NbClust Package. An examination of indices for determining the number of clusters

Malika Charrad, Nadia Ghazzali, Véronique Boiteau, Azam Niknafs

Package 'NbClust'

May 23, 2012

Type Package

Title An examination of indices for determining the number of clusters : NbClust Package

Version 1.0

Date 2012-05-01

Author Malika Charrad <malika.charrad.1@ulaval.ca> and Nadia Ghazzali <nadia.ghazzali@mat.ulaval.ca> and Veronique Boiteau <veronique.boiteau.1@ulaval.ca> and Azam Niknafs <azam.niknafs.1@ulaval.ca>

Maintainer : Nadia Ghazzali <nadia.ghazzali@mat.ulaval.ca>

Description This package provides most of the popular indices for cluster validation ready to use for the outputs produced by functions coming from the same package. It also proposes to user the best clustering scheme from the different results obtained by varying all combinations of number of clusters,distance measures, and clustering methods.

License GPL-2

Repository CRAN

Date/Publication 2012-05-23 16:21:05

R topics documented:

NbClust *An examination of indices for determining the number of clusters : NbClust Package*

Description

NbClust package provides 30 indices for determining the number of clusters and proposes to user the best clustering scheme from the different results obtained by varying all combinations of number of clusters, distance measures, and clustering methods.

Usage

```
NbClust(data, diss="NULL", distance = "euclidean",
            min.nc=2, max.nc=15, method = "ward",
            index = "all", alphaBeale = 0.1)
```

Arguments

data	matrix or dataset (the only mandatory argument)
diss	dissimilarity matrix to be used. By default, diss="NULL", but if it is replaced by a dissimilarity matrix, distance should be "NULL".
distance	the distance measure to be used to compute the dissimilarity matrix. This must be one of: "euclidean", "maximum", "manhattan", "canberra", "binary", "minkowski" or "NULL". By default, distance="euclidean". If the distance is "NULL", the dissimilarity matrix (diss) should be given by the user. If distance is not "NULL", the dissimilarity matrix should be "NULL".
min.nc	minimal number of clusters, between 2 and (number of objects - 1)
max.nc	maximal number of clusters, between 2 and (number of objects - 1), greater or equal to min.nc. By default, max.nc=15.
method	the cluster analysis method to be used. This should be one of: "ward", "single", "complete", "average", "mcquitty", "median", "centroid", "kmeans".
index	the index to be calculated. This should be one of : "kl", "ch", "hartigan", "ccc", "scott", "marriot", "trcovw", "tracew", "friedman", "rubin", "cindex", "db", "silhouette", "duda", "pseudot2", "beale", "ratkowsky", "ball", "ptbiserial", "gap", "frey", "mcclain", "gamma", "gplus", "tau", "dunn", "hubert", "sdindex", "dindex", "sdbw", "all" (all indices except GAP, Gamma, Gplus and Tau), "alllong" (all indices with GAP, Gamma, Gplus and Tau included).
alphaBeale	significance value for Beale's index.

Details

1. **Notes on the "Distance" argument**
   The following distance measures are written for two vectors **x** and **y**. They are used when the data is a **d**-dimensional vector arising from measuring **d** characteristics on each of **n** objects or individuals.

- **Euclidean distance** : Usual square distance between the two vectors (2 norm).

$$d(x,y) = \left(\sum_{j=1}^{d} (x_j - y_j)^2 \right)^{\frac{1}{2}}$$

- **Maximum distance**: Maximum distance between two components of **x** and **y** (supremum norm).

$$d(x,y) = \sup_{1 \leq j \leq d} |x_j - y_j|$$

- **Manhattan distance** : Absolute distance between the two vectors (1 norm).

$$d(x,y) = \sum_{j=1}^{d} |x_j - y_j|$$

- **Canberra distance** : Terms with zero numerator and denominator are omitted from the sum and treated as if the values were missing.

$$d(x,y) = \sum_{j=1}^{d} \frac{|x_j - y_j|}{|x_j| + |y_j|}$$

- **Binary distance** : The vectors are regarded as binary bits, so non-zero elements are "on" and zero elements are "off". The distance is the proportion of bits in which only one is on amongst those in which at least one is on.
- **Minkowski distance** : The **p** norm, the p^{th} root of the sum of the p^{th} powers of the differences of the components.

$$d(x,y) = \left(\sum_{j=1}^{d} |x_j - y_j|^p \right)^{\frac{1}{p}}$$

2. **Notes on the "method" argument**
   The following aggregation methods are available in this package.

   - **Ward** : Ward method minimizes the total within-cluster variance. At each step the pair of clusters with minimum cluster distance are merged. To implement this method, at each step find the pair of clusters that leads to minimum increase in total within-cluster variance after merging. This increase is a weighted squared distance between cluster centers. The initial cluster distances in Ward minimum variance method are defined to be the squared Euclidean distance between points:

$$D_{ij} = \|x_i - y_j\|^2$$

   - **Single** : The distance D_{ij} between two clusters C_i and C_j is the minimum distance between two points x and y, with $x \in C_i, y \in C_j$.

$$D_{ij} = min_{x \in C_i, y \in C_j} d(x,y)$$

     A drawback of this method is the so-called chaining phenomenon: clusters may be forced together due to single elements being close to each other, even though many of the elements in each cluster may be very distant to each other.

- **Complete** : The distance D_{ij} between two clusters C_i and C_j is the maximum distance between two points x and y, with $x \in C_i, y \in C_j$.

$$D_{ij} = max_{x \in C_i, y \in C_j} d(x, y)$$

- **Average** : The distance D_{ij} between two clusters C_i and C_j is the mean of the distances between the pair of points x and y, where $x \in C_i, y \in C_j$.

$$D_{ij} = sum_{x \in C_i, y \in C_j} \frac{d(x, y)}{n_i \times n_j}$$

where n_i and n_j are respectively the number of elements in clusters C_i and C_j. This method has the tendency to form clusters with the same variance and, in particular, small variance.

- **McQuitty** : The distance between clusters C_i and C_j is the weighted mean of the between-cluster dissimilarities:

$$D_{ij} = (D_{ik} + D_{il}) / 2$$

where cluster C_j is formed from the aggregation of clusters C_k and C_l.

- **Median** : The distance D_{ij} between two clusters C_i and C_j is given by the following formula:

$$D_{ij} = \frac{(D_{ik} + D_{il})}{2} - \frac{D_{kl}}{4}$$

where cluster C_j is formed by the aggregation of clusters C_k and C_l.

- **Centroid** : The distance D_{ij} between two clusters C_i and C_j is the squared euclidean distance between the gravity centers of the two clusters, i.e. between the mean vectors of the two clusters, $\bar{x_i}$ and $\bar{x_j}$ respectively.

$$D_{ij} = \|\bar{x_i} - \bar{x_j}\|^2$$

This method is more robust than others in terms of isolated points.

- **Kmeans** : This method is said to be a reallocation method. Here is the general principle:
  (a) Select as many points as the number of desired clusters to create initial centers.
  (b) Each observation is then associated with the nearest center to create temporary clusters.
  (c) The gravity centers of each temporary cluster is calculated and these become the new clusters centers.
  (d) Each observation is reallocated to the cluster which has the closest center.
  (e) This procedure is iterated until convergence.

3. **Notes on the "Index" argument**

3.1. CH index. Calinski and Harabasz (1974)

$$CH(q) = \frac{trace(B_q)/(q-1)}{trace(W_q)/(n-q)}$$

Where
$W_q = \sum_{k=1}^{q} \sum_{i \in C_k} (x_i - c_k)(x_i - c_k)^T$ is the within-group dispersion matrix for data clustered into q clusters.

$B_q = \sum_{k=1}^{q} n_k * (c_k - c)\,(c_k - c)^T$ is the between-group dispersion matrix for data clustered into q clusters.
x_i = p-dimensional vector of observations of the i^{th}object in cluster k.
c_k = centroid of cluster k
c = centroid of data matrix
n_k = number of objects in cluster C_k

The value of q ($q \in (2, ..., n-2)$), which maximizes CH(q), is regarded as specifying the number of clusters.
This index is calculated if *index = "ch" or "all" or "alllong"*.
The program for this index comes from the index.G1 function of the ClusterSim package. However, the program was slightly corrected to take into account clusters with only one observation.
References : Milligan and Cooper (1985), Calinski and Harabasz (1974), Gordon (1999) and Walesiak and Dudek (2011).

3.2. Duda index. Duda and Hart (1973)

$$duda = \frac{Je(2)}{Je(1)} = \frac{W_k + W_l}{W_m}$$

where
Je(2) is the sum of squared errors within cluster when the data are partitioned into two clusters and Je(1) gives the squared errors when only one cluster is present.
W_k, W_l, W_m are defined as W_q in CH Index.
It is assumed that clusters c_k and c_l are merged to form c_m.
$B_{kl} = W_m - W_k - W_l$, if $c_m = c_k \cup c_l$.
n_i = number of observations in cluster c_i, $i = k, l, m$.
The optimal number of clusters is the smallest q such that

$$duda \geq 1 - \frac{2}{\pi p} - z\sqrt{\frac{2\left(1 - \frac{8}{\pi^2 p}\right)}{n_m p}} = critValue_D uda$$

p is the number of variables in the data set
z is a standard normal score. Several values for the standard score were tested and the best results were obtained when the value was set to 3.20.
This index is calculated if *index = "duda" or "all" or "alllong"*.
References : Milligan and Cooper (1985), Duda and Hart (1973), Gordon (1999) and SAS/STAT(R) 9.2 User's Guide, Second Edition, the Cluster Procedure, Miscellaneous Formulas.

3.3. Pseudot2 index. Duda and Hart (1973)

$$pseudot2 = \frac{B_{kl}}{\frac{W_k + W_l}{n_k + n_l - 2}}$$

B_{kl}, W_k, W_l are defined in Duda index.
n_k and n_l are the number of objects in respectively C_k and C_l clusters.

The optimal number of clusters is the smallest q such that:

$$pseudot2 \leq \left(\frac{1 - critValue_Duda}{critValue_Duda} \right) \times (n_k + n_l - 2)$$

This index is calculated if *index = "pseudot2" or "all" or "alllong"*.
References : Milligan and Cooper (1985), Duda and Hart (1973), Gordon (1999) and SAS/STAT(R) 9.2 User's Guide, Second Edition, the Cluster Procedure, Miscellaneous Formulas.

3.4. C-index. Hubert and Levin (1976)

$$cindex = \frac{Du - (r \times D_{min})}{(r \times D_{max}) - (r \times D_{min})}$$

$D_{min} \neq D_{max}$
$cindex \in (0, 1)$
Du is the sum of all within-cluster dissimilarities
r = number of within-cluster dissimilarities
D_{min} = smallest within-cluster dissimilarity
D_{max} = largest within-cluster dissimilarity
The value of q ($q \in (2, ..., n-2)$) which minimizes cindex is considered as specifying the number of clusters.
This index is calculated if *index = "cindex" or "all" or "alllong"*.
The program for this index comes from the index.G3 function of the ClusterSim package.
References : Milligan and Cooper (1985), Hubert and Levin (1976), Gordon (1999) and Walesiak and Dudek (2011).

3.5. Gamma index. Baker and Hubert (1975)

$$gamma = \frac{s(+) - s(-)}{s(+) + s(-)}$$

where:
$s(+)$= number of concordant comparisons
$s(-)$= number of discordant comparisons

The maximum value across the hierarchy levels is used to select the optimal number of clusters.
In NbClust, this index is calculated only if *index = "gamma" or "alllong"* because of its high computational demand.
The program and the formulas for this index is based on the index.G2 function of the ClusterSim package, but the .C function was reprogrammed in R language.
References : Milligan and Cooper (1985), Baker and Hubert (1975), Milligan (1981), Gordon (1999) and Walesiak and Dudek (2011).

3.6. Beale index. Beale (1969)
Beale index is defined by the following formula :

$$beale = F \equiv \frac{\left(\frac{W_m-(W_k+W_l)}{W_k+W_l}\right)}{\left(\left(\frac{n_m-1}{n_m-2}\right)2^{\frac{2}{p}}-1\right)}$$

where
W_k, W_l, W_m are defined as W_q in Calinski and Harabasz index
n_m is the number of objects in cluster C_m.
It is assumed that clusters C_k and C_l are merged to form C_m.
The optimal number of clusters is obtained by comparing F with an $F_{p,(N_m-2)p}$ distribution.
The null hypothesis of a single cluster is rejected for significantly large values of F. By default, in **NbClust**, the 10% significance level was used to reject the null hypothesis.
This index is calculated if *index = "beale" or "all" or "alllong"*.
References : Milligan and Cooper (1985), Beale (1969) and Gordon (1999)

3.7. Cubic Clustering Criterion (CCC). Sarle (1983)

$$ccc = \ln\left[\frac{1-E\left(R^2\right)}{1-R^2}\right]\frac{\sqrt{\frac{np*}{2}}}{\left(0.001+E\left(R^2\right)\right)^{1.2}}$$

where

$$R^2 = 1-\frac{trace(W)}{trace(T)}$$

$T = X'X$ is the total-sample sum-of-squares and crossproducts (SSCP) matrix $(p \times p)$
W=T-B is the within-cluster SSCP matrix $(p \times p)$
$B = \bar{X}'Z'Z\bar{X}$ is between-cluster SSCP matrix $(p \times p)$
$\bar{X} = (Z'Z)^{-1}Z'X$
Z is a cluster indicator matrix $(n \times q)$ with element $z_{ik} = 1$ if the i^{th}observation belongs to the k^{th}cluster, 0 otherwise.

$$E(R^2) = 1-\left[\frac{\sum_{j=1}^{p*}\frac{1}{n+u_j}+\sum_{j=p^*+1}^{p}\frac{u_j^2}{n+u_j}}{\sum_{j=1}^{p}u_j^2}\right]\left[\frac{(n-q)^2}{n}\right]\left[1+\frac{4}{n}\right]$$

where :
$u_j = \frac{s_j}{c}$
s_j = square root of the j^{th} eigenvalue of $\frac{T}{(n-1)}$
$c = \left(\frac{v*}{q}\right)^{\frac{1}{p*}}$
$v^* = \prod_{j=1}^{p^*} s_j$
$p*$ is chosen to be the largest integer less than q such that u_p^* is not less than one.

The maximum values across the hierarchy levels is used to indicate the optimal number of clusters in the data.
This index is calculated if *index = "ccc" or "all" or "alllong"*.
References : Milligan and Cooper (1985) and Sarle (1983).

3.8. PtBiserial index. Examined by Milligan (1980,1981)

$$ptbiserial = \frac{\left(\bar{d_b} - \bar{d_w}\right)\left(f_w f_b / n_d^2\right)^{1/2}}{S_d}$$

where:
d_w = sum of within cluster distances
d_b = sum of between cluster distances
$\bar{d_w}$,$\bar{d_b}$ = respective means
S_d = standard deviation of all distances
n_d = total number of distances
f_w = number of within cluster distances
f_b = number of between cluster distances
This index is calculated if *index = "ptbiserial" or "all" or "alllong"*.
References : Milligan and Cooper (1985), Milligan (1980,1981), Kraemer (1982) and ltm package.

3.9. Gplus index. Reviewed by Rohlf (1974) and examined by Milligan (1981a)

$$G(+) = \frac{2s(-)}{n_d\left(n_d - 1\right)}$$

where:

s(-) is the number of discordant comparisons i.e. the number of times where two points which were in the same cluster had a larger distance than two points not clustered together.
n_d = total number of distances (which is the same as the total number of observations or objects under study).
Minimum values of the index are used to determine the optimal number of clusters in the data.
In **NbClust**, this index is calculated only if index = "gplus" or "alllong" because of its high computational demand.
References : Milligan and Cooper (1985), Rohlf (1974) and Milligan (1981a).

3.10. DB index. Davies and Bouldin (1979)

The Davies and Bouldin index is a function of the ratio of the sum of within-cluster scatter to between-cluster separation. It is calculated by the following formula :

$$DB(q) = 1/q \sum_{r=1}^{q} max_{s,r \neq s} \frac{S_r + S_s}{d_{rs}}$$

where
$r, s = 1, ..., q$=cluster number
q = number of clusters ($q \geq 2$)
C_r, C_s = r^{th}, s^{th} cluster
$d_{rs} = \sqrt[v]{\sum_{j=1}^{p} |c_{rj} - c_{sj}|^v}$ = distance between centroids of clusters C_r and C_s
(for $v = 2$, d_{rs} is the euclidean distance)
$c_r = (c_{r1}, ..., c_{rp})$= centroid of cluster C_r
$S_r = \sqrt[u]{\frac{1}{n_r} \sum_{i \in C_r} \sum_{j=1}^{p} |x_{ij} - c_{rj}|^u}$ = dispersion measure of a cluster C_r

(for u=2, S_r is the standard deviation of the distance of objects in cluster C_r to the centroid of cluster C_r)
n_r and n_s are respectively the number of objects in clusters C_r and C_s.
The value of q, which minimizes $DB(q)$, is regarded as specifying the number of clusters.
In **NbClust**, this index is calculated if *index = "db" or "all" or "alllong"*.
The program and the formulas for this index come from the index.DB function of the Cluster-Sim package.
References : Milligan and Cooper (1985), Davies and Bouldin (1979 and Walesiak and Dudek (2011).

3.11. Frey index. Frey and Van Groenewoud (1972)

Frey index is the ratio of difference scores from two successive levels in the hierarchy. The numerator is the difference between the mean outside-cluster distances, $\bar{d}_v$, from each of the two hierarchy levels (level j and level j+1). The denominator is the difference between the mean within cluster distances from the two levels (level j and level j+1). The authors proposed using a ratio score of 1.00 to identify the correct cluster level. The ratios often varied above and below 1.00. The best results occurred when clustering was continued until the ratio fell below 1.00 for the last series of times. At this point, the cluster level before this series was taken as the optimal partition. If the ratio never fell below 1.00, a one cluster solution was assumed.

$$K = \frac{\bar{d}_{v_{j+1}} - \bar{d}_{v_j}}{\bar{d}_{s_{j+1}} - \bar{d}_{s_j}}$$

where
$\bar{d}_v$ = mean outside-cluster distance
$\bar{d}_s$ = mean within-cluster distance
In **NbClust**, this index is calculated if index = "frey" or "all" or "alllong".
References : Milligan and Cooper (1985) and Frey and Van Groenewoud (1972).

3.12. Hartigan index. Hartigan (1975)

The Hartigan index is computed as follows :

$$hartigan = \left(\frac{trace(W_q)}{trace(W_{q+1})} - 1 \right) (n - q - 1)$$

where $W_q = \sum_{k=1}^{q} \sum_{i \in C_k} (x_i - \bar{x_k})(x_i - \bar{x_k})^T$ is the within-group dispersion matrix for data clustered into q clusters, $q \in (1, ..., n-2)$.

x_i = p-dimensional vector of objects of the i^{th} object in cluster C_k,
$\bar{x}_k$ = centroid of cluster k,
n is the number of observations in the data matrix.

Maximum value of the index is taken as indicating the correct number of clusters in the data (q in (1, ..., n - 2)).
This index is calculated if index = "hartigan" or "all" or "alllong".
The program and the formulas for this index come from the index.H function of the Cluster-Sim package.
References : Milligan and Cooper (1985), Hartigan (1975) and Walesiak and Dudek (2011).

3.13. Tau index. Reviewed by Rohlf (1974) and tested by Milligan (1981a)

Tau index is computed as follows:

$$Tau = \frac{s(+) - s(-)}{\left[\left(n_d\left(n_d-1\right)/2 - t\right)\left(n_d\left(n_d-1\right)/2\right)\right]^{1/2}}$$

where
s(+) is the number of concordant comparisons
s(-) is the number of discordant comparisons.
n_d is the total number of distances (which is the same as the total number of observations or objects under study)
t is the number of comparisons of two pairs of points where both pairs represent within cluster comparisons or both pairs are between cluster comparisons.
The maximum value in the hierarchy sequence was taken as indicating the correct number of clusters.
This index is calculated only if index = "tau" or "alllong" because of its high computational cost.
References : Milligan and Cooper (1985), Milligan (1981a) and Rohlf (1974).

3.14. Ratkowsky index. Ratkowsky and Lance (1978)

This index is based on this formula :

$$\frac{\bar{S}}{q^{1/2}}$$

.
The value of $\bar{S}$ is equal to the average of the ratios of B/T where B stands for the Sum of Squares Between the clusters for each variable and T for the Total Sum of Squares for each variable.
The optimal number of clusters is that value of q for which $\frac{\bar{S}}{q^{1/2}}$ has its maximum value.
If the value of q is made constant, the Ratkowsky and Lance criterion can be reduced from $\frac{\bar{S}}{q^{1/2}}$ to $\bar{S}$.
In **NbClust** package, Ratkowsky and Lance index is computed with the following formula :

$$ratkowsky = mean(\sqrt{B/T})$$

This index is calculated if index = "ratkowsky" or "all" or "alllong".
The program and the formulas for this index come from the clustIndex function of the cclust

package.
References : Milligan and Cooper (1985), Ratkowsky and Lance (1978), Hill (1980), Dimitriadou (2002) and Dimitriadou (2009).

3.15. Scott index. Scott and Symons (1971)

Scott index is based on the following formula :

$$nlog\left(|T| \,/\, |W|\right)$$

where

n is the number of elements in the data set,
T is the total sum of squares (see CCC index),
W is the sum of squares within the clusters (see CCC index).
The maximum difference between hierarchy levels was used to suggest the correct number of partitions.
This index is calculated if index = "scott" or "all" or "alllong".
The program for this index is based on the clustIndex function of the cclust package, but it is a little bit different. The difference comes from the definition of the W and T matrices.
References : Milligan and Cooper (1985), Scott and Symons (1971) and Dimitriadou (2009).

3.16. Marriot index. Marriot (1971)

$$marriot = k^2|W|$$

where W is defined as in CCC index.
The maximum difference between successive levels was used to determine the best partition level. This index is calculated if index = "marriot" or "all" or "alllong".
The program for this index is based on the clustIndex function of the cclust package, but it is a little bit different. The difference comes from the definition of the W matrix. **References :** Milligan and Cooper (1985), Marriot (1971), Dimitriadou (2002) and Dimitriadou (2009).

3.17. Ball index. Ball and Hall (1965)

This index is based on the average distance of the items to their respective cluster centroids.

$$ball = \frac{W}{q}$$

where q is the number of clusters and W is the sum of squares within the clusters.
The largest difference between levels was used to indicate the optimal solution.
This index is calculated if index = "ball" or "all" or "alllong".
The program for this index come from the clustIndex function of the cclust package.
References : Milligan and Cooper (1985), Ball and Hall (1965), Dimitriadou (2002) and Dimitriadou (2009).

3.18. TraceCovW index. Milligan and Cooper (1985)

This index represents the trace of the within clusters pooled covariance matrix.

$$trcovw = trace\left(cov\left(W\right)\right)$$

where W is defined as in CCC.
Maximum differences scores between levels were used to indicate the optimal solution.
This index is calculated if index = "tracecovw" or "all" or "alllong".
The program for this index is based on the clustIndex function of the cclust package, but it is a little bit different. The difference comes from the definition of the W and T matrices. **References :** Milligan and Cooper (1985) and cclust package.

3.19. TraceW index. Milligan and Cooper (1985)

$$tracew = trace(W)$$

where W is defined as in CCC index.

Given that the criterion increases monotonically with solutions containing fewer clusters, maximum of the second differences scores were used to determine the number of clusters in the data.
This index is calculated if index = "tracew" or "all" or "alllong".
The program for this index is based on the clustIndex function of the cclust package, but it is a little bit different. The difference comes from the definition of the W matrix. **References :** Milligan and Cooper (1985), Edwards and Cavalli-Sforza (1965), Friedman and Rubin (1967), Orloci (1967), Fukunaga and Koontz (1970) and Dimitriadou (2009).

3.20. Friedman index. Friedman and Rubin (1967)

This index was proposed as a basis for non hierarchical clustering method.

$$friedman = trace\left(W^{-1}B\right)$$

where B and W are defined as in CCC index.
The maximum difference in values of this criterion was used to indicate the optimal number of clusters. This index is calculated if index = "friedman" or "all" or "alllong".
The program for this index is based on the clustIndex function of the cclust package, but it is a little bit different. The difference comes from the definition of the W and B matrices. **References :** Milligan and Cooper (1985), Friedman and Rubin (1967) and Dimitriadou (2009).

3.21. Mcclain index. McClain and Rao (1975)

This index consists of the ratio of two terms. The first term is the average within cluster distance divided by the number of within cluster distances. The denominator value was the average between cluster distance divided by the number of cluster distances. It is computed as follows :

$$mcclain = \frac{mean\left(\sum_{q}^{k=1}\sum_{i=1}^{n_k}\sum_{j=i+1}^{n_k} d_{ij}\right)}{mean\left(\sum_{k=1}^{q}\sum_{i\in C_k}\sum_{l=k+1}^{q}\sum_{j\in C_l} d_{ij}\right)}$$

where

q is the Number of clusters,
n_k is the number of objects in the k^{th} cluster, $k \in [1...q]$k in [1...q]
d_{ij} = distance between i^{th} and j^{th} objects
The minimum value of the index is used to indicate the optimal number of clusters.
This index is calculated if index = "mcclain" or "all" or "alllong".
References : Milligan and Cooper (1985) and McClain and Rao (1975).

3.22. Rubin index. Friedman and Rubin (1967)

This index is based on the ratio of the determinant of the total sum of squares and cross products matrix to the determinant of the pooled within cluster matrix.

$$rubin = |T|/|W|$$

where T and W are defined as in CCC index.

The minimum value of second differences between levels was used.
This index is calculated if index = "rubin" or "all" or "alllong".
The program for this index is based on the clustIndex function of the cclust package. The difference in results comes from difference in definition of W and T matrices.
References : Milligan and Cooper (1985), Friedman and Rubin (1967) and Dimitriadou E (2009).

3.23. KL index. Krzanowski and Lai (1988)

$$KL(q) = \left|\frac{DIFF_q}{DIFF_{q+1}}\right|$$

where

$$DIFF_q = (q-1)^{2/p}\, trace\,(W_{q-1}) - q^{2/p} trace\,(W_q)$$

$W_q$is defined as in Hartigan.
The value of q, which maximizes $KL(q)$, is regarded as specifying the number of clusters.
This index is calculated if index = "kl" or "all" or "alllong".
The program and the formulas for this index come from the index.KL function of the ClusterSim package, but the program was corrected to take into account clusters with only one observation.

References : Krzanowski and Lai (1988) and Walesiak and Dudek (2011).

3.24. Silhouette index. Kaufman and Rousseeuw (1990)

$$silhouette = \sum_{i=1}^{n} S(i)/n, silhouette \in [-1,\ 1]$$

where
$S(i) = \frac{b(i)-a(i)}{\max\{a(i);b(i)\}}$
$a(i) = \frac{\sum_{j \in C_r \beta} d_{ij}}{n_r - 1}$ is the average dissimilarity of the i^{th} object to all other objects of C_r cluster
$b(i) = min_{s \neq r} d_{iC_s}$
$d_{iC_s} = \frac{\sum_{j \in C_s} d_{ij}}{n_s}$ is the average dissimilarity of the i^{th} object to all objects of C_s cluster
C_r and C_s are respectively r^{th} and s^{th} clusters
n_r and n_s are respectively the number of objects in clusters C_r and C_s.
Maximum values of the index are used to determine the optimal number of clusters in the data.
$S(i)$ is not defined for $k = 1$ (only one cluster). This index is calculated if index = "silhouette" or "all" or "alllong".
The program for this index comes from the index.S function of the ClusterSim package.
References : Kaufman and Rousseeuw (1990), Rousseeuw (1987) and Walesiak and Dudek (2011).

3.25. Gap index. Tibshirani et al. (2001)

The estimated gap statistic is computed as follows :
$Gap(q) = \frac{1}{B} \sum_{b=1}^{B} \log W_{qb} - \log W_q$
where
B is the number of reference data sets generated using uniform prescription
W_{qb} is the within-dispersion matrix defined as in Hartigan index.
The optimal number of clusters is choosen via finding the smallest q such that:
$Gap(q) \geq Gap(q+1) - s_{q+1}, \quad (q = 1, ..., n-2)$
where : $s_q = sd_q \sqrt{1 + 1/B}$
sd_q is the standard deviation of $\{\log W_{qb}\}, b = 1, ..., B$:

$$sd_q = \sqrt{\frac{1}{B} \sum_{b=1}^{B} (\log W_{qb} - \bar{l})^2}$$

$$\bar{l} = \frac{1}{B} \sum_{b=1}^{B} logW_{qb}$$

Which is the same as: $CritValue_Gap = Gap(q)???Gap(q+1) + s_{q+1} >= 0, \quad (q =$

$1, ..., n-2)$

In **NbClust**, Gap index is calculated only if ("index" = "gap" or "alllong") because of its high computational cost.
References : Tibshirani et al. (2001) and Walesiak and Dudek (2011).

3.26. Dindex. Lebart et al. (2000)

The Dindex is based on clustering gain on intra-cluster inertia. Intra-cluster inertia can be defined as:

$$W(P^q) = \frac{1}{q}\sum_{k=1}^{q}\frac{1}{n_k}\sum_{i=1}^{n_k} d(x_i, c_k)$$

Given two partitions, P^{k-1} composed of $k-1$ clusters and P^k composed of k clusters, the clustering gain on intra-cluster inertia is defined as :

$$Gain = W(P^{q-1}) - W(P^q)$$

This clustering gain should be minimized. The optimal cluster configuration can be identified by the sharp knee that corresponds to a significant decrease of the **first differences** of clustering gain versus the number of clusters. This knee or great jump of gain values can be identified by a significant peak in **second differences** of clustering gain.
Dindex is calculated if ("index" = "dindex" or "all" or "alllong"). **References :** Lebart et al. (2000).

3.27. Dunn index. Dunn(1974)

The Dunn index defines the ratio between the minimal intercluster distance to maximal intracluster distance. The index is given by:

$$Dunn = \frac{min_{1\leq i<j\leq q} d(C_i, C_j)}{max_{1\leq k\leq q} diam(C_k)}$$

Where q is the number of clusters,
$d(C_i, C_j)$ is the dissimilarity function between two clusters C_i and C_j defined as $d(C_i, C_j) = \min_{x\in C_i, y\in C_j} d(x, y)$
diam(C) is the diameter of a cluster. It can be defined as follows :

$$diam(C) = max_{x,y\in C} d(x, y)$$

Dunn index should be maximized.
Dunn is calculated if ("index" = "dunn" or "all" or "alllong"). **References :** Dunn (1974) and clValid package.

3.28. Hubert index. Hubert and Arabie 1985

Hubert Γ statistic is the point serial correlation coefficient between any two matrices. When the two matrices are symmetric, Γ can be written in its raw form as :

$$\Gamma(P,Q) = \frac{1}{M}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} P_{ij}Q_{ij}$$

where
$M = n(n-1)/2$,
P is the proximity matrix of the data set,
Q is an $n \times n$ matrix whose (i,j) element is equal to the distance between the representative points (v_{ci}, v_{cj}) of the clusters where the objects x_i and x_j belong.
We note, that for $q = 1$ or $q = n$, the index is not defined.

The definition of Hubert normalized Γ statistic is given by the following equation :

$$\bar{\Gamma} = \frac{(\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}(P_{ij}-\mu_P)(Q_{ij}-\mu_Q))}{\sigma_P\sigma_Q}$$

where μ_P, μ_Q, σ_P, σ_Q are the respective means and variances of P, Q matrices. This index takes values between -1 and 1. If P and Q are not symmetric then all summations are extended over all n^2 entries and $M = n^2$.
High values of normalized Γ statistic indicate the existence of compact clusters. Thus, in the plot of normalized Γ versus q, the number of clusters, we seek a significant knee that corresponds to a significant increase of normalized Γ as q varies from q_{max} to 2, where q_{max} is the maximum possible number of clusters.
The number of clusters at which the knee occurs is an indication of the number of clusters that underlie the data. In **NbClust**, second differences values of normalized Γ statistic are plotted to help distinguish the knee from other anomalies. A significant peak in this plot indicates the optimal number of clusters.
This index is computed if ("Index"= "hubert" or "all" or "alllong").
References : Hubert and Arabie (1985), Bezdek and Pal (1998) and Halkidi et al. (2001).

3.29. SDindex. Halkidi et al.(2000)

The SD validity index definition is based on the concepts of average scattering for clusters and total separation between clusters. It is computed as follows :

$$SDindex(q) = \alpha Scat(q) + Dis(q)$$

The first term ($Scat(q)$ indicates the average compactness of clusters (i.e. intra-cluster distance).

$$Scat(q) = \frac{1}{q}\sum_{k=1}^{q} \|\sigma(c_k)\| \,/\, \|\sigma(X)\|$$

where
q is the number of clusters, σ_X is the variance of the data set X,
$|X| = (X^T X)^2)^{1/2}$.

The second term $Dis(q)$ indicates the total separation between the q clusters (i.e. an indication of inter-cluster distance).

$$Dis(q) = \frac{D_{max}}{D_{min}} \sum_{k=1}^{q} \left(\sum_{z=1}^{q} \|c_k - c_z\| \right)^{-1}$$

where $D_{max} = max(|c_k - c_z|)\ \forall k, zin1, 2, 3, ..., q$ is the maximum distance between cluster centers.
The $D_{min} = min\left(\|c_k - c_z\|\right)\ \forall k, z \in \{1, 2, 3, ..., q\}$ is the minimum distance between cluster centers.
Alpha is a weightening factor equal to $Dis(q_{max})$ where q_{max} is the maximum number of input clusters.
The number of clusters, q, that minimizes the above index can be considered as an optimal value for the number of clusters present in the data set.
Unlike in clv package, where Alpha is equal to q_{max}, in **NbClust** package, Alpha is equal to $Dis(q_{max})$ as it is mentioned (Halkidi, 2000).
This index is computed if ("Index"= "SDindex" or "all" or "alllong").
References : (Halkidi, 2000)

3.30. SDbw. Halkidi et al.(2001)

The SDbw validity index definition is based on the criteria of compactness and separation between clusters. It is computed as follows:

$$SDbw(q) = Scat(q) + Density.bw(q)$$

The first term, $Scat(q)$, is the same computed in SDindex. The second term, $Density.bw(q)$, is the inter-cluster density. It evaluates the average density in the region among clusters in relation to the density of the clusters.

$$Density.bw(q) = \frac{1}{q(q-1)} \sum_{i=1}^{q} \left(\sum_{j=1, i \neq j}^{q} \frac{density(u_{ij})}{max(density(c_i), density(c_j))} \right)$$

where c_i and c_j are the centers of clusters and u_{ij} the middle point of the line segment defined by the clusters centers c_i, c_j.

$$density(u) = \sum_{l=1}^{n_{ij}} f(x_l, u)$$

$n_{ij}$is the number of tuples that belong to the clusters C_i and C_j. $f(x, u)$ is equal to 0 if d(x,u) > stdev and 1 otherwise.
Stdev is the average standard deviation of clusters.

$$stdev = \frac{1}{q} \sqrt{\sum_{i=1}^{q} \|\sigma(c_i)\|}$$

The number of clusters q that minimizes SDbw is considered as the optimal value for the number of clusters in the data set.
This index is computed if ("Index"= "SDbw" or "all" or "alllong").
References : Halkidi and Vazirgiannis (2001)

The table below summarizes indices implemented in NbClust and the criteria used to select the optimal number of clusters.

Index in literature	**Index in NbClust**	**Optimal number of clusters**
1. Krzanowski and Lai	"kl" or "all" or "alllong"	Maximum value of the index
2. Calinski and Harabasz	"ch" or "all" or "alllong"	Maximum value of the index
3. Hartigan	"hartigan" or "all" or "alllong"	Maximum difference between hierarchy levels of the index
4. Cubic Clustering Criterion	"ccc" or "all" or "alllong"	Maximum value of the index
5. $n \log(\lvert T\rvert/\lvert W\rvert)$	"scott" or "all" or "alllong"	Maximum difference between hierarchy levels of the index
6. $k^2\lvert W\rvert$	"marriot" or "all" or "alllong"	Max. value of second differences between levels of the index
7. Trace Cov W	"trcovw" or "all" or "alllong"	Maximum difference between hierarchy levels of the index
8. Trace W	"tracew" or "all" or "alllong"	Maximum value of absolute second differences between levels of the index
9. Trace $W^{-1}B$	"friedman" or "all" or "alllong"	Maximum difference between hierarchy levels of the index
10. $\lvert T\rvert/\lvert W\rvert$	"rubin" or "all" or "alllong"	Minimum value of second differences between levels of the index
11. C-index	"cindex" or "all" or "alllong"	Minimum value of the index
12. Davies and Bouldin	"db" or "all" or "alllong"	Minimum value of the index
13. Silhouette	"silhouette" or "all" or "alllong"	Maximum value of the index
14. $Je(2)/Je(1)$	"duda" or "all" or "alllong"	Smallest n_c such that index > criticalValue
15. $Pseudot^2$	"pseudot2" or "all" or "alllong"	Smallest n_c such that index < criticalValue
16. Beale	"beale" or "all" or "alllong"	n_c such that critical value of the index >= alpha
17. $\bar{c}/k^{.5}$	"ratkowsky" or "all" or "alllong"	Maximum value of the index
18. Ball and Hall	"ball" or "all" or "alllong"	Maximum difference between hierarchy levels of the index
19. Point-Biserial	"ptbiserial" or "all" or "alllong"	Maximum value of the index
20. Gap	"gap" or "alllong"	Smallest n_c such that criticalValue >= 0
21. Frey and Groenewood	"frey" or "all" or "alllong"	the cluster level before that index value < 1.00
22. McClain and Rao	"mcclain" or "all" or "alllong"	Minimum value of the index
23. Gamma	"gamma" or "alllong"	Maximum value of the index
24. G(+)	"gplus" or "alllong"	Minimum value of the index
25. Tau	"tau" or "alllong"	Maximum value of the index
26. Dunn	"dunn" or "all" or "alllong"	Maximum value of the index
27. Modified statistic of Hubert	"hubert" or "all" or "alllong"	Graphical method
28. SD	"sdindex" or "all" or "alllong"	Minimum value of the index
29. Lebart	"dindex" or "all" or "alllong"	Graphical method
30. SDbw	"sdbw" or "all" or "alllong"	Minimum value of the index

Value

`All.index` Values of indices for each partition of the dataset obtained with a number of clusters between min.nc and max.nc.

`All.CriticalValues` Critical values of some indices for each partition obtained with a number of clusters between min.nc and max.nc.

`Best.nc` Best number of clusters proposed by each index and the corresponding index value.

Author(s)

Malika Charrad, Nadia Ghazzali, Veronique Boiteau and Azam Niknafs

References

Baker FB, Hubert LJ (1975). "Measuring the Power of Hierarchical Cluster Analysis." Journal of the American Statistical Association, 70(349), 31-38. URL http://www.jstor.org/ stable/2285371.

Ball GH, Hall DJ (1965). "ISODATA, A novel method of data analysis and pattern classification". Menlo Park: Stanford Research Institute. (NTIS No. AD 699616).

Beale EML (1969). Cluster analysis. Scientific Control Systems, London.

Bezdek J, Pal N (1998). "Some new indexes of cluster validity". IEEE transactions on systems, man and cybernetics, 28(3).

Brock G, Pihur V, Datta S, Datta S (2008). "clvalid: Validation of Clustering Results". R package version 0.6-4, URL http://cran.r-project.org/web/packages/clvalid.

Calinski T, Harabasz J (1974). "A dendrite method for cluster analysis." Communications in Statistics - Theory and Methods, 3(1), 1-27. doi:10.1080/03610927408827101. URL http://dx.doi.org/10.1080/03610927408827101.

Chang F, Carey V, Qiu W, Zamar RH, Lazarus R, Wang X (2008). clues: Clustering Method Based on Local Shrinking. R package version 0.5-0, URL http://cran.r-project.org/ web/packages/clues.

Davies DL, Bouldin DW (1979). "A cluster separation measure". IEEE Transactions on Pattern Analysis and Machine Intelligence, 1(2), 224-227.

Dimitriadou E (2009). "cclust: Convex Clustering Methods and Clustering Indexes". R package version 0.6-16, URL http://cran.r-project.org/web/packages/cclust/.

Dimitriadou E, Dolnicar S, Weingessel A (2002). "An examination of indexes for determining the number of clusters in binary data sets". Psychometrika, 67(3), 137-160.

Duda RO, Hart PE (1973). "Pattern classification and scene analysis". John Wiley and Sons, Inc., New York, USA. ISBN 0-471-22361-1.

Dunn J (1974). "Well separated clusters and optimal fuzzy partitions". Journal Cybern, pp. 95-104.

Edwards AWF, Cavalli-Sforza L (1965). "A method for cluster analysis". Biometrics, 21(2), 362-375.

Frey T, Van Groenewoud H (1972). "A cluster analysis of the D-squared matrix of white spruce stands in Saskatchewan based on the maximum-minimum principle". Journal of Ecology, 60(3), 873-886.

Friedman HP, Rubin J (1967). "On some invariant criteria for grouping data". Journal of the American Statistical Association, 62(320), 1159-1178.

Fukunaga K, Koontz WLG (1970). "A criterion and an algorithm for grouping data". IEEE Transactions on Computers, C-19(10), 917-923.

Gordon A (1999). "Classification". 2nd edition. Chapman & Hall/CRC, London. ISBN 1-58488-013-9.

Halkidi M, Batistakis I, Vazirgiannis M (2001). "On clustering validation techniques". Journal of Intelligent Information Systems, 17(2/3), 107-145.

Halkidi M, Vazirgiannis M (2001). "Clustering validity assessment: finding the optimal partitioning of a data set". Proceeding ICDM '01 Proceedings of the 2001 IEEE International Conference on Data Mining, pp. 187-194.

Halkidi M, Vazirgiannis M, Batistakis I (2000). "Quality Scheme Assessment in the Clustering Process". PKDD2000, pp. 265-276.

Hartigan JA (1975). Clustering Algorithms. John Wiley & Sons, Inc., New York, NY, USA. ISBN 047135645X.

Hill RS (1980). "A stopping rule for partitioning dendrograms". Botanical Gazette, 141(3), 321-324.

Hubert LJ, Arabie P (1985). "Comparing partitions". Journal of Classification, 2, 193-218.

Hubert LJ, Levin JR (1976). "A General Statistical Framework for Assessing Categorical Clustering in Free Recall". Psychological Bulletin, 83(6), 1072-1080.
URL http://www.eric.ed.gov/ERICWebPortal/contentdelivery/servlet/ERICServlet?accno=ED116162.

Kaufman L, Rousseeuw P (1990). Finding groups in data: an introduction to cluster analysis. Wiley, New York, NY, USA.

Kraemer HC (1982). "Biserial Correlation", Encyclopaedia of Statistical Sciences, Volume 1. Wiley, pages 276-279.

KrzanowskiWJ, Lai YT (1988)."A Criterion for Determining the Number of Groups in a Data Set Using Sum-of-Squares Clustering". Biometrics, 44(1), 23-34. doi:10.2307/2531893.

Lebart L, Morineau A, Piron M (2000). Statistique exploratoire multidimensionnelle. Dunod, Paris, France. ISBN 2100053515.

Marriot FHC (1971). "Practical problems in a method of cluster analysis." Biometrics, 27(3), 501-514.

Milligan G (1981a). "A Monte Carlo study of thirty internal criterion measures for cluster analysis". Psychometrika, 46(2), 187-199.

Milligan G, Cooper M (1985). "An examination of procedures for determining the number of clusters in a data set." Psychometrika, 50(2), 159-179.

Milligan GW (1980). "An examination of the effect of six types of error perturbation on fifteen clustering algorithms". Psychometrika, 45(3), 325-342.

Nieweglowski L (2009). clv: Cluster Validation Techniques. R package version 0.3-2, URL http://cran.r-project.org/web/packages/clv.

Orloci L (1967). "An agglomerative method for classification of plant communities". Journal of Ecology, 55(1), 193-206.

Ratkowsky DA, Lance GN (1978). "A criterion for determining the number of groups in a classification". Australian Computer Journal, 10, 115-117.

Rholf F (1974). "Methods of comparing classifications". Annual Review of Ecology and Systematics, 5, 101-113.

Rousseeuw P (1987). "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis". Journal of Computational and Applied Mathematics, 20, 53-65.

Sarle WS (1983). "SAS Technical Report A-108, Cubic clustering criterion". Cary, N.C.: SAS Institute Inc.

Scott AJ, Symons MJ (1971). "Clustering methods based on likelihood ratio criteria". Biometrics, 27(2), 387-397.

Sheikholeslami C, Chatterjee S, Zhang A (1998)."WaveCluster: A-MultiResolution Clustering Approach for Very Large Spatial Database." Proceedings of 24th VLDB Conference.

Theodoridis S, Koutroubas K (1999). "Pattern recognition". Academic Press.

Tibshirani R, Walther G, Hastie T (2001). "Estimating the number of clusters in a data set via the gap statistic". Journal of the Royal Statistical Society: Series B (Statistical Methodology), 63(2), 411-423. doi:10.1111/1467-9868.00293. URL http://dx.doi.org/10.1111/1467-9868.00293.

Walesiak M, Dudek A (2011). "clusterSim: Searching for optimal clustering procedure for a data set". R package version 0.40-6, URL http://cran.r-project.org/web/packages/clusterSim.

Examples

```
## A 2-dimensional example
x<-rbind(matrix(rnorm(100,sd=0.1),ncol=2),
         matrix(rnorm(100,mean=1,sd=0.2),ncol=2),
         matrix(rnorm(100,mean=5,sd=0.1),ncol=2),
         matrix(rnorm(100,mean=7,sd=0.2),ncol=2))

NbClust(x, diss="NULL", distance = "euclidean", min.nc=2, max.nc=8,
            method = "complete", index = "ch", alphaBeale = 0.1)

## A 3-dimensional example
x<-rbind(matrix(rnorm(150,sd=0.3),ncol=3),
          matrix(rnorm(150,mean=3,sd=0.2),ncol=3),
          matrix(rnorm(150,mean=5,sd=0.3),ncol=3))
NbClust(x, diss="NULL", distance = "euclidean", min.nc=2, max.nc=10,
            method = "ward", index = "dindex", alphaBeale = 0.1)

## A 5-dimensional example
x<-rbind(matrix(rnorm(150,sd=0.3),ncol=5),
          matrix(rnorm(150,mean=3,sd=0.2),ncol=5),
          matrix(rnorm(150,mean=1,sd=0.1),ncol=5),
          matrix(rnorm(150,mean=6,sd=0.3),ncol=5),
          matrix(rnorm(150,mean=9,sd=0.3),ncol=5))
NbClust(x, diss="NULL", distance = "euclidean", min.nc=2, max.nc=10,
            method = "ward", index = "all", alphaBeale = 0.1)

## A real data example
data<-iris[,-c(5)]
NbClust(data, diss="NULL", distance = "euclidean", min.nc=2, max.nc=6,
            method = "ward", index = "kl", alphaBeale = 0.1) ## KL index

NbClust(data, diss="NULL", distance = "euclidean", min.nc=2, max.nc=6,
            method = "kmeans", index = "hubert", alphaBeale = 0.1)

NbClust(data, diss="NULL", distance = "manhattan", min.nc=2, max.nc=6,
            method = "complete", index = "all", alphaBeale = 0.1)
            ## Only indices with low computational cost (26 indices).
```

Index