Package: nomclust (via r-universe)

September 8, 2024

Title Hierarchical Cluster Analysis of Nominal Data

Author Zdenek Sulc [aut, cre], Jana Cibulkova [aut], Hana Rezankova [aut], Jaroslav Hornicek [aut]

Maintainer Zdenek Sulc <zdenek.sulc@vse.cz>

Version 2.8.0

Date 2023-8-18

Description Similarity measures for hierarchical clustering of objects characterized by nominal (categorical) variables. Evaluation criteria for nominal data clustering.

Depends cluster, methods, clValid

License GPL (>= 2)

RoxygenNote 7.2.3

NeedsCompilation yes

Encoding UTF-8

Imports Rcpp (>= 0.11.0)

LinkingTo Rcpp

Date/Publication 2023-08-18 10:12:38 UTC

Repository https://zdeneksulc.r-universe.dev

RemoteUrl https://github.com/cran/nomclust

RemoteRef HEAD

RemoteSha 63d7ad4664c5fe23ebb8bf1b961ae421b0329229

Contents

`anderberg` *Anderberg (AN) Measure*

Description

The function calculates a dissimilarity matrix based on the AN similarity measure.

Usage

```
anderberg(data)
```

Arguments

`data`	A data.frame or a matrix with cases in rows and variables in columns.

Details

The Anderberg similarity measure was presented in (Anderberg, 1973). The measure assigns higher weights to infrequent matches and mismatches. It takes on values from zero to one. The minimum similarity is attained when there are no matches and vice versa, see (Borian et al., 2008).

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Andergerg M.R. (1973). Cluster analysis for applications. Academic Press, New York.

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

See Also

burnaby, eskin, gambaryan, goodall1, goodall2, goodall3, goodall4, iof, lin, lin1, of, sm, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.anderberg <- anderberg(data20)
```

`as.agnes` *Convert Objects to Class agnes, twins*

Description

Converts objects of the class "nomclust" to the class "agnes, twins".

Usage

```
as.agnes(x, ...)
```

Arguments

x	The "nomclust" object containing components "dend" and "prox".
...	Further arguments passed to or from other methods.

Value

The function returns an object of class "agnes, twins".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

See Also

agnes, as.hclust and hclust.

Examples

```
# sample data
data(data20)

# creating an object with results of hierarchical clustering of
hca.object <- nomclust(data20, measure = "lin", method = "average",
 clu.high = 5, prox = TRUE)

# nomclust plot
plot(hca.object)

# obtaining the agnes, twins object
hca.object.agnes <- as.agnes(hca.object)

# agnes plot
plot(hca.object.agnes)

# obtaining the hclust object
hca.object.hclust <- as.hclust(hca.object)

# hclust plot
plot(hca.object.hclust)
```

---

burnaby *Burnaby (BU) Measure*

Description

The function calculates a dissimilarity matrix based on the BU similarity measure.

Usage

```
burnaby(data, var.weights = NULL)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in columns.
var.weights	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The Burnaby similarity measure was presented in (Burnaby, 1970). The measure assigns low similarity to mismatches on rare values and high similarity to mismatches on frequent values, see (Borian et al., 2008).

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Burnaby T. (1970). On a method for character weighting a similarity coefficient, employing the concept of information. Mathematical Geology, 2(1), 25-38.

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

See Also

anderberg, eskin, gambaryan, goodall1, goodall2, goodall3, goodall4, iof, lin, lin1, of, sm, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.burnaby <- burnaby(data20)

# dissimilarity matrix calculation with variable weights
weights.burnaby <- burnaby(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

CA.methods *Selected clustering algorithms*

Description

The dataset contains five different characteristics of 24 clustering algorithms. The "Type" variable expresses the principle on which the clustering is based. There are five possible categories: density, grid, hierarchical, model-based, and partitioning. The binary variable "OptClu" indicates if the clustering algorithm offers the optimal number of clusters. The variable "Large" indicates if the clustering algorithm was designed to cluster large datasets. The "TypicalType" variable presents the typical data type for which the clustering algorithm was determined. There are three possible categories: categorical, mixed, and quantitative. Since some clustering algorithms support more data types, the binary variable "MoreTypes" indicates this support.

Usage

```
data("CA.methods")
```

Format

A data frame containing 5 variables and 24 cases.

Source

created by the authors of the nomclust package

---

data20 *Artificial nominal dataset*

Description

This dataset consists of 5 nominal variables and 20 cases. Its main aim is to demonstrate the desired entry data structure for the nomclust package.

Usage

```
data(data20)
```

Format

A data frame containing 5 variables and 20 cases.

Source

created by the authors of the nomclust package

`dend.plot` *Visualization of Cluster Hierarchy using a Dendrogram*

Description

The function `dend.plot()` visualizes the hierarchy of clusters using a dendrogram. The function also enables a user to mark the individual clusters with colors. The number of displayed clusters can be defined either by a user or by one of the five evaluation criteria.

Usage

```
dend.plot(
  x,
  clusters = "BIC",
  style = "greys",
  colorful = TRUE,
  clu.col = NA,
  main = "Dendrogram",
  ac = TRUE,
  ...
)
```

Arguments

`x`	An output of the `nomclust()` or `nomprox()` functions containing the `dend` component.
`clusters`	Either a *numeric* value or a *character* string with the name of the evaluation criterion expressing the number of displayed clusters in a dendrogram. The following evaluation criteria can be used: `"AIC"`, `"BIC"`, `"BK"`, `"PSFE"` and `"PSFM"`.
`style`	A *character* string or a *vector* of colors defines a graphical style of the produced plots. There are two predefined styles in the **nomclust** package, namely `"greys"` and `"dark"`, but a custom color scheme can be set by a user as a vector of a length four.
`colorful`	A *logical* argument specifying if the output will be colorful or black and white.
`clu.col`	An optional *vector* of colors which allows a researcher to apply user-defined colors for displayed (marked) clusters in a dendrogram.
`main`	A *character* string with the chart title.
`ac`	A *logical* argument indicating if an agglomerative coefficient will be present in the output.
`...`	Other graphical arguments compatible with the generic `plot()` function.

Details

The function can be applied to a `nomclust()` or `nomprox()` output containing the `dend` component. This component is not available when the optimization process is used.

Value

The function returns a dendrogram describing the hierarchy of clusters that can help to identify the optimal number of clusters.

Author(s)

Jana Cibulkova and Zdenek Sulc.
Contact: <jana.cibulkova@vse.cz>

See Also

eval.plot, nomclust, nomprox.

Examples

```
# sample data
data(data20)

# creating an object with results of hierarchical clustering
hca.object <- nomclust(data20, measure = "iof", eval = TRUE)

# a basic plot
dend.plot(hca.object)

# a dendrogram with color-coded clusters according to the BIC index
dend.plot(hca.object, clusters = "BIC", colorful = TRUE)

# using a dark style and specifying own colors in a solution with three clusters
dend.plot(hca.object, clusters = 3, style = "dark", clu.col = c("blue", "red", "green"))

# a black and white dendrogram
dend.plot(hca.object, clusters = 3, style = "dark", colorful = FALSE)
```

---

eskin *Eskin (ES) Measure*

Description

The function calculates a dissimilarity matrix based on the ES similarity measure.

Usage

```
eskin(data, var.weights = NULL)
```

Arguments

`data`	A data.frame or a matrix with cases in rows and variables in columns.
`var.weights`	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The Eskin similarity measure was proposed by Eskin et al. (2002) and examined by Boriah et al., (2008). It is constructed to assign higher weights to mismatches on variables with more categories.

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

Eskin E., Arnold A., Prerau M., Portnoy L. and Stolfo S. (2002). A geometric framework for unsupervised anomaly detection. In D. Barbara and S. Jajodia (Eds): Applications of Data Mining in Computer Security, p. 78-100. Norwell: Kluwer Academic Publishers.

See Also

anderberg, burnaby, gambaryan, goodall1, goodall2, goodall3, goodall4, iof, lin, lin1, of, sm, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.eskin <- eskin(data20)

# dissimilarity matrix calculation with variable weights
weights.eskin <- eskin(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

`eval.plot` *Visualization of Evaluation Criteria*

Description

The function visualizes the values of up to eight evaluation criteria for the range of cluster solutions defined by the user in the **nomclust**, **evalclust** or **nomprox** functions. It also indicates the optimal number of clusters determined by these criteria. The charts for the evaluation criteria in the **nomclust** package.

Usage

```
eval.plot(
  x,
  criteria = "all",
  style = "greys",
  opt.col = "red",
  main = "Cluster Evaluation",
  ...
)
```

Arguments

`x`	An output of the "nomclust" object containing the `eval` and `opt` components.
`criteria`	A character string or character vector specifying the criteria that are going to be visualized. It can be selected one particular criterion, a vector of criteria, or all the available criteria by typing `"all"`.
`style`	A character string or a vector of colors defines the graphical style of the produced plots. There are two predefined styles in the **nomclust** package, namely `"greys"` and `"dark"`, but a custom color scheme can be set by a user as a vector of a length four.
`opt.col`	An argument specifying a color that is used for the optimal number of clusters identification.
`main`	A character string with the chart title.
`...`	Other graphical arguments compatible with the generic `plot()` function.

Details

The function can display up to eight evaluation criteria. Namely, Within-cluster mutability coefficient (WCM), Within-cluster entropy coefficient (WCE), Pseudo F Indices based on the mutability (PSFM) and the entropy (PSFE), Bayesian (BIC), and Akaike (AIC) information criteria for categorical data, the BK index, and the silhouette index (SI).

Value

The function returns a series of up to eight plots with evaluation criteria values and the graphical indication of the optimal numbers of clusters (for AIC, BIC, BK, PSFE, PSFM, SI).

Author(s)

Jana Cibulkova and Zdenek Sulc.
Contact: <jana.cibulkova@vse.cz>

See Also

dend.plot, nomclust, evalclust, nomprox.

Examples

```
# sample data
data(data20)

# creating an object with results of hierarchical clustering
hca.object <- nomclust(data20, measure = "iof", eval = TRUE)

# a default series of plots
eval.plot(hca.object)

# changing the color indicating the optimum number of clusters
eval.plot(hca.object, opt.col= "darkorange")

# selecting only AIC and BIC criteria with the dark style
eval.plot(hca.object, criteria = c("AIC", "BIC"), style = "dark")

# selecting only SI
eval.plot(hca.object, criteria = "SI")
```

---

evalclust *Cluster Quality Evaluation of Nominal Data Hierarchical Clustering*

Description

The function evaluates clustering results by a set of evaluation criteria (cluster validity indices).

Usage

```
evalclust(data, clusters, diss = NULL)
```

Arguments

`data`	A data.frame or a matrix with cases in rows and variables in columns.
`clusters`	A data.frame or a list of cluster memberships obtained based on the dataset defined in the parameter `data` in the form of a sequence from the two-cluster solution to the maximal-cluster solution.
`diss`	An optional parameter. A matrix or a dist object containing dissimilarities calculated based on the dataset defined in the parameter `data`.

Details

The function calculates a set of evaluation criteria if the original dataset and the cluster membership variables are provided. The function calculates up to 13 evaluation criteria described by (Sulc et al., 2018) and (Corter and Gluck, 1992) and provides the optimal number of clusters based on these criteria. It is primarily focused on evaluating hierarchical clustering results obtained by similarity measures different from those that occur in the nomclust package. Thus, it can serve for the comparison of various similarity measures for categorical data.

Value

The function returns a list with three components.

The `eval` component contains up to 13 evaluation criteria as vectors in a list. Namely, Within-cluster mutability coefficient (WCM), Within-cluster entropy coefficient (WCE), Pseudo F Indices based on the mutability (PSFM) and the entropy (PSFE), Bayesian (BIC), and Akaike (AIC) information criteria for categorical data, the BK index, Category Utility (CU), Category Information (CI), Hartigan Mutability (HM), Hartigan Entropy (HE) and, if the prox component is present, the silhouette index (SI) and the Dunn index (DI).

The `opt` component is present in the output together with the `eval` component. It displays the optimal number of clusters for the evaluation criteria from the `eval` component, except for WCM and WCE, where the optimal number of clusters is based on the elbow method.

The `call` component contains the function call.

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Corter J.E., Gluck M.A. (1992). Explaining basic categories: Feature predictability and information. Psychological Bulletin 111(2), p. 291–303.

Sulc Z., Cibulkova J., Prochazka J., Rezankova H. (2018). Internal Evaluation Criteria for Categorical Data in Hierarchical Clustering: Optimal Number of Clusters Determination, Metodoloski Zveski, 15(2), p. 1-20.

See Also

nomclust, nomprox, eval.plot.

Examples

```
# sample data
data(data20)

# creating an object with results of hierarchical clustering
hca.object <- nomclust(data20, measure = "iof", method = "average", clu.high = 7)

# the cluster memberships
data20.clu <- hca.object$mem

# obtaining evaluation criteria for the provided dataset and cluster memberships
data20.eval <- evalclust(data20, clusters = data20.clu)

# visualization of the evaluation criteria
eval.plot(data20.eval)

# silhouette index can be calculated if the dissimilarity matrix is provided
data20.eval <- evalclust(data20, clusters = data20.clu, diss = hca.object$prox)
eval.plot(data20.eval, criteria = "SI")
```

---

gambaryan *Gambaryan (GA) Measure*

Description

The function calculates a dissimilarity matrix based on the GA similarity measure.

Usage

```
gambaryan(data)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in columns.

Details

The Gambaryan similarity measure was presented in (Gambaryan, 1964). The measure assigns low weight to matches where the matching value occurs in about half the dataset, i.e., in between being frequent and rare, see (Borian et al., 2008).

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Gambaryan P. (1964). A mathematical model of taxonomy. SSR, 17(12), 47-53.

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

See Also

anderberg, burnaby, eskin, goodall1, goodall2, goodall3, goodall4, iof, lin, lin1, of, sm, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.gambaryan <- gambaryan(data20)
```

---

goodall1 *Goodall 1 (G1) Measure*

Description

The function calculates a dissimilarity matrix based on the G1 similarity measure.

Usage

```
goodall1(data, var.weights = NULL)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in column.
var.weights	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The Goodall 1 similarity measure was presented in (Boriah et al., 2008). It is a simple modification of the original Goodall measure (Goodall, 1966). The measure assigns higher weights to infrequent matches.

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

Goodall V.D. (1966). A new similarity index based on probability. Biometrics, 22(4), p. 882.

See Also

anderberg, burnaby, eskin, gambaryan, goodall2, goodall3, goodall4, iof, lin, lin1, of, sm, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.goodall1 <- goodall1(data20)

# dissimilarity matrix calculation with variable weights
weights.goodall1 <- goodall1(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

---

goodall2 *Goodall 2 (G2) Measure*

Description

The function calculates a dissimilarity matrix based on the G2 similarity measure.

Usage

```
goodall2(data, var.weights = NULL)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in columns.
var.weights	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The Goodall 2 similarity measure was presented in (Boriah et al., 2008). It is a simple modification of the original Goodall measure (Goodall, 1966). The measure assigns weight to infrequent matches under the condition that there are also other categories, which are even less frequent than the examined one.

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

Goodall V.D. (1966). A new similarity index based on probability. Biometrics, 22(4), p. 882.

See Also

anderberg, burnaby, eskin, gambaryan, goodall1, goodall3, goodall4, iof, lin, lin1, of, sm, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.goodall2 <- goodall2(data20)

# dissimilarity matrix calculation with variable weights
weights.goodall2 <- goodall2(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

goodall3 *Goodall 3 (G3) Measure*

Description

The function calculates a dissimilarity matrix based on the G3 similarity measure.

Usage

```
goodall3(data, var.weights = NULL)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in columns.
var.weights	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The Goodall 3 similarity measure was presented in (Boriah et al., 2008). It is a simple modification of the original Goodall measure (Goodall, 1966). The measure assigns higher weight if the infrequent categories match regardless on frequencies of other categories.

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

Goodall V.D. (1966). A new similarity index based on probability. Biometrics, 22(4), p. 882.

See Also

anderberg, burnaby, eskin, gambaryan, goodall1, goodall2, goodall4, iof, lin, lin1, of, sm, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.goodall3 <- goodall3(data20)

# dissimilarity matrix calculation with variable weights
weights.goodall3 <- goodall3(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

---

goodall4 *Goodall 4 (G4) Measure*

Description

The function calculates a dissimilarity matrix based on the G4 similarity measure.

Usage

```
goodall4(data, var.weights = NULL)
```

Arguments

`data`	A data.frame or a matrix with cases in rows and variables in columns.
`var.weights`	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The Goodall 4 similarity measure was presented in (Boriah et al., 2008). It is a simple modification of the original Goodall measure (Goodall, 1966). It assigns higher weights to the frequent categories matches.

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

Goodall V.D. (1966). A new similarity index based on probability. Biometrics, 22(4), p. 882.

See Also

anderberg, burnaby, eskin, gambaryan, goodall1, goodall2, goodall3, iof, lin, lin1, of, sm, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.goodall4 <- goodall4(data20)

# dissimilarity matrix calculation with variable weights
weights.goodall4 <- goodall4(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

---

iof *Inverse Occurence Frequency (IOF) Measure*

Description

The function calculates a dissimilarity matrix based on the IOF similarity measure.

Usage

```
iof(data, var.weights = NULL)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in columns.
var.weights	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The IOF (Inverse Occurrence Frequency) measure was originally constructed for the text mining tasks, see (Sparck-Jones, 1972), later, it was adjusted for categorical variables, see (Boriah et al., 2008). The measure assigns higher weight to mismatches on less frequent values and vice versa.

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

Spark-Jones K. (1972). A statistical interpretation of term specificity and its application in retrieval. In Journal of Documentation, 28(1), 11-21. Later: Journal of Documentation, 60(5) (2002), 493-502.

See Also

anderberg, burnaby, eskin, gambaryan, goodall1, goodall2, goodall3, goodall4, lin, lin1, of, sm, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.iof <- iof(data20)

# dissimilarity matrix calculation with variable weights
weights.iof <- iof(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

---

lin *Lin (LIN) Measure*

Description

The function calculates a dissimilarity matrix based on the LIN similarity measure.

Usage

```
lin(data, var.weights = NULL)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in columns.
var.weights	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The Lin measure was introduced by Lin (1998) and presented in (Boriah et al., 2008). The measure assigns higher weights to more frequent categories in case of matches and lower weights to less frequent categories in case of mismatches.

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

Lin D. (1998). An information-theoretic definition of similarity. In: ICML '98: Proceedings of the 15th International Conference on Machine Learning. San Francisco, p. 296-304.

See Also

anderberg, burnaby, eskin, gambaryan, goodall1, goodall2, goodall3, goodall4, iof, lin1, of, sm, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.lin <- lin(data20)

# dissimilarity matrix calculation with variable weights
weights.lin<- lin(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

lin1 *Lin 1 (LIN1) Measure*

Description

The function calculates a dissimilarity matrix based on the LIN1 similarity measure.

Usage

```
lin1(data, var.weights = NULL)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in columns.
var.weights	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The Lin 1 similarity measure was introduced in (Boriah et al., 2008) as a modification of the original Lin measure (Lin, 1998). In has a complex system of weights. In case of mismatch, lower similarity is assigned if either the mismatching values are very frequent or their relative frequency is in between the relative frequencies of mismatching values. Higher similarity is assigned if the mismatched categories are infrequent and there are a few other infrequent categories. In case of match, lower similarity is given for matches on frequent categories or matches on categories that have many other values of the same frequency. Higher similarity is given to matches on infrequent categories.

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

Lin D. (1998). An information-theoretic definition of similarity. In: ICML '98: Proceedings of the 15th International Conference on Machine Learning. San Francisco, p. 296-304.

See Also

anderberg, burnaby, eskin, gambaryan, goodall1, goodall2, goodall3, goodall4, iof, lin, of, sm, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.lin1 <- lin1(data20)

# dissimilarity matrix calculation with variable weights
weights.lin1 <- lin1(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

---

nomclust *Hierarchical Clustering of Nominal Data*

Description

The function performs and evaluates hierarchical cluster analysis of nominal data.

Usage

```
nomclust(
  data,
  measure = "lin",
  method = "average",
  clu.high = 6,
  eval = TRUE,
  prox = 100,
  var.weights = NULL
)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in columns.
measure	A character string defining the similarity measure used for computation of proximity matrix in HCA: "anderberg", "burnaby", "eskin", "gambaryan", "goodall1", "goodall2", "goodall3", "goodall4", "iof", "lin", "lin1", "of", "sm", "smirnov", "ve", "vm".
method	A character string defining the clustering method. The following methods can be used: "average", "complete", "single".
clu.high	A numeric value expressing the maximal number of cluster for which the cluster memberships variables are produced.

`eval`	A logical operator; if TRUE, evaluation of the clustering results is performed.
`prox`	A logical operator or a numeric value. If a logical value TRUE indicates that the proximity matrix is a part of the output. A numeric value (integer) of this argument indicates the maximal number of cases in a dataset for which a proximity matrix will occur in the output.
`var.weights`	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The function runs hierarchical cluster analysis (HCA) with objects characterized by nominal variables (without natural order of categories). It completely covers the clustering process, from the dissimilarity matrix calculation to the cluster quality evaluation. The function enables a user to choose from the similarity measures for nominal data summarized by (Boriah et al., 2008) and by (Sulc and Rezankova, 2019). Next, it offers to choose from three linkage methods that can be used for categorical data. It is also possible to assign user-defined variable weights. The obtained clusters can be evaluated by up to 13 evaluation criteria (Sulc et al., 2018) and (Corter and Gluck, 1992). The output of the nomclust() function may serve as an input for the visualization functions *dend.plot* and *eval.plot* in the nomclust package.

Value

The function returns a list with up to six components.

The `mem` component contains cluster membership partitions for the selected numbers of clusters in the form of a list.

The `eval` component contains up to 13 evaluation criteria as vectors in a list. Namely, Within-cluster mutability coefficient (WCM), Within-cluster entropy coefficient (WCE), Pseudo F Indices based on the mutability (PSFM) and the entropy (PSFE), Bayesian (BIC), and Akaike (AIC) information criteria for categorical data, the BK index, Category Utility (CU), Category Information (CI), Hartigan Mutability (HM), Hartigan Entropy (HE) and, if the prox component is present, the silhouette index (SI) and the Dunn index (DI).

The `opt` component is present in the output together with the `eval` component. It displays the optimal number of clusters for the evaluation criteria from the `eval` component, except for WCM and WCE, where the optimal number of clusters is based on the elbow method.

The `dend` component can be found in the output together with the `prox` component. It contains all the necessary information for dendrogram creation.

The `prox` component contains the dissimilarity matrix in the form of the "dist" object.

The `call` component contains the function call.

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Boriah S., Chandola V. and Kumar, V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

Corter J.E., Gluck M.A. (1992). Explaining basic categories: Feature predictability and information. Psychological Bulletin 111(2), p. 291–303.

Sulc Z., Cibulkova J., Prochazka J., Rezankova H. (2018). Internal Evaluation Criteria for Categorical Data in Hierarchical Clustering: Optimal Number of Clusters Determination, Metodoloski Zveski, 15(2), p. 1-20.

Sulc Z. and Rezankova H. (2019). Comparison of Similarity Measures for Categorical Data in Hierarchical Clustering. Journal of Classification, 35(1), p. 58-72. DOI: 10.1007/s00357-019-09317-5.

See Also

evalclust, nomprox, eval.plot, dend.plot.

Examples

```
# sample data
data(data20)

# creating an object with results of hierarchical clustering of
hca.object <- nomclust(data20, measure = "lin", method = "average",
 clu.high = 5, prox = TRUE)

# assigning variable weights
hca.weights <- nomclust(data20, measure = "lin", method = "average",
 clu.high = 5, prox = TRUE, var.weights = c(0.7, 1, 0.9, 0.5, 0))

# quick clustering summary
summary(hca.object)

# quick cluster quality evaluation
print(hca.object)

# visualization of the evaluation criteria
eval.plot(hca.object)

# a quick dendrogram
plot(hca.object)

# a dendrogram with three designated clusters
dend.plot(hca.object, clusters = 3)

# obtaining values of evaluation indices as a data.frame
data20.eval <- as.data.frame(hca.object$eval)
```

```
# getting the optimal numbers of clusters as a data.frame
data20.opt <- as.data.frame(hca.object$opt)

# extracting cluster membership variables as a data.frame
data20.mem <- as.data.frame(hca.object$mem)

# obtaining a proximity matrix
data20.prox <- as.matrix(hca.object$prox)

# setting the maximal number of objects for which a proximity matrix is provided in the output to 30
hca.object <- nomclust(data20, measure = "iof", method = "complete",
 clu.high = 5, prox = 30)

# transforming the nomclust object to the class "hclust"
hca.object.hclust <- as.hclust(hca.object)

# transforming the nomclust object to the class "agnes, twins"
hca.object.agnes <- as.agnes(hca.object)
```

---

nomprox *Hierarchical Clustering of Nominal Data Based on a Proximity Matrix*

Description

The function performs hierarchical cluster analysis based on a dissimilarity matrix.

Usage

```
nomprox(
  diss,
  data = NULL,
  method = "average",
  clu.high = 6,
  eval = TRUE,
  prox = 100
)
```

Arguments

diss	A proximity matrix or a dist object calculated based on the dataset defined in a parameter data.
data	A data.frame or a matrix with cases in rows and variables in columns.
method	A character string defining the clustering method. The following methods can be used: "average", "complete", "single".
clu.high	A numeric value that expresses the maximal number of clusters for which the cluster membership variables are produced.

eval	A logical operator; if TRUE, evaluation of clustering results is performed.
prox	A logical operator or a numeric value. If a logical value TRUE indicates that the proximity matrix is a part of the output. A numeric value (integer) of this argument indicates the maximal number of cases in a dataset for which a proximity matrix will occur in the output.

Details

The function performs hierarchical cluster analysis in situations when the proximity (dissimilarity) matrix was calculated externally. For instance, in a different R package, in an own-created function, or in other software. It offers three linkage methods that can be used for categorical data. The obtained clusters can be evaluated by up to 13 evaluation criteria (Sulc et al., 2018) and (Corter and Gluck, 1992).

Value

The function returns a list with up to six components:

The mem component contains cluster membership partitions for the selected numbers of clusters in the form of a list.

The eval component contains up to 13 evaluation criteria as vectors in a list. Namely, Within-cluster mutability coefficient (WCM), Within-cluster entropy coefficient (WCE), Pseudo F Indices based on the mutability (PSFM) and the entropy (PSFE), Bayesian (BIC), and Akaike (AIC) information criteria for categorical data, the BK index, Category Utility (CU), Category Information (CI), Hartigan Mutability (HM), Hartigan Entropy (HE) and, if the prox component is present, the silhouette index (SI) and the Dunn index (DI).

The opt component is present in the output together with the eval component. It displays the optimal number of clusters for the evaluation criteria from the eval component, except for WCM and WCE, where the optimal number of clusters is based on the elbow method.

The dend component can be found in the output only together with the prox component. It contains all the necessary information for dendrogram creation.

The prox component contains the dissimilarity matrix in the form of the "dist" object.

The call component contains the function call.

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Corter J.E., Gluck M.A. (1992). Explaining basic categories: Feature predictability and information. Psychological Bulletin 111(2), p. 291–303.

Sulc Z., Cibulkova J., Prochazka J., Rezankova H. (2018). Internal Evaluation Criteria for Categorical Data in Hierarchical Clustering: Optimal Number of Clusters Determination, Metodoloski Zveski, 15(2), p. 1-20.

See Also

nomclust, evalclust, eval.plot.

Examples

```
# sample data
data(data20)

# computation of a dissimilarity matrix using the iof similarity measure
diss.matrix <- iof(data20)

# creating an object with results of hierarchical clustering
hca.object <- nomprox(diss = diss.matrix, data = data20, method = "complete",
 clu.high = 5, eval = TRUE, prox = FALSE)

# quick clustering summary
summary(hca.object)

# quick cluster quality evaluation
print(hca.object)

# visualization of the evaluation criteria
eval.plot(hca.object)

# a dendrogram can be displayed if the object contains the prox component
hca.object <- nomprox(diss = diss.matrix, data = data20, method = "complete",
 clu.high = 5, eval = TRUE, prox = TRUE)

# a quick dendrogram
plot(hca.object)

# a dendrogram with three designated clusters
dend.plot(hca.object, clusters = 3)
```

---

of *Occurence Frequency (OF) Measure*

Description

The function calculates a dissimilarity matrix based on the OF similarity measure.

Usage

```
of(data, var.weights = NULL)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in columns.
var.weights	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The OF (Occurrence Frequency) measure was originally constructed for the text mining tasks, see (Sparck-Jones, 1972), later, it was adjusted for categorical variables, see (Boriah et al., 2008) It assigns higher weight to mismatches on less frequent values and otherwise.

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

Spark-Jones K. (1972). A statistical interpretation of term specificity and its application in retrieval. In Journal of Documentation, 28(1), p. 11-21. Later: Journal of Documentation, 60(5) (2002), p. 493-502.

See Also

anderberg, burnaby, eskin, gambaryan, goodall1, goodall2, goodall3, goodall4, iof, lin, lin1, sm, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.of <- of(data20)

# dissimilarity matrix calculation with variable weights
weights.of <- of(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

sm *Simple Matching Coefficient (SM)*

Description

The function calculates a dissimilarity matrix based on the SM similarity measure.

Usage

```
sm(data, var.weights = NULL)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in columns.
var.weights	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The simple matching coefficient (Sokal, 1958) represents the simplest way of measuring similarity. It does not impose any weights. By a given variable, it assigns the value 1 in case of match and value 0 otherwise.

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

Sokal R., Michener C. (1958). A statistical method for evaluating systematic relationships. In: Science bulletin, 38(22), The University of Kansas.

See Also

anderberg, burnaby, eskin, gambaryan, goodall1, goodall2, goodall3, goodall4, iof, lin, lin1, of, smirnov, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.sm <- sm(data20)

# dissimilarity matrix calculation with variable weights
weights.sm <- sm(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

`smirnov` *Smirnov (SV) Measure*

Description

The function calculates a dissimilarity matrix based on the SV similarity measure.

Usage

```
smirnov(data)
```

Arguments

`data`	A data.frame or a matrix with cases in rows and variables in columns.

Details

The Smirnov similarity measure was presented in (Smirnov, 1968). The measure assigns high similarity to matches when the frequency of the matching value is low, and the other values occur frequently, see (Borian et al., 2008).

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Smirnov E.S. (1968). On exact methods in systematics. Systematic Zoology, 17(1), 1-13.

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

See Also

anderberg, burnaby, eskin, gambaryan, goodall1, goodall2, goodall3, goodall4, iof, lin, lin1, of, sm, ve, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.smirnov <- smirnov(data20)
```

---

ve *Variable Entropy (VE) Measure*

Description

The function calculates a dissimilarity matrix based on the VE similarity measure.

Usage

```
ve(data, var.weights = NULL)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in columns.
var.weights	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The Variable Entropy similarity measure was introduced in (Sulc and Rezankova, 2019). It treats the similarity between two categories based on the within-cluster variability expressed by the normalized entropy. The measure assigns higher weights to rare categories.

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Boriah S., Chandola V., Kumar V. (2008). Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, p. 243-254.

Sulc Z. and Rezankova H. (2019). Comparison of Similarity Measures for Categorical Data in Hierarchical Clustering. Journal of Classification. 2019, 35(1), p. 58-72. DOI: 10.1007/s00357-019-09317-5.

See Also

anderberg, burnaby, eskin, gambaryan, goodall1, goodall2, goodall3, goodall4, iof, lin, lin1, of, sm, smirnov, vm.

Examples

```
# sample data
data(data20)

# dissimilarity matrix calculation
prox.ve <- ve(data20)

# dissimilarity matrix calculation with variable weighting
prox.ve.2 <- ve(data20, var.weights = c(1, 0.8, 0.6, 0.4, 0.2))

# dissimilarity matrix calculation with variable weights
weights.ve <- ve(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

---

vm *Variable Mutability (VM) measure*

Description

The function calculates a dissimilarity matrix based on the VM similarity measure.

Usage

```
vm(data, var.weights = NULL)
```

Arguments

data	A data.frame or a matrix with cases in rows and variables in columns.
var.weights	A numeric vector setting weights to the used variables. One can choose the real numbers from zero to one.

Details

The Variable Mutability similarity measure was introduced in (Sulc and Rezankova, 2019). It treats the similarity between two categories based on the within-cluster variability expressed by the normalized mutability. The measure assigns higher weights to rarer categories.

Value

The function returns an object of the class "dist".

Author(s)

Zdenek Sulc.
Contact: <zdenek.sulc@vse.cz>

References

Sulc Z. and Rezankova H. (2019). Comparison of Similarity Measures for Categorical Data in Hierarchical Clustering. Journal of Classification. 2019, 35(1), p. 58-72. DOI: 10.1007/s00357-019-09317-5.

See Also

anderberg, burnaby, eskin, gambaryan, goodall1, goodall2, goodall3, goodall4, iof, lin, lin1, of, sm, smirnov, ve,

Examples

```
#sample data
data(data20)

# dissimilarity matrix calculation
prox.vm <- vm(data20)

# dissimilarity matrix calculation with variable weights
weights.vm <- vm(data20, var.weights = c(0.7, 1, 0.9, 0.5, 0))
```

Index