiers et al. [16] used a hand-crafted method to validate their taxonomic classification of papillomaviruses. Later, Bernard et al. [17] revisited this same problem including some new samples of papillomaviruses. They also modified the previous method to improve the support given to the new classification. Although validation methods that do not make use of expert knowledge cannot fall in this kind of trap, they may sometimes overlook some meaningful clusters or may be more complex or time consuming than their counterparts.

: On the other hand, most general purpose clustering validation methods can be grouped by the type of output they provide. Graphical validation methods rely on different working principles but the nature of their output demands that the analyst selects a solution. A clear example of this kind of methods is the work by Yeung et al. [18]. The authors proposed FOM, a method that assumes that clusters can be validated by an unseen condition. To study the likeliness of a *c*-cluster model solution, the authors analyse the average response on unseen features. The unseen features are used in a validation step in order to provide an estimate of the error of the clustering solution on these conditions. As output the method draws a response curve, from which the user must select the most appropriate number of clusters. Using a different working principle, Ben-Hur et al. [10] proposed a method that does not depend of cluster shape. The authors link the notion of solution stability to the idea of "natural groups" by looking for stable structures under perturbation of the data to determine the most likely number of clusters of a clustering solution. Lange et al. [8] devised a validation method able to detect the number of arbitrary shaped clusters by using an idea called transference. This concept involves training a classifier that learns the structure found by a clustering algorithm, then "natural groups" arise as structures that can be generalized by the classifier. This last method has a similar approach to Clest [19], but Lange et al. showed that the performance of their method is not dependent of parameter setting as in Clest. Tibshirani et al. developed a method for clustering validation called Prediction Strength [20] which is based on applying two rounds of clustering, first to a training set and then to a test set. The resulting train and test labels are used to create a co-membership matrix, (also known as evidence accumulation matrix [21] or consensus matrix [22]) which is used by the authors to circumvent the problem of labels assignment between train and test sets. The prediction strength index is then calculated over the co-membership matrix. This measure represents the viability of the hypothesis that the data has c clusters. As pointed out by Lange et al. [8], a big disadvantage of both Prediction Strength and Clest is that they need to fix the sizes of test and train sets. Estimating the size of partition is not trivial and represents a source of variability that influences the stability index. Monti et al. developed a method named Consensus Clustering [22]. This validation procedure is based on many clustering rounds, i.e clustering ensembles, which allows the user to visualize the different clustering solutions in order to inspect its stability. This method uses an automatic ad hoc rule based on the difference in the area between successive cumulative density function defined by the clustering solution on the consensus matrix. The graphical output based on this rule is used to select the number of clusters. The authors showed that their results are comparable to the ones of the Gap Statistic [13]. However, in some examples the results of Monti et al. can be difficult to understand since the solution may not be unique. To analyse the reasons of multiple clustering solutions one would need to analyse consensus matrices or the cumulative density functions. In general, graphical output requires expert knowledge to interpret results and find an answer, even in the cases of automatic methods like Prediction Strength and Consensus Clustering.

: As a contrast to the methods described above, there is a group of validation methods that automatically detect the number of clusters [23]. These procedures look for the number of clusters, according to a given hypothesis, that best fit the data being analysed. To perform this task, they rely on a CVI that establishes the quality of the solution. There is a group of classical methods which use the clustering labels and spatial position or the relation between patterns that fall in this category: the Gap statistic [13], the methods developed by Calinski and Harabsz [24] (CH), Krzanowski and Lai [25] (KL), Hartigan [26] (H) and the Silhouette index [27]. All of these procedures rest on the hypothesis that clusters have spherical shapes. Each index presents a different formulation and thus they may detect different spatial configurations of clusters. The Gap statistic, KL and H use the within cluster sum of squares (WSS) as CVI, while CH and the silhouette use both WSS and the between cluster sum of squares (BSS). The variations among these methods account for different performances under diverse scenarios. Still, all the previous validation methods are biased towards the same form of cluster and tend to overlook groups of patterns with a different geometry. In the case of the Gap Statistic one possible way to overcome the shape bias could be using the Kernel trick. A kernel Gap Statistic method would require to change the within clusters sum of squares by a kernel to detect non globular clusters. Dhillon et al. [28] propose an interesting weighted kernel kmeans algorithm, which is also linked by the authors to spectral clustering [29]. The optimization rule that the authors describe for kernel k-means could be also used for a Kernel Gap Statistic method. Merging the optimization rule from Dhillon et al. with the Gap Statistic could be useful to find non globular clusters, although this modification comes with the challenge to set extra parameters. For instance, a Gaussian kernel needs an associated bandwidth. Any extra parameter has to be set by an unsupervised rule and its value has a direct impact on the findings of the kernel Gap method. Different values for these parameters may produce distinctive kernel gap evolutions pointing to diverse solutions.

: The key feature of these methods is that they have a simple automatic rule that informs its user the number of clusters detected. This quality is most important for users with little or no clustering knowledge.

: This paper explores the use of a MST to find clusters with arbitrary shapes. The idea of MST for clustering is not new and it can be tracked to Hierarchical Clustering (HC) using single linkage method [30] and most recently to Xu et al. [31] and Zhong et al. [32]. There are also some works done using this idea as a validation technique. Originally Smith and Jain [33] and later Jain et al. [34] used the Friedman-Rafsky test [35] combined with MST to test for data uniformly distributed in high dimensional spaces. Volkovich et al. [36] and Barzily et al. [37] pursued a similar idea than Smith and Jain but looking for cluster structure instead. Fischer et al. also explored the idea of characterizing the distance between patterns using the maximum edge length in a graph. This resulted in a novel clustering algorithm named "Path Based Clustering" [38], [39] which proved to be useful for texture segmentation. First they presented a clustering algorithm [38] that uses a stochastic optimization technique (like Simulated Annealing or Deterministic Annealing) to solve the data partitioning problem. A second version of the algorithm uses a Hierarchical Algorithm to partition the data [39]. In both cases the partitions are evaluated by a cost function that uses the maximum edge length concept. The work presented here uses instead a MST to automatically evaluate every partition of the data and finally obtain the most likely number of clusters.

: In the validation literature there are many avenues of research that aim to solve relevant validation problems. For instance, Datta and Datta [40] formulated three validation strategies that they used to analyse the performance of a group of clustering algorithms. Later, Datta and Datta [14] developed two measures based on stability and biological properties of the clusters and performed a new analysis of clustering algorithms. In both cases, based on their results, the authors conclude that there is no single "good" algorithm to perform clustering. As a result from their work the authors produced a set of guidelines to analyse clustering algorithms and select the most suited for a clustering task. Extending these previous work, Pihur et al. [41] devised an automatic method which makes use of a set of validation indexes to rank a group of clustering algorithms for a given clustering task. This new method, which automatically selects the best clustering algorithm by simultaneously testing multiple validation methods, can be more beneficial to the user than a single test method. The present work has a narrower focus, aiming at solving some deficiencies of previous validation methods. We limit ourselves to the comparison of single validation methods.

# III. A NEW VALIDATION INDEX: THE AVERAGE INTRA-CLUSTER DISTANCE

: The set  $X \in \mathbb{R}^d$  consists of N data points  $\{x_1, x_2, ..., x_N\}$ , where each point  $x_i$  is described by d features  $x_{ij}$  (j = 1, 2, ..., d). The set  $\mathcal{L}$  is a clustering solution, a set of labels that divides X in c clusters  $\{L_1, L_2, ..., L_c\}$ . Both sets X and  $\mathcal{L}$  can be used as arguments of a function that measures the quality of the solution  $\mathcal{L}$  in X. A common example of this is the WSS. This measure estimates clusters tightness by assuming that clusters are spherical. The WSS is defined as:

WSS = 
$$\sum_{r=1}^{c} \frac{1}{n_r} \sum_{i,k \in L_r} d^2(x_i, x_k),$$
 (1)

where d(.,.) is commonly the Euclidean distance, but it can be any distance, and  $n_r$  is the cardinality of cluster  $L_r$ .

: The index presented in this work also estimates cluster tightness, but instead of assuming spherical shape, it assumes that clusters are connected structures with arbitrary shape.

: Patterns from dataset X are used to build an undirected complete graph G(V, E) where  $x_i \equiv v_i$  and edge  $e_{ik}$  corresponds to the pairwise distance between vertex  $v_i$  and  $v_k$ . The idea followed by the new validation index is to use local relations between points to evaluate the global tightness between the clusters formed in X. The most local relation between points is the one of first neighbors and the connected graph that better follows this concept is the minimum spanning tree. A  $MST(V, E^t)$ , where  $E^t \subset E$ , uses local information to form a graph that joins all vertex. This structure restricts the original Euclidean space to the paths formed by the edges of the MST. From the set of edges  $E^t$ , it is possible to derive a pairwise distance to detect non spherical groups of clusters more accurately. In this work, the pairwise distance between vertex is defined by the longest edge in the path joining a pair of points. Formalizing,  $P_{ik} = G'(V_p, E_p^t)$  is a sub-graph from  $MST(V, E^t)$  where  $V_p \subset V$  and  $E_p^t \subset E^t$  are the subsets of vertex and edges that form the path between vertex  $v_i$  and  $v_k$  where  $v_i, v_k \in V_p$ , i.e.  $v_i$  and  $v_k$  are part of the path. Using the previous notation a new distance named Maximum Edge Distance (MED) is defined as:

$$d^{MED}(v_i, v_k) = d^{MED}_{ik} = \{ E^t_p \in P_{ik} / max(E^t_p) \}, \quad (2)$$

which represents the longest edge in a path  $P_{ik}$ . This distance is symmetric,  $d_{ik}^{MED} = d_{ki}^{MED}$ ; always positive  $d_{ik}^{MED} \ge 0$ ; satisfies identity,  $d_{ii}^{MED} = 0$  and also the triangle inequality (which is analysed in the Additional File 1).

: The intra-cluster tightness among the members of a partition can be defined as the average of the pairwise MED distance among those members. This leads to the definition of the average intra-cluster gap (IC-av):

IC-av = 
$$\sum_{r=1}^{c} \frac{1}{n_r} \sum_{i,k \in L_r} d_{MED}^2(x_i, x_k).$$
 (3)

: Equation 3 and Equation 1 are the same, they only differ in the averaged metric. While the WSS has a simple interpretation, it is the squared distance towards the clusters center, the IC-av represents the average sum of the maximum edges between all pairs defined along the MST. This measure can be interpreted as an estimation of the tightness of the points in each cluster. It is a measure that considers local relations and that makes no assumptions about clusters shape. Thus, it can be used to detect the correct number of clusters in many arbitrary shaped and globular clustering configurations. To actually detect the number of clusters the new metric is combined with the Gap Statistic [13]. This method was proven useful by the authors to detect spherical shaped clusters. The use of the new IC-av will improve the previous method so it can detect groups of patterns with different shapes.

# A. Detecting arbitrary shaped clusters

: This subsection describes the detection mechanism behind the new IC-av validation index. For this purpose an abstract example composed by a group of  $c^*$  arbitrary shaped clusters is described step by step. A similar analysis was provided by Tibshirani et al. [13] when they explained the Gap Statistic working principle. Here the analysis is presented from the point of view of the IC-av index instead of the WSS. This will hopefully help to better understand the new metric and validation index. The only restriction placed to the clusters of the example is that the minimum separation between them has to be at least the maximum first neighbor distance (this restriction is related to the use of the MST). Subsection III-B discusses this restriction by analyzing a group of experiments based on simple artificial problems.

: In this work the Gap Statistic is combined with the ICav index as a way to detect the number of arbitrary shaped clusters present in the data. In this method the number of clusters c increases from 1 to  $c_{max}$ , where  $c_{max}$  is greater than  $c^*$  $(c_{max} > c^*)$  and  $c^*$  represents the true number of clusters that the method is trying to detect. The average sum given by Eq. 3 measures the changes in the clustering solutions as the number of partitions c varies from 1 to  $c_{max}$ . In the interval where  $c < c^*$ , an arbitrary pair of points i and k can be clustered together though in fact they may belong to different structures. For instance, assume that a clustering solution in the example from Figure 1.a merges the red and black ellipses as a single cluster. Two arbitrary points *i* and *k* from the same cluster can actually belong to different ellipses. Then any point i from the red ellipse is at a maximum distance to any other point k from the black ellipse in the sense of the MED distance. In this case the IC-av has a value that is dominated by the large edge value separating the red and black ellipses. Since clusters are well separated in terms of the maximum first neighbor distance when c increases large edges can be cut and IC-av values will be reduced. When  $c = c^*$ , assume that a new clustering solution maps each ellipse to a cluster. As a result any pair of points i and k belong to the same ellipse. In this case IC-av is the sum of each ellipse intra-cluster MED distance, intra-cluster distances are much smaller than distances between clusters. As result, the value  $c = c^*$  produces an alteration in the IC-av evolution which is marked by a change in the slope of the curve, i.e. an elbow point is formed at that value of c. As c keeps increasing,  $c > c^*$ , the IC-av value decreases monotonically at a very slow rate compared to the interval where  $c < c^*$ . The reason is that there are no more large edges to cut, clusters separation becomes smaller since compact structures are being broken to form new spurious clusters. It is then simple to see that structures are more difficult to detect when they are not well separated. The breaking point is reached when the separation between clusters becomes smaller than the maximum first neighbor distance.

: Additional file 6 includes figures showing multidimensional scaling (MDS) projections of some relevant artificial examples. These examples are intended to clarify Equation 2 and Equation 3, which define the MED distance and the new ICav validation index respectively. The MDS projections of the Maximum Edge Distance illustrate the transformation undergone by the data when the MED distance is applied to them. These examples show that the clusters are represented as tight and well separated groups when they meet a minimum separation.

: Panel a of Figure 1 presents an artificial example dataset of three elongated clusters with Gaussian distribution and different densities. Later on, this example will be further analysed with other methods but at the moment it is presented to illustrate the detection process described before. The black curve in Figure 1.d, shows the evolution of the IC-av index as a function of the number of clusters c, where c = 1, 2, ..., 10. This curve follows the behavior explained above and when c = 3, it is easy to see that there is a change in the slope. This change indicates the number of clusters present in the dataset shown by Figure 1.a. Figure 1.b is an MDS projection of the MED distance using the first two components. The projection shows that most of the points of each cluster form a relative tight structure with exception of a few points from the green and black clusters that are the result of the dispersion of each cluster. The ripple in Figure 1.d in the range c = [4 - 10] can be explained by the many small intra-cluster separation in each cluster.

: The number of clusters in a dataset can be detected by contrasting a clustering solution against a null clustering hypothesis. Smith and Jain [33] studied data uniformity in a multidimensional space using the Friedman-Rafsky test combined with MST. Later, Tibshirani et al. [13], following a different approach, used a uniform distribution as null hypothesis to contrast against the

observed data. Both studies concur in the use of an uniform distribution as a proper null clustering hypothesis to represent the lack of structure. Also the method developed by Tibshirani et al. [13] explained how to contrast the null hypothesis against observed data to support the existence of a c-cluster solution. The change of the WSS for the new IC-av index in the Gap statistic allows to find groups of patterns with arbitrary shapes. The method was not able to find this type of clusters before because of the bias toward spherical clusters present in the WSS. The green curve in Figure 1.b shows the evolution of the ICav index as a function of the number of clusters c for the null hypothesis which corresponds to a uniform distribution of points in  $\mathcal{R}^2$ . The index shows monotonically decreasing values as c increases which reveals the relations between graph components in a null distribution, the slope in this case is constant. Ideally, the number of clusters is detected when there is a great reduction in the slope of the observed data which precedes an elbow point in the curve. The gap for the IC-av index and c clusters can be rewritten as:

$$G_{\text{IC-av}}(c) = \log(\text{IC-av}^{obs}) - \frac{1}{B} \sum_{b=1}^{B} \log(\text{IC-av}_{b}^{null}).$$
(4)

where *B* is the number of times the null distribution is repeated, IC-av<sup>*obs*</sup> is the index calculated using the clustering solutions from the observed data and IC-av<sup>*null*</sup> is the index calculated using the clustering solutions from the null data. The standard deviation of Equation 4 can be computed as:

$$sd_{\text{IC-av}}(c) = \sqrt{\left(\frac{1}{B}\sum_{b=1}^{B} (\log(\text{IC-av}^{null} - \bar{\mu}_{null}))^2\right)}.$$
 (5)

where  $\bar{\mu}_{null} = \frac{1}{B} \sum_{b=1}^{B} log(\text{IC-av}^{null})$  and the criterion for selecting the number of clusters is the minimum *c* satisfying:

$$G_{\text{IC-av}}(c) \ge G_{\text{IC-av}}(c+1) - sd_{\text{IC-av}}(c+1)\sqrt{(1+\frac{1}{B})}.$$
 (6)

: The term  $sd_{IC-av}(c+1)\sqrt{(1+\frac{1}{B})}$  is an empirical bound for the null hypothesis rejection introduced by Tibshirani et al. [13]. Equation 6 tells that the reduction in the index has to be bigger than what it is expected from partitioning a uniform distribution in order to accept that the change in the index is due to the split of a real cluster. When the reduction is due in fact to the split of a "compact" structure, the change should be of the same order of what it is observed on the null hypothesis. Figure 1.d shows the values calculated using Equation 4 (black curve) where the error bars were estimated as  $sd_{IC-av}(c+1)\sqrt{(1+\frac{1}{B})}$ . The number of clusters indicated by Equation 6 is three, which concurs with data depicted in Figure 1.a.

# B. Breaking point for Clustering Detection

: In the beginning of this section it was mentioned that the only restriction placed to the clusters, in order to detect them, is that the minimum separation between them has to be at least the maximum first neighbor distance. This subsection presents a simple set of examples that explores the separation limit imposed to IC-av.



Fig. 1. Three clusters datasets with non-identity covariance matrix. Panel a) shows the data forming three clusters, each with a different color. Panel b) presents Multidimensional scaling projections using MED distance. The figure uses the first two components of MDS, x-axis corresponds to the first MDS component and y-axis to the second component. See Additional file 6 for more detail. Panel c) displays the evolution of the IC-av index when the number of clusters *c* varies from 1 to 10. The curve in black is the evolution of the observed data while the green curve is the evolution of the null distribution. The curve in panel d) is obtained by applying Equation 4 and the error bars are obtained by applying Equation 5 corrected by the term  $\sqrt{(1 + \frac{1}{B})}$ .

: The dataset from Figure 2.a depicts two clusters composed by a black and a red bar generated by two independent uniform distributions. The green line in the figure shows the minimum cluster separation between the two closest point from each cluster. IC-av is applied to different configurations of this dataset where the varying parameter is the minimum cluster separation between the two closest points. This distance was fixed to a set of fractional values of the 1-knn maximum distance to analyse the response of the IC-av validation method as a function of the clusters separation.

: For these simulations the following values: { 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3, 1.5, 2.0} were used as fractions of the 1-knn maximum distance. Each setting is evaluated 50 times by accumulating the outputs of GAP + IC-av and dividing the values by 50 (the number of experiments), which is a hit/miss ratio. As a result of these experiments only two values were obtained as output: one and two clusters. Figure 2.b presents the results in the form of two curves where the x-axis shows the clusters separation and the y-axis shows the mean value of the hit ratio for one and two clusters. Figure 2.b has three gray guiding lines, two horizontal guides at ratios 0 and 1 and one vertical guide at the 1-knn maximum distance separation. The figure clearly shows that when separation increases the detection of the two clusters improves, having a limit point at the intersection of ratio 1 and the 1-knn maximum distance separation. All examples were clustered using Evidence Accumulation combined with average linkage hierarchical clustering (EAC-av) [21] (the results of the algorithm were verified in order to ensure that the algorithm always detected correctly both clusters). These results show that for separation values below the 1-knn maximum distance, where EAC-av still



Fig. 2. The Two bars Dataset is used to test the minimum separation between cluster where IC-av fails to detect two clusters. a) The Two bars Dataset: the green line joins the closest points from the black and red cluster. This line represents the clusters separation expressed as fraction of the maximum first nearest neighbor (1-knn) distance. b) This figure shows the detection ratios for one and two clusters as a function of the clusters separation. The y-axis scale is set as the ratio of the number of times 1 cluster (red curve) or 2 clusters (black curve) were detected divided by the total number of experiments. Each batch of experiments is performed 11 times, one for a different cluster separation. The x-axis represents the clusters separation as a fraction of the 1-knn maximum. These fractions are  $\{0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3, 1.5, 2.0\}$ .

finds the two clusters, the performance of IC-av begins to degrade. In Additional File 6, Figures 13.a-c correspond to an example of Two Bars with big separation between clusters while Figures 13.d-f show the Two Bars data with a small separation. The projections using multidimensional scaling of the MED distance (Figures 13.c and Figures 13.d) present a good example on the importance of cluster separation for MED and therefore for IC-av. In the example corresponding to Figures 13.d-f it is not possible to model clusters as tight structures with MED. As a result IC-av is not able to find the Two Bars.

# IV. SIMULATIONS AND RESULTS

#### A. Artificial Data

: This section presents a number of challenging artificial datasets. These problems are used to compare and contrast a group of clustering validation methods against the new IC-av index. The goal of this comparison is to show the differences between the indexes and to exhibit how the new index can solve some of the shortcomings of the competing validation methods. The analysis presented here compares the IC-av index against classical automatic methods, i.e. methods that use both spatial and label information to find the number of clusters by means of an automatic rule.

: The simulations presented here have the following layout: for each artificial dataset 100 repetitions were created. A repetition consisted in generating the dataset with a different seed each time, the data was then clustered and the clustering solution was fed to different automatic validation methods used to estimate the number of clusters. In all the cases of this subsection the clustering algorithms used a range  $c = \{1 - 10\}$ . As a result of this process, a distribution of values estimating the number of clusters was obtained from each validation method. To present the results three values representing each distribution were selected: the  $10^{th}$  percentile, the  $50^{th}$  percentile or median value and the  $90^{th}$  percentile. The median value was used to decide on the number of clusters that a validation method estimated to be true for a set of clustering solutions. This value is robust to outliers and allows to describe the distribution center when there is no prior knowledge to make any assumptions about the resulting distribution. The  $10^{th}$  and  $90^{th}$  percentile were used as estimators of the distribution dispersion. These parameters were selected following the same idea that lead to the choice of the media. The use of the standard deviation was discarded to estimate the dispersion since it is not possible to make assumptions regarding the normality or symmetry of the distribution describing the number of clusters. Results from this subsection and from the following one show that the distributions tend to be highly asymmetrical having sometimes upper or lower tails. The choice of the  $10^{th}$  and  $90^{th}$  percentile allowed to properly display the variability of unbalanced distributions, also this two values and the median make results more robust to outliers.

: The first example included here is a dataset that consists of 200 sample points uniformly distributed in a 10 dimensional cube where each side of the cube has one unit of length. This set of data has no meaningful clustering structure. The new validation index is applied in this case to verify if it can detect the lack of structure represented by an uniform distribution of patterns. The following artificial dataset, depicted in Figure 3.a, consists of three Gaussian clouds drawn from Normal distributions with identity covariance matrix. This Three Gaussian clusters example is presented to show that IC-av can also detect clusters with spherical shape. The covariance matrix of the previous dataset was changed to transform spheres into ellipses, (Figure 1.a) producing the Three ellipses dataset. This last set is formed by three clusters with a different number of points: 80 (black cluster), 125 (red cluster) and 70 (green cluster). Also the covariance matrices used to create each cloud are different. The Three ellipses dataset was then extended to ten dimensions, each cloud in the new 10D Ellipses dataset has 150 points sampled from a diagonal covariance matrix with its diagonal given by the clusters the constant  $\sqrt{10} \times 2.5 \alpha_{10}^{diag}$  was added to the 9<sup>th</sup> component of two of the clouds. Finally the data was rotated by a random 10 by 10 base which correlated all variables.

: The two final problems presented in this section are not solvable by the WSS rule. The Three ring dataset consists of three concentric rings of radii:  $r_3 > r_2 > r_1$ , where points corresponding to radius  $r_1$  form a single cluster and points corresponding to  $r_2$  and  $r_3$  form two different clusters each. The inner ring cluster has 75 points and both outer rings clusters  $r_2$  and  $r_3$  have 250 points each. Figure 3.b shows this dataset. The last example is the Fractal dataset, depicted by Figure 3.c, which has seven clusters with recurrent structure. Each one of the clusters is sampled from a Gaussian distribution with covariance matrix  $\alpha_i I$ . The first level of recurrence is formed by clusters { E, F, G}, the centers of these clusters form an equilateral triangle and the sum of the points in the three clouds is equal to the number of points in clusters C or D. The following level of recurrence is formed by the center of mass of clusters { E, F, G} and centers of clusters { C, D} which form an equilateral triangle. The sum of the points in clusters { C, D, E, F, G} is equal to the number of points in clusters A or B. Finally, the last level of the structure is formed by the center of mass of clusters { C, D, E, F, G} and centers of clusters { A, B} that together form an equilateral triangle. The variance of the seven clouds follow the relation:  $\alpha_{E,F,G} = \frac{1}{2} \alpha_{C,D} = \frac{1}{4} \alpha_{A,B}$  and the number of points in each 6



Fig. 3. Three different artificial datasets. Panel a) shows three spherical shaped clusters, panel b) presents five arbitrary shaped clusters formed by 3 concentric rings and panel c) shows a seven spherical shaped clusters problem with a recurrent geometric structure.

cloud is given by  $N_{E,F,G} = \frac{1}{3}N_{C,D} = \frac{1}{9}N_{A,B}$ .

: The six artificial datasets were clustered using five different algorithms. Columns one and two of Table I show which method was used to find clusters in each case. The selection of the five algorithms was based only on the ability of each method to find the correct number of clusters, except for the 10d Noise dataset where there is no structure to be found. In that example K-means was used as clustering algorithm. In the three clouds data, PAM [27] was used to find clusters. This algorithm is based on the same minimization rule as K-means but it uses a distance matrix as input instead of the data itself. Fractal and Three Rings datasets were also clustered with PAM but in both cases the algorithm was combined with the PKNNG metric [42]. This metric allows the algorithm to find high dimensional arbitrary shaped clusters that otherwise plain Euclidean distance would normally miss. The Three Ellipses dataset was clustered using Model Based Clustering [43] (MBC), which can easily find the covariance structure of three ellipses. Finally the 10D Ellipses data was clustered using Evidence Accumulation combined with average linkage hierarchical clustering (EAC-av) [21].

: Columns four through nine in Table I show the estimates of the number of clusters for each dataset using procedures based on different automatic validation methods. The silhouette index (Sil), KL index and CH index are not defined for a one cluster case, so none of them can evaluate correctly the structure of the 10D Noise dataset. The Fractal dataset, which is composed by 7 Gaussian clusters with diagonal correlation matrix of the form  $\alpha_i I$ , is a difficult example, not because of the shape of the clusters but because of their arrangement. This last example cannot be solved by an algorithm that minimizes the sum of squares. Given the particular layout of the clusters and their number of points the decreasing rate of the WSS is greatly reduced when c > 3. This phenomenon repeats again for c > 5 making yet more difficult to find all 7 clusters. Therefore it is expected that any method relying only on this rule like the Gap statistic, the KL index or the H statistic would not be able to find all seven clusters. On the other hand, the silhouette index detects the seven cluster structure because it considers the inter-cluster distance too. The CH index

Results for six artificial datasets clustered with different algorithms. The clustering algorithms are named in the second column. The third column accounts for the true number of clusters in each case,  $c^*$ . The following six columns show the distribution of the estimation on the number of clusters for different automatic methods, as discussed in the text.

TABLE I

Dataset	Algorithm	$c^*$	Н	KL	СН	Sil	Gap	k-Gap	IC
10D Noise	K-means	1	1:1:1	2:2:2	2:2:2	9:10:10	1:1:1	1:1:1	1:1:1
3 Clouds	PAM	3	3:3:3	2:2:2	3:3:3	3:3:3	3:3:3	3:3:3	3:3:3
3 Ellipses	MBC	3	3:4:5	3:7:9	3:5:10	2:2:5	1:2:2	3:3:3	3:3:3
3 Ellipses 10d	EAC	3	3:5:8	4:7:9	9:10:10	2:7:9	1:1:1	3:3:3	3:3:3
3 Rings	PAM-PKNNG	5	1:1:1	7:7:9	10:10:10	10:10:10	1:1:1	1:1:1	5:5:5
Fractal	PAM-PKNNG	7	5:5:5	2:2:2	5:5:5	7:7:7	3:3:3	7:7:7	7:7:7

also combines the inter and intra cluster distances, but it includes a (c-1) term dividing the inter-cluster distance which penalizes solutions with many clusters. Results with both Ellipses datasets exhibit that the competing methods have severe problems when Gaussian data has a correlation matrix different than  $\alpha I$  unless they are very far apart. The last example, Three Rings, is a set of non cloud-shaped clusters separated by a big gap, compared to the inter point distance within each cluster. In this case all the other methods are expected to fail since they are biased to the globular cluster shape.

: The kernel Gap method (k-Gap) was left out from the previous paragraph since it deserves some special attention due to its complexity. In order to implement k-Gap the paper by Dhillon et al. [28] was followed. More precisely, Equations 1 and 2 which describe the optimization rule of kernel k-means. In all k-Gap experiments the Gaussian kernel was used. One of the most troublesome points was to select a value for the kernel width ( $\sigma$ ) since all the results of k-Gap depend on this value. Given that one is looking for tight arbitrary shaped structures, the kernel width  $(\sigma)$  has to be set to a value that captures the local information of each data point but that also makes possible to find the global clustering structure. As a general rule for all the experiments from this section sigma ( $\sigma$ ) was set to the 5th percentile of the distribution of the pairwise distances. The results from Table I show that k-Gap detected most of the clustering structures except in the 3 Rings dataset. A more thorough analysis (the results of these analysis are not presented in this work ) which included a hand tuning of  $\sigma$  for each dataset showed that it was possible to find better validation results for the 3 Rings Dataset too. The major limitation for k-gap is the setting of  $\sigma$  since each value selected for this parameter results in a new kernel GAP curve. A second limitation, which is discussed in the Spectral Clustering community (see Zelnik-Manor et al. [44], Azran et al. [45] and Nadler et al. [46]) is related to varying scales or clusters densities that limit the ability of the Gaussian Kernel to describe the data. This in turn leads Spectral Clustering to bad clustering solutions. For these cases kernel gap may also experience problems in combination with a Gaussian Kernel because of the direct relation to Spectral Clustering. Examples of this kind are not included here since they exceed the scope of this work.

: In all five datasets with cluster structure from Table I, the selected clustering algorithms always find the clusters that match the classes of the data. Yet most of the validation methods do not validate these results because they are biased to a fixed shape of clusters, i.e spherical ones. In the case of the 10D Noise dataset any clustering algorithm can be used because there is no structure in the data. All results from Table I show that the IC-av not

only estimates the correct number of clusters but it consistently outperforms all other competing methods. These results using artificial data uncover some of the shortcomings that have the competing automatic validation methods. In the case of the Three Clouds data, the result emphasizes that the new index can detect spherical shaped data if clusters are adequately separated. Both ellipses datasets are used to point out that classic methods from Table I cannot solve correlation structures nor complicated layouts like the one in the Fractal dataset even if they are made by spherical clouds. The last dataset, Three Rings data, is used to show the performance of the new index when clusters are non Gaussian. One important fact illustrated by Table I is that the new validation measure does not present any of the deficiencies of previous methods on these datasets.

: Additional file 7 and 8 include five tables similar to Table I. Additional file 7 presents a set of simulations where K-means is always used as clustering algorithm. The validation of the four remaining clustering algorithms are presented in Additional file 8. Results were divided for the sake of clarity since the clustering solutions from K-means need more effort to be interpreted. Both files help to give a more complete vision of the Gap + IC-av. These tables show validation results when some of the clusters or none of them are found by the clustering algorithm, i.e. these tables include some results where the clustering algorithm cannot discover the clusters formed by the geometric figures of the datasets

: Regarding non automatic or graphical methods there are at least two of them mentioned in this work, Model Explorer (ME) and Cluster Stability (CS), that can also find the correct number of clusters for the five examples with structure. Lange et al. presented a similar example, Three Rings, when they tested CS with artificial data and obtained also promising results with arbitrary shaped cluster datasets. Full tests with ME and CS were not included because the aim here is to compare only with automatic methods of similar characteristics, i.e. methods using both spatial and clustering label information. The next section presents an analysis on real data which includes both graphical and automated methods.

# B. Real Data

: The purpose of this subsection is to explore if the new index can also find the number of clusters in real gene expression datasets. With that goal in mind, six public available gene expression datasets were tested. The data was first clustered with three different methods and then the solution of each clustering algorithm was evaluated with seven different validation methods.

#### TABLE II

MAIN FEATURES OF GENE EXPRESSION DATASETS. SIX GENE EXPRESSION DATASETS DESCRIBED BY FOUR MAIN FEATURES: TYPE OF THE MICROARRAY CHIP TECHNOLOGY (CDNA OR OLIGONUCLEOTIDE), NUMBER OF SAMPLES (N), NUMBER OF CLASSES AND NUMBER OF EXAMPLES PER CLASS (CLASSES), AND NUMBER OF GENES OR CONDITION IN THE CASE OF YEAST DATASET (P).

Dataset	Туре	n	Classes	р
Thyroid [48] (THY)	Oligo	18	2 (9-9)	2000
AML-ALL [2] (ALB)	Oligo	38	3 (11-8-19)	1000
Lung [52]	Oligo	197	4 (139-17-21-20)	1000
Multi-A [49]	Oligo	103	4 (26-26-28-23)	1000
Lymphoma [50] (ALI)	cDNA	62	3 (42-9-11)	1000
Yeast [1] (Y)	cDNA	208	4 (41-121-35-11)	79

Both graphical and automated methods were included in this section. In all the cases of this section samples are clustered (not genes). Then the number of clusters in each solution is validated and later contrasted against the known number of classes of the data, in order to evaluate the accuracy and stability of each validation method.

: The use of clustering algorithms based on different working principles avoids any possible bias towards a single method. For that matter, the following algorithms were included in this section: Evidence Accumulation combined with average linkage hierarchical clustering (EAC-av) [21], PKNNG metric [42] combined with average linkage hierarchical clustering (PKNNGav) and Spectral Clustering [47] using a Gaussian Kernel. This three algorithms have different working principles and have been previously introduced in the literature and proved to work efficiently in gene expression data clustering [42].

: To analyse the validation methods a group of six gene expression datasets both from oligonucleotides and cDNA technology were selected. Table II describes the main attributes of these datasets. For the Lung Tumors dataset the version by Monti and Tamayo [22] was used. For ALB [2], THY [48] and Multi-A [49] datasets the procedure described by Monti and Tamayo [22] was applied to the original versions of the data to select a number of discriminant genes. All four datasets were normalized by genes to zero mean and unitary standard deviation. In the case of the Lymphoma dataset [50] (ALI), the top 1000 genes with highest standard deviation were selected from the version of De Souto et al. [4]. Finally, the last example presented is the Yeast dataset (Y), which comes from an experiment performed by Eisen et al. [1] conducted to study the response of this organism to different stimulus or stress conditions. From this data Brown et al. [51] selected five functional groups to determine if they could be learnt by an SVM. This last version of selected genes compiled by Brown et al. was used for clustering and clustering validation.

: Previous to the validation analysis used to detect the number of clusters, clustering solutions of each dataset were contrasted against the golden rule, i.e. the original class labels. Figure 2 in the Additional File 2 shows the average corrected Rand index [53] (cRand or ARI) obtained after running 100 data sub-samples for all datasets and all clustering algorithms. This procedure is explained in the paragraph below. In all cases the cRand peaks points to the number of classes of each dataset, which means that given each set of examples the different clustering algorithms tend to gather these examples by their biological classes as the more likely solution. This suggests that the biological classes can be thought as natural structures. Given the existence of these groups one could expect that a good validation method could point to the existence of these clusters linked to the biological classes. The sub-sampling process can also show how the different validation methods respond to the variation of the input data making it an interesting parameter of quality to evaluate these methods.

: A byproduct of the previous analysis of natural structures is solution invariance of the validation method. This considers the behavior of the final validation output when the input data suffers from minor variations or perturbation. The goal of this test is to evaluate if small variations in the input data modify the solution found by the clustering validity method. Perturbation of the input data can be achieved by sub-sampling or by adding noise to the data. Then, to perform an invariance analysis, the output (in this case, the number of clusters) is monitored over a number of iterations. In this work a sub-sampling scheme is adopted. All simulations made with automatic methods were performed 100 times, each one using a 95% sub-sample of the original data. Also, to eliminate any possible differences in the results due to the sampling process, each dataset was clustered with the same 100 sub-samples, which were then analysed with a validation method to find the number of clusters. Every sub-sampling iteration is then composed by first applying a clustering method and then by calculating the number of clusters of that solution. The end result of this process is a distribution describing the number of clusters for each algorithm and validity method. Table III uses the same notation as Table I from the artificial data subsection, where the estimated value of the number of clusters was informed by the median value of the distribution and the dispersion was described by the  $10^{th}$  and  $90^{th}$  percentiles. In all cases from Table III the clustering algorithms looked for solutions in the range c = [1 - 10]. In all examples from Table IV the range was the same with exception of the THY dataset for which the range had to be restricted due to artifacts in the solutions of FOM, CS an ME. In the case of CS the range of clustering solutions was set to c = [1 - 7] and for ME and FOM it was set to c = [1 - 5].

: Table III displays the results of the number of clusters estimated by each automatic method. Results are underlined when the estimation made by the validation method matches the correct number of classes. The KL index worked in two out of six cases for each of the clustering algorithms. It should be noted that in half of those cases the results showed high variability. The CH index also had a poor performance; it only showed good results for the THY dataset. The Silhouette Index only estimated correctly one case for Eac-av and PKNNG-av and two cases for Spectral Clustering algorithm. On the other hand, the Gap statistic obtained very good results with Eac-av but not with the other two algorithms, where it only estimated the correct number of clusters in one case. Though the previous five validation methods look for spherical shaped groups, the differences in their mathematical formulation make them consider different structures which explains why their estimation vary so much between one another. The k-Gap method exhibited a good performance, slightly better than the Gap but clearly below ICav. The bandwidth of the kernel was set in each case to the value selected for the Spectral Clustering kernel. Even if those bandwidth values showed a good performance for clustering, the method has difficulties finding the right number of clusters, which again points out the drawback of k-Gap. The estimation made by

#### TABLE III

VALIDATION RESULTS FOR SIX GENE EXPRESSION DATASETS CLUSTERED WITH THREE ALGORITHMS: EAC-AV, PKNNG-AV AND SPECTRAL CLUSTERING WITH GAUSSIAN KERNEL . FROM LEFT TO RIGHT, THE FIRST TWO COLUMNS INDICATE THE NAME OF THE DATASET AND THE TRUE NUMBER OF CLUSTERS. THE FOLLOWING SIX COLUMNS CORRESPOND TO AN AUTOMATED VALIDITY METHOD USED TO ESTIMATE THE NUMBER OF CLUSTERS. THE MEDIAN VALUE IS USED TO DECIDE THE NUMBER OF CLUSTER IN EACH CASE.

PKNNG-av								
Dataset	$c^*$	Н	KL	CH	Sil	Gap	k-Gap	IC
THY	2	1:1:1	3:4:4	2:2:2	3:4:4	1:1:2	1:1:1	1:2:3
ALB	3	1:1:1	6:7:8	2:2:3	6:7:7	1:1:1	3: <b>3</b> :3	1:3:3
LUNG	4	1:1:1	3:4:6	2:2:2	2:2:2	2:3:4	3:3:3	3: <b>4</b> :4
Multi-A	4	1:1:1	4:4:4	2:2:2	4: <b>4</b> :4	4: <b>4</b> :4	4:5:5	3:4:4
ALI	3	1:1:1	2:2:2	2:2:2	2:2:2	2:2:3	2:2:2	1:3:3
Y	4	1:1:1	3:3:6	3:5:6	3:3:3	1:1:1	4:4:4	4: <b>4</b> :4
EAC-av								
Dataset	$c^*$	Н	KL	CH	Sil	Gap	k-Gap	IC
THY	2	1:1:1	3:3:4	<u>2:2:2</u>	3:3:4	1:1:3	<u>2:2:2</u>	<u>1:2:3</u>
ALB	3	1:1:1	6:6:7	2:2:3	6:6:7	<u>1:3:3</u>	<u>3:3:3</u>	<u>3:3:3</u>
LUNG	4	1:1:1	<u>4:4:4</u>	2:2:2	2:2:2	<u>4:4:4</u>	<u>3:4:4</u>	4:4:4
Multi-A	4	1:1:1	4:4:8	2:2:2	<u>4:4:4</u>	4: <b>4</b> :5	3:3:3	3:4:4
ALI	3	1:1:1	2:2:2	2:2:2	2:2:2	2:3:4	<u>3:3:4</u>	1: <b>3</b> :3
Y	4	2:2:2	2:5:9	2:2:2	3:3:3	<u>3:4:5</u>	<u>4:4:4</u>	<u>4:4</u> :4
Spectral Clustering								
Dataset	$c^*$	Н	KL	CH	Sil	Gap	k-Gap	IC
THY	2	1:1:1	2:2:2	2:2:8	2:10:10	2: <b>2</b> :2	2: <b>2</b> :2	2: <b>2</b> :2
ALB	3	1:1:1	2:3:8	2:2:3	<u>2:3:3</u>	1:2:3	3: <b>3</b> :3	2: <b>3</b> :3
LUNG	4	1:1:1	4:6:9	2:2:2	2:2:2	4:5:6	6:6:6	4:4:4
Multi-A	4	1:1:1	4:5:8	2:2:2	4:4:4	4:5:5	5:5:5	4: <b>4</b> :4
ALI	3	1:1:1	3: <b>3</b> :7	2:2:2	2:2:2	2:2:2	4:4:4	2:3:4
Y	4	1:1:1	2:2:2	2:2:2	3:3:3	5:6:6	<u>4:4:4</u>	4: <b>4</b> :4

the IC-av index agrees in all cases with the number of classes in the data and it only exhibits a moderate dispersion in three cases, ALI and ALB for PKNNG-av and ALI for Eac-av.

: As part of this work other non automated validation procedures were included. These methods have a graphical output which need human experts to determine their results. From this family three graphical clustering validation procedures, FOM, CS and ME, were selected. In the case of CS and ME, both methods include sampling procedures defined by their respective authors, while FOM does not use any data perturbation procedure. CS and ME have a sound formulation. ME is a validation method strictly based on stability and CS is specially careful in the treatment of sampling data. The division in train and test in the last method was set to a fixed value by the authors since this value can be source of instability if it is left as free parameter. A similar method, Clest, presented some instability in their results related to varying partition values of the train and test sets. Finally, both methods were used following the suggestions of their respective authors, since they provide an explanation for the setting of their parameters and show results supporting these values. On the other hand, FOM has no sampling procedure, for that reason the method was slightly modified to analyse the output under input perturbation. Similar to the automatic methods previously discussed, 100 iterations of FOM were performed using a 95% sub-sample of the data. Instead of the original FOM result of one point per cluster solution, with the new variation a distribution of values was obtained per cluster solution (one point per iteration). To provide an output similar to the original method, boxplots were used to show the distribution for each solution and a curve was obtained by joining the median values of the boxplots. Finally,

#### TABLE IV

RESULTS FROM CLUSTER STABILITY (CS), MODEL EXPLORER (ME) AND FOM USED TO ESTIMATE THE NUMBER OF CLUSTERS FOR SIX GENE EXPRESSION DATASETS CLUSTERED WITH THREE ALGORITHMS: EAC-AV, PKNNG-AV AND SPECTRAL CLUSTERING WITH GAUSSIAN KERNEL. NA INDICATES THAT THE RESULT WAS NOT AVAILABLE BECAUSE COMPUTING TIME WAS EXCESSIVE.

- 1	PKNNG-av							
	Dataset	$c^*$	CS	ME	FOM			
	THY	2	<u>2</u> , 3	<u>2</u> , 3, 4	<u>2</u> , 3			
	ALB	3	3	3	3			
	LUNG	4	2	2	4			
	Multi-A	4	3, <u>4</u> , 5	3, <u>4</u>	4			
	ALI	3	2	<u>3</u>	4			
	Y	4	3	2, 3	3			
	EAC-av							
	Dataset	$c^*$	CS	ME	FOM			
	THY	2	<u>2</u> , 3, 4	<u>2</u> , 3	NA			
	ALB	3	<u>3</u>	<u>3</u>	NA			
	LUNG	4	2	2	NA			
	Multi-A	4	3	3	NA			
	ALI	3	2	2, <u>3</u>	NA			
	Y	4	3, <u>4</u>	3, <u>4</u>	3			
	Spectral Clustering							
	Dataset	$c^*$	CS	ME	FOM			
	THY	2	7	2	2			
	ALB	3	<u>3,</u> 4, 5	2, <u>3</u>	<u>3</u> ,4			
	LUNG	4	<u>4</u> ,5	2	4			
	Multi-A	4	2	2, 3, 5	4			
	ALI	3	2	3	4			
	Y	4	2, 4	2, 3, 4	3			

the analysis of invariance of the method can be performed by inspecting the sizes of the individual boxplots and by analyzing the evolution of their size.

: The results from CS, ME and FOM are graphical in nature. The output of these three methods are shown in Figures 3 to 11 in the Additional File 3, 4 and 5. Table IV presents a clustering estimation based on the previous figures. Cluster Stability (CS) was set to use the same three clustering algorithms from Table III; EAC-av, PKNNG-av and Spectral Clustering for the first step, then for the second step each algorithm was combined with a knearest neighbor predictor (k-nnp) following Lange et al.

: Tables III and IV have similar layouts. Table IV is divided in three parts, one for each clustering algorithm. The columns of the table report the performance of each validation method. Results of CS are presented in the first column, in each case a clustering method is combined with a 2-nearest neighbor predictor or 1-nearest neighbor predictor if a tie occurs. These results were obtained from Figures 3 to 5 in the Additional File 3, which show the full output from CS. This method does not always find the right number of clusters but it is usually close to the optimal value. There are just a few examples where the method does not behave well: one is THY, with all three clustering algorithms, and the other one is ALB with Spectral Clustering. Results in these cases are too variable and therefore the number of clusters are not reliable. One possible reason for this variability could be the small number of examples in these datasets. CS divides the data in two sets, train and test, each with half of the examples and both THY and ALB may not have enough points to apply the method. Model explorer output was also included in the Additional File 4, Figures 6 to 8. The second column of Table IV, marked as ME, summarizes the results and presents an estimation of the number of clusters using the same convention as with CS. ME

Fig. 4. This figure presents a series of difficult examples for CS, ME and FOM validation methods. Figures 3 to 11 in the Additional Files 3, 4 and 5 present the full output of the three methods. These nine different examples are grouped by row, the ones corresponding to the first row were analysed with CS, the ones from the second row were analysed with ME and the last row examples were analysed with FOM. The following items describe the name of the dataset and the name of the clustering algorithm used: a) THY data clustered with EAC-av. b) Multi-A data clustered with PKNNG-av. c) ALB data clustered with Spectral Clustering. d) THY dataset clustered with PKNNG-av. e) ALI dataset clustered with Spectral Clustering. f) Y data clustered with EAC-av. g) THY data clustered with PKNNG-av. h) ALB data clustered with Spectral Clustering. i) ALB data clustered with PKNNG-av.



shows a better overall performance than CS. Also, ME does not have the same problem with THY as CS, probably because the sub-sampling is less severe.

: The adjusted figure of merit index (A-FOM) is a measure that informs the quality of a clustering solution based on how well that solution can explain an unseen condition previously removed from the dataset. Yeung et al. [18] defined the 2-norm FOM as the root mean square deviation in the left-out condition and they used the evolution of this figure for a varying number of clusters to estimate the most likely value. FOM index is defined as:  $FOM(k) = \sum_{e=1}^{m} FOM(e, k)$  where e is the left out condition and k is the number of clusters, then if FOM(k) is applied to each sub-sample one obtains a distribution of values as result. The adjusted FOM is corrected to consider the number of points in the dataset and the number of clusters in the solution. The last column from Table IV displays results of FOM. The number of clusters on each dataset was obtained considering the lowest median value and variability from the A-FOM index distribution. Due to the heavy computational burden of FOM, when it was combined with EAC-av, it was only possible to finish the analysis

for the Yeast dataset (Y). In the rest of the cases, FOM exceeded the available running time in our cluster. In those cases the results were informed as not available (NA). Comparing Tables III and IV one can see that IC-av performs much better than CS, ME and FOM.

# C. Variability in validation methods

: An analysis of the variability of the validation methods is difficult since outputs from them are different. While CS, ME and FOM consider the variability of the validity index itself within a curve, the IC-av index and the rest of the methods from Table III consider variability in terms of number of clusters. At every subsample iteration an automatic rule using the IC-av index finds the number of clusters while CS, ME and FOM use all iterations to find the answer. The variability of the validity indexes of the last three methods are extra parameters that also need to be accounted to find the number of clusters. It is clear then that there is no direct comparison between these methods other than their final estimations. : CS, ME and FOM may have difficulties when the clustering validity index distributions are similar for different number of clusters. Figure 4 shows a group of nine solutions, three using CS, three using ME and three solutions using FOM. All of these cases have some particular index distribution that makes difficult to decide the actual number of clusters. This problem is not related to the method used to display the results but rather to the index itself, which gives similar scores to different solutions.

: The graphics from Figure 4 were extracted from the additional files (Figures 3 to 11), which show the results of CS, ME and FOM in full length. This last set of figures presents also a brief explanation about the methods.

: The first row of Figure 4 corresponds to CS. The authors of the method suggest to use the minimum value of the green curve as estimation of the number of clusters. Figure 4.a corresponds to THY data clustered with EAC-av. Based on the mean value the recommended solution could be 2, 3 or 4 clusters. But even if a different parameter is considered, like the median value, this example is still difficult to solve. More than 50% of the distribution of the 3 and 4 clusters solutions have lower values compared to the 2 clusters solution, yet the dispersion of the 3 and 4 clusters solutions are several times greater than the one of the 2 clusters solution. Figure 4.b, Multi-A data clustered with PKNNG-av, presents similar solutions for 4 and 5 clusters, similar media values but with a smaller dispersion in the case of the 5 clusters solution. A second look at this example shows also that the 3 clusters solution has more than 50% of its distribution below than those of the 4 and 5 clusters solutions but because of some extreme values its mean is bigger than the ones of the 4 and 5 clusters solutions. The third CS example, depicted by Figure 4.c corresponds to ALB data clustered with Spectral Clustering. This case presents similar distributions for 3, 4 and 5 clusters solutions, all three distributions are similar and have close mean values which prevents the selection of a unique solution.

: The second row from Figure 4 corresponds to solutions of ME. Ben-Hur et al. [10] propose to look for groups of solution with a tight distribution of values near one and suggest to select the solution with the highest number of clusters. In their work the authors show that there is a gap separating good solution with tight distributions near one from bad solutions that exhibited more variability and values closer to zero. Figure 4.d, THY dataset clustered with PKNNG-av, has almost the same distribution for the 2, 3 and 4 clusters solutions. Following the recommendations from Ben-Hur et al. the most likely solution should be the one with four clusters, which differs from the number of classes. In Figure 4.e, ALI dataset clustered with Spectral Clustering, again it is difficult to place the gap between good and bad solutions. The gap could be placed between 3 and 4, 4 and 5 or 5 and 6 clusters. In this analysis the gap was fixed between 3 and 4 clusters and the estimated number of clusters was set to 3. In the example depicted in Figure 4.f, Y data clustered with EAC-av, it is easy to place the gap between 4 and 5 clusters solutions and to estimate the number of clusters as four. Though, it should be noted that despite Ben-Hur et al. suggest that four should be regarded as the best choice, which also coincides with the number of classes, the distribution for three clusters has less variability (all its values equal to one), which means that 3 clusters present a more stable solution.

: Finally, the last row from Figure 4 presents three results of FOM. Figure 4.g shows THY data clustered with PKNNG-av and

Figure 4.h shows ALB data clustered with Spectral Clustering. Both panels show a similar issue which makes difficult the accurate selection of the number of clusters based on the FOM curves. Figure 4.g shows that the 3 clusters solution presents a bigger percentage (more than 50%) of lower index values than the 2 clusters solution but on the other hand the 2 clusters solution has a much lower variability. The choice then is to select a more stable solution with 2 clusters or one that exhibits great variability but with a great percentage of values indicating good quality. Figure 4.h shows a different trend in the curve values. Usually FOM values tend to decrease when the number of clusters increases. This change in the curve can make more difficult to decide between two similar distributions, as the 4 and 5 clusters solutions, given the sudden inversion in the slope. Figure 4.i shows a similar phenomenon but with a more severe slope between 6 to 7 clusters. Moreover, the slope between 6 and 7 clusters is more abrupt than the slope from 2 to 3 clusters. Both variations point out a change in the clustering solution.

# V. CONCLUSIONS

: This paper presented the new IC-av index for the validation of clustering solutions. This new measure is based on the local relations between patterns and their clustering labels, evaluated with a metric based on the use of the MST of the data. The IC-av index proved to be useful in many cases when compared to other methods, showing its best results with arbitrary shaped clusters. Several simulations were performed to show that the proposed index is superior to previous well-known methods. First, the new method was applied to artificial data. Simple geometrical problems were used to illustrate the new idea in controlled scenarios. The results of these tests were compared against other similar automatic cluster validity methods also based on geometrical properties. The following step was to test the new index with real data. This new set of simulations also included graphical cluster validity methods to contrast against the IC-av index. The results of the new index in both artificial and real data were better than previous methods, not only because it found the correct number of clusters more times than the competing methods, but also because it showed a low variability when input data was perturbed by sub-sampling. Also, the use of diverse clustering algorithms showed that the good performance of the IC-av index does not depend on the working principles of the different clustering methods.

: The experiments using gene expression data included a sub-sampling process, whose main goal was to show the existence of natural classes and its relation to the biological classes. Following this premise the clustering results were contrasted to their biological classes, i.e the golden rule. One of the questions that this work tried to answer is if a group of validation methods could detect the underlying structure of a problem. In other words, knowing the biological classes and a set of clustering solutions that are fed to the validation methods, can they detect which solution is similar to the golden rule? Our results show that under this setting not all methods can find the biological classes. Also, it should be noted that while under similar conditions the results from some automatic methods vary much, the results from Gap + IC-av remain stable and accurate.

: The sub-sampling process used in the real data section was devised to link the biological classes to natural structures. This process removes some examples from the dataset but without altering the underlying structure in the data. A heavier subsampling was not consider since one may risk to lose part of the structure of the data. There is one scenario that was regarded as pathological for our method. Consider an example where mild sub-sampling changes the number of clusters in the data by creating a gap and fragmenting a single cluster into two clusters. For instance, assume a sub-sampling experiment that has two outcomes i) points are removed and one cluster breaks into two clusters and ii) points are removed and clustering structure does not change. In that event, Gap + IC-av can become unstable because of the two possible solutions.

: The response of the three graphical methods included in our experiments deserves some comments. In the case of FOM, the average of the generalization measure of the single left-out condition did not seem to be a good method to estimate the number of clusters. Perhaps the datasets presented here include a high number of uninformative genes that reduce the overall quality of the index, as in some sense uninformative genes are adding noise to this method. Also, it should be noted that FOM may have problems when dealing with non globular clusters since FOM measures the within-cluster similarity in the left out conditions. The last issue with FOM is that the computation of the index scales linearly with the number of input variables, which becomes prohibitive when working with thousands of genes as input. The two remaining graphical methods, ME and CS, are not biased to any cluster shape, nonetheless neither of them performed better than the IC-av index. In the case of CS, performance problems could be linked to sample sizes. The train and test sample size divide data in two halves, which leads to clustering and later generalization of high dimensional datasets with perhaps not enough examples. On the other hand, results with ME are clearly better than with CS, but the problem of how to determine the correct answer with this method in some datasets remains open.

: There is a big difference between the two classes of validation methods applied in this work. Graphical methods use replicated clustering experiments to obtain a distribution of values, from which they extract a single answer to the question of how many clusters there are in the data. Automatic methods, on the other side, produce a single answer to the question with each replication, producing a distribution of answers instead of a single one. This last kind of response is more informative about the data under analysis, as it also shows the robustness of the answer. The main problem with previous automatic methods is that their performance on non-spherical clusters, and in particular on gene expression problems, is poor. The new IC-av index introduces a solution to this problem, combining the efficiency of graphical methods with the ease-of-use and more informative response of automatic methods.

: In several places within this paper it has been pointed out many deficiencies of non-automatic methods. In a few occasions these deficiencies were related to the parameter of these methods, the partitions size of ME and CS. Also in the case of kernel Gap, which is an automatic method, some concern was expressed regarding the tuning of the parameter ( $\sigma$ ). It is not the purpose of this work to point that automatic methods without parameters should be in general preferred to other methods. The results presented here cannot support such claim. The preference toward automatic methods without parameters stems from the difficulty of tuning these parameters in a non supervised environment. Nev: In this work the main goal was to show that the new validation method could perform well in different settings. The simulations used to compare the methods were always challenging but remained controlled and far from real experiment conditions. This work intended to show the capabilities and limitations of the new method, though future work is needed to produce a validation on real world situations.

# REFERENCES

- M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein, "Cluster analysis and display of genome-wide expression patterns," *Proceedings* of the National Academy of Sciences USA, vol. 95, pp. 14863–14868, 1998.
- [2] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander, "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring." *Science*, vol. 286, no. 5439, pp. 531–537, 1999.
- [3] S. K. Archer, D. Inchaustegui, R. Queiroz, and C. Clayton, "The cell cycle regulated transcriptome of trypanosoma brucei," *PLoS ONE*, vol. 6, no. 3, 2011.
- [4] M. de Souto, I. Costa, D. de Araujo, T. Ludermir, and A. Schliep, "Clustering cancer gene expression data: a comparative study," *BMC Bioinformatics*, vol. 9, no. 1, 2008.
- [5] J. Wang, "A fast hierarchical clustering algorithm for functional modules discovery in protein interaction networks," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, pp. 607–620, 2011.
- [6] S. C. Li, "Clustering 100,000 protein structure decoys in minutes," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, pp. 765–773, 2012.
- [7] J. Wang, M. Li, H. Wang, and Y. Pan, "Identification of essential proteins based on edge clustering coefficient," *IEEE/ACM Transactions* on Computational Biology and Bioinformatics, vol. 9, pp. 1070–1080, 2012.
- [8] T. Lange, V. Roth, M. L. Braun, and J. M. Buhmann, "Stability-based validation of clustering solutions," *Neural Comp.*, vol. 16, no. 6, pp. 1299–1323, 2004.
- [9] G. Getz, E. Levine, and E. Domany, "Coupled two-way clustering analysis of gene microarray data." *Proceedings of the National Academy* of Sciences of the United States of America, vol. 97, no. 22, pp. 12079– 12084, 2000.
- [10] A Ben-Hur and A Elisseeff and I Guyon, "A stability based method for discovering structure in clustered data," in *Pacific Symposium on Biocomputing Lihue: 3-7 January 2002; Hawaii*, R. B. Altman, A. K. Dunker, L. Hunter, K. Lauderdale, and T. E. Klein, Eds., 2002, pp. 6–17.
- [11] J. Handl, J. Knowles, and D. B. Kell, "Computational cluster validation in post-genomic data analysis," *Bioinformatics*, vol. 21, no. 15, pp. 3201– 3212, 2005.
- [12] Y. B. M Halkidi and M. Vazirgiannis, "On clustering validation techniques," *Journal of Intelligent Information Systems*, vol. 17, no. 2-3, pp. 107–145, 2001.
- [13] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a dataset via the gap statistic," *Journal of the Royal Statistical Society B*, vol. 63, pp. 411–423, 2003.
- [14] S. Datta and S. Datta, "Evaluation of clustering algorithms for gene expression data," *BMC Bioinformatics*, vol. 7, no. Suppl 4, 2006.
- [15] G. Stegmayer, D. H. Milone, L. Kamenetzky, M. G. Lpez, and F. Carrari, "A biologically inspired validity measure for comparison of clustering methods over metabolic data sets," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, pp. 706–716, 2012.
- [16] E.-M. M. de Villiers, C. Fauquet, T. R. Broker, H.-U. U. Bernard, and H. zur Hausen, "Classification of papillomaviruses." *Virology*, vol. 324, pp. 17–27, 2004.

- [17] H.-U. U. Bernard, R. D. Burk, Z. Chen, K. van Doorslaer, H. zur Hausen, and E.-M. M. de Villiers, "Classification of papillomaviruses (pvs) based on 189 pv types and proposal of taxonomic amendments." *Virology*, vol. 401, pp. 70–79, 2010.
- [18] K. Y. Yeung, D. R. Haynor, and W. L. Ruzzo, "Validating clustering for gene expression data," *Bioinformatics*, vol. 17, no. Suppl 4, 2001.
- [19] S. Dudoit and J. Fridlyand, "A prediction-based resampling method for estimating the number of clusters in a dataset." *Genome biology*, vol. 3, no. 7, 2002.
- [20] R. Tibshirani and G. Walther, "Cluster validation by prediction strength," *Journal of Computational & Graphical Statistics*, vol. 14, no. 3, pp. 511–528, 2005.
- [21] A. Fred and A. K. Jain, "Combining multiple clusterings using evidence accumulation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 6, pp. 835–850, 2005.
- [22] S. Monti, P. Tamayo, J. P. Mesirov, and T. R. Golub, "Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data," *Machine Learning*, vol. 52, no. 1-2, pp. 91–118, 2003.
- [23] M. G. Cardoso and A. P. de Leon F. de Carvalho, "Quality indices for (practical) clustering evaluation," *Intelligent Data Analysis*, vol. 13, pp. 725–740, 2009.
- [24] T. Calinski and J. Harabasz, "A dendrite method for cluster analysis," *Communications in Statistics - Theory and Methods*, vol. 32, no. 1, pp. 1–27, 1974.
- [25] W. J. Krzanowski and Y. T. Lai, "A criterion for determining the number of groups in a data set using sum-of-squares clustering," *Biometrics*, vol. 44, no. 1, 1988.
- [26] J. Hartigan, Clustering Algorithms. Wiley, 1975.
- [27] L. Kaufman and P. J. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis. Hoboken, New Jersey: John Wiley & Sons, 1990.
- [28] I. Dhillon, Y. Guan, and B. Kulis, "Kernel k-means, spectral clustering and normalized cuts," 2004, pp. 551–556.
- [29] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Neural Information Processing Systems 14*, T. Dietterich, S. Becker, and Z. Ghahramani, Eds., MIT Press. MIT Press, 2001, pp. 849–856.
- [30] S. Johnson, "Hierarchical clustering schemes," *Psychometrika*, vol. 32, no. 3, pp. 241–254, 1967.
- [31] Y. Xu, V. Olman, and D. Xu, "Clustering gene expression data using a graph-theoretic approach: an application of minimum spanning trees," *Bioinformatics*, vol. 18, no. 4, pp. 536–545, 2002.
- [32] C. Zhong, D. Miao, and R. Wang, "A graph-theoretical clustering method based on two rounds of minimum spanning trees," *Pattern Recognition*, vol. 43, pp. 752–766, 2010.
- [33] S. P. Smith and A. K. Jain, "Testing for uniformity in multidimensional data," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 6, no. 1, pp. 73–81, 1984.
- [34] A. K. Jain, X. Xu, T. K. Ho, and F. Xiao, "Uniformity testing using minimal spanning tree," in *International Conference on Pattern Recognition*. IEEE Computer Society, 2002, pp. 281–284.
- [35] J. H. Friedman and L. C. Rafsky, "Multivariate generalizations of the wald-wolfowitz and smirnov two-sample tests," *The Annals of Statistics*, vol. 7, no. 4, 1979.
- [36] Z. V. Volkovich, Z. Barzily, G.-W. Weber, and D. Toledano-Kitai, "Cluster stability estimation based on a minimal spanning trees approach," *AIP Conference Proceedings*, vol. 1159, no. 1, pp. 299–305, 2009.
- [37] Z. Barzily, Z. Volkovich, B. A. Öztürk, and G.-W. Weber, "On a minimal spanning tree approach in the cluster validation problem," *Informatica*, vol. 20, no. 2, pp. 187–202, 2009.
- [38] T. Z. B Fischer and J. M. Buhmann, "Path based pairwise data clustering with application to texture segmentation," *Energy Minimization Methods in Computer Vision and Pattern Recognition*, vol. 2134, pp. 235–250, 2001.
- [39] B. Fischer and J. M. Buhmann, "Bagging for path-based clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 11, pp. 1411–1415, 2003.
- [40] S. Datta and S. Datta, "Comparisons and validation of statistical clustering techniques for microarray gene expression data," *Bioinformatics*, vol. 19, no. 4, 2003.
- [41] V. Pihur, S. Datta, and S. Datta, "Weighted rank aggregation of cluster validation measures: A monte carlo cross-entropy approach," *Bioinformatics*, vol. 23, no. 13, pp. 1607–1615, 2007.
- [42] A. E. Baya and P. M. Granitto, "Clustering gene expression data with a penalized graph-based metric," *BMC Bioinformatics*, vol. 12, 2011.

- [43] G. J. McLachlan, R. W. Bean, and D. Peel, "A mixture model-based approach to the clustering of microarray expression data," *Bioinformatics*, vol. 18, no. 3, pp. 413–422, 2002.
- [44] Z.-M. L. and P. P., "Self-tuning spectral clustering," in *Neural Informa*tion Processing Systems 14, S. L., W. Y., and B. L., Eds., MIT Press. MIT Press, 2004, pp. 1601–1608.
- [45] A. A. and G. Z., "Spectral methods for automatic multiscale data clustering," in *Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, 2006, pp. 190–197.
- [46] N. B. and G. M., in Neural Information Processing Systems 14, Sch Ed.
- [47] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 888–905, 2000.
- [48] H. He, K. Jazdzewski, W. Li, S. Liyanarachchi, R. Nagy, S. Volinia, G. A. Calin, C. G. Liu, K. Franssila, S. Suster, R. T. Kloos, C. M. Croce, and A. de la Chapelle, "The role of microrna genes in papillary thyroid carcinoma." *Proceedings of the National Academy of Sciences* of the United States of America, vol. 102, no. 52, pp. 19075–19080, 2005.
- [49] Y. Hoshida, J.-P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, "Subclass mapping: Identifying common subtypes in independent disease data sets," *PLoS ONE*, vol. 2, no. 11, 2007.
- [50] A. A. Alizadeh, M. B. Eisen, R. E. Davis, C. Ma, I. S. Lossos, A. Rosenwald, J. C. Boldrick, H. Sabet, T. Tran, X. Yu, J. I. Powell, L. Yang, G. E. Marti, T. Moore, J. Hudson, L. Lu, D. B. Lewis, R. Tibshirani, G. Sherlock, W. C. Chan, T. C. Greiner, D. D. Weisenburger, J. O. Armitage, R. Warnke, R. Levy, W. Wilson, M. R. Grever, J. C. Byrd, D. Botstein, P. O. Brown, and L. M. Staudt, "Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling," *Nature*, vol. 403, no. 6769, pp. 503–511, 2000.
- [51] M. P. S. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. Sugnet, T. S. Furey, J. M.Ares, and D. Haussler, "Knowledge-based analysis of microarray gene expression data using support vector machines," *Proceedings of the National Academy of Sciences USA*, pp. 262–267, 2000.
- [52] A. Bhattacharjee, W. G. Richards, J. Staunton, C. Li, S. Monti, P. Vasa, C. Ladd, J. Beheshti, R. Bueno, M. Gillette, M. Loda, G. Weber, E. J. Mark, E. S. Lander, W. Wong, B. E. Johnson, T. R. Golub, D. J. Sugarbaker, and M. Meyerson, "Classification of human lung carcinomas by mrna ex-pression profiling reveals distinct adenocarcinomas subclasses," *Proceedings of the National Academy of Sciences USA*, vol. 98, pp. 13 790–13 795, 2001.
- [53] G. Milligan and M. Cooper, "A study of comparability of external criteria for hierarchical cluster analysis," *Multivariate Behavioral Research*, vol. 21, pp. 441–458, 1986.



ical data.

Ariel E. Bayá received his degree in Electronic Engineering from Universidad Nacional de Rosario, Argentina, in 2003. From 2004 to 2005 he was an intern with the MEC group at Cardiff University, Wales. In 2011 he received his Ph.D in Information Science from Universidad Nacional de Rosario, Argentina. He is currently a staff member of the CIFASIS Research Institute, CONICET, where he holds the position of Assistant Researcher. His main research interests are clustering methods, clustering validation methods, and their applications to biolog-



**Pablo M. Granitto** was born in Rosario, Argentina, in 1970. He received the Degree in physics, and the Ph.D. degree, also in physics, in 1997 and 2003, respectively, both from Universidad Nacional de Rosario (UNR), Rosario, Argentina. He was a Post-Doctoral Researcher at Istituto Agrario San Michele allAddige, Trento, Italy. Since 2006, he has been a full-time Researcher at Consejo Nacional de Investigaciones Cientcas y Tecnolgicas (CONICET), Rosario, and UNR. He leads the Machine Learning Group at CIFASIS, Rosario. His current research

interests include application of modern machine learning techniques to agroindustrial and biological problems, involving feature selection, clustering, and ensemble methods.