Package ‘NbClust’

April 13, 2015

Type Package
Title Determining the Best Number of Clusters in a Data Set
Version 3.0
Depends R (>= 3.1.0)
Date 2015-04-13
Author Malika Charrad and Nadia Ghazzali and Veronique Boiteau and Azam Niknafs
Maintainer Malika Charrad <malika.charrad.1@ulaval.ca>
Description It provides 30 indexes for determining the optimal number of clusters in a data set and offers the best clustering scheme from different results to the user.
URL https://sites.google.com/site/malikacharrad/research/nbclust-package
License GPL-2
NeedsCompilation no
Repository CRAN
Date/Publication 2015-04-13 22:54:43

R topics documented:

NbClust

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

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

Arguments

data matrix or dataset.
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 1 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.D",
"ward.D2", "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", "sil-
houette", "duda", "pseudot2", "beale", "rakowsky", "ball", "ptbiserial", "gap",
"frey", "mcclain", "gamma", "gplus", "tau", "dunn", "hubert", "sdindex", "din-
dex", "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. Two different algorithms are found in the literature for Ward clustering. The one used by option "ward.D" (equivalent to the only Ward option "ward" in R versions <= 3.0.3) does not implement Ward’s (1963) clustering criterion, whereas option "ward.D2" implements that criterion (Murtagh and Legendre 2013). With the latter, the dissimilarities are squared before cluster updating.

• **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 \(x \in C_i, y \in C_j\).

\[ D_{ij} = \frac{\sum_{x \in C_i, y \in C_j} d(x, y)}{n_i \times n_j} \]
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:
  \[
  D_{ij} = \frac{(D_{ik} + D_{jl})}{2}
  \]
  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:
  \[
  D_{ij} = \frac{(D_{ik} + D_{jl})}{2} - \frac{D_{kl}}{4}
  \]
  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

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

<table>
<thead>
<tr>
<th>Index in NbClust</th>
<th>Optimal number of clusters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. &quot;kl&quot; or &quot;all&quot; or &quot;alllong&quot; (Krzanowski and Lai 1988)</td>
<td>Maximum value of the index</td>
</tr>
<tr>
<td>2. &quot;ch&quot; or &quot;all&quot; or &quot;alllong&quot; (Calinski and Harabasz 1974)</td>
<td>Maximum value of the index</td>
</tr>
<tr>
<td>3. &quot;hartigan&quot; or &quot;all&quot; or &quot;alllong&quot; (Hartigan 1975)</td>
<td>Maximum difference between hierarchy levels of the index</td>
</tr>
<tr>
<td>4. &quot;ccc&quot; or &quot;all&quot; or &quot;alllong&quot; (Sarle 1983)</td>
<td>Maximum value of the index</td>
</tr>
<tr>
<td>5. &quot;scott&quot; or &quot;all&quot; or &quot;alllong&quot; (Scott and Symons 1971)</td>
<td>Maximum difference between hierarchy levels of the index</td>
</tr>
<tr>
<td>6. &quot;marriot&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
<td>Max. value of second differences</td>
</tr>
<tr>
<td></td>
<td>Description</td>
</tr>
<tr>
<td>---</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>1</td>
<td>NbClust</td>
</tr>
<tr>
<td>2</td>
<td>&quot;tcovw&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>3</td>
<td>Maximum difference between levels of the index</td>
</tr>
<tr>
<td>4</td>
<td>&quot;tracew&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>5</td>
<td>Maximum value of absolute second differences between levels of the index</td>
</tr>
<tr>
<td>6</td>
<td>&quot;friedman&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>7</td>
<td>Maximum difference between hierarchy levels of the index</td>
</tr>
<tr>
<td>8</td>
<td>&quot;rubin&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>9</td>
<td>Minimum value of second differences between levels of the index</td>
</tr>
<tr>
<td>10</td>
<td>&quot;cindex&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>11</td>
<td>Minimum value of the index</td>
</tr>
<tr>
<td>12</td>
<td>&quot;db&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>13</td>
<td>Minimum value of the index</td>
</tr>
<tr>
<td>14</td>
<td>&quot;silhouette&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>15</td>
<td>Maximum value of the index</td>
</tr>
<tr>
<td>16</td>
<td>&quot;duda&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>17</td>
<td>Smallest ( n_c ) such that index ( &gt; ) criticalValue</td>
</tr>
<tr>
<td>18</td>
<td>&quot;pseudot2&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>19</td>
<td>Smallest ( n_c ) such that index ( &lt; ) criticalValue</td>
</tr>
<tr>
<td>20</td>
<td>&quot;beale&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>21</td>
<td>( n_c ) such that critical value of the index ( \geq ) alpha</td>
</tr>
<tr>
<td>22</td>
<td>&quot;ratkowsky&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>23</td>
<td>Maximum value of the index</td>
</tr>
<tr>
<td>24</td>
<td>&quot;ball&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>25</td>
<td>Maximum difference between hierarchy levels of the index</td>
</tr>
<tr>
<td>26</td>
<td>&quot;ptbiserial&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>27</td>
<td>Maximum value of the index</td>
</tr>
<tr>
<td>28</td>
<td>&quot;gap&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>29</td>
<td>Smallest ( n_c ) such that criticalValue ( \geq 0 )</td>
</tr>
<tr>
<td>30</td>
<td>&quot;frey&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>31</td>
<td>the cluster level before that index value ( &lt; 1.00 )</td>
</tr>
<tr>
<td>32</td>
<td>&quot;mcclain&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>33</td>
<td>Minimum value of the index</td>
</tr>
<tr>
<td>34</td>
<td>&quot;gamma&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>35</td>
<td>Maximum value of the index</td>
</tr>
<tr>
<td>36</td>
<td>&quot;gplus&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>37</td>
<td>Minimum value of the index</td>
</tr>
<tr>
<td>38</td>
<td>&quot;tau&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>39</td>
<td>Maximum value of the index</td>
</tr>
<tr>
<td>40</td>
<td>&quot;dunn&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>41</td>
<td>Maximum value of the index</td>
</tr>
<tr>
<td>42</td>
<td>&quot;hubert&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>43</td>
<td>Graphical method</td>
</tr>
<tr>
<td>44</td>
<td>&quot;sindex&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>45</td>
<td>Minimum value of the index</td>
</tr>
<tr>
<td>46</td>
<td>&quot;dindex&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>47</td>
<td>Graphical method</td>
</tr>
<tr>
<td>48</td>
<td>&quot;sdbw&quot; or &quot;all&quot; or &quot;alllong&quot;</td>
</tr>
<tr>
<td>49</td>
<td>Minimum value of the index</td>
</tr>
</tbody>
</table>
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.

Best.partition Partition that corresponds to the best number of clusters

Author(s)

Malika Charrad, Nadia Ghazzali, Veronique Boiteau and Azam Niknafs

References


Examples

```r
## DATA MATRIX IS GIVEN

## A 2-dimensional example

set.seed(1)
x<-cbind(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))
res<-NbClust(x, distance = "euclidean", min.nc=2, max.nc=8,
             method = "complete", index = "ch")
res$All.index
res$Best.nc
res$Best.partition

## A 5-dimensional example

set.seed(1)
x<-cbind(matrix(rnorm(150, sd=0.3), ncol=5),
           matrix(rnorm(150, mean=3, sd=0.2), ncol=5),
           matrix(rnorm(150, mean=7, sd=0.2), ncol=5),
           matrix(rnorm(150, mean=10, sd=0.2), ncol=5),
           matrix(rnorm(150, mean=10, sd=0.3), ncol=5))
res<-NbClust(x, distance = "euclidean", min.nc=2, max.nc=8,
             method = "complete", index = "ch")
res$All.index
res$Best.nc
res$Best.partition
```
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))

res<-NbClust(x, distance = "euclidean", min.nc=2, max.nc=10,
method = "ward.D", index = "all")

res$All.index
res$Best.nc
res$All.CriticalValues
res$Best.partition

## A real data example

data<-iris[-c(5)]
res<-NbClust(data, diss=NULL, distance = "euclidean", min.nc=2, max.nc=6,
method = "ward.D2", index = "kl")

res$All.index
res$Best.nc
res$Best.partition

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

res$All.index

res<-NbClust(data, diss=NULL, distance = "manhattan", min.nc=2, max.nc=6,
method = "complete", index = "all")

res$All.index
res$Best.nc
res$All.CriticalValues
res$Best.partition

## Examples with a dissimilarity matrix

## Data matrix is given

set.seed(1)
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))
diss_matrix<- dist(x, method = "euclidean", diag=FALSE)
res<-NbClust(x, diss=diss_matrix, distance = NULL, min.nc=2, max.nc=6,
method = "ward.D", index = "ch")

res$All.index
res$Best.nc
res$Best.partition

data<-iris[-c(5)]
diss_matrix<- dist(data, method = "euclidean", diag=FALSE)
NbClust(data, diss=diss_matrix, distance = NULL, min.nc=2, max.nc=6,
method = "ward.D2", index = "all")

res$All.index
res$Best.nc
res$All.CriticalValues
res$Best.partition

set.seed(1)
x<-cbind(matrix(rnorm(20, sd=0.1), ncol=2),
         matrix(rnorm(20, mean=1, sd=0.2), ncol=2),
         matrix(rnorm(20, mean=5, sd=0.1), ncol=2),
         matrix(rnorm(20, mean=7, sd=0.2), ncol=2))
diss_matrix<- dist(x, method = "euclidean", diag=FALSE)
res<-NbClust(x, diss=diss_matrix, distance = NULL, min.nc=2, max.nc=6,
             method = "ward.2", index = "alllong")
res$All.index
res$Best.nc
res$All.CriticalValues
res$Best.partition

## Data matrix is not available. Only the dissimilarity matrix is given
## In this case, only these indices can be computed : frey, mcclain, cindex, silhouette and dunn
res<-NbClust(diss=diss_matrix, distance = NULL, min.nc=2, max.nc=6,
             method = "ward.2", index = "silhouette")
res$All.index
res$Best.nc
res$All.CriticalValues
res$Best.partition
Index

*Topic Number of clusters
   NbClust, 1
*Topic R packages
   NbClust, 1
*Topic Validity Indices
   NbClust, 1
*Topic cluster validity
   NbClust, 1
*Topic clustering algorithms
   NbClust, 1
*Topic clustering validation
   NbClust, 1