

Package ‘smacof’

August 1, 2009

Type Package

Title SMACOF for Multidimensional Scaling.

Version 1.0-0

Date 2008-07-27

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

Date/Publication 2009-08-01 13:32:44

R topics documented:

smacof-package	2
bread	3
breakfast	4
ekman	5
kinshipdelta	5
perception	6
plot.smacof	7

plot3d.smacof	8
residuals.smacof	10
sim2diss	11
smacofConstraint	12
smacofIndDiff	14
smacofRect	16
smacofSphere.dual	17
smacofSym	19
summary.smacofB	20
trading	21

Index 23

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) ...
```

`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)
```

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

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

```
residuals.smacof    Residuals
```

Description

Computes the residuals by subtracting the configuration dissimilarities 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	<i>Converts similarites to dissimilarities</i>
----------	--

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 dissimilarities 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 = 1e-06)
```

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 specified. For `constraint = "linear"` it has to be of dimension $(n \times p)$. For `constraint = "unique"` it is typically of the form $X = (YD)$ with D as $(n \times n)$ diagonal matrix and Y $(n \times p)$ configuration matrix. Hence X is of dimension $(n \times (n + p))$. For `constraint = "diagonal"` it is typically of dimension $(n \times q)$ 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 $(n \times q)$. 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

<code>obsdiss</code>	Observed dissimilarities, normalized
<code>confdiss</code>	Configuration dissimilarities
<code>conf</code>	Matrix of final configurations
<code>stress.m</code>	stress value for metric MDS
<code>stress.nm</code>	stress value for non-metric MDS (if computed)
<code>ndim</code>	Number of dimensions
<code>model</code>	Type of smacof model
<code>niter</code>	Number of iterations
<code>nobj</code>	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 <code>dist</code>
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 being 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 <code>dist</code>
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 <code>FALSE</code> non-metric MDS is performed
ties	Tie specification for non-metric MDS only
verbose	If <code>TRUE</code> , intermediate stress is printed out

<code>relax</code>	Relaxed smacof update
<code>modulus</code>	Number of smacof iterations per monotone regression call
<code>itmax</code>	Maximum number of iterations
<code>eps</code>	Convergence criterion

Value

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

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
```

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)
```

Index

*Topic **datasets**

bread, 3
breakfast, 3
ekman, 4
kinshipdelta, 5
perception, 6
trading, 20

*Topic **hplot**

plot.smacof, 6
plot3d.smacof, 8

*Topic **methods**

residuals.smacof, 9
summary.smacofB, 19

*Topic **models**

sim2diss, 10
smacofConstraint, 11
smacofIndDiff, 13
smacofRect, 15
smacofSphere.dual, 16
smacofSym, 18

*Topic **package**

smacof-package, 2

bread, 3

breakfast, 3

ekman, 4

kinshipdelta, 5

kinshipscales (*kinshipdelta*), 5

perception, 6

plot, 7, 9

plot.smacof, 6, 9

plot.smacofID (*plot.smacof*), 6

plot.smacofR (*plot.smacof*), 6

plot3d.smacof, 7, 8

plot3d.smacofID (*plot3d.smacof*), 8

plot3d.smacofR (*plot3d.smacof*), 8

plot3dstatic (*plot3d.smacof*), 8

print.smacofB (*summary.smacofB*),
19

print.smacofID (*summary.smacofB*),
19

print.smacofR (*summary.smacofB*),
19

residuals.smacof, 9

residuals.smacofID
(*residuals.smacof*), 9

residuals.smacofR
(*residuals.smacof*), 9

sim2diss, 10

smacof, 2

smacof (*smacof-package*), 2

smacof-package, 2

smacofConstraint, 11, 14, 16, 17, 19

smacofIndDiff, 2, 12, 13, 16, 17, 19

smacofRect, 2, 12, 14, 15, 17, 19

smacofSphere.dual, 2, 12, 14, 16, 16, 19

smacofSphere.primal, 2, 12, 14, 16, 19

smacofSphere.primal
(*smacofSphere.dual*), 16

smacofSym, 2, 12, 14, 16, 17, 18

summary.smacofB, 19

summary.smacofID
(*summary.smacofB*), 19

summary.smacofR
(*summary.smacofB*), 19

trading, 20