Package 'smacof'

April 9, 2010

Type Package

Title SMACOF for Multidimensional Scaling.

Version 1.0-1

Date 2010-04-08

Author Jan de Leeuw, Patrick Mair

Maintainer Jan de Leeuw <deleeuw@stat.ucla.edu>

Description This package provides the following approaches of multidimensional scaling (MDS) based on stress minimization by means of majorization (smacof): Simple smacof on symmetric dissimilarity matrices, smacof for rectangular matrices (unfolding models), smacof with constraints on the configuration, three-way smacof for individual differences (including constraints for idioscal, indscal, and identity), and spherical smacof (primal and dual algorithm). Each of these approaches is implemented in a metric and nonmetric manner including primary, secondary, and tertiary approaches for tie handling.

Imports graphics, stats

Depends R (>= 2.8.0), polynom, rgl, scatterplot3d

License GPL

Repository CRAN

Repository/R-Forge/Project psychor

Repository/R-Forge/Revision 112

Date/Publication 2010-04-09 17:30:51

R topics documented:

`smacof-package` *SMACOF for Multidimensional Scaling.*

Description

This package provides the following approaches of multidimensional scaling (MDS) based on stress minimization by means of majorization (smacof): Simple smacof on symmetric dissimilarity matrices, smacof for rectangular matrices (unfolding models), smacof with constraints on the configuration, three-way smacof for individual differences (including constraints for idioscal, indscal, and identity), and spherical smacof (primal and dual algorithm). Each of these approaches is implemented in a metric and nonmetric manner including primary, secondary, and tertiary approaches for tie handling.

Details

Package:	smacof
Type:	Package
Version:	1.0-0
Date:	2008-07-31
License:	GPL

The function for basic SMACOF on symmetric dissimilarity matrices is `smacofSym()`. For rectangular input matrices (unfolding model) `smacofRect()` is appropriate and by means of `smacofIndDiff()` individual difference models (three-way MDS) can be computed.

Author(s)

Jan de Leeuw, Patrick Mair

Maintainer: Jan de Leeuw <deleeuw@stat.ucla.edu>

References

de Leeuw, J. \& Mair, P. (2009). Multidimensional scaling using majorization: The R package smacof. Journal of Statistical Software, 31(3), 1-30, http://www.jstatsoft.org/v31/i03/

See Also

smacofSym, smacofRect, smacof, smacofIndDiff, smacofSphere.primal, smacofSphere.dual

Examples

```
data(trading)
res <- smacofSym(trading)
res
```

---

bread *Breakfast preferences*

Description

The data set is described in Bro (1998). The raw data consist of ratings of 10 breads on 11 different attributes carried out by 8 raters. Note that the bread samples are pairwise replications: Each of the 5 different breads, which have a different salt content, was presented twice for rating.

Usage

```
data(bread)
```

Format

A list of length 8 with elements of class "dist". The attributes are bread odor, yeast odor, off-flavor, color, moisture, dough, salt taste, sweet taste, yeast taste, other taste, and total taste.

References

Bro, R. (1998). Multi-way Analysis in the Food Industry: Models, Algorithms, and Applications. Ph.D. thesis, University of Amsterdam (NL) \& Royal Veterinary and Agricultural University (DK).

Examples

```
data(bread)
```

`breakfast` *Breakfast preferences*

Description

42 individuals were asked to order 15 breakfast items due to their preference.

Usage

```
data(breakfast)
```

Format

Data frame with students in the rows and breakfast items in the columns.

toast: toast pop-up

butoast: buttered toast

engmuff: English muffin and margarine

jdonut: jelly donut

cintoast: cinnamon toast

bluemuff: blueberry muffin and margarine

hrolls: hard rolls and butter

toastmarm: toast and marmalade

butoastj: buttered toast and jelly

toastmarg: toast and margarine

cinbun: cinnamon bun

danpastry: Danish pastry

gdonut: glazed donut

cofcake: coffee cake

cornmuff: corn muffin and butter

References

Green, P. E. \& Rao, V. (1972). Applied multidimensional scaling. Hinsdale, IL: Dryden.

Examples

```
data(breakfast)
```

ekman *Ekman data set*

Description

Ekman dissimilarities

Usage

```
data(ekman)
```

Format

Object of class `dist`

Details

Ekman presents similarities for 14 colors which are based on a rating by 31 subjects where each pair of colors was rated on a 5-point scale (0 = no similarity up to 4 = identical). After averaging, the similarities were divided by 4 such that they are within the unit interval. Similarities of colors with wavelengths from 434 to 674 nm.

References

Ekman, G. (1954). Dimensions of color vision. Journal of Psychology, 38, 467-474.

Examples

```
data(ekman)
## maybe str(ekman) ; plot(ekman) ...
```

EW_ger *Work values*

Description

Intercorrelations of 13 working values for former West (first list element) and East Germany.

Usage

```
data(EW_ger)
```

Format

Object of class `dist`

Details

Note that the labels are given in German. For smacof, the data must be converted into a dissimilarity matrix by applying the `sim2diss()` function to each list element.

References

ALLBUS 1991, German General Social Survey.

Borg, I., Groenen, P. J. F., & Mair, P. (2010). Multidimensionale Skalierung. Muenchen: Hampp Verlag.

Examples

```
data(EW_ger)
```

---

kinshipdelta *Kinship Terms*

Description

Percentages of how often 15 kinship terms were not grouped together by college students including three external scales.

Usage

```
data(kinshipdelta)

data(kinshipscales)
```

Format

Dissimilarity matrix of 15 kinship terms and data frame with the following external scales:

Gender (1 = male, 2 = female, 9 = missing)

Generation (-2 = two back, -1 = one back, 0 = same generation, 1 = one ahead, 2 = two ahead)

Degree (1 = first, 2 = second, 3 = third, 4 = fourth)

References

Rosenberg, S. & Kim, M. P. (1975). The method of sorting as a data gathering procedure in multivariate research. Multivariate Behavioral Research, 10, 489-502.

Examples

```
data(kinshipdelta)
data(kinshipscales)
```

partypref *Party preferences*

Description

Artificial dataset containing the judges in the rows and the parties in the columns.

Usage

```
data(partypref)
```

Format

Matrix of party preferences.

References

Borg, I., Groenen, P. J. F., & Mair, P. (2010). Multidimensionale Skalierung. Muenchen: Hampp Verlag.

Examples

```
data(partypref)
```

perception *Rectangle Perception Data*

Description

42 subjects are assigned to two groups of 21 persons. 120 stimulus pairs of rectangles are presented. One group judges on a scale from 0 to 9 the perception in terms of width-height (WH), the other group in terms of size-shape (SS).

Usage

```
data(perception)
```

Format

List of subject dissimilarities for WH and SS group.

References

Borg, I. & Leutner, D. (1983). Dimensional models for the perception of rectangles. Perception and Psychophysics, 34, 257-269.

Examples

```
data(perception)
```

`plot.smacof` *2D SMACOF plots*

Description

These methods provide various 2D plots for SMACOF models.

Usage

```
## S3 method for class 'smacof':
plot(x, plot.type = "confplot", plot.dim = c(1,2), sphere = TRUE,
main, xlab, ylab, xlim, ylim, ...)

## S3 method for class 'smacofR':
plot(x, plot.type = "confplot", joint = FALSE, plot.dim = c(1,2),
main, xlab, ylab, xlim, ylim, ...)

## S3 method for class 'smacofID':
plot(x, plot.type = "confplot", plot.dim = c(1,2), main, xlab, ylab, xlim, ylim, ..
```

Arguments

`x`	Object of class `"smacof"`, `"smacofR"`, and `"smacofID"` (see details)
`plot.type`	String indicating which type of plot to be produced: `"confplot"`, `"resplot"` `"Shepard"`, `"stressplot"` (see details)
`plot.dim`	Vector with dimensions to be plotted.
`main`	Plot title.
`xlab`	Label of x-axis.
`ylab`	Label of y-axis.
`xlim`	Scale x-axis.
`ylim`	Scale y-axis.
`sphere`	In case of spherical smacof, whether sphere should be plotted or not.
`joint`	If `TRUE`, the configurations are plotted jointly in rectangular smacof.
`...`	Further plot arguments passed: see `plot` in package `scatterplot3d` for detailed information.

Details

smacofSym() creates object of class "smacof", whereas smacofRect() produces "smacofR" and smacofIndDiff() generates "smacofID".

Plot description:

- Configuration plot (plot.type = "confplot"): Plots the MDS configurations.

- Residual plot (plot.type = "resplot"): Plots the configuration distances against the corresponding residuals.

- Shepard diagram (plot.type = "Shepard"): Diagram with the observed against the fitted distances including isotonic regression line.

- Stress decomposition plot (plot.type = "stressplot"): Plots the stress contribution in of each observation.

For smacofIndDiff() the residual plot, Shepard diagram, and stress plot are based on the sum of the residuals across individuals/ways. The configuration plot represents the group stimulus space (i.e., joint configurations).

See Also

plot3d.smacof

Examples

```
## 2D plots for spherical SMACOF
data(trading)
res <- smacofSym(trading)
plot(res, plot.type = "confplot")
plot(res, plot.type = "Shepard")
plot(res, plot.type = "stressplot")

## Joint configuration plot and row/column stressplots for rectangular SMACOF
data(breakfast)
res <- smacofRect(breakfast)
plot(res, plot.type = "confplot", joint = TRUE)
plot(res, plot.type = "stressplot")
```

---

plot3d.smacof *3D SMACOF plots*

Description

These methods produce static and dynamic 3D configuration plots for SMACOF models.

Usage

```
## S3 method for class 'smacof':
plot3d(x, plot.dim = c(1,2,3), sphere = FALSE, xlab, ylab, zlab,
col, main, bgpng = NULL, ax.grid = TRUE, sphere.rgl = FALSE,...)

## S3 method for class 'smacofR':
plot3d(x, plot.dim = c(1,2,3), joint = FALSE, xlab, ylab, zlab,
col, main, bgpng = NULL, ax.grid = TRUE, sphere.rgl = FALSE,...)

## S3 method for class 'smacofID':
plot3d(x, plot.dim = c(1,2,3), xlab, ylab, zlab,
col, main, bgpng = NULL, ax.grid = TRUE, sphere.rgl = FALSE,...)

## S3 method for class 'smacof':
plot3dstatic(x, plot.dim = c(1,2,3), main, xlab, ylab, zlab, col, ...)

## S3 method for class 'smacofR':
plot3dstatic(x, plot.dim = c(1,2,3), main, xlab, ylab, zlab, col, joint = FALSE, ..

## S3 method for class 'smacofID':
plot3dstatic(x, plot.dim = c(1,2,3), main, xlab, ylab, zlab, col, ...)
```

Arguments

x	Object of class "smacof", "smacofR", and "smacofID" (see details)
plot.dim	Vector of length 3 with dimensions to be plotted.
sphere	Spherical SMACOF: Whether sphere should be plotted or not.
joint	Rectangular SMACOF: If TRUE, the configurations are plotted jointly.
main	Plot title.
xlab	Label of x-axis.
ylab	Label of y-axis.
zlab	Label of z-axis.
col	Color of the text labels.
bgpng	Background image from rgl library; NULL for white background
ax.grid	If TRUE, axes grid is plotted.
sphere.rgl	If TRUE, rgl sphere (background) is plotted.
...	Further plot arguments passed: see plot in package scatterplot3d for detailed information.

Details

smacofSym() creates object of class "smacof", whereas smacofRect() produces "smacofR" and smacofIndDiff() generates "smacofID".

For smacofIndDiff() the configuration plot represents the group stimulus space (i.e., joint configurations).

See Also

plot.smacof

Examples

```
## 3D plot for spherical SMACOF
data(trading)
res <- smacofSphere.dual(trading, ndim = 3)
plot3d(res, plot.type = "confplot", sphere = TRUE, sphere.rgl = FALSE)
plot3dstatic(res)

## Group stimulus space for rectangular SMACOF
data(breakfast)
res <- smacofRect(breakfast, ndim = 3)
plot3d(res, joint = TRUE)
```

---

rectangles *Rectangles*

Description

These data are based on an experiment by Borg and Leutner (1983). They constructed rectangles based on a specific grid structure. The size of the rectangles is given in the constraints matrix. In total, we have 16 rectangles. 21 subjects had to rate twice the similarity of each pair of rectangles on a scale from 0 (identical) to 9 (very different). The dist object contains the average ratings which are dissimilarities.

Usage

```
data(rectangles)
data(rect_constr)
```

Format

The rectangles are object of class `dist`, the constraints are given as matrix

Details

Note that the labels are given in German.

References

Borg, I., & Leutner, D. (1983). Dimensional models for the perception of rectangles. Perception and Psychophysics, 34, 257-269.

Borg, I., Groenen, P. J. F., & Mair, P. (2010). Multidimensionale Skalierung. Muenchen: Hampp Verlag.

Examples

```
data(rectangles)
data(rect_constr)
```

`residuals.smacof` *Residuals*

Description

Computes the residuals by subtracting the configuration dissimilarites from the observed dissimilarities.

Usage

```
## S3 method for class 'smacof':
residuals(object, ...)
## S3 method for class 'smacofR':
residuals(object, ...)
## S3 method for class 'smacofID':
residuals(object, ...)
```

Arguments

`object`	Object of class `smacof`, `smacofR` (rectangular), or `smacofID` (individual differences)
`...`	Ignored

Examples

```
data(kinshipdelta)
res <- smacofSym(kinshipdelta)
residuals(res)
```

`sim2diss` *Converts similarites to dissimilarities*

Description

Utility function for converting similarities into dissimilarities. Different methods are provided.

Usage

```
sim2diss(similmat, method = "corr", to.dist = TRUE)
```

Arguments

similmat	Similarity matrix (not necessarily symmetric, nor quadratic)
method	Various methods for converting similarities into dissimilarities: "corr", "neglog", "counts", or an integer value (see details)
to.dist	If TRUE, object of class dist is produced

Details

We provide the following methods for converting similarities S into dissimilarities D: "corr" is suited for correlation matrices and takes D = sqrt(1-S). "neglog" takes the negative logarithm in terms of -log(S). Having frequencies, "counts" is appropriate which does -log((S[i,j]*S[j,i)/(S[i,i]*S[j,j])). The user can specify also an integer value v. In this case sim2diss() computes v-S.

Value

Returns dissimiarities either as matrix or as dist object.

Examples

```
## Converting Ekman data (similarities) into dissimilarities by subtraction from 1
data(ekman)
ekman.diss <- sim2diss(ekman, method = 1)
res <- smacofSym(ekman.diss)
```

---

smacofConstraint *SMACOF Constraint*

Description

SMACOF with constraints on the configuration

Usage

```
smacofConstraint(delta, constraint = "linear", external, ndim = 2, weightmat = NULL
metric = TRUE, ties = "primary", verbose = FALSE, modulus = 1, itmax = 1000, eps =
```

Arguments

delta	Either a symmetric dissimilarity matrix or an object of class "dist"
constraint	Type of constraint: "linear", "unique", "diagonal", or a user-specified function (see details)
external	Data frame or matrix with external covariates, or list for simplex and circumplex (see details)
ndim	Number of dimensions

weightmat	Optional matrix with dissimilarity weights
startconf	Optional matrix with starting values for configurations (see details)
metric	If FALSE non-metric MDS is performed
ties	Tie specification for non-metric MDS only: "primary", "secondary", or "tertiary"
verbose	If TRUE, intermediate stress is printed out
modulus	Number of smacof iterations per monotone regression call
itmax	Maximum number of iterations
eps	Convergence criterion

Details

The user can specify a function with the following arguments: configuration matrix with starting values, matrix V (based on the weight matrix, see package vignette), external scale matrix. The function must return a matrix of resulting configurations.

A matrix with starting configurations can be specifiied. For `constraint = "linear"` it has to be of dimension (n x p). For `constraint = "unique"` it is typically of the form X = (Y|D) with D as (n x n) diagonal matrix and Y (n x p) configuration matrix. Hence X is of dimension (n x (n + p)). For `constraint = "diagonal"` it is typically of dimension (n x q) where q is the number of columns of the external scale matrix (and thus number of dimensions). If `constraint` is user-specified the specification of `startconf` is mandatory.

The argument `external` allows for the specification of a covariate data frame (or matrix) of dimension (n x q). Alternatively, for simplex fitting the user can specify a list of the following structure: `external = list("simplex", dim2)` with `dim2` denoting the dimension of the simplex with dim2 < n. For a circumplex the list has to be of the following form: `external = list("circumplex", dim2, k1, k2)` with 1 <= k1 <= k2 <= n (see also examples section). k1 and k2 denote the circumplex width.

Value

obsdiss	Observed dissimilarities, normalized
confdiss	Configuration dissimilarities
conf	Matrix of final configurations
stress.m	stress value for metric MDS
stress.nm	stress value for non-metric MDS (if computed)
ndim	Number of dimensions
model	Type of smacof model
niter	Number of iterations
nobj	Number of objects

Author(s)

Jan de Leeuw and Patrick Mair

References

de Leeuw, J. \& Mair, P. (2009). Multidimensional scaling using majorization: The R package smacof. Journal of Statistical Software, 31(3), 1-30, http://www.jstatsoft.org/v31/i03/

de Leeuw, J., \& Heiser, W. (1980). Multidimensional scaling with restrictions on the configurations. In P. R. Krishnaiah (eds.), Multivariate Analysis V, pp. 501-522. North-Holland.

See Also

smacofSym, smacofRect, smacofIndDiff, smacofSphere.primal, smacofSphere.dual

Examples

```
## SMACOF with linear configuration constraints
data(kinshipdelta)
data(kinshipscales)
res.lin1 <- smacofConstraint(kinshipdelta, constraint = "linear", external = kinshipscales)

## SMACOF with unique constraints
res.unique <- smacofConstraint(kinshipdelta, constraint = "unique", external = kinshipscales

## SMACOF with diagonal constraints
res.diag <- smacofConstraint(kinshipdelta, constraint = "diagonal", external = kinshipscales

## Fitting a simplex with q = 14 (i.e., n-1), diagonal constraints
res.simp <- smacofConstraint(kinshipdelta, constraint = "diagonal", external = list("simplex

## Fitting a circumplex with q = 10, k1 = 2, k2 = 8, diagonal constraints
res.circ <- smacofConstraint(kinshipdelta, constraint = "diagonal", external = list("circump
```

---

smacofIndDiff *smacof for Individual Differences*

Description

Performs smacof for individual differences also known as Three-Way smacof on a list of dissimilarity matrices. Various restrictions decompositions and restrictions on the weight matrix are provided.

Usage

```
smacofIndDiff(delta, ndim = 2, weightmat = NULL, init = NULL, metric = TRUE,
              ties = "primary", constraint = NULL, verbose = FALSE, modulus = 1,
              itmax = 1000, eps = 1e-6)
```

Arguments

delta	A list of dissimilarity matrices or a list objects of class dist
ndim	Number of dimensions
weightmat	Optional matrix with dissimilarity weights
init	Matrix with starting values for configurations (optional)
metric	If FALSE non-metric MDS is performed
ties	Tie specification for non-metric MDS
constraint	Either NULL, "idioscal", "diagonal", or "identity" (see details)
verbose	If TRUE, intermediate stress is printed out
modulus	Number of smacof iterations per monotone regression call
itmax	Maximum number of iterations
eps	Convergence criterion

Details

If the constraint is NULL, INDSCAL is performed with identity configuration weight matrices. An additional restriction can be imposed with "identity" which restricts the configurations across individuals/replications/ways to be equal. More unrestricted models are "diagonal" which restricts only the configuration weight matrices to be diagonal and "idioscal" is unrestricted.

Value

obsdiss	List of observed dissimilarities, normalized
confdiss	List of configuration dissimilarities
conf	List of matrices of final configurations
gspace	Joint configurations aka group stimulus space
cweights	Configuration weights
stress.m	stress value for metric MDS
stress.nm	stress value for non-metric MDS (if computed)
stress.co	Constrained stress value
ndim	Number of dimensions
model	Type of smacof model
niter	Number of iterations
nobj	Number of objects

Author(s)

Jan de Leeuw and Patrick Mair

References

de Leeuw, J. \& Mair, P. (2009). Multidimensional scaling using majorization: The R package smacof. Journal of Statistical Software, 31(3), 1-30, http://www.jstatsoft.org/v31/i03/

See Also

smacofConstraint, smacofSym, smacofRect, smacofSphere.primal, smacofSphere.dual

Examples

```
data(perception)
res <- smacofIndDiff(perception)
res
summary(res)

res.id <- smacofIndDiff(perception, constraint = "identity")
res.diag <- smacofIndDiff(perception, constraint = "diagonal")
res.idio <- smacofIndDiff(perception, constraint = "idioscal")
```

---

smacofRect *Rectangular smacof*

Description

Variant of smacof for rectangular matrices (typically ratings, preferences) which is also known as metric unfolding.

Usage

```
smacofRect(delta, ndim = 2, weightmat = NULL, init = NULL, verbose = FALSE,
           itmax = 1000, reg = 1e-6, eps = 1e-6)
```

Arguments

delta	Data frame or matrix of preferences, ratings, dissimilarities.
ndim	Number of dimensions
weightmat	Optional matrix with dissimilarity weights
init	Matrix with starting values for configurations (optional)
verbose	If TRUE, intermediate stress is printed out
itmax	Maximum number of iterations
reg	Regularization factor, prevents distances from beeing 0
eps	Convergence criterion

Details

Creates an object of class smacofR.

Value

obsdiss	Observed dissimilarities
confdiss	Configuration dissimilarities
conf.row	Matrix of final row configurations
conf.col	Matrix of final column configurations
stress	Final stress value
ndim	Number of dimensions
model	Type of smacof model
niter	Number of iterations
nind	Number of individuals (rows)
nobj	Number of objects (columns)

Author(s)

Jan de Leeuw and Patrick Mair

References

de Leeuw, J. \& Mair, P. (2009). Multidimensional scaling using majorization: The R package smacof. Journal of Statistical Software, 31(3), 1-30, http://www.jstatsoft.org/v31/i03/

See Also

smacofConstraint, smacofSym, smacofIndDiff, smacofSphere.primal, smacofSphere.dual

Examples

```
data(breakfast)
res <- smacofRect(breakfast)
res
summary(res)
```

---

smacofSphere.dual *Spherical SMACOF*

Description

Dual and primal approach for spherical SMACOF.

Usage

```
smacofSphere.dual <- function(delta, penalty = 100, ndim = 2, weightmat = NULL, ini
metric = TRUE, ties = "primary", verbose = FALSE, relax = 1, modulus = 1, itmax = 1

smacofSphere.primal (delta, ndim = 2, weightmat = NULL, init = NULL,
metric = TRUE, ties = "primary", verbose = FALSE, modulus = 1, itmax = 100, eps = 1
```

Arguments

delta	Either a symmetric dissimilarity matrix or an object of class dist
penalty	Penalty parameter for dual algorithm (larger 0)
ndim	Number of dimensions
weightmat	Optional matrix with dissimilarity weights
init	Matrix with starting values for configurations (optional)
metric	If FALSE non-metric MDS is performed
ties	Tie specification for non-metric MDS only
verbose	If TRUE, intermediate stress is printed out
relax	Relaxed smacof update
modulus	Number of smacof iterations per monotone regression call
itmax	Maximum number of iterations
eps	Convergence criterion

Value

obsdiss	Observed dissimilarities, normalized
obsdiss1	Dual SMACOF: Observed dissimilarities
obsdiss2	Dual SMACOF: Restriction matrix
confdiss	Configuration dissimilarities
conf	Matrix of final configurations
stress.m	stress value for metric MDS
stress.nm	stress value for non-metric MDS (if computed)
ndim	Number of dimensions
dummyvec	Dummy vector of restriction matrix
model	Type of smacof model
niter	Number of iterations
nobj	Number of objects

Author(s)

Jan de Leeuw and Patrick Mair

References

de Leeuw, J. \& Mair, P. (2008). Multidimensional scaling using majorization: The R package smacof.

See Also

smacofRect, smacofIndDiff, smacofSym,smacofConstraint

Examples

```
## spherical SMACOF solution for trading data
data(trading)
res <- smacofSphere.dual(trading)
res
summary(res)
```

`smacofSym` *Symmetric smacof*

Description

Basic smacof on symmetric dissimilarity matrix

Usage

```
smacofSym(delta, ndim = 2, weightmat = NULL, init = NULL, metric = TRUE, ties = "pr
```

Arguments

`delta`	Either a symmetric dissimilarity matrix or an object of class `"dist"`
`ndim`	Number of dimensions
`weightmat`	Optional matrix with dissimilarity weights
`init`	Matrix with starting values for configurations (optional)
`metric`	If `FALSE` non-metric MDS is performed
`ties`	Tie specification for non-metric MDS only: `"primary"`, `"secondary"`, or `"tertiary"`
`verbose`	If `TRUE`, intermediate stress is printed out
`relax`	Relaxed smacof update
`modulus`	Number of smacof iterations per monotone regression call
`itmax`	Maximum number of iterations
`eps`	Convergence criterion

Value

`obsdiss`	Observed dissimilarities, normalized
`confdiss`	Configuration dissimilarities
`conf`	Matrix of final configurations
`stress.m`	stress value for metric MDS

stress.nm	stress value for non-metric MDS (if computed)
ndim	Number of dimensions
model	Type of smacof model
niter	Number of iterations
nobj	Number of objects

Author(s)

Jan de Leeuw and Patrick Mair

References

de Leeuw, J. \& Mair, P. (2009). Multidimensional scaling using majorization: The R package smacof. Journal of Statistical Software, 31(3), 1-30, http://www.jstatsoft.org/v31/i03/

See Also

smacofConstraint, smacofRect, smacofIndDiff, smacofSphere.primal, smacofSphere.dual

Examples

```
## simple SMACOF solution for kinship data
data(kinshipdelta)
res <- smacofSym(kinshipdelta)
res
summary(res)

## 3D nonmetric SMACOF solution for trading data
data(trading)
res <- smacofSym(trading, ndim = 3, metric = FALSE, ties = "secondary")
res
```

---

stardist *Distances among stars in zodiac signs*

Description

A distance matrix for the 10 brightest stars in each of the 12 zodiac signs was computed. Astronomers measure the projected positions of objects on the celestial sphere in two angles, i.e. right ascension α and declination δ. For every zodiac sign, the projected distances on the sky between individual stars S_i and S_j have been calculated in decimal degrees by means of the Pythagorean theorem

$$d_{i,j} = \sqrt{(\alpha_i - \alpha_j)\,2 + (\delta_i - \delta_j)\,2}$$

assuming planar geometry. Since the zodiac signs are relatively small compared to the whole celestial sphere and the computation is only done for illustrative purposes, such a simplified assumption is appropriate.

Usage

```
data(stardist)
```

Format

A dist object contaning the star distances.

Note

Thanks to Paul Eigenthaler, Department of Astronomy, University of Vienna for calculating the distances.

Examples

```
data(stardist)
```

---

summary.smacofB *S3 methods for smacof*

Description

Print and summary methods for objects of class `smacofB`, `smacofR` (rectangular), and `smacofID` (individual differences).

Usage

```
## S3 method for class 'smacofB':
summary(object, ...)
## S3 method for class 'smacofB':
print(x, ...)
## S3 method for class 'smacofR':
summary(object, ...)
## S3 method for class 'smacofR':
print(x, ...)
## S3 method for class 'smacofID':
summary(object, ...)
## S3 method for class 'smacofID':
print(x, ...)
```

Arguments

`object`	Object of class `smacofB`, `smacofR`, `smacofID`
`x`	Object of class `smacofB`, `smacofR`, `smacofID`
`...`	Ignored

Examples

```
data(kinshipdelta)
res <- smacofSym(kinshipdelta)
res
summary(res)
```

`trading` *Trading data*

Description

Data from the New Geographical Digest (1986), analysed in Cox and Cox (2001), on which countries traded with other countries. For 20 countries the main trading partners are dichotomously scored (1 = trade performed, 0 = trade not performed). Based on this dichotomous matrix the dissimilarities are computed using the Jaccard coefficient.

Usage

```
data(trading)
```

Format

Object of class `"dist"` with dissimilarities of the following countries:

Arge: Argentina

Aust: Australia

Braz: Brazil

Cana: Canada

Chin: China

Czec: Czechoslovakia

Egyp: Egypt

E.Ge: East Germany

Fran: France

Hung: Hungary

Indi: India

Ital: Italy

Japa: Japan

N.Ze: New Zealand

Pola: Poland

Swed: Sweden

USA

USSR: Soviet Union

U.K.: United Kingdom

W.Ge: West Germany

References

Cox, T.F., Cox, M.A.A. (1991). Multidimensional scaling on a sphere. Communications in Statistics: Theory and Methods, 20, 2943-2953.

Examples

```
data(trading)
```

`wish_ger` *Wish dataset*

Description

Similarity ratings for 12 countries. There were no instructions concerning the characteristics on which these similarity judgements were to be made, this was information to discover rather than to impose.

Usage

```
data(wish_ger)
```

Format

Object of class `dist`

Details

Note that the country labels are given in German. For smacof, the data must be converted into a dissimilarity matrix (see examples).

References

Borg, I., Groenen, P. J. F., & Mair, P. (2010). Multidimensionale Skalierung. Muenchen: Hampp Verlag.

Wish, M. (1971). Individual differences in perceptions and preferences among nations. In C. W. King and D. Tigert (Eds.), Attitude research reaches new heights, pp. 312-328. Chicago: American Marketing Association.

Examples

```
data(wish_ger)
sim2diss(wish_ger)
```

Index