Package ‘asymmetry’

September 9, 2017

Type Package
Title Visualizing Asymmetric Data
Version 1.2.3
Date 2017-09-08
Author Berrie Zielman
Maintainer Berrie Zielman <berrie.zielman@gmail.com>
Description Multidimensional scaling models and methods for the visualization for asymmetric data. A matrix is asymmetric if the number of rows equals the number of columns, and these rows and columns refer to the same set of objects. An example is a student migration table, where the rows correspond to the countries of origin of the students and the columns to the destination countries. This package provides the slide-vector model, a scaling model with unique dimensions and the asymscal model for asymmetric multidimensional scaling. Furthermore, a heat map for skew-symmetric data, and the decomposition of asymmetry are provided for the analysis of asymmetric tables.

License GPL (>= 2)
Imports rJava, gplots, stats, methods, smacof
Suggests knitr, rmarkdown, RColorBrewer
VignetteBuilder knitr
NeedsCompilation yes
SystemRequirements Java JDK 1.2 or higher (for JRI/REngine JDK 1.4 or higher), GNU make

Repository CRAN
Date/Publication 2017-09-09 18:47:47 UTC

R topics documented:

asymmetry-package .................................................. 2
asymscal ................................................................. 3
asymscalexample ....................................................... 5
decomposition ......................................................... 5
Englishtowns ........................................................... 5
Description

Multidimensional scaling models and methods for the visualization for asymmetric data. A matrix is asymmetric if the number of rows equals the number of columns, and these rows and columns refer to the same set of objects. An example is a student migration table, where the rows correspond to the countries of origin of the students and the columns to the destination countries. This package provides the slide-vector model, a scaling model with unique dimensions and the asymscal model for asymmetric multidimensional scaling. Furthermore, a heat map for skew-symmetric data, and the decomposition of asymmetry are provided for the analysis of asymmetric tables.

Author(s)

Berrie Zielman

Maintainer: Berrie Zielman <berrie.zielman@gmail.com>

References

Zielman, B., and Heiser, W. J. (1993), The analysis of asymmetry by a slide-vector, Psychometrika, 58, 101-114.
Description

This function fits the multidimensional scaling model known as the asymscal model. This model is an extension of the symmetric Euclidean distance model proposed by Young (1975). The model is fitted in a stress majorization framework called SMACOF, whereas Young fitted this model using a least squares algorithm. Asymmetry is modelled by differential weighting of the dimensions of a multidimensional scaling configuration. When a subject compares object \( i \) to \( j \) he or she may use different weights when comparing object \( j \) to \( i \). In addition to these weights, the locations of the objects are jointly estimated from the data.

\[
d_{ij}(X) = \sqrt{\sum_{s=1}^{p} v_{is}(x_{is} - x_{js})^2}
\]

Usage

asymscal(data, ndim = 2, start=NULL, itmax = 10000, eps = 1e-10)

Arguments

data A matrix with the same number of rows and columns
ndim The number of dimensions
start An optional configuration with starting values
itmax The maximum number of iterations
eps Convergence criterion for Stress

Details

This function exploits a connection between the INDSCAL model and the asymscal model. This method inherits the methods for plotting an printing from the smacofIndDiff in the smacof package. Basically, the asymscal takes two steps. First, this function sets up the appropriate dissimilarity and missing data structure for a three-way multidimensional scaling model, then a call to the method smacofIndDiff in the imported package smacof is made. After correcting for the normalization applied to the data by smacofIndDiff, the results can be displayed and plotted by the methods in the package smacof.

Value

delta Observed dissimilarities
obsdiss List of observed dissimilarities, normalized
gspace Joint configurations aka group stimulus space
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cweights</td>
<td>Configuration weights</td>
</tr>
<tr>
<td>stress</td>
<td>Stress-1 value</td>
</tr>
<tr>
<td>resmat</td>
<td>Matrix with squared residuals</td>
</tr>
<tr>
<td>rss</td>
<td>Residual sum-of-squares</td>
</tr>
<tr>
<td>spp</td>
<td>Stress per point</td>
</tr>
<tr>
<td>ndim</td>
<td>Number of dimensions</td>
</tr>
<tr>
<td>model</td>
<td>Type of smacof model</td>
</tr>
<tr>
<td>niter</td>
<td>Number of iterations</td>
</tr>
<tr>
<td>nobj</td>
<td>Number of objects</td>
</tr>
</tbody>
</table>

**Note**

The original algorithm for fitting the asymscal model fits squared distances. This function fits distances and not squared distances. The configuration matrix is normalized to one. Although my experience with fitting this model in the majorization framework is limited, it appears to converge slowly. I recommend keeping the convergence criteria in the example.

**Author(s)**

Berrie Zielman

**References**


**Examples**

```r
## Not run:
data("asymscalexample")t <- asymscal(asymscalexample, ndim = 2, itmax = 10000, eps = 1e-10)
t$cweightsround(t$cweights, 3)plot(t, plot.type = "confplot")plot(t, plot.type = "bubbleplot")plot(t, plot.type = "stressplot")
## End(Not run)
```
**asymscalexample**  

**Description**  

This is an artificial dataset. The data are distances from a two-dimensional model, and because of this construction the asymscal model fit this data exactly. In addition, two rows of this matrix have weights different from (1,1). The fifth subject has weights (1.35,.25), and the 15th subject has weights (1.65,.425).

**Usage**

data("asymscalexample")

**Format**

A matrix with 15 rows and 15 columns.

---

**decomposition**  

**Description**  

Deprecated

---

**Englishtowns**  

**Description**  

A datamatrix with 8 rows and 8 columns. The data are distances between eight English towns, this datamatrix is made asymmetric by adding linear skew-symmetric matrix. In this dataset, asymmetry is imposed by perturbing the data.

**Usage**

data("Englishtowns")

**References**


**Examples**

data(Englishtowns)
hmap

Heat map for skew-symmetric data

Description

This heatmap displays the values of a skew-symmetric matrix by colors. The option dominance orders the rows and columns of the matrix in such a way that the values in the upper triangle are positive and the values in the lower triangle are negative. The order is calculated from the row-sums of the signs obtained from the skew-symmetric matrix.

Usage

hmap(x, dominance = FALSE, ...)

Arguments

x
A skew-symmetric matrix, an asymmetric matrix, or an object of class decomposition
If an asymmetric matrix is given, the skew-symmetric part is computed.
dominance
If true the signs of the skew-symmetric matrix are shown in the heatmap, if set to false the values in this matrix are shown.
...
Further plot arguments: see heatmap.2 for detailed information.

Author(s)

Berrie Zielman

Examples

data(studentmigration)
hmap(studentmigration, dominance=TRUE, col=c("red","white","blue"))

mdsunique

MDS Model with Unique Dimensions

Description

This MDS model has both common and unique dimensions. The common dimensions are shared by all other objects, whereas unique dimension apply to one object. A unique dimension has a non zero value for only one object, the coordinates for the other objects are zero. There are as many unique dimensions as there are objects. An asymmetric version of this model has two sets of unique dimensions: one for the rows and one for the columns. The distance in this model is defined as:

\[
d_{ij}(X) = \sqrt{\sum_{s=1}^{p} (x_{is} - x_{js})^2 + r_i^2 + c_j^2}
\]
mdsunique

Usage

mdsunique(data, dim = 2, verbose = FALSE, itmax = 125, eps = 1e-12)

Arguments

data A data matrix
dim The number of dimensions
verbose If true, prints the iteration history to screen
itmax Maximum number of iterations
eps Convergence criterion for Stress

Value

ndim The number of dimensions
fulldim The number of dimensions of the full model
stress The raw stress for this model
confi Returns the configuration matrix of shared dimensions of this multidimensional scaling model
X Returns the configuration matrix of the full model consisting of shared and unique dimensions
niter The number of iterations for the algorithm to converge
nobs The number of observations in this model
resid A matrix with raw residuals

Author(s)

Berrie Zielman

Examples

## Not run:
data("studentmigration")
mm<-studentmigration
mm[mm==0]<-.5 # replace zeroes by a small number
mm <- -log(mm/sum(mm)) # convert similarities to dissimilarities
v<-mdsunique(mm,dim = 2,itmax = 2100,verbose=TRUE, eps = .000000001)
plot(v, yplus = .3)

## End(Not run)
Configuration Plot for mdsunique

Description

Method for a two-dimensional plot of the model. Available rownames are plotted as labels above the points.

Usage

```
## S3 method for class 'mdsunique'
plot(x, plot.dim = c(1, 2), yplus = 0, xlab, ylab, ...)
```

Arguments

- **x**: Object of class `mdsunique`
- **plot.dim**: A vector with dimensions to be plotted
- **yplus**: Parameter to adjust the vertical position of the label
- **xlab**: Label of x-axis.
- **ylab**: Label of y-axis.
- **...**: Further plot arguments: see `plot` for detailed information.

Examples

```r
## Not run:
data("studentmigration")
mm<-studentmigration
mm[mm==0]<-.5 # replace zeroes by a small number
mm <- -log(mm/sum(mm)) # convert similarities to dissimilarities
v<-mdsunique(mm, dim = 2, itmax = 2100, verbose=TRUE, eps = .000000001)
plot(v, yplus = .3)
## End(Not run)
```

Plotting method for the skew-symmetric part of an asymmetric matrix

Description

This plotting method provides a multidimensional representation of skew-symmetry based on the singular value decomposition (SVD). The properties of the SVD of a skew-symmetric matrix and a novel way of plotting skew-symmetry was given by Gower (1977). The singular vectors come in pairs with equal singular values. These diagrams are not interpreted by comparing distances between point, but by comparing areas formed by two points and the origin. These singular vectors span a plane, and the area of the triangle between two points and the origin represents skew-symmetry. The signs of the skew-symmetry is modelled by a direction in the plane.
plot.slidevector

Usage

## S3 method for class 'skewsymmetry'
plot(x, plot.plane = 1, yplus = 0, xlab, ylab, ...)

Arguments

x An object of class skewsymmetry
plot.plane Integer indicating which plane to plot
yplus Offset for the labels above the object points
xlab Label for the x-axis
ylab Label for the y-axis
... Further plot arguments

References


plot.slidevector  

Configuration Plot for the Slide-vector Model

Description

Method for a two-dimensional plot of the model. Available rownames are plotted as labels above the points. The slide-vector is shown as an arrow.

Usage

## S3 method for class 'slidevector'
plot(x, plot.dim = c(1,2), yplus = 0, xlab, ylab, ...)

Arguments

x Object of class slidevector
plot.dim A vector with dimensions to be plotted
yplus Parameter to adjust the vertical position of the label
xlab Label of x-axis.
ylab Label of y-axis.
... Further plot arguments: see plot for detailed information.
Examples

```
# 2D plot for the slide-vector model
dis <- matrix(c(1,2,3,4,5,6,2,8,9,3), nrow = 5, ncol = 2) # configuration
a <- rbind(dis,dis+1.5) # generate slide-vector
test <- as.matrix(dist(a))[1:5,6:10] # extract data
v <- slidevector(test, dim = 2, itmax = 250, eps = .001)
plot(v)
```

### print.mdsunique

**Print Method for the MDS Model with unique dimensions**

**Description**

Print method for objects of class mdsunique.

**Usage**

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

**Arguments**

- `x` Object of class mdsunique
- `...` Further arguments

### print.slidevector

**Print Method for the Slide-vector Model**

**Description**

Print method for objects of class slidevector.

**Usage**

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

**Arguments**

- `x` Object of class slidevector
- `...` Further arguments
Description

The Rapid Alert System for dangerous non-food products (RAPEX) notifies EU member states about risks of products to the health and safety of consumers. Risks for the consumer include choking, strangulation and fire, to name just a few. Examples of products in this database are powerbanks, clothing, toys, lighters, among others. Dozens of products in the EU are withdrawn from the market every month because they pose a risk to users health and safety. Market surveillance authorities in EU member states are expected to inform other countries about dangerous products, so that they are removed from the market in other countries. These data are maintained in an exchange system known as RAPEX. Countries can register unsafe products in the RAPEX database, this process is called notification. Other countries may then act on a notification made by one of the other countries. This table is derived from the RAPEX database. The entries in the table give the number of products removed from the row country, that is acted upon by the column country.

References

https://english.rekenkamer.nl/publications/reports/2017/01/19/products-sold-on-the-european-market-unraveling-the-system-of-ce-marking

Description

The decomposition of an asymmetric matrix into a symmetric matrix and a skew-symmetric matrix is an elementary result from mathematics that is the cornerstone of this package. The decomposition into a skew-symmetric and a symmetric component is written as: \( Q = S + A \), where \( Q \) is an asymmetric matrix, \( S \) is a symmetric matrix, and \( A \) is a skew-symmetric matrix. This decomposition provides a justification for separate analyses of \( S \) and \( A \). This decomposition is a useful tool for data analysis, and a second application is to the study of an asymmetric matrix of residuals, obtained after fitting a model.

Usage

skewsymmetry(x)

Arguments

\( x \quad \text{An asymmetric matrix} \)
Value

- **S**: The symmetric part of the matrix
- **A**: The skew-symmetric part of the matrix
- **linear**: The linear part of the skew-symmetric matrix
- **sv**: The singular vectors of the skew-symmetric matrix
- **nobj**: The number of objects

Author(s)

Berrie Zielman

See Also

- `plot.skewsymmetry`

Examples

```r
data("Englishtowns")
Q <- skewsymmetry(Englishtowns)
# the skew-symmetric part
Q%*%A
```

---

**slidevector**  
*The slide-vector model*

Description

The slide-vector model is a multidimensional scaling model for asymmetric data. Here, a distance model is fitted to the symmetric part of the data whereas the asymmetric part of the data is represented by projections of the coordinates onto the slide-vector. The slide-vector points in the direction of large asymmetries in the data. The interpretation of asymmetry in this model is aided by the use of projections of points onto the slide-vector. The distance from $i$ to $j$ is larger if the point $i$ has a higher projection onto the slide-vector than the distance from $j$ to $i$. If the line connecting two points is perpendicular to the slide-vector the difference between the two projections is zero. In this case the distance between the two points is symmetric. The algorithm for fitting this model is derived from the majorization approach to multidimensional scaling.

$$
d_{ij}(X) = \sqrt{\sum_{s=1}^{n} (x_{is} - x_{js} + z_s)^2}
$$

Usage

```r
slidevector(data, dim = 2, verbose = FALSE, itmax = 125, eps = 1e-12)
```
Arguments

- **data**: An asymmetric matrix
- **dim**: The number of dimensions for this model
- **verbose**: Print the history of iterations
- **itmax**: The maximum number of iterations
- **eps**: The convergence criterion for the algorithm

Details

The slide-vector model is a special case of the unfolding model. Therefore, the algorithm for fitting this model is a constrained unfolding model. The coordinates of the objects are calculated by minimizing a least squares loss function. This loss function is called stress in the multidimensional scaling literature. The stress is minimized by a version of the SMACOF algorithm. The main output are the configuration of points and the slide-vector.

Value

- **ndim**: The number of dimensions
- **stress**: The raw stress for this model
- **confi**: Returns the configuration matrix of this multidimensional scaling model
- **niter**: The number of iterations for the algorithm to converge
- **nobs**: The number of observations in this model
- **resid**: A matrix with raw residuals
- **slvec**: Coordinates of the slide-vector

Author(s)

Berrie Zielman

References

Zielman, B., and Heiser, W. J. (1993), The analysis of asymmetry by a slide-vector, Psychometrika, 58, 101-114.

See Also

- `plot.slidevector`

Examples

```r
## asymmetric distances between English towns
data(Englishtowns)
v <- slidevector(Englishtowns, dim = 2, itmax = 250, eps = .001)
plot(v)
```
Description

The table lists the home and destination country of 268,142 students in the academic year 2012-2013 participating in the Erasmus program. The 33 rows of this table refer to the home country whereas the 33 columns refer to the destination countries. The table gives the number of inbound and outbound students between every pair of countries, and the entries in the table are read as follows: 32 students from Bulgaria studied in The Netherlands, 18 students from the Netherlands studied in Bulgaria. Macedonia (MK) was excluded from the published table because only one student from Macedonia studied abroad and this country did not receive any students.

Usage

data(studentmigration)

Format

A matrix of 33 rows by 33 columns

Details

The Erasmus program is a student exchange program from the European Union. Three million students had taken part since the start of the program in 1987. To join the program a student has study at least three months or do an internship of at least two months in another country. The 2-letter codes shown below are supplied by the ISO (International Organization for Standardization). Country codes are given here: Countrycodes

Note

Macedonia has been removed from this table because only one student from this country participated in the program, and no students moved to Macedonia.

Source


Examples

data(studentmigration)
hmap(studentmigration)
Summary method of the decomposition

**Description**

Deprecated, use skewsymmetry

Summary method of the decomposition

**Description**

Prints a decomposition of the sum of squares of an asymmetric matrix. The first column gives the sum of squares, and the second column gives the percentages of the two components. This decomposition can be applied to data, but also to a matrix of residuals obtained from a fitted model.

**Usage**

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

**Arguments**

- `object` An object of class decomposition
- `...` Further parameters

**Author(s)**

Berrie Zielman

**Examples**

```r
data(Enlishtowns)
q <- skewsymmetry(Enlishtowns)
summary(q)
```
summary.slidevector  Summary method for the slide-vector model

Description
This function extracts the configuration matrix and the slide-vector from the object.

Usage
## S3 method for class 'slidevector'
summary(object, ...)

Arguments
- object: An object of class slidevector
- ...: Further arguments

Examples
```r
data(Englishtowns)
v <- slidevector(Englishtowns, dim = 2, itmax = 250, eps = .001)
summary(v)
```
Index

*Topic **Datasets**
  - asymscalexample, 5
  - Englishtowns, 5
  - productsafety, 11
  - studentmigration, 14

*Topic **MDS**
  - asymscal, 3
  - mdsunique, 6
  - slidevector, 12

*Topic **Methods**
  - hmap, 6
  - plot.skewsymmetry, 8
  - skewsymmetry, 11

*Topic **Models**
  - asymscal, 3
  - slidevector, 12

asymmetry (asymmetry-package), 2
asymmetry-package, 2
asymscal, 3
asymscalexample, 5
decomposition, 5
Englishtowns, 5
heatmap.2, 6
hmap, 6

mdsunique, 6
plot, 8, 9
plot.mdsunique, 8
plot.skewsymmetry, 8, 12
plot.slidevector, 9, 13
print.mdsunique, 10
print.slidevector, 10
productsafety, 11

skewsymmetry, 11
slidevector, 12

studentmigration, 14
summary.decomposition, 15
summary.skewsymmetry, 15
summary.slidevector, 16