Available online at www.sciencedirect.com
ScienceDirect
Computational Statistics & Data Analysis 52 (2007) 258–271
COMPUTATIONAL STATISTICS & DATA ANALYSIS
www.elsevier.com/locate/csda

Cluster-wise assessment of cluster stability

Christian Hennig*

Department of Statistical Science, University College London, Gower St., London WC1E 6BT, UK

Available online 12 December 2006

Abstract

Stability in cluster analysis is strongly dependent on the data set, especially on how well separated and how homogeneous the clusters are. In the same clustering, some clusters may be very stable and others may be extremely unstable. The Jaccard coefficient, a similarity measure between sets, is used as a cluster-wise measure of cluster stability, which is assessed by the bootstrap distribution of the Jaccard coefficient for every single cluster of a clustering compared to the most similar cluster in the bootstrapped data sets. This can be applied to very general cluster analysis methods. Some alternative resampling methods are investigated as well, namely subsetting, jittering the data points and replacing some data points by artificial noise points. The different methods are compared by means of a simulation study. A data example illustrates the use of the cluster-wise stability assessment to distinguish between meaningful stable and spurious clusters, but it is also shown that clusters are sometimes only stable because of the inflexibility of certain clustering methods.

Keywords: Cluster validation; Bootstrap; Robustness; Clustering with noise; Jaccard coefficient

1. Introduction

Validation is very important in cluster analysis, because clustering methods tend to generate clusterings even for fairly homogeneous data sets. Most clustering methods assume a certain model or prototype for clusters, and this may be adequate for some parts of data, but not for others. Cluster analysis is often carried out in an exploratory manner, and the patterns found by cluster analysis are not necessarily meaningful.

An important aspect of cluster validity is stability. Stability means that a meaningful valid cluster should not disappear easily if the data set is changed in a non-essential way. There could be several conceptions what a "non-essential change" of the data set is. In terms of statistical modelling it could be demanded that a data set drawn from the same underlying distribution should give rise to more or less the same clustering (though the true underlying distribution is unknown). It could also be of interest whether clusterings remain stable under the addition of outliers, under subsetting or under "jittering", i.e., the addition of a random error to every point to simulate measurement errors.

* Tel.: +44 20 7679 1698; fax: +44 20 7383 4703.
E-mail address: chrish@stats.ucl.ac.uk.
URL: http://www.homepages.ucl.ac.uk/∼ucakche.

0167-9473/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.csda.2006.11.025

Given a clustering on a data set generated by a clustering method, the following principle is discussed in the present paper:

- Interpret the Jaccard coefficient (Jaccard, 1901) as a measure of similarity between two subsets of a set based on set membership.
- Resample new data sets from the original one (using various strategies) and apply the clustering method to them.
- For every given cluster in the original clustering find the most similar cluster in the new clustering and record the similarity value.
- Assess the cluster stability of every single cluster by the mean similarity taken over the resampled data sets.

It appears to be quite natural to assess cluster stability by resampling methods, and this has been done in several recent papers, most of them related to the analysis of gene expression data. Examples are Ben-Hur et al. (2002), Bryan (2004), Dudoit and Fridlyand (2002), Grün and Leisch (2004), Lange et al. (2004), Monti et al. (2001) and Tibshirani and Walther (2005). Many of these papers use stability or prediction strength measurements as a tool to estimate the true number of clusters.

The approach taken in the present article has the following two important characteristics:

- It is applicable to very general clustering methods including methods based on (not necessarily metric) dissimilarity measures, non-partitioning methods and methods that include an estimator of the number of clusters (so that the determination of this number is not an aim of the present approach), as well as conventional methods based on Euclidean data with a fixed number of clusters such as k-means. No particular cluster model is assumed.
- The approach is cluster-wise. The idea behind this is that many data sets contain meaningful clusters for which a certain cluster model is adequate, but they do not necessarily consist *only* of such clusters. Therefore, the result of a clustering method could find some important meaningful patterns in the data set, while other clusters in the same clustering can be spurious. The reason for this is not necessarily the choice of the wrong clustering method; it may well be that no single method delivers a satisfactory result for the whole data set. Note that none of the approaches in the literature cited above is cluster-wise.

As an example consider the data set in Fig. 1, which is described in more detail in Section 5. The data consists of 366 points in four dimensions and has been generated by classical multidimensional scaling on a dissimilarity matrix. Therefore the first two dimensions shown in Fig. 1 are the first two principal components. The plot suggests that there are some patterns in the data set, but many points do not seem to belong to such a pattern. Neither the more nor the less clustered parts clearly suggest a fit with a standard parametric distribution such as the normal or the uniform distribution. This impression can be backed up by more sophisticated visual analyses (some patterns become a bit clearer if all dimensions are considered; not shown).

The clustering shown in Fig. 1 has been obtained by a normal mixture model with unrestricted covariance matrices for the mixture components and a noise component modelled as a uniform distribution on the convex hull of the data. The number of clusters has been estimated by the Bayesian information criterion. The procedure is explained in Fraley and Raftery (1998) and implemented in the package MCLUST for the statistical software R. A tuning constant for the initial estimation of the noise component has to be specified and was chosen as $h = 10$, so that the distinction between noise and non-noise points has been made based on the 10th nearest neighbor of every point, see Byers and Raftery (1998). This is implemented in the R-package PRABCLUS. Several clustering methods have been carried out on this data set, but none of these leads to more convincing results.

Usually, in such an analysis, the normal components are interpreted as clusters, but this does not seem to be reasonable for all components in the given data set. This motivates the cluster-wise approach: it would be very helpful to know to what extent the normal components can be interpreted as stable patterns of the data, and it can reasonably be suspected that this applies to some but not all of the components. The methods suggested in the present paper confirm stability only for the cluster nos. 1, 7 and 8, see Section 5.

Stability is not the only aspect of cluster validity, and therefore a stable cluster is not guaranteed to be a meaningful pattern. With another clustering of the same data set, it will be illustrated why meaningless clusters sometimes are stable.

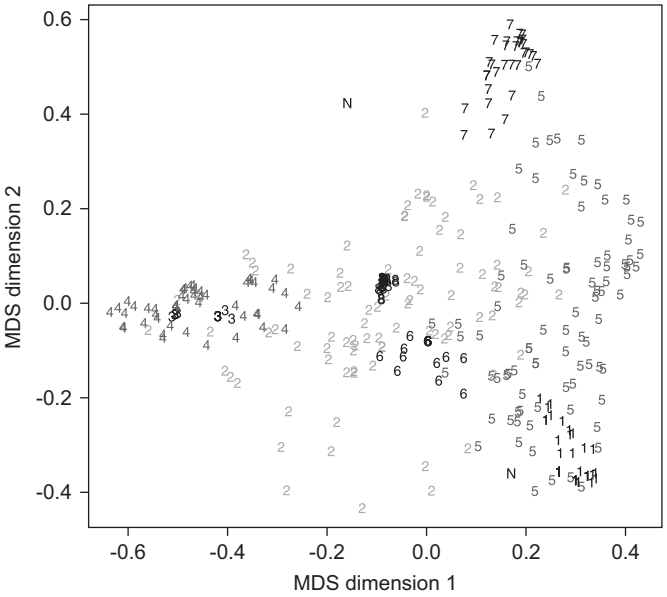


Fig. 1. First two MDS dimensions of snails distribution ranges data set with clustering generated by normal mixture clustering with BIC and noise (cluster no. 8 is the packed one in the center). "N" indicates points estimated as noise.

Some alternative methods of cluster validation are homogeneity and/or separation-based validation indexes, comparison of different clustering methods on the same data, visual cluster validation, tests of homogeneity of the data set against a clustering alternative and use of external information, see Gordon (1999), Haldiki et al. (2002), Hennig (2004b, 2005), Milligan and Cooper (1985) and the references given therein.

The analysis of the sensitivity of a clustering against perturbations of the data has a long history as well, see, e.g., Rand (1971) and Milligan (1996). The adjusted Rand index (Hubert and Arabie, 1985) has been used often to measure the similarity between two complete clusterings.

Some work on robustness properties in cluster analysis (e.g., Garcia-Escudero and Gordaliza, 1999; Hennig, 2004a) is also related to the assessment of stability in cluster analysis. It turns out in this work that classical robustness concepts such as the finite sample breakdown point (Donoho and Huber, 1983) are heavily data dependent when applied to cluster analysis.

The paper proceeds as follows. The basic method, based on a non-parametric bootstrap, is introduced in Section 2. Section 3 discusses some alternative approaches to carry out the resampling. The approaches are compared in Section 4 by means of a simulation study. Section 5 applies the methodology to the snails distribution ranges data and a concluding discussion is given in Section 6.

2. Bootstrapping the Jaccard coefficient

A sequence of mappings $E = (E_n)_{n\in\mathbb{N}}$ is called a general clustering method, if E_n maps a set of entities $\mathbf{x}_n = \{x_1, \ldots, x_n\}$ (this is how $\mathbf{x}_n$ is always defined throughout the paper) onto a collection of subsets $\{C_1, \ldots, C_s\}$ of $\mathbf{x}_n$. Note that it is assumed that entities with different indexes can be distinguished. This means that the elements of $\mathbf{x}_n$ are interpreted as data points and that $|\mathbf{x}_n| = n$ is even if, for example, for $i \neq j$, $x_i = x_j$ in terms of their values. It is not assumed how the entities are defined. This could be, e.g., via a dissimilarity matrix or via p Euclidean variables.

Most clustering methods generate disjoint clusterings, i.e., $C_i \cap C_j = \emptyset$ for $i \neq j \leqslant k$. A partition is defined by $\bigcup_{j=1}^{k} C_j = \mathbf{x}_n$. The methodology defined here does not necessarily assume that the clustering method is disjoint or a partition, but the interpretation of similarity values between clusters is easier for methods that do not generate a too rich clustering structure. For example, if the clustering method generates a full hierarchy, every subset containing only one point is always a cluster and these clusters will be perfectly stable, though totally meaningless.

To assess the stability of a cluster of the initial clustering with respect to a new clustering, a similarity measure between clusters is needed. Because the measure should be applicable to general clustering methods (even methods that do not operate on the Euclidean space), it has to be based on set memberships.

There exist many similarity measures between sets, see e.g., Gower and Legendre (1986). I suggest the Jaccard coefficient, which originated in the analysis of species distribution data (Jaccard, 1901):

$$\gamma(C, D) = \frac{|C \cap D|}{|C \cup D|}, \quad C, D \subseteq \mathbf{x}_n.$$

The Jaccard coefficient gives the proportion of points belonging to both sets of all the points involved in at least one of the sets, and it is therefore easily directly interpretable. It has several good properties, e.g., being independent of the number of points not belonging to any of the two sets. $1-\gamma$ is a metric (Gower and Legendre, 1986). Hennig (2006) suggested the use of the Jaccard coefficient to compare cluster analysis methods theoretically, and he defined the value $\frac{1}{2}$ as a critical value for so-called "dissolution" of a cluster under addition of points to the data set. It can be shown that $\frac{1}{2}$ is the smallest value so that every cluster in a partition consisting of more than one cluster can be dissolved by a new partition, and it is also the smallest value so that whenever an initial cluster has s clusters and a new clustering has $r < s$ clusters, then at least $s - r$ clusters of the original clustering are dissolved (equivalent results are shown in Hennig, 2004a). The Jaccard coefficient has also been used by Ben-Hur et al. (2002) in the context of cluster validation with resampling methods, though not for cluster-wise evaluation.

The idea behind the use of the non-parametric bootstrap for the assessment of cluster stability is the following: assume that there is an underlying mixture distribution $P = \sum_{i=1}^{s} \varepsilon_i P_i$ where $P_i,\ i = 1, \ldots, s$, are the distributions generating s "true" clusters, and ε_i is the probability that a point from P_i is drawn. For a given data set with n points, the "true" clustering would then consist of s clusters each of which contains exactly the points generated by $P_i,\ i = 1, \ldots, s$. When a data set generated from P is clustered, the found clusters differ from the "true" clusters, because the clustering method introduces a certain bias and variation. This can depend on the cluster P_i, for example, if two different clusters are weakly separated or if P_i deviates strongly from the cluster model assumed by the clustering method. Bias and variation can be expressed by the maximum Jaccard coefficient between the set of all the points generated by P_i and the most similar cluster in the actually obtained clustering.

The bootstrap is usually used to give an idea of bias and variation caused by a certain statistical method, because in reality no true underlying distribution and no true clustering is known. The empirical distribution of the observed data set is then taken to simulate P. Points can be drawn from the data set and the originally found clusters can be treated as the "true" ones. The mean maximal Jaccard coefficient can be interpreted as indicating the stability of the original clusters.

Given a number B of bootstrap replications and a cluster C from the original clustering $E_n(\mathbf{x})$, the scheme works as follows. Repeat for $i = 1, \ldots, B$:

(1) Draw a bootstrap sample $\mathbf{x}_n^i$ of n points with replacement from the original data set $\mathbf{x}_n$.
(2) Compute the clustering $E_n(\mathbf{x}_n^i)$.
(3) Let $\mathbf{x}_*^i = \mathbf{x}_n \cap \mathbf{x}_n^i$ be the points of the original data set that are also in the bootstrap sample. Let $C_*^i = C \cap \mathbf{x}_*^i,\ \Delta = E_n(\mathbf{x}_n^i) \cap \mathbf{x}_*^i$.
(4) If $C_*^i \neq \emptyset$, compute the maximum Jaccard similarity between the induced cluster C_*^i and the induced new clustering Δ on $\mathbf{x}_*^i$: $\gamma_{C,i} = \max_{D \in \Delta} \gamma(C_*^i, D)$ (i.e., D is the maximizer of $\gamma(C_*^i, D)$; else $\gamma_{C,i} = 0$).

This generates a sequence $\gamma_{C,i},\ i = 1, \ldots, B$. I suggest the mean

$$\bar{\gamma}_C = \frac{1}{B^*} \sum_{i=1}^{B} \gamma_{C,i}$$

as stability measure (B^* being the number of bootstrap replications for which $C_*^i \neq \emptyset$ and is used here because in all other cases $\gamma_{C,i} = 0$).

Other summary statistics such as the median, a trimmed mean or the number of dissolutions ($\gamma_{C,i} \leqslant 0.5$) or good recoveries ($\gamma_{C,i} > 0.75$, say) can be used as well. Experience suggests that the mean is a good choice here; in all

examples in which I examined further statistics, I did not find any results that deviated strongly from what could be expected by looking at the mean alone. The value range and therefore also the size of possible outliers affecting the mean are restricted to [0, 1] and if moderate outliers occur, they may be treated as informative and need presumably not to be downweighted or trimmed.

The given scheme compares clusters based on the data set $\mathbf{x}_*^i$ in which every point of the original data set appears only once. An alternative would be to compare clusters on the bootstrap data set $\mathbf{x}_n^i$ with a version of the original cluster in which points are repeatedly included if they also occur repeatedly in the bootstrap sample. This, however, would have the disadvantage that clusters dominated by multiple points would be upweighted in the computation of $\bar{\gamma}_C$.

Note that a parametric bootstrap does not suggest itself for the aim of the present paper, because parametric methods discover structures really generated by the underlying model (what the resampled data sets in parametric bootstrap are) much better than patterns in real data for which the models do not hold. Therefore, the parametric bootstrap can be expected to yield much too optimistic stability assessments at least for methods based on the used parametric model.

3. Alternative resampling and simulation schemes

The non-parametric bootstrap is not the only possibility to generate new similar but somewhat distorted data sets from the original data set, which can be used to assess stability. As already observed by Monti et al. (2001), a disadvantage of non-parametric bootstrap particularly in connection with cluster analysis is the occurrence of multiple points in the bootstrapped data set. Multiple points can be seen as miniclusters in itself. For some implementations of clustering and multidimensional scaling methods multiple points cause numerical problems. The following schemes avoid multiple points. Note that subsetting is as general as bootstrapping because it can be applied with arbitrary data formats including dissimilarity matrices. The other two alternative schemes, replacing points by noise and jittering, can in the present state only be applied to p-dimensional Euclidean data (though suitable versions for dissimilarity data are conceivable).

3.1. Subsetting

The simplest idea is to draw a subsample of $\mathbf{x}_n$ without replacement instead of a bootstrap sample. This avoids multiple points and shortens computation times, which can be an issue with large data sets. The scheme of Section 2 can be carried out as before with $\mathbf{x}_*^i$ now being the drawn subsample of $\mathbf{x}_n$.

Subsetting requires the choice of the size $m < n$ of the subsample. If m is too large, subsetting will not generate enough variation to be informative. If m is too small, the clustering results can be expected to be much worse than that obtained from the original data set. I always worked with $m = [n/2]$, where "$[x]$" denotes the integer part of x.

An alternative subsetting method would be to discard multiple points in a bootstrap scheme.

3.2. Replacing points by noise

The instability of statistical methods can often be demonstrated by replacing some points in the data set by "noise points" or outliers. The definition of the finite sample replacement breakdown point (Donoho and Huber, 1983) is based on this idea as well. In cluster analysis, the replacement of points by noise can be seen as an exploration of the strength of a pattern. It is stable in this sense if it can still easily be found by a clustering algorithm in spite of the contamination of the data set.

Instead of drawing a bootstrap sample, a certain number m of points from $\mathbf{x}_n$ can be drawn without replacement. These points then have to be replaced by points drawn from a noise distribution.

The basic scheme from Section 2 can again be applied. The subset $\mathbf{x}_*^i$, on which the clusterings are to be compared, is now the set of the remaining $n - m$ non-noise points.

Two choices have to be made, namely the number m of points to be replaced, and the noise distribution. The noise distribution is difficult to choose. Noise points should be allowed to lie far away from the bulk (or bulks) of the data, but it may also be interesting to have noise points in between the clusters, possibly weakening their separation.

I suggest the following strategy:

(1) Sphere and center the data set so that it has the identity matrix as covariance matrix and the zero vector as mean vector (this is done for notational convenience in the next step).
(2) Draw noise points from a uniform distribution on a hyperrectangle with a not too small range in all directions. I used two versions in the simulations, namely a range of $[-3, 3]$ together with $m = 0.05n$ and $[-4, 4]$ together with $m = 0.2n$.
(3) If a clustering method is used that is not affine equivariant, rotate the data back to the original coordinate system.

This is based on the classical covariance matrix, which I suggest because it contains information of all points including different clusters and outliers in the data, so that this scheme usually can generate extreme points as well as points between clusters. An alternative would be the convex hull of the data set (possibly blown up by a factor), which is more likely to generate only outliers in the case that there are already extreme outliers in the data.

3.3. Jittering and bootstrap/jittering

"Jittering" means that a small amount of noise is added to every single point. This represents the idea that all points may include measurement errors and the information in them is therefore somewhat fuzzy. It is then interesting whether clusters are stable with respect to the addition of further measurement error.

In the scheme from Section 2, $\mathbf{x}_n^i = \{y_1, \ldots, y_n\}$ has to be the jittered data set with $y_k = x_k + e_k,\ k = 1, \ldots, n$, e_k being the simulated measurement error. In Step (3), $C_*^i = \{k : \ x_k \in C\}$ is now the set of index numbers of points in cluster C and Δ is a clustering of index numbers consisting of sets $D_* = \{k : \ y_k \in D\}$ for all clusters $D \in E_n(\mathbf{x}_n^i)$. An additional set $\mathbf{x}_*^i$ is not needed.

By analogy to Section 3.2, a measurement error distribution has to be chosen. Here is a suggestion:

(1) Sphere and center the data set so that it has the identity matrix as covariance matrix and the zero vector as mean vector.
(2) For all p dimensions, compute the $n - 1$ differences d_{ij} between (one-dimensionally) neighboring points: for $i = 1, \ldots, n - 1,\ j = 1, \ldots, p,\ d_{ij}$ is the difference between the $(i + 1)$th and the ith order statistic of the jth component of the data set $\mathbf{x}_n$. For $j = 1, \ldots, p$, let q_j be the empirical q-quantile of the d_{ij} (q is a tuning constant).
(3) Draw noise $e_k,\ k = 1, \ldots, n$, i.i.d. from a p-dimensional normal distribution with a zero mean vector and a diagonal matrix as covariance matrix with diagonal elements $\sigma_1^2 = q_1^2, \ldots, \sigma_p^2 = q_p^2$ and compute $y_k = x_k + e_k$ for $k = 1, \ldots, n$.
(4) If a clustering method is used that is not affine equivariant, rotate the data back to the original coordinate system.

The normal distribution is traditionally used as measurement error distribution. Two versions for the component-wise standard deviation have been chosen for the simulations, namely the 0.1- and 0.25-quantile of the one-dimensional distances between neighboring points. It is important that the chosen quantile is smaller than the distance between two points from different clusters which are well separated from each other. The idea is that the order of points (along a single dimension) should change from time to time but not often by the introduction of the measurement error.

The jittering idea can also be combined with the bootstrap scheme from Section 2, so that a simulated measurement error is added to the bootstrap data set. This avoids the problem of multiple points in the bootstrap scheme. The choice of the small 0.1-quantile above may be suitable for this application.

4. A simulation study

To assess the performance of a method for cluster stability assessment, it is necessary to find out whether the method can distinguish "better" from "worse" clusters in the same clustering. Therefore data sets have to be constructed in which there are well-defined "true" clusters. Clustering methods have to be applied which make some sense for this kind of data, but which do not necessarily find all of the clusters, and which do not necessarily have to be the best methods for these data.

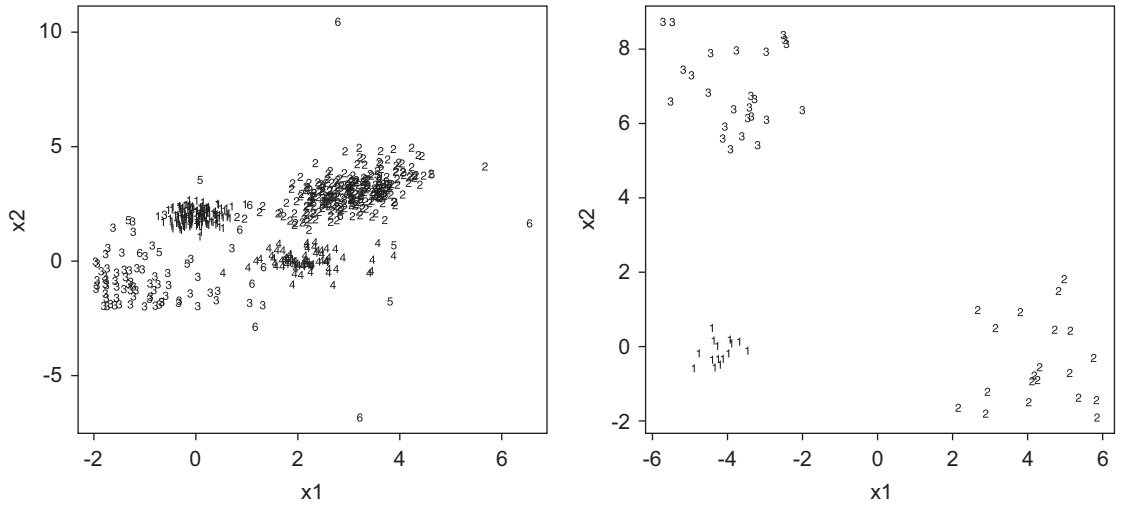


Fig. 2. Simulated data from model 1 (dimensions 1 and 2, left side) and model 2 (right side).

Data have been simulated from two models:

Model 1: This model is designed to generate data in which a clustering is clearly visible, but the data should include some realistic problems such as outliers and cluster distributions with different shapes. The first two dimensions of data generated from model 1 are shown on the left side of Fig. 2. Dimensions 3 and 4 look similarly.

The model generates six-dimensional data. The dimensions 5 and 6 are noise, dimension 5 distributed i.i.d. $N(0, 1)$, dimension 6 i.i.d. t_2, so that large outliers may occur in dimension 6. The clustering is present in the dimensions 1–4.

The model for these dimensions consists of six submodels. Submodels 1–4 correspond to four clusters, submodels 5 and 6 generate "noise" not belonging to any cluster.

Submodel (*cluster*) 1 (150 *points*): Normal distribution with mean vector (0, 2, 0, 2) and covariance matrix $0.1\mathbf{I}_4$.

Submodel (*cluster*) 2 (250 *points*): Normal distribution with mean vector (3, 3, 3, 3) and a covariance matrix with diagonal elements 0.5 and covariances 0.25 in all off-diagonals.

Submodel (*cluster*) 3 (70 *points*): A skew cluster with all four dimensions distributed independently exponentially (1) shifted so that the mean vector is $(-1, -1, -1, -1)$.

Submodel (*cluster*) 4 (70 *points*): 4-variate t_2-distribution with mean vector (2, 0, 2, 0) and covariance matrix $0.1\mathbf{I}_4$ (this is the covariance matrix of the normal distribution involved in the definition of the multivariate t-distribution).

Submodel (*noise*) 5 (10 *points*): Uniform distribution on $[-2, 5]^4$.

Submodel (*noise*) 6 (10 points): 4-variate t_2-distribution with mean vector (1.5, 1.5, 1.5, 1.5) and covariance matrix $2\mathbf{I}_4$.

Model 2: This is a two-dimensional toy example which is designed to find out whether "real" clusters that are found easily are more stable than clusters which are erroneously split up by the clustering method. Data generated from model 2 are shown on the right side of Fig. 2.

Cluster 1 (15 *points*): Normal distribution with mean vector $(-4, 0)$ and covariance matrix $0.1\mathbf{I}_2$.

Cluster 2 (20 *points*): Uniform distribution on $[2, 6] \times [-2, 2]$.

Cluster 3 (25 *points*): Uniform distribution on $[-6, -2] \times [5, 9]$.

Model 1 has been analyzed by three clustering methods, namely the normal mixture plus noise approach (Fraley and Raftery, 1998) as explained in the Introduction, 10% trimmed 5-means on data scaled to variance 1 for all variables (Cuesta-Albertos et al., 1997; the best out of four local minima iterated from random initial partitions has been taken) and 4-means on the unscaled data (for k-means only one iteration has been carried out—this method has been included to show the performance on a fast and simple but not very adequate standard method).

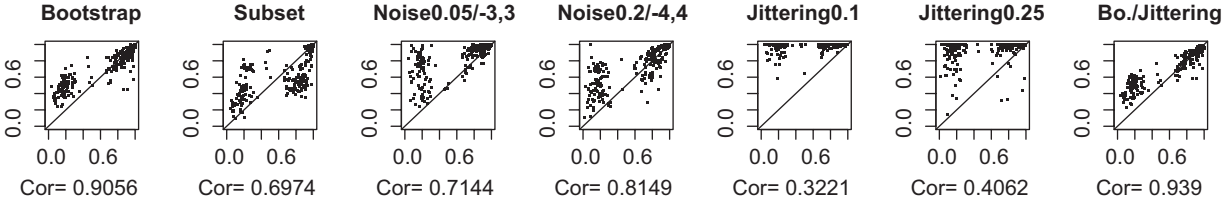
Model 2 has been analyzed by 6-means and between-groups average linkage clustering on Euclidean distances from scaled data. For average linkage, a partition was obtained by cutting the tree so that there are five clusters. As an example for overlapping clustering, the resampling schemes have also been applied to the full cluster hierarchy from average linkage, i.e., all clusters occurring in any stage of the hierarchy have been used, omitting only the trivial clusters with all points and only one point.

For a given model and cluster analysis method, 50 data sets have been generated from the model (called "model data" from now on), and then $B = 50$ repetitions have been taken for every resampling scheme. For each scheme, the average maximum Jaccard coefficients $\bar{\gamma}_C$ have been recorded for all clusters found in the model data. Furthermore, for every "true" cluster from the model, the most similar cluster found in the model data has been determined to find out how well the true clusters have been recovered by the clustering methods. The desired result is then that the found clusters in the model data that match a true cluster well yield high-stability values $\bar{\gamma}_C$, while "meaningless" clusters not corresponding well to any true submodel should yield low-stability values. Note, though, that $\bar{\gamma}_C$ is not an "estimator" for the similarity of C to a true cluster in any technical sense. It may happen that true clusters are inherently unstable (generally, or with respect to a particular clustering method), and in this case $\bar{\gamma}_C$ should be small even if a found C matches the truth almost perfectly. It may also happen that the underlying cluster model of some methods do not match the true cluster model, and such methods may generate quite stable partitions of the data which do not match the true clusters. Note that, without any parametric assumption, "true clusters" could even be mixtures and are therefore, strictly speaking, not uniquely identified.

For every simulation setup, the following information is shown:

- Correlation and scatterplot of $\bar{\gamma}_C$ (y-axis) versus the Jaccard coefficient between C and the most similar true cluster over all clusters found in the model data (x-axis). The identity line is added to help the eye. Note that more than 50 clusters are present in each scatter diagram, because more than one cluster has usually been found in the model data. Therefore there is dependence between results for clusters belonging to the same model data set. In model 1 the submodels 5 and 6 were allowed to be found as "most similar true cluster" as well.
- A "true cluster"-wise table shows the mean Jaccard coefficient between a true cluster and the most similar cluster found in the model data ("Best") and the mean of the $\bar{\gamma}_C$ values for the corresponding most similar clusters in the 50 model data sets. "Boot" refers to bootstrap, "Sub" refers to subsetting, "N0.05/3" refers to an addition of noise with $m = 0.05n$ and range $[-3, 3]$ for the uniform distribution ("N0.2/4" by analogy), "J0.1" refers to jittering with 0.1-quantile ("J0.25" by analogy) and "B/J" refers to bootstrapping plus jittering with 0.1-quantile.

4.1. Model 1, normal mixture plus noise

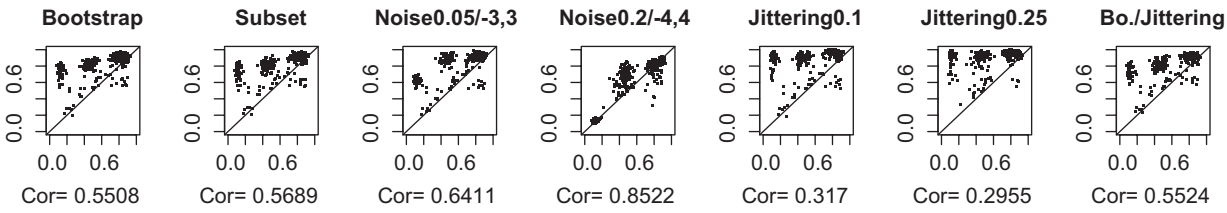


True	Best	Boot	Sub	N0.05/3	N0.2/4	J0.1	J0.25	B/J
1	0.949	0.944	0.95	0.973	0.973	1	1	0.943
2	0.935	0.937	0.955	0.957	0.957	0.998	0.998	0.928
3	0.771	0.764	0.568	0.813	0.813	0.978	0.978	0.776
4	0.778	0.8	0.569	0.888	0.888	0.98	0.98	0.82

The true clusters 1 and 2 were generally found successfully by the clustering method and this has been confirmed by all resampling schemes. The true clusters 3 and 4 have been identified fairly well, but this was often unstable under subsetting. Bootstrap and bootstrap/jittering performed best in terms of correlation. Jittering alone did not produce useful results. Note that the scatter diagrams are clustered along the x-axis, because a clear distinction in terms of the

Jaccard similarity can be made between properly found true clusters in the model data and some meaningless clusters occurring in the mixture clusterings.

4.2. Model 1, 10%-trimmed 5-means

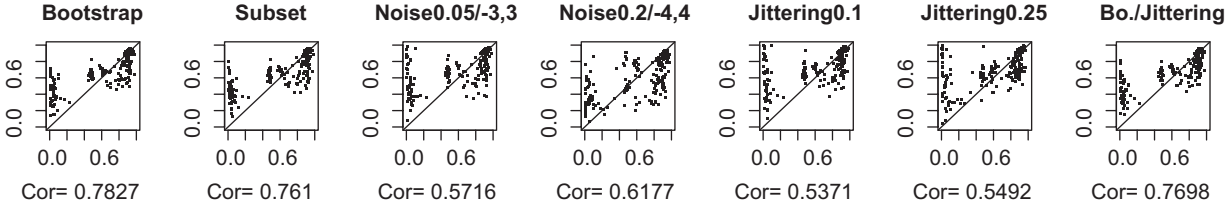


True	Best	Boot	Sub	N0.05/3	N0.2/4	J0.1	J0.25	B/J
1	0.832	0.851	0.855	0.87	0.78	0.889	0.894	0.858
2	0.562	0.789	0.782	0.83	0.706	0.851	0.875	0.793
3	0.761	0.885	0.886	0.889	0.799	0.938	0.95	0.888
4	0.784	0.886	0.886	0.893	0.742	0.935	0.939	0.887

Trimmed k-means assumes all within-cluster covariance matrices to be spherical and equal. Therefore it tends to split up cluster 2 into two parts. This was done quite consistently, and therefore the stability values corresponding to the true cluster 2 look better than the recovery of cluster 2 really is. However, the schemes detect that the model data clusters corresponding to cluster 2 are less stable than the others. In terms of correlation, noise $0.2/[-4, 4]$ is the best scheme; jittering alone performs badly again.

The three clusters along the x-axis of the scatter diagram now correspond to meaningless clusters in the trimmed k-means solution, parts of the cluster 2 and clusters corresponding to the model clusters 1, 3 and 4. The $\bar{\gamma}_C$ values are rather higher than the values for true cluster recovery. Only in some cases (points close to the identity line) their values are similar.

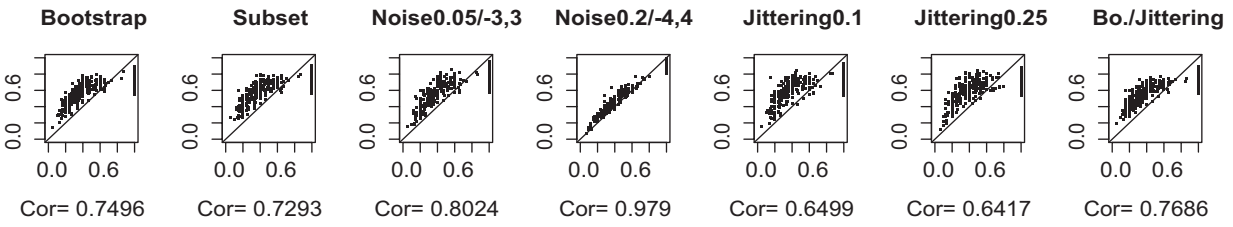
4.3. Model 1, 4-means



True	Best	Boot	Sub	N0.05/3	N0.2/4	J0.1	J0.25	B/J
1	0.749	0.74	0.74	0.743	0.662	0.761	0.721	0.746
2	0.918	0.913	0.92	0.927	0.86	0.937	0.919	0.915
3	0.576	0.643	0.644	0.646	0.615	0.659	0.652	0.65
4	0.489	0.631	0.629	0.636	0.576	0.654	0.644	0.641

The 4-means seems to find cluster 2 reliably, the recovery of cluster 1 is fairly good and the clusters 3 and 4 are not found properly. Generally, this is detected by all resampling schemes, with bootstrap, subsetting and bootstrap/jittering performing better than noise and jittering alone. The four clusters along the x-axis of the scatter diagrams correspond to meaningless clusters, versions of cluster 4, versions of cluster 3, versions of cluster 1 and 2.

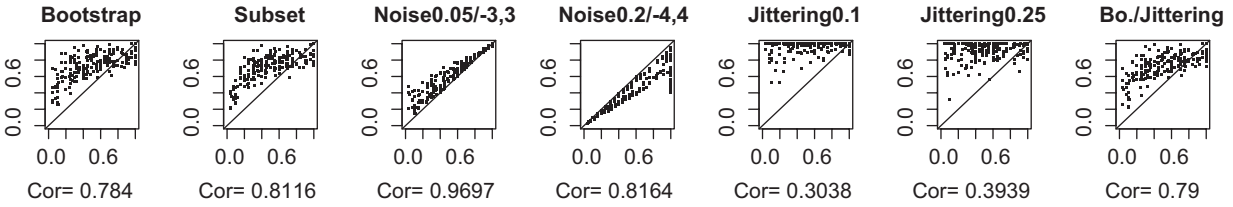
4.4. Model 2, 6-means



True	Best	Boot	Sub	N0.05/3	N0.2/4	J0.1	J0.25	B/J
1	0.85	0.75	0.754	0.793	0.808	0.762	0.755	0.755
2	0.661	0.654	0.674	0.673	0.679	0.673	0.658	0.658
3	0.598	0.644	0.638	0.657	0.629	0.656	0.667	0.64

The 6-means does not do a very good job here. Cluster 1 is found often, but sometimes split up. Even if it is found perfectly, this is often not stable under the resampling schemes. Therefore its stability is underestimated. However, the stability values from noise $0.2/[-4, 4]$ correlate almost perfectly with the similarities to the true clusters. In the scatter diagrams it can be seen that model clusters have been perfectly recovered several times (value 1 on the x-axis), but even these clusters were not perfectly stable under resampling.

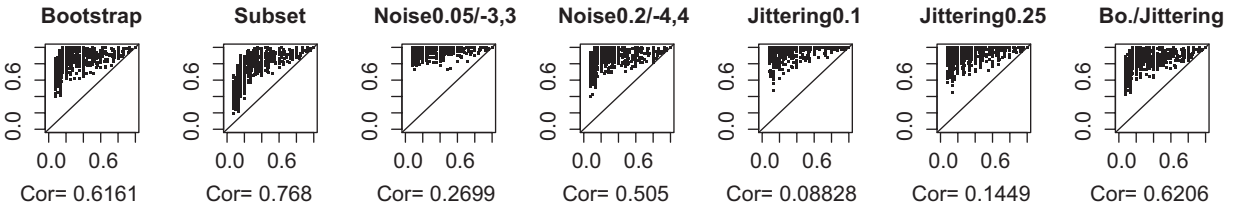
4.5. Model 2, average linkage partition



True	Best	Boot	Sub	N0.05/3	N0.2/4	J0.1	J0.25	B/J
1	1	1	1	0.992	0.48	1	1	1
2	0.698	0.812	0.813	0.753	0.651	0.972	0.929	0.813
3	0.741	0.798	0.793	0.784	0.477	0.972	0.929	0.795

The average linkage partition always finds the true cluster 1 perfectly, and this is reproduced by the resampling schemes except the addition of noise points. Replacing 20% of the points by noise seems to destroy this cluster completely, indicating a serious robustness problem of average linkage (with fixed number of clusters) in this situation. This is informative, even though noise $0.05/[-3, 3]$ is the best method in terms of correlation. Subsetting is better than the bootstrap schemes here and jittering is bad, as almost always.

4.6. Model 2, average linkage full hierarchy



The means table is omitted in this situation because it consists only of ones. This means that the true clusters have always been found somewhere in the hierarchy, for the model data and for the resampled data. The problem is that it is difficult for the user to figure out which of the many sets in the hierarchy the good clusters are (given that it is not as easy to visualize the data in reality as it is in the given toy example). Therefore the correlation analysis is still informative, and it seems that almost all $\bar{\gamma}_C$ values are larger than or about equal to the similarities of the found clusters to the true clusters. Therefore, a small $\bar{\gamma}_C$ value is a reliable indicator that a cluster is not meaningful here, while meaningless clusters could easily produce high-stability values because of the richness of the clustering structure. In terms of correlation, subsetting is the best scheme here, followed by bootstrap/jittering, bootstrap, the (somewhat disappointing) noise schemes and jittering.

4.7. Simulation results: summary

Generally, large stability values do not necessarily indicate valid clusters, but small stability values are informative. Either they correspond to meaningless clusters (in terms of the true underlying models), or they indicate inherent instabilities in clusters or clustering methods.

The two "jittering alone" schemes were always outperformed by the other schemes. The ranking among the other schemes depends on data and clustering method, so that no clear recommendation can be given. The noise methods brought different (but valuable) information to light than the other methods. The correlations between bootstrap and subsetting (not shown) have generally been 0.8 and higher, so that it may be worthwhile to use one noise scheme and one scheme out of bootstrap, bootstrap/jittering and subsetting in practice. Subsetting was the best of these three for the two setups with average linkage; bootstrap/jittering was usually a bit better than bootstrap alone, but the latter can be applied more generally.

5. Data example

Every point in the data set shown in Fig. 1 represents a distribution range of a species of snails in North-Western Europe. The data have been generated from a 0–1 matrix indicating whether each of the 366 species (data points) involved are present on each of 306 grid squares of a grid spanning a map of North-Western Europe. Clustering of such distribution ranges is interesting because some theories about the species differentiation process predict the occurrence (and a particular decomposition) of such clusters. Dissimilarities between the distribution ranges (i.e., a $366 * 366$-dissimilarity matrix) have been computed by the Kulczynski coefficient, see Hausdorf and Hennig (2003) and Hennig and Hausdorf (2004) for details.

Table 1 gives stability results for the eight clusters and the noise component estimated by a normal mixture method with noise as explained in the Introduction and shown in Fig. 1. The initial data set consists of dissimilarities. Therefore, the applied clustering method is in fact a two-step method, the first step being the application of a multidimensional scaling method. Because the Kulczynski coefficient is not a metric (Gower and Legendre, 1986), Kruskal's non-metric

Table 1
Stability results for normal mixture with noise solution for snails data

Cluster	Boot	Boot(Kr)	Sub	Boot(Eu)	N0.05/3	N0.2/4	B/J
1	0.769	0.653	0.773	0.743	0.843	0.834	0.785
2	0.439	0.467	0.417	0.42	0.57	0.556	0.457
3	0.516	0.416	0.325	0.494	0.725	0.537	0.532
4	0.59	0.546	0.634	0.596	0.726	0.673	0.591
5	0.502	0.447	0.49	0.471	0.63	0.697	0.536
6	0.56	0.442	0.482	0.494	0.688	0.514	0.501
7	0.819	0.786	0.86	0.869	0.923	0.928	0.905
8	0.867	0.845	0.88	0.845	0.995	0.872	0.932
Noise	0.0768	0.0547	0.0546	0.121	0.223	0.162	0.0684

Cluster numbers as in Fig. 1. "Noise" denotes the estimated noise by the clustering method. "Boot(Kr)" refers to the use of Kruskal's non-metric MDS ("Boot" and "Sub" have been computed with classical metric MDS), "Boot(Eu)", the noise schemes and "B/J" have been applied to Euclidean data, i.e., the stability of the MDS has been taken for granted.

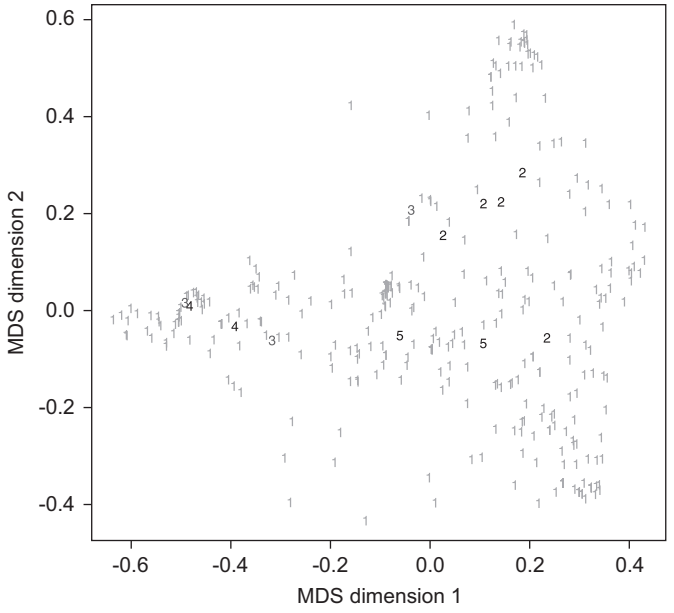


Fig. 3. First two MDS dimensions of snails distribution ranges data set with clustering generated by complete linkage (five-cluster partition).

MDS as implemented in the R-package MASS suggests itself. After we published results from these data first (Hausdorf and Hennig, 2003), it turned out that this method for the given data suffers from numerical unstabilities. Sometimes, numerical precisions of different machines led to quite different MDS solutions. Therefore, and because Kulczynski dissimilarities for this kind of data usually do not contain dangerous outliers, I now prefer the classical metric MDS for these data. This can be backed up by the stability results in Table 1, which show that the bootstrapped stability $\bar{\gamma}_C$ is clearly higher for the classical MDS for all but one cluster and especially for the stable ones. Note that on my computer Kruskal's MDS leads to the same clustering as classical MDS, but I have seen machines on which this does not hold.

Noise and bootstrap/jittering cannot be carried out on distance data. Therefore these schemes have been applied only to the Euclidean data which resulted from the MDS. This means that they diagnose the stability of the normal mixture with noise clustering, but cannot detect unstabilities in the MDS (jittering results were not very informative). "Boot(Eu)" gives the corresponding bootstrap results, so that the differences between this column and "Boot" illustrate the unstabilities that stem from the MDS.

The stability results confirm the cluster nos. 1, 7 and 8, which also have a clear biogeographical interpretation (1: species concentrated in the Eastern Alps, 7: widespread species present almost everywhere, 8: species concentrated in the Carpathes). Some further clusters can be interpreted biogeographically (3: Western European, 4: Pyrenean), but unfortunately they do not turn out to be stable. A deeper analysis of the clustering results under resampling shows that sometimes 3 and 4 are put together and sometimes the demarcation between them is drawn differently, so that there is certainly a pattern corresponding to 3 and 4, but it is very unstable whether this is only one or two clusters (and if two, which species belongs to which one).

The decision that only two points are assigned to the noise component is obviously very unstable and it seems to make sense to interpret more species as not belonging to any meaningful cluster.

Fig. 3 provides an illustration why a high-stability value does not necessarily guarantee a meaningful cluster. The clustering shown there has been generated by complete linkage carried out directly on the Kulczynski dissimilarities (the MDS is only used for visualization here). A partition was obtained by cutting the tree so that there are five clusters. This is certainly not a very useful partition, with 354 out of 366 points belonging to cluster 1. But cluster 1 reaches a bootstrap $\bar{\gamma}_C$ value 0.9225 ($\bar{\gamma}_C$ for the other four clusters is smaller than 0.15—they are obviously unstable; other resampling schemes lead to similar results). What is stable is the fact that complete linkage with number of clusters fixed to five consistently produces one very large cluster. But this is rather due to the inflexibility of the method than

due to the meaning of this cluster. Generally, inflexible methods can produce very stable but meaningless clusters, and stability alone is not enough to make a valid pattern.

6. Discussion

The simulation study and the example suggest that the various schemes to measure the stability of the clusters by computing the average maximum Jaccard coefficient over resampled (or modified) data sets can be very informative. Only the "jittering alone" schemes cannot be recommended. A good strategy in practice can be the use of one of the schemes bootstrap, bootstrap/jittering and subsetting together with one of the noise schemes. The number of bootstrap replications B does not have to be very large. A mean of Jaccard coefficients between 0 and 1 (often with small standard deviations, at least for the larger means) can be fairly precisely estimated with 50 replications. In data mining applications with large data sets, even five replications may be informative.

It is important to keep in mind that stability alone is not sufficient to validate a cluster. Inflexible methods can yield meaningless but stable clusters. Therefore, it is recommended to complement stability analyses with other cluster validation methods such as visual or subject-matter-based validation. On the other hand, detected unstabilities almost always point to serious problems so that unstable clusters should not be interpreted or only with caution. Cluster validity cannot be sufficiently assessed without reference to the aim of clustering, so that even unstable clusters may be accepted if there is no need for stability (for example in "organizational clustering" such as for the location of storerooms to serve groups of shops).

The issue of stability in cluster analysis is complex. Instabilities can stem from inherent instabilities in the data, lack of robustness of the clustering method or just an unfortunate choice of a generally good clustering method which is inadequate for the data at hand.

An implementation of the suggested resampling schemes will be included in the R-package FPC available as all other packages mentioned in this article on CRAN (www.R-project.org) in the near future.

References

Ben-Hur, A., Elisseeff, A., Guyon, I., 2002. A stability based method for discovering structure in clustered data. In: Proceedings of the Pacific Symposium on Biocomputing, 2002, pp. 6–17.

Bryan, J., 2004. Problems in gene clustering based on gene expression data. J. Multivariate Anal. 90, 67–89.

Byers, S., Raftery, A.E., 1998. Nearest-neighbor clutter removal for estimating features in spatial point processes. J. Amer. Statist. Assoc. 93, 577–584.

Cuesta-Albertos, J.A., Gordaliza, A., Matran, C., 1997. Trimmed k-means: an attempt to robustify quantizers. Ann. Statist. 25, 553–576.

Donoho, D.L., Huber, P.J., 1983. The notion of breakdown point. In: Bickel, P.J., Doksum, K., Hodges Jr.,J.L. (Eds.), A Festschrift for Erich L. Lehmann. Wadsworth, Belmont, CA, pp. 157–184.

Dudoit, S., Fridlyand, J., 2002. A prediction-based resampling method to estimate the number of clusters in a dataset. Genome Biol. 3, 0036.1–0036.21.

Fraley, C., Raftery, A.E., 1998. How many clusters? Which clustering method? Answers via model based cluster analysis. Comput. J. 41, 578–588.

Garcia-Escudero, L.A., Gordaliza, A., 1999. Robustness properties of k means and trimmed k means. J. Amer. Statist. Assoc. 94, 956–969.

Gordon, A.D., 1999. Classification. second ed. Chapman & Hall, Boca Raton, FL.

Gower, J.C., Legendre, P., 1986. Metric and Euclidean properties of dissimilarity coefficients. J. Classification 3, 5–48.

Grün, B., Leisch, F., 2004. Bootstrapping finite mixture models. In: Antoch,J. (Ed.), COMPSTAT 2004. Physica, Heidelberg, pp. 1115–1122.

Haldiki, M., Batistakis, Y., Vazirgiannis, M., 2002. Cluster validity methods, Part I. SIGMOD Record 31, 40–45.

Hausdorf, B., Hennig, C., 2003. Biotic element analysis in biogeography. Systematic Biol. 52, 717–723.

Hennig, C., 2004a. Breakdown points for ML estimators of location-scale mixtures. Ann. Statist. 32, 1313–1340.

Hennig, C., 2004b. Asymmetric linear dimension reduction for classification. J. Comput. Graph. Statist. 13, 930–945.

Hennig, C., 2006. Dissolution point and isolation robustness: robustness criteria for general cluster analysis methods. Research Report no. 272, Department of Statistical Science, University College London, submitted for publication. URL: http://www.ucl.ac.uk/Stats/research/Resrprts/psfiles/rr272.pdf

Hennig, C., 2005. A method for visual cluster validation. In: Weihs,C., Gaul, W. (Eds.), Classification—The Ubiquitous Challenge. Springer, Heidelberg, pp. 153–160.

Hennig, C., Hausdorf, B., 2004. Distance-based parametric bootstrap tests for clustering of species ranges. Comput. Statist. Data Anal. 45, 875–896.

Hubert, L., Arabie, P., 1985. Comparing partitions. J. Classification 2, 193–218.

Jaccard, P., 1901. Distribution de la florine alpine dans la Bassin de Dranses et dans quelques regiones voisines. Bull. Soc. Vaud. Sci. Nat. 37, 241–272.

Lange, T., Roth, V., Braun, M.L., Buhmann, J.M., 2004. Stability-based validation of clustering solutions. Neural Comput. 16, 1299–1323.

Milligan, G.W., 1996. Clustering validation: results and implications for applied analyses. In: Arabie, P., Hubert, L.J., De Soete, G. (Eds.), Clustering and Classification. World Scientific, Singapore, pp. 341–375.

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

Monti, S., Tamayo, P., Mesirov, J., Golub, T., 2001. Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. Mach. Learning 52, 91–118.

Rand, W.M., 1971. Objective criteria for the evaluation of clustering methods. J. Amer. Statist. Assoc. 66, 846–850.

Tibshirani, R., Walther, G., 2005. Cluster validation by prediction strength. J. Comput. Graph. Statist. 14, 511–528.