Package ‘fmds’

July 22, 2025

Type Package

Title Multidimensional Scaling Development Kit

Version 0.1.5

Author Frank M.T.A. Busing [aut, cre] (ORCID: <https://orcid.org/0000-0002-8062-538X>), Juan Claramunt Gonzalez [aut, com] (ORCID: <https://orcid.org/0009-0009-5387-6341>)

Maintainer Frank M.T.A. Busing <busing@fsw.leidenuniv.nl>

Description Multidimensional scaling (MDS) functions for various tasks that are beyond the beta stage and way past the alpha stage.
Currently, options are available for weights, restrictions, classical scaling or principal coordinate analysis, transformations (linear, power, Box-Cox, spline, ordinal), outlier mitigation (rdop), out-of-sample estimation (predict), negative dissimilarities, fast and faster executions with low memory footprints, penalized restrictions, cross-validation-based penalty selection, supplementary variable estimation (explain), additive constant estimation, mixed measurement level distance calculation, restricted classical scaling, etc. More will come in the future.
References. Busing (2024) ``A Simple Population Size Estimator for Local Minima Applied to Multidimensional Scaling''. Manuscript submitted for publication.
Busing (2025) ``Node Localization by Multidimensional Scaling with Iterative Majorization''. Manuscript submitted for publication.
Busing (2025) ``Faster Multidimensional Scaling''. Manuscript in preparation.
Barroso and Busing (2025) ``e-RDOP, Relative Density-Based Outlier Probabilities, Extended to Proximity Mapping''. Manuscript submitted for publication.

Depends R (>= 3.5.0)

Imports graphics, stats

Suggests ggplot2, microbenchmark

License BSD_2_clause + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

LazyData true

NeedsCompilation yes

Repository CRAN

Date/Publication 2025-06-03 07:00:01 UTC

Contents

`addconst` *Additive Constant Function for Classical Multidimensional Scaling*

Description

`addconst` returns the smallest additive constant which, added to the dissimilarities, makes the data true Euclidean distances. Note: NA's are not allowed.

Usage

```
addconst(delta, faster = FALSE, error.check = FALSE)
```

Arguments

`delta`	an n by n square symmetric hollow matrix containing (non-negative) dissimilarities.
`faster`	logical indicating faster but less precise procedure
`error.check`	extensive check validity input (data) parameters (default = FALSE).

Value

additive constant

Author(s)

Frank M.T.A. Busing

References

Cailliez (1983)

`asymmetry` *Asymmetry Function*

Description

`asymmetry` return statistics from asymmetry analyses.

Usage

```
asymmetry(delta, z, error.check = FALSE, echo = FALSE)
```

Arguments

`delta`	an n by n square asymmetric hollow matrix containing dissimilarities.
`z`	coordinates (n by p) after symmetric analysis
`error.check`	extensive check validity input parameters (default = FALSE).
`echo`	print (intermediate) results (default = FALSE).

Value

ssaf skew-symmetric sum-of-squares-accounted-for

vaf skew-symmetric variance-accounted-for

drifts drift vectors starting from coordinates in z (n by p matrix)

radii n vector with circle radii

Author(s)

Frank M.T.A. Busing

References

Gower (1968).

`colors` *Color data*

Description

Dissimilarities are one minus the original ratings. 31 subjects rated 91 combinations for similarity on a 5-point scale (0-4). The ratings were averaged and divided by 4 to obtain similarities between 0 and 1. The 14 colors had wavelengths 434, 445, 465, 472, 490, 504, 537, 555, 54, 600, 610, 628, 651, and 674.

Usage

```
colors
```

Format

14 x 14 dissimilarity matrix

- V1: dissimilarities for V1.
- V2: dissimilarities for V2.
- V3: dissimilarities for V3.
- V4: dissimilarities for V4.
- V5: dissimilarities for V5.
- V6: dissimilarities for V6.
- V7: dissimilarities for V7.
- V8: dissimilarities for V8.
- V9: dissimilarities for V9.
- V10: dissimilarities for V10.
- V11: dissimilarities for V11.
- V12: dissimilarities for V12.
- V13: dissimilarities for V13.
- V14: dissimilarities for V14.

References

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

`cv.fastmds` *Repeated Cross-Validation Penalized Restricted Multidimensional Scaling Function*

Description

`cv.fastmds` performs repeated cross-validation for a penalized restricted multidimensional scaling model.

Usage

```
cv.fastmds(
  delta,
  w = NULL,
  p = 2,
  q = NULL,
  b = NULL,
  lambda = 0,
  alpha = 1,
  grouped = FALSE,
  NFOLDS = 10,
  NREPEATS = 30,
  MAXITER = 1024,
  FCRIT = 1e-08,
  ZCRIT = 1e-06,
  error.check = FALSE,
  echo = FALSE
)
```

Arguments

`delta`	an n by n symmatric and hollow matrix containing dissimilarities.
`w`	an identical sized matrix containing nonnegative weights (all ones when omitted).
`p`	dimensionality (default = 2).
`q`	independent variables (n by h).
`b`	initial regression coefficients (h by p).
`lambda`	regularization penalty parameter(s) (default = 0.0: no penalty).
`alpha`	elastic-net parameter (default = 1.0: lasso only).
`grouped`	boolean for lasso penalty (default = FALSE: ordinary lasso).
`NFOLDS`	number of folds for the k-fold cross-validation.
`NREPEATS`	number of repeats for the repeated k-fold cross-validation.
`MAXITER`	maximum number of iterations (default = 1024).
`FCRIT`	relative convergence criterion function value (default = 0.00000001).

`ZCRIT`	absolute convergence criterion coordinates (default = 0.000001).
`error.check`	extensive check validity input parameters (default = FALSE).
`echo`	print intermediate algorithm results (default = FALSE).

Value

mserrors mean squared errors for different values of lambda.

stderrors standard errors for mean squared errors.

varnames labels of independent row variables.

coefficients list with final h by p matrices with regression coefficients (lambda order).

lambda sorted regularization penalty parameters.

alpha elastic-net parameter (default = 1.0: lasso only).

grouped boolean for lasso penalty (default = FALSE: ordinary lasso).

References

de Leeuw, J., and Heiser, W. J. (1980). Multidimensional scaling with restrictions on the configuration. In P.R. Krishnaiah (Ed.), Multivariate analysis (Vol. 5, pp. 501–522). Amsterdam, The Netherlands: North-Holland Publishing Company.

Heiser,W. J. (1987a). Joint ordination of species and sites: The unfolding technique. In P. Legendre and L. Legendre (Eds.), Developments in numerical ecology (pp. 189–221). Berlin, Heidelberg: Springer-Verlag.

Busing, F.M.T.A. (2010). Advances in multidimensional unfolding. Unpublished doctoral dissertation, Leiden University, Leiden, the Netherlands.

`dilation` *Dilation*

Description

`dilation` returns dilation or scale factor, such that || factor * source - target ||^2_weights is minimal.

Usage

```
dilation(source, weights = NULL, target = NULL, error.check = FALSE)
```

Arguments

`source`	n x m source matrix
`weights`	weights matrix, size n
`target`	if NULL: rotate source to principal axes; otherwise: rotate source to n x m target
`error.check`	extensive check validity input parameters (default = FALSE).

Value

dilation factor

Author(s)

Frank M.T.A. Busing

References

Gower (1968). Commandeur (1991)

---

explain.fmds *Explain method for all fmds objects*

Description

`explain` fits variables to coordinate configuration

Usage

```
## S3 method for class 'fmds'
explain(
  x,
  q,
  w = NULL,
  level = c("absolute", "ratio", "linear", "ordinal", "nominal"),
  MAXITER = 1024,
  FCRIT = 1e-08,
  error.check = FALSE,
  echo = FALSE,
  ...
)
```

Arguments

`x`	fmds object
`q`	variables
`w`	weights
`level`	transformation level
`MAXITER`	maximum number of iterations
`FCRIT`	convergence criterion
`error.check`	error chacking
`echo`	intermediate output
`...`	additional arguments to be passed.

Value

data data

weights weights

transformed.data transformed data

approach approach

degree zero

ninner zero

iknots NULL

anchor anchor

knotstype zero

coordinates NULL

coefficients coefficients

distances distances

last.iteration last.iteration

last.difference last.difference

n.stress n.stress

stress.1 stress.1

call call

`expressions` *Facial Expressions Data*

Description

Dissimilarities represent the correspondence in facial expressions. 30 students rated the dissimilarity between 13 female portraits (photographs) on a 9-point scale. The dissimilarities are the means of the re-scaled values obtained by the method of successive intervals.

Usage

```
expressions
```

Format

13 x 16 matrix. The first 13 x 13 matrix is a dissimilarity matrix

- V1: dissimilarities for V1.
- V2: dissimilarities for V2.
- V3: dissimilarities for V3.
- V4: dissimilarities for V4.
- V5: dissimilarities for V5.

- V6: dissimilarities for V6.
- V7: dissimilarities for V7.
- V8: dissimilarities for V8.
- V9: dissimilarities for V9.
- V10: dissimilarities for V10.
- V11: dissimilarities for V11.
- V12: dissimilarities for V12.
- V13: dissimilarities for V13.
- P1: Property 1.
- P2: Property 2.
- P3: Property 3.

References

Abelson and Sermat (1962). Multidimensional scaling of facial expressions. Journal of experimental psychology, 63(6), 546-554. Diederich, Messick, and Tucker (1957). A general least squares solution for successive intervals. Psychometrika, 22(2), 159-173. Woodworth (1938). Experimental psychology. New York, Holt. Engen, Levy, and Schlosberg (1958). The dimensional analysis of a new series of facial expressions. Journal of Experimental Psychology, 55(5), 454-458.

---

fastboxcoxmds *Box-Cox Multidimensional Scaling Function*

Description

`fastlinearmds` performs Box-Cox multidimensional scaling. The function follows algorithms given by de Leeuw and Heiser (1980). The data, dissimilarities and weights, are either symmetric or asymmetric. The dissimilarities are may contain negative values, the weights may not. The configuration is either unrestricted, (partly) fixed, or a linear combination of independent variables. The dissimilarities are optimally Box-Cox transformed.

Usage

```
fastboxcoxmds(
  delta,
  w = NULL,
  p = 2,
  z = NULL,
  r = NULL,
  b = NULL,
  MAXITER = 1024,
  FCRIT = 1e-08,
  ZCRIT = 1e-06,
  rotate = TRUE,
```

```
  faster = FALSE,
  error.check = FALSE,
  echo = FALSE
)
```

Arguments

`delta`	an n by n squares hollow matrix containing dissimilarities.
`w`	an identical sized matrix containing non-negative weights (all ones when omitted).
`p`	dimensionality (default = 2).
`z`	n by p matrix with initial coordinates.
`r`	restrictions on the configuration, either an n by p matrix with booleans indicating free (false) and fixed (true) coordinates or an n by h numerical matrix with h independent variables.
`b`	h by p matrix with initial regression coefficients.
`MAXITER`	maximum number of iterations (default = 1024).
`FCRIT`	relative convergence criterion function value (default = 0.00000001).
`ZCRIT`	absolute convergence criterion coordinates (default = 0.000001).
`rotate`	if TRUE: solution is rotated to principal axes.
`faster`	boolean specifying faster but less precise procedure.
`error.check`	extensive validity check input parameters (default = FALSE).
`echo`	print intermediate algorithm results (default = FALSE).

Value

data original n by n matrix with dissimilarities.

weights original n by n matrix with weights.

transformed.data final n by n matrix with transformed dissimilarities.

coordinates final n by p matrix with coordinates.

restriction either the fixed coordinates or the independent variables.

coefficients final h by p matrix with regression coefficients.

distances final n by n matrix with Euclidean distances between n rows of coordinates.

last.iteration final iteration number.

last.difference final function difference used for convergence testing.

n.stress final normalized stress value.

rotate if solution is rotated to principal axes.

faster if a faster procedure has been used.

Author(s)

Frank M.T.A. Busing

References

de Leeuw, J., and Heiser, W. J. (1980). Multidimensional scaling with restrictions on the configuration. In P.R. Krishnaiah (Ed.), Multivariate analysis (Vol. 5, pp. 501–522). Amsterdam, The Netherlands: North-Holland Publishing Company.

Heiser, W.J. (1991). A generalized majorization method for least squares multidimensional scaling of pseudo-distances that may be negative. Psychometrika, 55, pages 7-27.

Busing, F.M.T.A. (submitted). Node Localization by Multidimensional Scaling with Iterative Majorization: A Psychometric Perspective. Signal Processing, Elsevier.

---

fastermds *Stochastic Iterative Majorization Multidimensional Scaling Function*

Description

`fastermds` performs multidimensional scaling using a stochastic iterative majorization algorithm. The data are either dissimilarities (full or only lower triangular part) or multivariate data. The dissimilarities and the weights may not contain negative values. The configuration is either unrestricted or (partly) fixed. Local multidimensional scaling is performed when a boundary is provided. Interval multidimensional scaling is performed with a full dissimilarity matrix, using the lower triangular part for the lower bound and the upper triangular part for the upper bound.

Usage

```
fastermds(
  delta = NULL,
  lower = NULL,
  data = NULL,
  w = NULL,
  p = 2,
  z = NULL,
  fixed = NULL,
  linear = FALSE,
  boundary = NULL,
  interval = FALSE,
  NCYCLES = 32,
  MINRATE = 0.01,
  error.check = FALSE,
  test = 0
)
```

Arguments

`delta`	dissimilarity matrix, non-negative, square, and hollow.
`lower`	lower triangular part of dissimilarity matrix.
`data`	multivariate data matrix.

`w`	non-negative weights per dissimilarity for delta and lower, and per object for data
`p`	dimensionality (default = 2).
`z`	n by p matrix with initial coordinates.
`fixed`	n by p matrix with booleans indicating free (FALSE) or fixed (TRUE) coordinates.
`linear`	boolean indicating whether linear is used.
`boundary`	boundary value for local mds.
`interval`	interval measurements for interval mds, requires delta data format.
`NCYCLES`	number of cycles taken by the algorithm (default = 32).
`MINRATE`	criterion rate of convergence (default = 0.01).
`error.check`	extensive validity check input parameters (default = FALSE).
`test`	indicates which test is applied.

Details

One of the following three data formats need to be specified:

Value

n by p matrix with final coordinates.

Author(s)

Frank M.T.A. Busing

References

Agrafiotis, and others, and Busing

Examples

```
n <- 1000
m <- 10
delta <- as.matrix( dist( matrix( runif( n * m ), n, m ) ) )
p <- 2
zinit <- matrix( runif( n * p ), n, p )
# r <- fastermds( delta = delta, p = p, z = zinit, error.check = TRUE )
```

`fasterstress` *Faster Stress Function*

Description

`fasterstress` calculates stochastic normalized stress. Neither data nor distances based on z are optimally scaled.

Usage

```
fasterstress(data = NULL, z = NULL, nsamples = 100, samplesize = 30)
```

Arguments

`data`	an n by m multivariate data matrix.
`z`	n by p matrix with coordinates.
`nsamples`	number of samples
`samplesize`	sample size

Value

n.stress normalized stress, mean over samples and observations

se standard error of se, standard deviation over samples

Author(s)

Frank M.T.A. Busing

References

agrafiotis, and others, and busing

Examples

```
n <- 10000
m <- 10
data <- matrix( runif( n * m ), n, m )
p <- 2
zinit <- matrix( runif( n * p ), n, p )
# r <- fastermds( data = data, p = p, z = zinit )
# s <- fasterstress( data = data, z = r )
```

fastlinearmds *Linear Multidimensional Scaling Function*

Description

fastlinearmds performs linear multidimensional scaling. The function follows algorithms given by de Leeuw and Heiser (1980). The data, dissimilarities and weights, are either symmetric or asymmetric. The dissimilarities are may contain negative values, the weights may not. The configuration is either unrestricted, (partly) fixed, or a linear combination of independent variables. The dissimilarities are optimally linearly transformed.

Usage

```
fastlinearmds(
  delta,
  w = NULL,
  p = 2,
  z = NULL,
  r = NULL,
  b = NULL,
  anchor = TRUE,
  MAXITER = 1024,
  FCRIT = 1e-08,
  ZCRIT = 1e-06,
  rotate = TRUE,
  faster = FALSE,
  error.check = FALSE,
  echo = FALSE
)
```

Arguments

delta	an n by n squares hollow matrix containing dissimilarities.
w	an identical sized matrix containing non-negative weights (all ones when omitted).
p	dimensionality (default = 2).
z	n by p matrix with initial coordinates.
r	restrictions on the configuration, either an n by p matrix with booleans indicating free (false) and fixed (true) coordinates or an n by h numerical matrix with h independent variables.
b	h by p matrix with initial regression coefficients.
anchor	boolean indicating the use of an intercept
MAXITER	maximum number of iterations (default = 1024).
FCRIT	relative convergence criterion function value (default = 0.00000001).

`ZCRIT`	absolute convergence criterion coordinates (default = 0.000001).
`rotate`	if TRUE: solution is rotated to principal axes.
`faster`	logical indicating faster but less precise procedure
`error.check`	extensive validity check input parameters (default = FALSE).
`echo`	print intermediate algorithm results (default = FALSE).

Value

data original n by n matrix with dissimilarities.

weights original n by n matrix with weights.

transformed.data final n by n matrix with transformed dissimilarities.

anchor whether an intervept was used or not.

coordinates final n by p matrix with coordinates.

restriction either the fixed coordinates or the independent variables.

coefficients final h by p matrix with regression coefficients.

distances final n by n matrix with Euclidean distances between n rows of coordinates.

last.iteration final iteration number.

last.difference final function difference used for convergence testing.

n.stress final normalized stress value.

rotate if solution is rotated to principal axes.

faster if a faster procedure has been used.

Author(s)

Frank M.T.A. Busing

References

de Leeuw, J., and Heiser, W. J. (1980). Multidimensional scaling with restrictions on the configuration. In P.R. Krishnaiah (Ed.), Multivariate analysis (Vol. 5, pp. 501–522). Amsterdam, The Netherlands: North-Holland Publishing Company.

Heiser, W.J. (1991). A generalized majorization method for least squares multidimensional scaling of pseudo-distances that may be negative. Psychometrika, 55, pages 7-27.

Busing, F.M.T.A. (submitted). Node Localization by Multidimensional Scaling with Iterative Majorization: A Psychometric Perspective. Signal Processing, Elsevier.

Examples

```
data( "colors" )
delta <- as.matrix( colors^3 )
n <- nrow( delta )
w <- 1 - diag( n )
p <- 2
z <- matrix( runif( n * p ), n, p )
#r <- fastlinearmds( delta, w, p, z, echo = TRUE )
```

fastmds *Multidimensional Scaling Function*

Description

fastmds performs multidimensional scaling. The function follows algorithms given by de Leeuw and Heiser (1980). The data, dissimilarities and weights, are either symmetric or asymmetric. The dissimilarities are may contain negative values, the weights may not. The configuration is either unrestricted, (partly) fixed, or a linear combination of independent variables, penalized or not.

Usage

```
fastmds(
  delta,
  w = NULL,
  p = 2,
  z = NULL,
  r = NULL,
  b = NULL,
  anchor = 0,
  lambda = 0,
  alpha = 1,
  grouped = FALSE,
  MAXITER = 1024,
  FCRIT = 1e-08,
  ZCRIT = 1e-06,
  rotate = TRUE,
  faster = FALSE,
  error.check = FALSE,
  echo = FALSE
)
```

Arguments

delta	an n by n squares hollow matrix containing dissimilarities.
w	an identical sized matrix containing non-negative weights (all ones when omitted).
p	dimensionality (default = 2).
z	n by p matrix with initial coordinates.
r	restrictions on the configuration, either an n by p matrix with booleans indicating free (false) and fixed (true) coordinates or an n by h numerical matrix with h independent variables.
b	h by p matrix with initial regression coefficients.
anchor	used as additive constant, but estimated for pcoa only when anchor == NA.
lambda	regularization penalty parameter (default = 0.0: no penalty).

`alpha`	elastic-net parameter (default = 1.0: lasso only).
`grouped`	boolean for grouped lasso penalty (default = FALSE: ordinary lasso).
`MAXITER`	maximum number of iterations (default = 1024).
`FCRIT`	relative convergence criterion function value (default = 0.00000001).
`ZCRIT`	absolute convergence criterion coordinates (default = 0.000001).
`rotate`	if TRUE: solution is rotated to principal axes.
`faster`	logical indicating faster but less precise procedure
`error.check`	extensive validity check input parameters (default = FALSE).
`echo`	print intermediate algorithm results (default = FALSE).

Value

data original n by n matrix with dissimilarities.

weights original n by n matrix with weights.

coordinates final n by p matrix with coordinates.

restriction either the fixed coordinates or the independent variables.

coefficients final h by p matrix with regression coefficients.

lambda (optimal) penalty parameter.

alpha elastic-net penalty parameter.

grouped common or grouped lasso penalty.

distances final n by n matrix with Euclidean distances between n rows of coordinates.

last.iteration final iteration number.

last.difference final function difference used for convergence testing.

n.stress final normalized stress value.

rotate if solution is rotated to principal axes.

Author(s)

Frank M.T.A. Busing

References

de Leeuw, J., and Heiser, W. J. (1980). Multidimensional scaling with restrictions on the configuration. In P.R. Krishnaiah (Ed.), Multivariate analysis (Vol. 5, pp. 501–522). Amsterdam, The Netherlands: North-Holland Publishing Company.

Heiser, W.J. (1991). A generalized majorization method for least squares multidimensional scaling of pseudo-distances that may be negative. Psychometrika, 55, pages 7-27.

Busing, F.M.T.A. (submitted). Node Localization by Multidimensional Scaling with Iterative Majorization: A Psychometric Perspective. Signal Processing, Elsevier.

Examples

```
data( "colors" )
delta <- as.matrix( ( colors )^3 )
n <- nrow( delta )
w <- 1 - diag( n )
p <- 2
zinit <- matrix( runif( n * p ), n, p )
#r <- fastmds( delta, w, p, z = zinit, echo = TRUE )
```

---

fastordinalmds *Ordinal Multidimensional Scaling Function*

Description

`fastordinalmds` performs ordinal multidimensional scaling. The function follows algorithms given by de Leeuw and Heiser (1980). The data, dissimilarities and weights, are either symmetric or asymmetric. The dissimilarities are may contain negative values, the weights may not. The configuration is either unrestricted, (partly) fixed, or a linear combination of independent variables. The dissimilarities are optimally monotone transformed.

Usage

```
fastordinalmds(
  delta,
  w = NULL,
  p = 2,
  z = NULL,
  r = NULL,
  b = NULL,
  approach = 1,
  MAXITER = 1024,
  FCRIT = 1e-08,
  ZCRIT = 1e-06,
  rotate = TRUE,
  faster = FALSE,
  error.check = FALSE,
  echo = FALSE
)
```

Arguments

`delta`	an n by n squares hollow matrix containing dissimilarities.
`w`	an identical sized matrix containing non-negative weights (all ones when omitted).
`p`	dimensionality (default = 2).

`z`	n by p matrix with initial coordinates.
`r`	restrictions on the configuration, either an n by p matrix with booleans indicating free (false) and fixed (true) coordinates or an n by h numerical matrix with h independent variables.
`b`	h by p matrix with initial regression coefficients.
`approach`	approach to ties: 1 = untie ties, 2 = keep ties tied.
`MAXITER`	maximum number of iterations (default = 1024).
`FCRIT`	relative convergence criterion function value (default = 0.00000001).
`ZCRIT`	absolute convergence criterion coordinates (default = 0.000001).
`rotate`	if TRUE: solution is rotated to principal axes
`faster`	logical indicating faster but less precise procedure.
`error.check`	extensive check validity input parameters (default = FALSE).
`echo`	print intermediate algorithm results (default = FALSE).

Value

data original n by n matrix with dissimilarities.

weights original n by n matrix with weights.

transformed.data final n by n matrix with transformed dissimilarities.

approach apporach to ties: 1 = untie ties, 2 = keep ties tied.

coordinates final n by p matrix with coordinates.

restriction either the fixed coordinates or the independent variables.

coefficients final h by p matrix with regression coefficients.

distances final n by n matrix with Euclidean distances between n rows of coordinates.

last.iteration final iteration number.

last.difference final function difference used for convergence testing.

n.stress final normalized stress value.

rotate if solution is rotated to principal axes.

faster if a faster procedure has been used.

Author(s)

Frank M.T.A. Busing

References

de Leeuw, J., and Heiser, W. J. (1980). Multidimensional scaling with restrictions on the configuration. In P.R. Krishnaiah (Ed.), Multivariate analysis (Vol. 5, pp. 501–522). Amsterdam, The Netherlands: North-Holland Publishing Company.

Heiser, W.J. (1991). A generalized majorization method for least squares multidimensional scaling of pseudo-distances that may be negative. Psychometrika, 55, pages 7-27.

Busing, F.M.T.A. (submitted). Node Localization by Multidimensional Scaling with Iterative Majorization: A Psychometric Perspective. Signal Processing, Elsevier.

fastpowermds *Power Multidimensional Scaling Function*

Description

fastpowermds performs power multidimensional scaling. The function follows algorithms given by de Leeuw and Heiser (1980). The data, dissimilarities and weights, are either symmetric or asymmetric. The dissimilarities are may contain negative values, the weights may not. The configuration is either unrestricted, (partly) fixed, or a linear combination of independent variables. The dissimilarities are optimally power transformed.

Usage

```
fastpowermds(
  delta,
  w = NULL,
  p = 2,
  z = NULL,
  r = NULL,
  b = NULL,
  MAXITER = 1024,
  FCRIT = 1e-08,
  ZCRIT = 1e-06,
  rotate = TRUE,
  faster = FALSE,
  error.check = FALSE,
  echo = FALSE
)
```

Arguments

delta	an n by n squares hollow matrix containing dissimilarities.
w	an identical sized matrix containing non-negative weights (all ones when omitted).
p	dimensionality (default = 2).
z	n by p matrix with initial coordinates.
r	restrictions on the configuration, either an n by p matrix with booleans indicating free (false) and fixed (true) coordinates or an n by h numerical matrix with h independent variables.
b	h by p matrix with initial regression coefficients.
MAXITER	maximum number of iterations (default = 1024).
FCRIT	relative convergence criterion function value (default = 0.00000001).
ZCRIT	absolute convergence criterion coordinates (default = 0.000001).
rotate	if TRUE: solution is rotated to principal axes.

`faster`	logical indicating faster but less precise procedure
`error.check`	extensive validity check input parameters (default = FALSE).
`echo`	print intermediate algorithm results (default = FALSE).

Value

data original n by n matrix with dissimilarities.

weights original n by n matrix with weights.

transformed.data final n by n matrix with transformed dissimilarities.

coordinates final n by p matrix with coordinates.

restriction either the fixed coordinates or the independent variables.

coefficients final h by p matrix with regression coefficients.

distances final n by n matrix with Euclidean distances between n rows of coordinates.

last.iteration final iteration number.

last.difference final function difference used for convergence testing.

n.stress final normalized stress value.

rotate if solution is rotated to principal axes.

faster if a faster procedure has been used.

Author(s)

Frank M.T.A. Busing

References

de Leeuw, J., and Heiser, W. J. (1980). Multidimensional scaling with restrictions on the configuration. In P.R. Krishnaiah (Ed.), Multivariate analysis (Vol. 5, pp. 501–522). Amsterdam, The Netherlands: North-Holland Publishing Company.

Heiser, W.J. (1991). A generalized majorization method for least squares multidimensional scaling of pseudo-distances that may be negative. Psychometrika, 55, pages 7-27.

Busing, F.M.T.A. (submitted). Node Localization by Multidimensional Scaling with Iterative Majorization: A Psychometric Perspective. Signal Processing, Elsevier.

`fastsplinemds` *Polynomial Multidimensional Scaling Function*

Description

`fastpolynomialmds` performs monotone polynomial multidimensional scaling. The function follows algorithms given by de Leeuw and Heiser (1980). The data, dissimilarities and weights, are either symmetric or asymmetric. The dissimilarities are may contain negative values, the weights may not. The configuration is either unrestricted, (partly) fixed, or a linear combination of independent variables. The dissimilarities are optimally monotone transformed following a polynomial function.

Usage

```
fastsplinemds(
  delta,
  w = NULL,
  p = 2,
  z = NULL,
  r = NULL,
  b = NULL,
  anchor = TRUE,
  ninner = 0,
  degree = 2,
  knotstype = c("none", "uniform", "percentile", "midpercentile"),
  iknots = NULL,
  MAXITER = 1024,
  FCRIT = 1e-08,
  ZCRIT = 1e-06,
  rotate = TRUE,
  faster = FALSE,
  error.check = FALSE,
  echo = FALSE
)
```

Arguments

`delta`	an n by n squares hollow matrix containing dissimilarities.
`w`	an identical sized matrix containing non-negative weights (all ones when omitted).
`p`	dimensionality (default = 2).
`z`	n by p matrix with initial coordinates.
`r`	restrictions on the configuration, either an n by p matrix with booleans indicating free (false) and fixed (true) coordinates or an n by h numerical matrix with h independent variables.
`b`	h by p matrix with initial regression coefficients.
`anchor`	boolean indicating the use of an intercept
`ninner`	number of interior knots.
`degree`	spline degree.
`knotstype`	type of knots, either a vector with knots or the type uniform, percentile, or midpercentile.
`iknots`	user-provided interior knots
`MAXITER`	maximum number of iterations (default = 1024).
`FCRIT`	relative convergence criterion function value (default = 0.00000001).
`ZCRIT`	absolute convergence criterion coordinates (default = 0.000001).
`rotate`	if TRUE: solution is rotated to principal axes.
`faster`	logical indicating faster but less precise procedure.

`error.check` extensive validity check input parameters (default = FALSE).

`echo` print intermediate algorithm results (default = FALSE).

Value

data original n by n matrix with dissimilarities.

weights original n by n matrix with weights.

transformed.data final n by n matrix with transformed dissimilarities.

anchor whether an intercept was used or not.

degree spline degree.

ninner number of interior knots.

knotstype type of procedure creating the interior knot sequence.

iknots interior knots sequence.

coordinates final n by p matrix with coordinates.

restriction either the fixed coordinates or the independent variables.

coefficients final h by p matrix with regression coefficients.

distances final n by n matrix with Euclidean distances between n rows of coordinates.

last.iteration final iteration number.

last.difference final function difference used for convergence testing.

n.stress final normalized stress value.

rotate if solution is rotated to principal axes.

faster if a faster procedure has been used.

Author(s)

Frank M.T.A. Busing

References

de Leeuw, J., and Heiser, W. J. (1980). Multidimensional scaling with restrictions on the configuration. In P.R. Krishnaiah (Ed.), Multivariate analysis (Vol. 5, pp. 501–522). Amsterdam, The Netherlands: North-Holland Publishing Company.

Heiser, W.J. (1991). A generalized majorization method for least squares multidimensional scaling of pseudo-distances that may be negative. Psychometrika, 55, pages 7-27.

Busing, F.M.T.A. (submitted). Node Localization by Multidimensional Scaling with Iterative Majorization: A Psychometric Perspective. Signal Processing, Elsevier.

faststress — *Fast Stress Function*

Description

`faststress` calculates value for normalized stress with different input parameters. Distances d are optimally scaled.

Usage

```
faststress(
  lower = NULL,
  delta = NULL,
  data = NULL,
  w = NULL,
  z = NULL,
  d = NULL
)
```

Arguments

`lower`	an n x (n - 1) / 2 vector containing lower triangular part of dissimilarity matrix.
`delta`	an n by n square hollow matrix containing dissimilarities.
`data`	an n by m multivariate data matrix.
`w`	an identical sized matrix containing non-negative weights (all ones when omitted).
`z`	n by p matrix with coordinates.
`d`	distances between the rows of z, an n by n square hollow matrix containing Euclidean distances.

Value

n.stress normalized stress value

Author(s)

Frank M.T.A. Busing

food *Food data*

Description

38 subjects places 45 food items in categories based on similarity. The dissimilarities are the proportions of combinations NOT placed in the same category.

Usage

```
food
```

Format

45 x 45 dissimilarity matrix

- V1: dissimilarities for V1.
- V2: dissimilarities for V2.
- V3: dissimilarities for V3.
- V4: dissimilarities for V4.
- V5: dissimilarities for V5.
- V6: dissimilarities for V6.
- V7: dissimilarities for V7.
- V8: dissimilarities for V8.
- V9: dissimilarities for V9.
- V10: dissimilarities for V10.
- V11: dissimilarities for V11.
- V12: dissimilarities for V12.
- V13: dissimilarities for V13.
- V14: dissimilarities for V14.
- V15: dissimilarities for V15.
- V16: dissimilarities for V16.
- V17: dissimilarities for V17.
- V18: dissimilarities for V18.
- V19: dissimilarities for V19.
- V20: dissimilarities for V20.
- V21: dissimilarities for V21.
- V22: dissimilarities for V22.
- V23: dissimilarities for V23.
- V24: dissimilarities for V24.

- V25: dissimilarities for V25.
- V26: dissimilarities for V26.
- V27: dissimilarities for V27.
- V28: dissimilarities for V28.
- V29: dissimilarities for V29.
- V30: dissimilarities for V30.
- V31: dissimilarities for V31.
- V32: dissimilarities for V32.
- V33: dissimilarities for V33.
- V34: dissimilarities for V34.
- V35: dissimilarities for V35.
- V36: dissimilarities for V36.
- V37: dissimilarities for V37.
- V38: dissimilarities for V38.
- V39: dissimilarities for V39.
- V40: dissimilarities for V40.
- V41: dissimilarities for V41.
- V42: dissimilarities for V42.
- V43: dissimilarities for V43.
- V44: dissimilarities for V44.
- V45: dissimilarities for V45.

References

Ross and Murphy (1999). Food for thought: Cross-classification and category organization in a complex real-world domain. Cognitive psychology, 38(4), 495-553. Brusco and Stahl (2000). Using Quadratic Assignment Methods to Generate Initial Permutations for Least-Squares Unidimensional Scaling of Symmetric Proximity Matrices. Journal of Classification, 17(2).

`fullmds` *Full Multidimensional Scaling Function*

Description

`fullmds` performs a complete multidimensional scaling analysis. The function follows algorithms given by de Leeuw and Heiser (1980). The data, dissimilarities and weights, are either symmetric or asymmetric. The dissimilarities are may contain negative values, the weights may not. The configuration is either unrestricted, (partly) fixed, or a linear combination of independent variables. The dissimilarities may be transformed by different functions, with related parameters.

Usage

```
fullmds(
  delta,
  w = NULL,
  p = 2,
  z = NULL,
  r = NULL,
  b = NULL,
  level = c("none", "linear", "power", "box-cox", "spline", "ordinal"),
  anchor = TRUE,
  ninner = 0,
  degree = 2,
  knotstype = c("none", "uniform", "percentile", "midpercentile"),
  iknots = NULL,
  approach = 1,
  lambda = 0,
  alpha = 1,
  grouped = FALSE,
  MAXITER = 1024,
  FCRIT = 1e-08,
  ZCRIT = 1e-06,
  rotate = TRUE,
  faster = FALSE,
  error.check = FALSE,
  echo = FALSE
)
```

Arguments

`delta`	an n by n squares hollow matrix containing dissimilarities.
`w`	an identical sized matrix containing non-negative weights (all ones when omitted).
`p`	dimensionality (default = 2).
`z`	n by p matrix with initial coordinates.
`r`	restrictions on the configuration, either an n by p matrix with booleans indicating free (false) and fixed (true) coordinates or an n by h numerical matrix with h independent variables.
`b`	h by p matrix with initial regression coefficients.
`level`	type of dissimilarity transformation
`anchor`	boolean indicating the use of an intercept
`ninner`	number of interior knots.
`degree`	spline degree.
`knotstype`	type of knots, either a vector with knots or the type uniform, percentile, or midpercentile.
`iknots`	user-provided interior knots

`approach`	approach to ties: 1 = untie ties, 2 = keep ties tied.
`lambda`	regularization penalty parameter (default = 0.0: no penalty).
`alpha`	elastic-net parameter (default = 1.0: lasso only).
`grouped`	boolean for grouped lasso penalty (default = FALSE: ordinary lasso).
`MAXITER`	maximum number of iterations (default = 1024).
`FCRIT`	relative convergence criterion function value (default = 0.00000001).
`ZCRIT`	absolute convergence criterion coordinates (default = 0.000001).
`rotate`	if TRUE: solution is rotated to principal axes.
`faster`	logical indicating faster but less precise procedure.
`error.check`	extensive validity check input parameters (default = FALSE).
`echo`	print intermediate algorithm results (default = FALSE).

Value

data original n by n matrix with dissimilarities.

weights original n by n matrix with weights.

transformed.data final n by n matrix with transformed dissimilarities.

anchor whether an intercept was used or not.

degree spline degree.

ninner number of interior knots.

knotstype type of procedure creating the interior knot sequence.

iknots interior knots sequence.

approach apporach to ties: 1 = untie ties, 2 = keep ties tied.

coordinates final n by p matrix with coordinates.

restriction either the fixed coordinates or the independent variables.

coefficients final h by p matrix with regression coefficients.

lambda (optimal) penalty parameter.

alpha elastic-net penalty parameter.

grouped common or grouped lasso penalty.

distances final n by n matrix with Euclidean distances between n rows of coordinates.

last.iteration final iteration number.

last.difference final function difference used for convergence testing.

n.stress final normalized stress value.

rotate if solution is rotated to principal axes.

faster if a faster procedure has been used.

Author(s)

Frank M.T.A. Busing

References

de Leeuw, J., and Heiser, W. J. (1980). Multidimensional scaling with restrictions on the configuration. In P.R. Krishnaiah (Ed.), Multivariate analysis (Vol. 5, pp. 501–522). Amsterdam, The Netherlands: North-Holland Publishing Company.

Heiser, W.J. (1991). A generalized majorization method for least squares multidimensional scaling of pseudo-distances that may be negative. Psychometrika, 55, pages 7-27.

Busing, F.M.T.A. (submitted). Node Localization by Multidimensional Scaling with Iterative Majorization: A Psychometric Perspective. Signal Processing, Elsevier.

---

kabah — *Kabah data*

Description

Data description

Usage

```
kabah
```

Format

17 x 17 dissimilarity matrix

---

mdist — *Mixed Measurement Level Euclidean Distances Function*

Description

`fastmixed` returns Euclidean distances for variables from mixed measurement levels.

Usage

```
mdist(
  data,
  level = rep("numeric", ncol(data)),
  scale = FALSE,
  error.check = FALSE
)
```

Arguments

`data`	an n (objects) by m (variables) numerical data matrix .
`level`	measurement level variables: 1:numerical, 2:ordinal, 3:nominal (default = 1).
`scale`	boolean specifying scaling of distances such that sum-of-squares are n times n.
`error.check`	extensive check validity input parameters (default = FALSE).

Value

'dist' object with Euclidean distances between objects.

Author(s)

Frank M.T.A. Busing

References

Busing (2025). A Consistent Distance Measure for Mixed Data: Bridging the Gap between Euclidean and Chi-Squared Distances. Manuscript in progress.

---

pcoa — *Classical Multidimensional Scaling Function*

Description

pcoa performs classical multidimensional scaling or principal coordinates analysis. The function uses an eigenvalue decomposition on a Gramm matrix. The data are supposed to be distances, but often dissimilarities will do fine. The data matrix contains nonnegative values, is square, symmetric, and hollow. NA's are not allowed. An additive constant may be provided, which is added to the dissimilarities. This constant might be obtained optimally with the function fastaddconst(). Error checking focuses on the data requirements.

Usage

```
pcoa(
  delta = NULL,
  lower = NULL,
  data = NULL,
  p = 2,
  k = NULL,
  ac = 0,
  q = NULL,
  faster = FALSE,
  error.check = FALSE
)
```

Arguments

`delta`	dissimilarity matrix, non-negative, square, and hollow.
`lower`	lower triangular part of dissimilarity matrix.
`data`	multivariate data matrix.
`p`	dimensionality (default = 2).
`k`	number of landmark points (default = NULL, i.e., no landmarks).

ac	additive constant (default = 0.0, i.e., no additive constant). An additive constant can be obtained with the function fastaddconst(d) or can be user specified.
q	matrix with h independent n-sized variables (nrow(q) >= p), specifying the linear restriction z = qb (coordinates = variables times coefficients)
faster	logical indicating faster but less precise procedure
error.check	extensive check validity input parameters (default = FALSE).

Value

either n by p coordinates matrix (if q == NULL) or h by p coefficients matrix b (if q != NULL), in which case z = qb

Author(s)

Frank M.T.A. Busing

References

Young and Householder (1938) Torgerson (1952, 1958) Gower (1966) Carroll, Green, and Carmone (1976) De Leeuw and Heiser (1982) Ter Braak (1992) Borg and Groenen (2005)

---

petersen — *Mark-Recapture Population Size Estimator*

Description

`petersen` returns the estimated population size based on two independent equally-sized samples

Usage

```
petersen(first, second)
```

Arguments

first	vector with first sample identifiers (local minima)
second	vector with second sample identifiers (local minima)

Value

population size estimate

Author(s)

Frank M.T.A. Busing

References

Busing (2025). A Simple Population Size Estimator for Local Minima Applied to Multidimensional Scaling.

plot.fmds *Visualisation of an fmds object*

Description

Plot function for a fmds object. The plot shows the result of fmds.

Usage

```
## S3 method for class 'fmds'
plot(
  x,
 type = c("configuration", "transformation", "fit", "residuals", "shepard", "stress",
    "biplot", "dendrogram", "threshold", "neighbors"),
  markers = NULL,
  labels = NULL,
  ...
)
```

Arguments

x	An fmds object .
type	type of plot (configuration by default)
markers	vector or matrix for pie markers
labels	vector with labels
...	additional arguments to pass

Value

none

predict.fmds *Predict method for all fmds objects*

Description

predict provides locations for additional objects. Function is under construction.

Usage

```
## S3 method for class 'fmds'
predict(
  object,
  delta,
  w = NULL,
  level = c("absolute", "ratio", "linear", "ordinal"),
  MAXITER = 1024,
  FCRIT = 1e-08,
  error.check = FALSE,
  echo = FALSE,
  ...
)
```

Arguments

`object`	object of class `fmds`.
`delta`	an n by n squares hollow matrix containing dissimilarities.
`w`	an identical sized matrix containing non-negative weights (all ones when omitted).
`level`	parameter indicating whether absolute, ratio, linear, or ordinal level to be used.
`MAXITER`	maximum number of iterations (default = 1024).
`FCRIT`	relative convergence criterion function value (default = 0.00000001).
`error.check`	extensive validity check input parameters (default = FALSE).
`echo`	print intermediate algorithm results (default = FALSE).
`...`	additional arguments to be passed.

Value

data original n by n matrix with dissimilarities.

weights original n by n matrix with weights.

transformed.data final n by n matrix with transformed dissimilarities.

approach apporach to ties: 1 = untie ties, 2 = keep ties tied.

degree spline degree.

ninner number of interior knots.

iknots interior knots sequence.

anchor whether an intervept was used or not.

knotstype type of procedure creating the interior knot sequence.

coordinates final n by p matrix with coordinates.

coefficients final h by p matrix with regression coefficients.

distances final n by n matrix with Euclidean distances between n rows of coordinates.

last.iteration final iteration number.

last.difference final function difference used for convergence testing.

n.stress final normalized stress value.

Author(s)

Frank M.T.A. Busing

---

print.fmds *Print method for all fmds objects*

Description

`print` produces output for fmds object.

Usage

```
## S3 method for class 'fmds'
print(x, ...)
```

Arguments

`x`	object of class `fmds`.
`...`	additional arguments to be passed.

Value

none

Author(s)

Frank M.T.A. Busing

---

rdop *Relative Density-based Outlier Probabilities Function*

Description

`rdop` returns the relative density-based outlier probabilities according to Barroso and Busing (2025).

Usage

```
rdop(data, k = 0, lambda = 3, extended = FALSE, alpha = 0.2, beta = 0.25)
```

Arguments

`data`	a (rectangular, multivariate, n by m) data matrix or a (n by n) distance matrix, in either case, the function continues with a full distance matrix
`k`	number of neighbors (default: sqrt(2n))
`lambda`	multiple of standard distance deviations to get probabilistic distances
`extended`	extended relative density-based probabilities
`alpha`	steepness parameter turning scores into weights
`beta`	halfway parameter turning scores into weights

Value

if (extended == FALSE): outlier scores; else: weights matrix

Author(s)

Frank M.T.A. Busing

References

Barroso and Busing (2025).

`rotation` *Rotation*

Description

`rotation` returns rotation matrix, such that || rotation * source - target ||^2_weights is minimal.

Usage

```
rotation(source, weights = NULL, target = NULL, error.check = FALSE)
```

Arguments

`source`	n x m source matrix
`weights`	diagonal of weights matrix, size n
`target`	if NULL: rotate source to principal axes; otherwise: rotate source to n x m target
`error.check`	extensive check validity input parameters (default = FALSE).

Value

rotation matrix

Author(s)

Frank M.T.A. Busing

References

Gower (1968). Commandeur (1991).

`summary.fmds` *Summary method for all fmds objects*

Description

`summary` produces an output summary for fmds objects.

Usage

```
## S3 method for class 'fmds'
summary(object, ...)
```

Arguments

`object`	object of class `fmds`.
`...`	additional arguments to be passed.

Value

none

`tortula` *Tortula data*

Description

Morphological data on 10 taxa of Tortula labeled C, CC, CCS, I, I, N, N, RF, RH, RS, R, R, V, V.

Usage

```
tortula
```

Format

Multivariate data set with 14 cases and 7 variables.

- hydroids: (nominal)
- leaves: (nominal)
- hairs: (ordinal)
- apex: (ordinal)
- length: (numerical)
- diameter: (numerical)
- papillae: (numerical)

References

Podani (1999). Extending Gower's general coefficient of similarity to ordinal characters. Taxon, 48(2), 331-340.

Index