Decomposition of the Rand index in order to assess both the stability and the number of clusters of a partition

Ghazi Bel Mufti, Patrice Bertrand, Lassad El Moubarki

Decomposition of the Rand index in order to assess both the stability and the number of clusters of a partition

G. Bel Mufti[a], P. Bertrand[b], L. El Moubarki[c]

[a] *ESSEC-Tunis, Tunisia*
[b] *CEREMADE, Université Paris-Dauphine, France*
[c] *Faculté des sciences économiques et de gestion de Sfax, Tunisia*

Abstract

During the last decade, stability-based measures became popular in order to validate the results of partitioning methods. A number of different procedures were proposed in order to compute a measure of partitioning stability, each of them requiring intensive computations. Among these procedures, those are the simplest and most commonly used estimate the stability of a partition by performing a large number of comparisons between two partitions achieved on distinct perturbed sets (*e. g.* Ben-Hur *et al.* (2002)). The Rand index provides a simple and easily interpretable way to achieve such comparisons, and thus is commonly employed to assess partitioning stability. The contribution of this paper is twofold. We first propose an additive decomposition of a partitioning stability measure which is directly derived from the Rand index, and we interpret the factors of this decomposition as stability-based measures of the cohesion and the isolation of each cluster. Then, we derive a bi-criterion, which takes account of both the isolation and the cohesion of each individual cluster, in order to assess the optimal number of clusters of a partition. Based on simulated data sets, we compare our approach with the most successful methods for predicting the number of clusters.

Keywords: Partition stability, cluster isolation, cluster cohesion, resampling, random noise

1. Introduction

Partitioning stability is usually requested when assessing the validity of a partition that was obtained by running a clustering algorithm on a real dataset. Recently, several cluster stability based methods were proposed in order to

select the optimal number of clusters of a partition. Another approch, for both assessing the validity of a partition and the optimal nomber of clusters, consists in assessing the degree of adequacy between the dataset and the cluster structure generated by the partitioning algorithm. Measures of this type, which are based on geometrical properties of the clusters, were, in the chronological order, among the first cluster validation measures to be proposed in the litterature on cluster analysis. As examples of such measures, one can cite the indices of Calinski and Harabasz [2], and of Krzanowski and Lai [3] that have been repeatedly chosen as benchmarks [1], [4]. Note that these indices are both based on the adequacy criterion of minimal cluster inertia, so that they have a tendacy to favor spherical clusters.

A more recent type of approach validates clusters on the basis of their stability (see for example, Levine and Domany [5], Ben-Hur *et al.* [1], Tibshirani *et al.* [4], Kapp and Tibshirani [6], Bertrand and Bel Mufti [7]). Stability measurements are nowadays also employed in practice for selecting the optimal number of clusters: the optimal number, say k, is then chosen such that the k-partition stability measure is maximum among all values of k for which a k-partitioning of the data set was performed.

Cluster stability is generally supposed to hold when small changes in the dataset have no significant effect on membership of the clusters. In order to obtain these changes, various perturbation methods were investigated, like resampling the dataset [1], [8], [9], or adding a noise to the descriptors or to the data objects [10], [11]. The resulting data sets are clustered using the same clustering algorithm. Then, a standard way to measure partitioning stability is to compare the partitions of two perturbed data sets, and repeat N times (with N large) this procedure of comparison. In this paper, we consider a variant of this procedure: instead of comparing, at each of the N iterations, the partitions of two perturbed data sets, we will compare the examined partition of the whole data set with the partition obtained on some perturbed data set. As a result, an indicator that synthesizes the N comparisons is computed in order to assess the stability of the examined partition.

In what follows, we first consider the average of the well known Rand index, for pairs of partitions, as a measure of partition stability, and show that this index can be expressed as a weighted mean of two indices that estimate, respectively, the isolation and the cohesion of the whole partition. These two indices can be decomposed, in their turn, into k terms that are stability-based measures of cohesion and isolation of each of the k clusters. The two-level decomposition led us to propose a bi-criterion-based measure that estimates

the optimal number of clusters. Finally, we illustrate our bi-criterion approach for cluster validation both on real data sets and on simulated data sets taken from Kapp [12].

2. Decomposition of a partitioning stability measure based on the Rand index

We consider a dataset X and a partition P that was obtained by applying to X a partitioning algorithm $\mathcal{A}$ into k clusters. In the sequel, a partitioning algorithm into k clusters will be called, simply, a k-partitioning algorithm. We will also denote as $\mathcal{A}(S)$ the partition obtained by applying a k-partitioning algorithm $\mathcal{A}$ to an arbitrary set S. Consequently $P = \mathcal{A}(X)$. Moreover, if S is a subset of X or if $S = X$, the partition of S whose clusters coincide with the intersections between S and any cluster of P is denoted $P_{|S}$ and is said to be the *trace of P on S*. Formally $P_{|S}$ is then defined by

$$P_{|S} = \{C \cap S \ : \ C \in P\}.$$

We aim to estimate the stability of both the partition P and the clusters of P. Our approach is based on the well known Rand index which enables to compare any two partitions. Let P_1 and P_2 be two partitions defined on dataset X and let, as usual, N_{11} (resp. N_{00}) be the number of pairs clustered together (resp. not clustered together) both in P_1 and in P_2, Then, the Rand index R is defined as follows:

$$\mathrm{R}(P_1, P_2) \ = \ \frac{N_{11} + N_{00}}{\binom{n}{2}}, \tag{1}$$

where $n = |\, X \,|$. Then R is the empirical frequency that the cluster memberships of partitions P_1 and P_2 agree, so that R is a measure of similarity between P_1 and and P_2. Formally, the Rand index is the empirical frequency that Several measures of partitioning stability can be derived from the Rand index. The next procedure gives an example of such a Rand-index based measure.

1. Compute the partition $P = \mathcal{A}(X)$ to be validated.
2. Generate N i.i.d. random sets, denoted as $S_1, \ldots, S_N$, such that each S_j $(j = 1, \ldots, N)$ is a perturbed version of dataset X
3. Compute the N partitions $\mathcal{A}(S_1), \ldots, \mathcal{A}(S_N)$.

4. Compute the measure of stability of partition P here defined as the arithmetic mean $\overline{\mathrm{R}}_N(P)$ of the values $\mathrm{R}(P_{|S_j}, \mathcal{A}(S_j))$ for $j = 1, \ldots, N$.

Notice that this procedure can be seen as a variant of the procedure proposed by Ben-Hur *et al.* [1]. The sets $S_1, \ldots, S_N$ are perturbed versions of the dataset X which are assumed to be of equal size, say m. Moreover, $S_1, \ldots, S_N$ are i. i. d. random sets, so that according to the law of large numbers, whenever N is large enough, $\overline{\mathrm{R}}_N(P)$ is a faithful estimate of the probability, say p, that memberships of P_1 and P_2 agree. Now, the Central Limit Theorem provides a way to compute the minimum value N_ϵ of N such that the Confidence Interval at threshold 95% that estimates this probability p has a length less or equal than a precision of 2ϵ.

Let us now investigate the extent with which the cluster-isolation degree and the cluster-cohesion degree of each cluster of P contribute to the overall measure of partitioning stability $\overline{\mathrm{R}}_N(P)$:

$$\overline{\mathrm{R}}_N(P) \;=\; \frac{1}{N}\sum_{j=1}^{N} \mathrm{R}(P_{|S_j}, \mathcal{A}(S_j)). \tag{2}$$

Then, we consider both an arbitrary cluster C of P and an arbitrary perturbed version S of the dataset X. If D is some cluster of $\mathcal{A}(S)$, we denote as $m_{C,D}$ the number of objects that are clustered both in cluster $C \in P$ and in cluster $D \in \mathcal{A}(S)$, so that $m_{C,D} = |\, C \cap D \,|$. In addition, for all $C \in P$ and $D \in \mathcal{A}(S)$, we denote:

$$m_C = \sum_{D \in \mathcal{A}(S)} m_{C,D} = |\, C \cap S \,| \qquad \text{and} \qquad m_D = \sum_{C \in P} m_{C,D} = |\, D \,|\,.$$

Consequently,

$$m \;=\; \sum_{C \in P} m_C \;=\; \sum_{D \in \mathcal{A}(S)} m_D \;=\; \sum_{C \in P} \sum_{D \in \mathcal{A}(S)} m_{C,D}.$$

With these notations, we decompose $\mathrm{R}(P_{|S}, \mathcal{A}(S))$ according to the contributions of the clusters of P by means of decompositions of the terms N_{11}

and N_{00} (see (1)):

$$N_{11} = \sum_{C\in P}\sum_{D\in\mathcal{A}(S)}\binom{m_{C,D}}{2} = \sum_{C\in P}\binom{m_C}{2}\sum_{D\in\mathcal{A}(S)}\frac{\binom{m_{C,D}}{2}}{\binom{m_C}{2}}. \quad (3)$$

$$N_{00} = \frac{1}{2}\sum_{C\in P}\sum_{D\in\mathcal{A}(S)}|C\cap D||\overline{C}\cap\overline{D}|\,.$$

Since $|\overline{C}\cap\overline{D}| = |\overline{C\cup D}| = m-(m_C+m_D-m_{C,D})$, we have:

$$N_{00} = \frac{1}{2}\sum_{C\in P}\sum_{D\in\mathcal{A}(S)} m_{C,D}(m-m_C-m_D+m_{C,D}),$$

$$N_{00} = \sum_{C\in P}\frac{1}{2}\,m_C(m-m_C)\sum_{D\in\mathcal{A}(S)}\frac{m_{C,D}(m-m_C-m_D+m_{C,D})}{m_C(m-m_C)}. \quad (4)$$

As a direct consequence of (1), (3) and (4), it follows:

$$\mathrm{R}(P_{|S},\mathcal{A}(S)) = \sum_{C\in P}\Big[\alpha(C)\,\mathrm{R}_{\mathrm{co}}(C;S) + \beta(C)\,\mathrm{R}_{\mathrm{is}}(C;S)\Big], \quad (5)$$

where $\alpha(C), \beta(C), \mathrm{R}_{\mathrm{co}}(C;S)$ and $\mathrm{R}_{\mathrm{is}}(C;S)$ are defined by:

$$\alpha(C) = \frac{\binom{m_C}{2}}{\binom{m}{2}},$$

$$\beta(C) = \frac{\frac{1}{2}\,m_C(m-m_C)}{\binom{m}{2}},$$

$$\mathrm{R}_{\mathrm{co}}(C;S) = \frac{\displaystyle\sum_{D\in\mathcal{A}(S)}\binom{m_{C,D}}{2}}{\binom{m_C}{2}}, \quad (6)$$

$$\mathrm{R}_{\mathrm{is}}(C;S) = \frac{\displaystyle\sum_{D\in\mathcal{A}(S)}^{k} m_{C,D}(m-m_C-m_D+m_{C,D})}{m_C\,(m-m_C)}. \quad (7)$$

Since $\sum_{C \in P} m_C = m$, we have:

$$\sum_{C \in P} \big(\alpha(C) + \beta(C)\big) \;=\; \frac{\displaystyle\sum_{C \in P} \tbinom{m_C}{2} + \frac{1}{2}\, m_C(m - m_C)}{\tbinom{m}{2}} \;=\; 1.$$

From equation (5) we deduce that $\mathrm{R}(P_{|\,S}, \mathcal{A}(S))$ is the weighted mean, for all C in P, of $\mathrm{R}_{\mathrm{co}}(C; S)$ and of $\mathrm{R}_{\mathrm{is}}(C; S)$, with weights being defined as $\alpha(C), \beta(C)$ respectively. Note that the weights $\alpha(C), \beta(C)$ are independent from the sample S.

Interpretation of $\mathrm{R}_{\mathrm{co}}(C; S)$ *for* $C \in P$. Given an arbitrary cluster C of P, $\mathrm{R}_{\mathrm{co}}(C; S)$ is the proportion, among the pairs of objects of $C \cap S$, of those that are clustered together by the partition $\mathcal{A}(S)$ (see eq. (6)). Hence $\mathrm{R}_{\mathrm{co}}(C; S)$ is the empirical estimation of the conditional probability that two objects of a sample S are clustered together by partition $\mathcal{A}(S)$ given that they are in cluster C. As a consequence, $\mathrm{R}_{\mathrm{co}}(C; S)$ is the confidence index value of the following (association) rule:

(Co) Given two objects of sample S of X, if they are clustered together in cluster C then they are in the same cluster of $\mathcal{A}(S)$.

Since rule (Co) expresses the degree cohesion of cluster C, $\mathrm{R}_{\mathrm{co}}(C; S)$ is a measure of the degree cohesion of cluster C.

Interpretation of $\mathrm{R}_{\mathrm{is}}(C; S)$ *for* $C \in P$. We proceed as for the interpretation of $\mathrm{R}_{\mathrm{co}}(C; S)$. Given an arbitrary cluster C of P, $\mathrm{R}_{\mathrm{is}}(C; S)$ is the proportion of pairs of objects that are not clustered together by partition $\mathcal{A}(S)$, among those pairs of objects of S for which only one belongs to C (see eq. (7). It results that $\mathrm{R}_{\mathrm{is}}(C; S)$ is the empirical estimation of the conditional probability that two objects of a sample S are are not clustered together by partition $\mathcal{A}(S)$ given that only one of them belongs to C. Therefore, $\mathrm{R}_{\mathrm{is}}(C; S)$ is the confidence index value of the following (association) rule:

(Is) Given two objects of sample S of X, if only one is in C, then the two objects are not in the same cluster of $\mathcal{A}(S)$.

Since rule (Is) expresses the degree of isolation of cluster C, $\mathrm{R}_{\mathrm{is}}(C; S)$ is a measure of the degree isolation of cluster C.

As an immediate consequence of the above interpretations, $\mathrm{R}_{\mathrm{co}}(C;S)$ and of $\mathrm{R}_{\mathrm{is}}(C;S)$ are empirical estimations of two conditional probabilities, these estimations being dependent from the sample S under consideration. Applying the standard law of large numbers, the two means $\overline{\mathrm{R}}_{\mathrm{co},N}(C)$ and as $\overline{\mathrm{R}}_{\mathrm{is},N}(C)$ defined by:

$$\begin{aligned}\overline{\mathrm{R}}_{\mathrm{co},N}(C) &= \frac{1}{N}\sum_{j=1}^{N}\mathrm{R}_{\mathrm{co}}(C;S_j),\\ \overline{\mathrm{R}}_{\mathrm{is},N}(C) &= \frac{1}{N}\sum_{j=1}^{N}\mathrm{R}_{\mathrm{is}}(C;S_j),\end{aligned}$$

are faithful statistical estimations of the corresponding conditional probabilities, provided that they are computed for a large number N of random i.i.d. samples $S_1,\ldots,S_N$. Furthermore, recall that the global stability of partition P is here estimated by $\overline{\mathrm{R}}_N(P)$ defined as mean of $\mathrm{R}(P_{\mid S_j},\mathcal{A}(S_j))$ for $j=1,\ldots,N$ (cf. eq. (2)). Since the coefficients $\alpha(C)$ and $\beta(C)$, for $C\in P$, don't depend from the sample S_j being considered, it results from eq. (5) that:

$$\overline{\mathrm{R}}_N(P) = \sum_{C\in P}\Big[\alpha(C)\,\overline{\mathrm{R}}_{\mathrm{co},N}(C) + \beta(C)\,\overline{\mathrm{R}}_{\mathrm{is},N}(C)\Big]. \tag{8}$$

As a consequence of the previous decomposition given by (8) we obtain the next proposition.

Proposition. *Assuming that a partitioning method provides a partition P of some dataset, the measure of partitional stability, $\overline{\mathrm{R}}_N(P)$, is the weighted mean, for all clusters C in P, of $\overline{\mathrm{R}}_{\mathrm{co},N}(C)$ and $\overline{\mathrm{R}}_{\mathrm{is},N}(C)$ where :*

- *the measures $\overline{\mathrm{R}}_{\mathrm{co},N}(C)$ and $\overline{\mathrm{R}}_{\mathrm{is},N}(C)$ are stability-based estimations of the cohesion degree and of the isolation degree of cluster C, respectively;*
- *the weight of $\overline{\mathrm{R}}_{\mathrm{co},N}(C)$ is the proportion of pairs of objects of any sample under consideration that are in cluster C.*
- *the weight of $\overline{\mathrm{R}}_{\mathrm{is},N}(C)$ is the proportion of pairs of objects of any sample under consideration for which only one object belongs to cluster C.*

3. Partitioning stability measure based on individual-cluster stability

The aim of this section is to derive, from the decomposition obtained in the previous section, a partitioning stability measure that can be used in order to identify the optimal number of clusters of a partition P of the dataset X. Recall first that Ben David *et al.* [13] and Shamir and Tishby [14] pointed out the importance of a low density of objects on the frontier between each two clusters in order to insure an optimal stability of the partitioning. According to this theoretical result, we propose to estimate the stability of a partition by the minimum of its cluster cohesion degrees and its cluster isolation degrees. Thus denoting as $\mathrm{ICM}(P)$ the stability measure of partition P defined in such a way, we have:

$$\mathrm{ICM}(P) = \min_{C \in P} \{\overline{\mathrm{R}}_{\mathrm{co},N}(C), \overline{\mathrm{R}}_{\mathrm{is},N}(C)\}.$$

This definition of the stability measure $\mathrm{ICM}(P)$ requires to use a procedure that defines and generates perturbed datasets S from the dataset X. One approach may be to define S by adding a random noise to the value of each variable for each object of X (typically, 5% of twice the standard deviation for each variable). Another approach is to draw a random sample S from the data set X using a sample rate f (typically, $f = 0.80$). In this case, remark that each cluster of P may be under-represented in a given random sample S, in particularly if some cluster sizes are less than $(1 - f)|X|$ then it may happen that no objects of such clusters belong to the sample S. In such a case, it is clear that whatever the quality of partition P, the comparison of $\mathcal{A}(S)$ with P cannot be in favor of the stability of P. We then propose to use the so called *proportionate stratified sampling* procedure (see for example Hansen et al. ([15], p. 140)) which consists in selecting randomly and without replacement m_C elements in each cluster C of P, where m_C is defined as the value $f \mid C \mid$ rounded down to the nearest integer. Without any specific indication, each sample S of X that we consider will be drawn according to this proportionate stratified sample procedure. It results that all the samples are of size equal to $\sum_{C \in P} m_C$. Denoting as m this common size, we then have:

$$m = \sum_{C \in P} m_C \approx \lfloor f \mid X \mid \rfloor = \lfloor f\, n \rfloor.$$

The proposed stability measure ICM aims at validating both the choice of the clustering criterion and the choice of the number of clusters. In what follows, we propose a general algorithm, denoted as Algorithm 1, that estimates the stability measure ICM(P) of a partition P, together with the two degrees of cluster cohesion and cluster isolation for each cluster of P. Algorithm 1 requires to know the values of cluster-cohesion measure $\overline{\mathrm{R}}_{\mathrm{co},N}(C)$ and cluster-isolation measure $\overline{\mathrm{R}}_{\mathrm{is},N}(C)$ for each cluster C of P: these values are indeed computed hereafter by Algorithm 2.

Algorithm 1.

Require:

- X: reference dataset,
- k_{max}: maximum number of clusters to be tested,
- $\mathcal{A}_k$: clustering algorithm into k clusters,
- γ: threshold value.

Ensure: k^*: optimal number of clusters.

1: **for** $k = 2$ to k_{max} **do**

2: Partition the reference dataset X into k clusters:

$$P_k = \mathcal{A}_k(X) = \{C_1, \ldots, C_k\}.$$

3: For each cluster $C_i (i = 1, \ldots, k)$ of P_k, compute the cluster cohesion measure $\overline{\mathrm{R}}_{\mathrm{co},N}(C_i)$ and cluster isolation measure $\overline{\mathrm{R}}_{\mathrm{is},N}(C_i)$.

4: Compute the minimum stability value:

$$\mathrm{ICM}(P_k) = \min_{i=1,\ldots,k} [\min(\overline{\mathrm{R}}_{\mathrm{co},N}(C_i), \overline{\mathrm{R}}_{\mathrm{is},N}(C_i))].$$

5: **end for**

6: The optimal number of clusters is:

$$k^* = \max\{k = 2, \ldots, k_{max} \ : \ \mathrm{ICM}(P_k) > \gamma\}.$$

If all the values of ICM(P_k) are less then the specified value of γ, then the optimal number of clusters is 1.

As mentioned in step 6 of the algorithm, a given partition P_k is stable if all its cluster cohesion degrees and all its cluster isolation degrees are greater than the specified value γ. In the experiences presented in next section 4, the value of γ is fixed at 0.95.

Now, Algorithm 1 which computes our index ICM, requires to estimate with a sufficient precision the stability-based measures $\overline{\mathrm{R}}_{\mathrm{co},N}(C)$ and $\overline{\mathrm{R}}_{\mathrm{is},N}(C)$

for each $C \in P$. Next algorithm, denoted as Algorithm 2, computes such estimations, together with the value of $\overline{\mathrm{R}}_N(P)$, providing these estimates using a precision of 2ϵ at a confidence level of 95%. Notice that $IC_{95\%}$ stands here for the Confidence Interval at the level 95% of the stability-based measure considered, this interval being computed by using the Central Limit theorem. It is worth to notice also that Algorithm 2 runs with perturbed datasets that can be defined either by addition of random noise or by any type of random sampling.

Algorithm 2.

Require:

- X: reference dataset,
- $\mathcal{A}_k$: clustering algorithm,
- $P = \mathcal{A}_k(X)$,
- ϵ: precision,
- $\mathrm{I}(S)$: stability-based measure given a single perturbed dataset S.

$$\mathrm{I}(S) \in \{\mathrm{R}(P,S), \mathrm{R}_{\mathrm{is}}(C,S), \mathrm{R}_{\mathrm{co}}(C,S)\}.$$

- $\bar{\mathrm{I}}_N = \dfrac{1}{N}\displaystyle\sum_{j=1}^{N} \mathrm{I}(S_j)$: mean of the stability-based measure for N perturbed datasets $S_1, \ldots, S_N$.

Ensure: $\bar{\mathrm{I}}_{N_\epsilon}$ estimate of the stability-based measure I with a precision less or equal than ϵ

1: **while** $(IC_{95\%}(\bar{\mathrm{I}}_N) > 2\epsilon)$ & $(N < 500)$ **do**
2: $\quad N = N + 1$
3: $\quad$ Compute the N th perturbed dataset S_N derived from dataset X.
4: $\quad$ Compute $\bar{\mathrm{I}}_N$
5: $\quad$ if $(N > 30)$, *compute* $IC_{95\%}(\bar{\mathrm{I}}_N)$
6: **end while**
7: $N_\epsilon = N$
8: $\bar{\mathrm{I}}_{N_\epsilon} = \dfrac{1}{N_\epsilon}\displaystyle\sum_{i=j}^{N_\epsilon} \mathrm{I}(S_j)$

---

4. Experimental results

In this section we compare the ICM criterion with nine among the most successfull methods for determining the optimal number of clusters (cf. Milligan and Cooper [10], Hardy [16] and Tibshirani and Walter [9]). This comparison includes the In-group proportion (IGP) method proposed by Kapp and Tibshirani [6], the Calinski and Harabasz method (CH) [2], the Krzanowski and Lai method (KL) [3], the Silhouette statistic proposed by Rousseeuw [17], the Gap statistic proposed by Tibshirani *et al.* [4]), the Prediction strength (PS) method of Tibshirani and Walter [9], the Jump statistic of Sugar et James [18] and the Clest method due to Dudoit and Fridlyand [19].
We compare the ICM criterion with all these criteria on both real datasets and artificial datasets generated from ten models that are specified in Kapp [12]. The used parameters of the clustering validation methods considered above are selected as in the paper which initially presented the method, in particular two null reference distributions are considered for the Gap statistic: the uniform reference distribution over the range of each observed feature and the uniform reference in the principal components orientation.

4.1. Simulation results

We consider a dataset that consists of a $n \times p$ matrix $X = (X_{ij})_{\substack{i=1,\dots,n \\ j=1,\dots,p}}$ of real values. Let $X_i = (X_{i1}, \dots, X_{ip})$ be the i^{th} row of X. The simulated datasets were generated from ten models: seven low-dimensional models and three hight-dimensional models that are presented below.

1. Uniform null in 10 dimensions: 200 data points uniformly distributed over the unit square in 10 dimensions.

2. Gaussian null in 10 dimensions: 200 data points of a standard multivariate normal distribution in 10 dimensions.

3. Four evenly-sized Gaussian clusters with identity covariance in 2 dimensions: a mixture of 4 bivariate normal distributions with (25 25 25 25) observations and means and variance-covariance matrix given by

$$\boldsymbol{\mu}_1 = \begin{bmatrix} -3 \\ 3 \end{bmatrix},\ \boldsymbol{\mu}_2 = \begin{bmatrix} 4 \\ 4 \end{bmatrix},\ \boldsymbol{\mu}_3 = \begin{bmatrix} 5 \\ 5 \end{bmatrix},\ \boldsymbol{\mu}_4 = \begin{bmatrix} -6 \\ 6 \end{bmatrix} \text{ and } \boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

4. Four unevenly-sized Gaussian clusters with identity covariance in 2 dimensions: a mixture of 4 bivariate normal distributions with (10 20 30 40)

observations and means and variance-covariance matrix given by

$$\boldsymbol{\mu}_1 = \begin{bmatrix} -3 \\ 3 \end{bmatrix},\ \boldsymbol{\mu}_2 = \begin{bmatrix} 4 \\ 4 \end{bmatrix},\ \boldsymbol{\mu}_3 = \begin{bmatrix} 5 \\ 5 \end{bmatrix},\ \boldsymbol{\mu}_4 = \begin{bmatrix} -6 \\ 6 \end{bmatrix} \text{ and } \boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

5. Four unevenly-sized Gaussian clusters with non-identity covariance in 2 dimensions: a mixture of 4 bivariate normal distributions with (10 20 30 40) observations and means and variance-covariance matrix given by

$$\boldsymbol{\mu}_1 = \begin{bmatrix} -3 \\ 3 \end{bmatrix},\ \boldsymbol{\mu}_2 = \begin{bmatrix} 4 \\ 4 \end{bmatrix},\ \boldsymbol{\mu}_3 = \begin{bmatrix} 5 \\ 5 \end{bmatrix},\ \boldsymbol{\mu}_4 = \begin{bmatrix} -6 \\ 6 \end{bmatrix},$$

$$\text{with } \boldsymbol{\Sigma}_1 = \begin{bmatrix} 1 & -0.7 \\ -0.7 & 1 \end{bmatrix},\ \boldsymbol{\Sigma}_2 = \begin{bmatrix} 1 & -0.3 \\ -0.3 & 1 \end{bmatrix}, \boldsymbol{\Sigma}_3 = \begin{bmatrix} 1 & 0.3 \\ 0.3 & 1 \end{bmatrix},$$

$$\text{and } \boldsymbol{\Sigma}_4 = \begin{bmatrix} 1 & 0.7 \\ 0.7 & 1 \end{bmatrix}.$$

6. Two elongated clusters in 3 dimensions: for $i = 1, \ldots, 100$ and $j = 1, 2, 3$

$$X_{ij} = Y_{ij} + N(0, 0.1) \text{ with } Y_i = \begin{bmatrix} -0.5 + \frac{i-1}{99} \\ -0.5 + \frac{i-1}{99} \\ -0.5 + \frac{i-1}{99} \end{bmatrix}$$

and for $i = 101, \ldots, 200$ and $j = 1, 2, 3$

$$X_{ij} = Y_{ij} + N(0, 0.1) \text{ with } Y_i = \begin{bmatrix} -0.5 + \frac{i-101}{99} \\ -0.5 + \frac{i-101}{99} \\ -0.5 + \frac{i-101}{99} \end{bmatrix}$$

7. Four exponential clusters in 2 dimensions: for $i = 1, \ldots, 100$ and $j = 1, 2$, let $Y_{ij} \overset{iid}{\sim} exp\,(1)$ and

$$X_i = \begin{cases} Y_i + (2, -2) & \text{for } i = 1, \ldots, 25, \\ Y_i + (2, 2) & \text{for } i = 26, \ldots, 50, \\ Y_i + (-2, 2) & \text{for } i = 51, \ldots, 75, \\ Y_i + (-2, -2) & \text{for } i = 76, \ldots, 100. \end{cases}$$

In order to take account of the effect of the high dimensionalilty of the data sets on the results, the last three models are generated in 1000 dimensions.

8. Uniform null in 1000 dimensions: 100 data points uniformly distributed over the unit square in 1000 dimensions.

9. Three evenly-sized clusters separated in 900 dimensions: initially, let $X_{ij} = 0$ for all $i = 1, \ldots, 150$ and $j = 1, \ldots, 1000$. Then, set

$$X_{ij} = 5 \begin{cases} \text{for } i = 1, \ldots, 50 \text{ and } j = 51, \ldots, 350, \\ \text{for } i = 51, \ldots, 100 \text{ and } j = 351, \ldots, 650, \\ \text{for } i = 101, \ldots, 150 \text{ and } j = 651, \ldots, 950. \end{cases}$$

To add noise to the data let the n_i be the results from randomly drawing 150 elements from $\{1, 2, \ldots, 100\}$ with replacement. For each i, generate n_i random elements from a $U[-1; 1]$ distribution and randomly add them to n_i elements among the X_{ij}s.

10. Three unevenly-sized clusters separated in 900 dimensions: initially, let $X_{ij} = 0$ for all $i = 1, \ldots, 175$ and $j = 1, \ldots, 1000$. Then, set

$$X_{ij} = 5 \begin{cases} \text{for } i = 1, \ldots, 25 \text{ and } j = 51, \ldots, 350, \\ \text{for } i = 26, \ldots, 75 \text{ and } j = 351, \ldots, 650, \\ \text{for } i = 76, \ldots, 175 \text{ and } j = 651, \ldots, 950. \end{cases}$$

To add noise to the data let the n_i be the results from randomly drawing 175 elements from $\{1, 2, \ldots, 100\}$ with replacement. For each i, generate n_i random elements from a $U[-1; 1]$ distribution and randomly add them to n_i elements among the X_{ij}s.

Procedure	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7
	k-means						
CH	-	-	1.00	0.99	0.69	0.00	0.34
KL	-	-	0.89	0.87	0.86	0.98	0.75
Silhouette	-	-	1.00	0.99	0.99	1.00	0.62
Gap/Unif.	0.01	0.09	0.04	0.96	0.67	0.00	0.18
Gap/PC	0.02	0.21	0.22	0.93	0.73	1.00	0.13
Clest	0.90	0.99	0.09	0.14	0.06	0.77	0.66
Jump	0.00	0.00	1.00	0.99	0.98	0.00	0.81
PS	1.00	1.00	1.00	0.90	0.75	0.43	0.85
IGP	1.00	1.00	0.80	0.15	0.10	0.76	0.11
Rand	1.00	1.00	0.02	0.01	0.01	0.00	0.41
ICM/Str. Sample	1.00	1.00	1.00	0.95	0.70	0.22	0.72
ICM/Noise	0.85	0.86	1.00	0.16	0.95	1.00	0.50
	AHC (***average linkage***)						
CH	-	-	1.00	0.99	0.97	0.11	0.31
KL	-	-	0.75	0.88	0.64	1.00	0.26
Silhouette	-	-	1.00	0.99	1.00	1.00	0.29
Gap/Unif.	0.78	0.95	0.89	0.96	0.98	0.03	0.06
Gap/PC	0.89	0.99	0.60	0.97	1.00	1.00	0.07
Clest	0.86	0.14	0.12	0.18	0.19	0.85	0.16
Jump	0.05	1.00	1.00	0.99	0.87	0.00	0.42
PS	0.10	0.04	0.45	0.15	0.36	0.96	0.21
IGP	1.00	1.00	0.64	0.25	0.12	0.64	0.16
Rand	1.00	1.00	0.95	0.94	0.93	0.33	0.32
ICM/Str. Sample	1.00	1.00	1.00	0.95	1.00	1.00	0.26
ICM/Noise	1.00	1.00	1.00	1.00	1.00	1.00	0.10

Table 1: Low-dimensional simulations: % of success for finding the number of clusters.

For each entry in Table 1, one hundred realizations were generated from the model and the fraction of correct estimates for the cluster number estimation method was computed.

With models 1 and 2, if we exclude the CH, KL and Silhouette methods which are not defined for data with a single cluster, ICM, IGP and the Rand index did fairly well whether hierarchical clustering or k-means clustering was used. The Gap and Jump methods did well in the case of a single Gaussian cluster when hierarchical clustering was used. In contrast, the PS and Clest methods performed much better with k-means.

With model 3, all the estimates of the CH, KL, Silhouette, Jump and ICM methods were correct. The PS method did much better with k-means clustering and the Gap, IGP and specially Rand performed better using hierarchical clustering. The Clest method did poorly with both clustering methods.

With model 4, the ICM method with stratified sampling, CH, Silhouette and Jump did fairly well for the two used clustering methods and the fact that the four clusters of the model are not sized equally improved drastically the Gap method especially when k-means clustering was used. Furthermore, like the Rand index, the ICM method with random noise did better with hierarchical clustering compared with k-means clustering. The methods that did poorly with both clustering methods were Clest and IGP.

With model 5, the additional complexity of nonidentity covariance matrices adversely affected the performance of most of the methods when compared to those of model 4. The notable exceptions were the silhouette and ICM method with random noise, both of whom performed well with both clustering methods.

With model 6, the methods which almost always identified the two elongated clusters for the two used clustering methods were KL, Silhouette, Gap (PC reference distribution), and ICM with random noise. The ICM with stratified sampling and PS did much better for the hierarchical clustering method than for the k-means method. The CH, Gap (Uniform reference distribution), Jump and Rand index did poorly especially for the k-means method (0.00% of correct estimates).

With model 7, the performance of most of the methods declined: no method correctly estimated the number of clusters present in model 7 every time.

	Model 8	Model 9	Model 10	Model 8	Model 9	Model 10
Procedure	***AHC***			***k-means***		
CH	-	1.00	1.00	-	1.00	1.00
KL	-	1.00	1.00	-	1.00	1.00
Silhouette	-	1.00	1.00	-	1.00	1.00
Gap/Unif.	0.08	0.26	0.12	0.00	0.20	0.06
Gap/PC	0.64	1.00	0.48	0.45	1.00	1.00
Jump	1.00	1.00	1.00	1.00	1.00	1.00
PS	0.00	0.00	0.00	0.38	0.00	0.00
IGP	1.00	1.00	1.00	1.00	1.00	1.00
Rand	1.00	1.00	1.00	1.00	0.99	0.98
ICM/Str. Sample	1.00	1.00	1.00	1.00	1.00	0.95
ICM/Noise	1.00	1.00	1.00	1.00	1.00	1.00

Table 2: High-dimensional simulations: % of success for finding the number of clusters. For each entry, one hundred realizations were generated from the model.

For the high-dimensional simulations, all data were in one thousand dimensions. The Clest procedures were not considered in any high-dimensional simulations because they required too much computation time.

With model 8, where a single uniform cluster was present, the Jump method, IGP, Rand and ICM did well with the two used clustering method. The methods that almost always overestimated the number of clusters present were the PS and Gap (uniform reference distribution).

With model 9, when three clusters were present in the dataset, except the Gap (uniform reference distribution) and PS, all the methods performed perfectly in all cases.

With model 10, the results revealed that once the three clusters are not sized equally, all of the methods that correctly estimated the number of clusters in model 9 did also well with model 10 except the Gap (PC reference distribution) method, when hierarchical clustering was used, which did poorly with 48% off correct estimates.

4.2. Real datasets

We have also compared the ICM method with the other methods on three real datasets including the famous Fishers's Iris.

The second dataset is the Wisconsin Breast Cancer Database and is constituted of 699 instances described by ten attributes. Each instance has one of two possible classes: benign or malignant. We should note that owing to the presence of missing values, 18 instances were removed from the original dataset.

The third dataset is called Sonar and is constituted of 208 patterns: 111 patterns obtained by bouncing sonar signals off a metal cylinder at various angles and under various conditions and 97 patterns obtained from rocks under similar conditions. Each pattern is described by a set of 60 numbers in the range 0.0 to 1.0. Each number represents the energy within a particular frequency band, integrated over a certain period of time. A brief description of these datasets is presented in Table 3. More details on these datasets are available on the UCI Machine Learning Repository website. The results of this comparison are given in Table 4.

Data	Number of variables	Number of objects	True number of clusters
Fisher's Iris	4	150	3
Wisconsin Breast Cancer	10	681	2
Sonar	60	208	2

Table 3: A description of real datasets.

With the Fisher's Iris, the ICM method was among the six methods that detected the 3 clusters of this dataset. The methods that failed in finding this optimal number of clusters are Silhouette, Gap, KL and Jump.

With the Wisconsin Breast Cancer database, the methods that estimated correctly the number of clusters are ICM, KL, Silhouette and Gap.

Finally, with the Sonar database, only three methods have succeeded in estimating correctly the optimal number of clusters: ICM with stratified sampling, Silhouette and the CH index.

Procedure	Fisher's Iris	Breast-Cancer	Sonar	**Number of success**
CH	3	11	2	2
KL	9	2	3	1
Silhouette	2	2	2	2
Gap/Unif	2	2	1	1
Gap/PC	4	2	3	1
Clest	3	3	3	1
Jump	7	10	9	0
PS	3	5	1	1
IGP	3	8	1	1
ICM/Str. Sample	3	2	2	3
ICM/Noise	3	2	3	2
Number of success	6	6	3	

Table 4: Real datasets : each entry in the table is the optimal number of clusters estimated by the method.

4.3. Discussion

The simulation results showed that estimating the number of clusters present in a dataset is a hard task in so far as it depends at the same time on the characteristics of the dataset (shape of the clusters, dimensionality...), the classification algorithm and the parameters of the estimation method. This remark is illustrated by the important differences on the methods performances from a model to another and a classification algorithm to another. Nevertheless, based on the results presented in Table 5, our method, and more precisely ICM with stratified sampling, was ranked first with 89% of correct estimates when used with hierarchical clustering and was ranked fourth with 80% when used with k-means algorithm. This result is due to not only the good performances of this method on almost all the models but also because it always estimates the correct number of clusters in at least 22% of the datasets for each model with both clustering algorithms. A real advantage of this method is that it does not require selecting or using any specific defi-

nition of the notion of cluster.
The results of the PS method that was ranked third (resp. 15th) on the 7 low-dimensional models when k-means (resp. hierarchical clustering) was used, showed the extreme dependency of this method to the used classification algorithm. This was predictable since this method is based on a cross validation approach where the test set objects are classified to the group whose center is the nearest to this object.
The Silhouette method performed very well ont models 3 to 7 and was ranked first on these models. Nevertheless it has the major disadvantage of not being defined for data with a single cluster whereas our methods has always detected the single cluster with both clustering algorithms.

	k-means	*AHC*	Average success$^{(1)}$	Rank$^{(1)}$	Average success$^{(2)}$	Rank$^{(2)}$
ICM/Str. Sample		*	0.89	1	0.84	4
ICM/Rand. Noise		*	0.87	2	0.82	5
PS	*		0.85	3	0.78	6
ICM/Str. Sample	*		0.80	4	0.71	10
Gap/PC		*	0.79	5	0.72	8
Rand		*	0.78	6	0.69	12
ICM/Rand. Noise	*		0.76	7	0.72	8
Gap/Unif.		*	0.66	8	0.58	17
Jump		*	0.62	9	0.65	14
IGP	*		0.56	10	0.38	20
Jump	*		0.54	11	0.75	7
IGP		*	0.54	12	0.36	21
Clest	*		0.52	13	0.34	22
Gap/PC	*		0.46	14	0.60	15
Clest		*	0.36	15	0.30	23
Rand	*		0.35	16	0.09	24
PS		*	0.32	17	0.42	18
Gap/Unif.	*		0.28	18	0.37	19
Silhouette	*		-	-	0.92	1
KL	*		-	-	0.87	2
Silhouette		*	-	-	0.85	3
KL		*	-	-	0.70	11
CH		*	-	-	0.67	13
CH	*		-	-	0.60	15

Table 5: The column number three and four gives respectively the average percentage of success of each estimation procedure for the models (1-7) and the place of the estimation procedure based on the fraction of correct estimates for all 700 artificial datasets. The column number five and six gives respectively the average percentage of success of each estimation procedure for the models (3-7) and the place of the estimation procedure based on the fraction of correct estimates for these models.

References

[1] A. Ben-Hur, A. Elisseeff, I. Guyon, A stability based method for discovering structure in clustered data, in: Pacific Symposium on Biocomputing, 2002.

[2] R. B. Calinski, J. Harabasz, A dendrite method for cluster analysis, Communications in Statistics 3 (1974) 1–27.

[3] W. J. Krzanowski, Y. T. Lai, A criterion for determing the number of groups in data set using sum of squares clustering, Biometrics 44 (1985) 23–34.

[4] R. Tibshirani, G. Walther, T. Hastie, Estimating the number of clusters in a dataset via the gap statistic, Journal of Royal Statistical Society 32 (2) (2001) 411–423.

[5] E. Levine, E. Domany, Resampling method for unsupervised estimation of cluster validity, Neural Computation 13 (2001) 2573–2593.

[6] A. V. Kapp, R. Tibshirani, Are clusters found in one dataset present in another dataset?, Biostatistics 8 (1) (2007) 9–31.

[7] P. Bertrand, G. Bel Mufti, Loevinger's measures of rule quality for assessing cluster stability, Computational Statistics & Data Analysis 50 (4) (2006) 992–1015.

[8] J. N. Breckenridge, Replication cluster analysis: Method, consistency, and validity, Multivariate Behavior Research 24 (1989) 147–161.

[9] R. Tibshirani, G. Walther, Cluster validation by prediction strength, Journal of Computational & Graphical Statistics 14 (3) (2005) 511–528.

[10] G. W. Milligan, M. C. Cooper, An examination of procedures for determining the number of clusters in a data set, Psychometrika 50 (4) (1985) 159–179.

[11] M. Kerr, G. Churchill, Bootstrapping cluster analysis: assessing the reliability of conclusions from microarray experiments, in: the National Academy of Sciences of the United States of America, Vol. 98, Natl Acad. Sci USA, 2001, pp. 8961–8965.

[12] A. Kapp, Cluster analysis of microarray data using the in-group proportion, Ph.D. thesis, Stanford University (2007).

[13] S. Ben-David, U. von Luxburg, D. Pal, A sober look at clustering stability, Springer, Berlin, Germany, 2006, pp. 5–19.

[14] O. Shamir, N. Tishby, On the reliability of clustering stability in the large sample regime, in: NIPS, 2008, pp. 1465–1472.

[15] H. W. Hansen, M.H., W. Madow, Sample Survey Methods and Theory, Methods and Applications, Wiley, New York, 1993.

[16] A. Hardy, On the number of clusters, Computational Statistics and Data Analysis 23 (1996) 83–96.

[17] P. J. Rousseeuw, Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, Journal of Computational and Applied Mathematics 20 (1987) 53–65.

[18] C. A. Sugar, G. M. James, Finding the number of clusters in a data set: An information theoretic approach, Journal of the American Statistical Association 98 (463) (2003) 750–763.

[19] S. Dudoit, J. Fridlyand, A prediction-based resampling method for estimating the number of clusters in a dataset, Genome Biology (7).