Package 'ca'

October 10, 2018

Version 0.71

Date 2018-10-10

Title Simple, Multiple and Joint Correspondence Analysis

Author Michael Greenacre [aut],
Oleg Nenadic [aut, cre],
Michael Friendly [ctb]

Maintainer Oleg Nenadic <onenadi@uni-goettingen.de>

Depends R (>= 3.0.0)

Suggests rgl (>= 0.64-10), vcd

Description Computation and visualization of simple, multiple and joint correspondence analysis.

LazyLoad yes

LazyData yes

License GPL

URL http://www.carme-n.org/

Repository CRAN

Repository/R-Forge/Project ca0

Repository/R-Forge/Revision 39

Repository/R-Forge/DateTimeStamp 2018-10-10 17:12:30

Date/Publication 2018-10-10 18:10:02 UTC

NeedsCompilation no

R topics documented:

`author` *Author dataset*

Description

This data matrix contains the counts of the 26 letters of the alphabet (columns of matrix) for 12 different novels (rows of matrix). Each row contains letter counts in a sample of text from each work, excluding proper nouns.

Usage

```
data("author")
```

Format

Data frame containing the 12 x 26 matrix.

Source

Larsen, W.A. and McGill, R., unpublished data collected in 1973.

`ca` *Simple correspondence analysis*

Description

Computation of simple correspondence analysis.

Usage

```
ca(obj, ...)

## S3 method for class 'matrix'
ca(obj, nd = NA, suprow = NA, supcol = NA,
   subsetrow = NA, subsetcol = NA, ...)

## S3 method for class 'data.frame'
ca(obj, ...)

## S3 method for class 'table'
ca(obj, ...)

## S3 method for class 'xtabs'
ca(obj, ...)

## S3 method for class 'formula'
ca(formula, data, ...)
```

Arguments

obj,formula	The function is generic, accepting various forms of the principal argument for specifying a two-way frequency table. Currently accepted forms are matrices, data frames (coerced to frequency tables), objects of class "xtabs" or "table" and one-sided formulae of the form ~ F1 + F2, where F1 and F2 are factors.
nd	Number of dimensions to be included in the output; if NA the maximum possible dimensions are included.
suprow	Indices of supplementary rows.
supcol	Indices of supplementary columns.
subsetrow	Row indices of subset.
subsetcol	Column indices of subset.
data	A data frame against which to preferentially resolve variables in the formula
...	Other arguments passed to the ca.matrix method

Details

The function ca computes a simple correspondence analysis based on the singular value decomposition.
The options suprow and supcol allow supplementary (passive) rows and columns to be specified. Using the options subsetrow and/or subsetcol result in a subset CA being performed.

Value

sv	Singular values
nd	Dimenson of the solution

rownames	Row names
rowmass	Row masses
rowdist	Row chi-square distances to centroid
rowinertia	Row inertias
rowcoord	Row standard coordinates
rowsup	Indices of row supplementary points
colnames	Column names
colmass	Column masses
coldist	Column chi-square distances to centroid
colinertia	Column inertias
colcoord	Column standard coordinates
colsup	Indices of column supplementary points
N	The frequency table

References

Nenadic, O. and Greenacre, M. (2007). Correspondence analysis in R, with two- and three-dimensional graphics: The ca package. *Journal of Statistical Software*, **20 (3)**, http://www.jstatsoft.org/v20/i03/

Greenacre, M. (2007). *Correspondence Analysis in Practice*. Second Edition. London: Chapman & Hall / CRC. Blasius, J. and Greenacre, M. J. (1994), Computation of correspondence analysis, in *Correspondence Analysis in the Social Sciences*, pp. 53-75, London: Academic Press.

Greenacre, M.J. and Pardo, R. (2006), Subset correspondence analysis: visualizing relationships among a selected set of response categories from a questionnaire survey. *Sociological Methods and Research*, **35**, pp. 193-218.

See Also

svd, plot.ca, plot3d.ca, summary.ca, print.ca

Examples

```
data("author")
ca(author)
plot(ca(author))

# table method
haireye <- margin.table(HairEyeColor, 1:2)
haireye.ca <- ca(haireye)
haireye.ca
plot(haireye.ca)
# some plot options
plot(haireye.ca, lines=TRUE)
plot(haireye.ca, arrows=c(TRUE, FALSE))
```

caconv *Converting data types in CA and MCA*

Description

Conversion from and to a number of different data types commonly used in CA and MCA (frequency tables, response pattern matrices, indicator matrices and Burt matrices).

Usage

```
caconv(x, from = c("freq", "rpm", "ind", "Burt"), to = c("rpm", "ind", "Burt", "freq"),
              nlev = NA, vars = c(1,2), ...)
```

Arguments

x	A matrix (two-way frequency table, indicator matrix, or Burt matrix) or data frame (response pattern matrix).
from	The type of input data in x: a frequency table ("freq"), or a response pattern matrix ("rpm"), or an indicator matrix ("ind"), or a Burt matrix ("Burt").
to	The data type into which x should be converted.
nlev	A vector containing the number of levels for each categorical variable (for from="ind" or from="Burt"). If NA, nlev is computed from the data.
vars	A vector of length 2 specifying the index of the variables to use for converting to "freq" (i.e. to a regular two-way frequency table).
...	Further arguments (ignored).

Details

The function caconv converts between data types in CA and MCA. Note that a conversion from from="Burt" to to="ind" or to="rpm" is not supported.

Value

A matrix or data frame containing the converted data (with the type specified in to).

See Also

ca,mjca

cacoord *Extracting coordinates from ca and mjca objects.*

Description

Extracting standard and principal coordinates as well as various row and column scaling configurations for visual display from ca and mjca objects.

Usage

```
cacoord(obj,
                type = c("standard", "principal",
                       "symmetric", "rowprincipal", "colprincipal", "symbiplot",
                         "rowgab", "colgab", "rowgreen", "colgreen"),
                dim  = NA,
                rows = NA,
                cols = NA,
                ...)
```

Arguments

obj	A ca or mjca object returned by ca or mjca.
type	The type of coordinates to extract ("standard" or "principal"). The remaining options ("symmetric", ..., "colgreen") return the corresponding row/column coordinate configuration for the map scaling options described in plot.ca where the corresponding argument is map.
dim	The dimensions to return. If NA, all available dimensions are returned.
rows	Logical indicating whether to return the row coordinates (see below for details).
cols	Logical indicating whether to return the column coordinates (see below for details).
...	Further arguments (ignored).

Details

The function cacoord returns the standard or principal coordinates of a CA or MCA solution. Additionally, row and column scaling configurations for plotting methods can be computed (see plot.ca for details).
Note that by default row and column coordinates are computed (i.e. for (rows=NA&cols=NA) | (rows=TRUE&cols=TRUE)). Using rows=TRUE (and cols=NA or cols=FALSE) returns a matrix with the row coordinates, and for cols=TRUE (and cols=NA or cols=FALSE) a matrix with the column coordinates is returned.

Value

A list with the slots rows (row coordinates) and columns (column coordinates). When computing only row or only column coordinates, a matrix (with the corresponding row or column coordinates) is returned.

See Also

ca,mjca,plot.ca,plot.mjca

`iterate.mjca` *Updating a Burt matrix in Joint Correspondence Analysis*

Description

Updating a Burt matrix in Joint Correspondence Analysis based on iteratively weighted least squares.

Usage

```
iterate.mjca(B, lev.n, nd = 2, maxit = 50, epsilon = 0.0001)
```

Arguments

`B`	A Burt matrix.
`lev.n`	The number of levels for each factor from the original response pattern matrix.
`nd`	The required dimensionality of the solution.
`maxit`	The maximum number of iterations.
`epsilon`	A convergence criterion for the maximum absolute difference of updated values compared to the previous values. The iteration is completed when all differences are smaller than `epsilon`.

Details

The function `iterate.mjca` computes the updated Burt matrix. This function is called from the function `mjca` when the option `lambda="JCA"`, i.e. when a Joint Correspondence Analysis is performed.

Value

`B.star`	The updated Burt matrix
`crit`	Vector of length 2 containing the number of iterations and epsilon

See Also

mjca

mjca *Multiple and joint correspondence analysis*

Description

Computation of multiple and joint correspondence analysis.

Usage

```
mjca(obj, ...)

## S3 method for class 'data.frame'
mjca(obj, ...)
## S3 method for class 'table'
mjca(obj, ...)
## S3 method for class 'array'
mjca(obj, ...)

## Default S3 method:
mjca(obj, nd = 2, lambda = c("adjusted", "indicator", "Burt", "JCA"),
     supcol = NA, subsetcat = NA,
     ps = ":", maxit = 50, epsilon = 0.0001, reti = FALSE, ...)
```

Arguments

obj	A response pattern matrix (data frame containing factors), or a frequency table (a "table" object) or an integer array.
nd	Number of dimensions to be included in the output; if NA the maximum possible dimensions are included.
lambda	Gives the scaling method. Possible values include "indicator", "Burt", "adjusted" and "JCA". Using lambda = "JCA" results in a joint correspondence analysis using iterative adjusment of the Burt matrix in the solution space. See Details for descriptions of these options.
supcol	Indices of supplementary columns.
subsetcat	Indices of subset categories (previously subsetcol).
ps	Separator used for combining variable and category names.
maxit	The maximum number of iterations (Joint Correspondence Analysis).
epsilon	A convergence criterion (Joint Correspondence Analysis).
reti	Logical indicating whether the indicator matrix should be included in the output.
...	Arguments passed to mjca.default

Details

The function mjca computes a multiple or joint correspondence analysis based on the eigenvalue decomposition of the Burt matrix. The lambda option selects the scaling variant desired for reporting inertias.

- lambda="indicator" gives multiple correspondence analysis based on the correspondence analysis of the indicator matrix, with corresponding inertias (eigenvalues).
- lambda="Burt" gives the version of multiple correspondence analysis based on the correspondence analysis of the Burt matrix, the inertias of which are the squares of those for the indicator option.
- lambda="adjusted" is the default option, giving improved percentages of inertia based on fitting the off-diagonal submatrices of the Burt matrix by rescaling the multiple correspondence analysis solution. All these first three options give the same standard coordinates of the categories.
- lambda="JCA" gives a joint correspondence analysis, which uses an iterative algorithm that optimally fits the off-diagonal submatrices of the Burt matrix. The JCA solution does not have strictly nested dimensions, so the percentage of inertia explained is given for the whole solution of chosen dimensionality, not for each dimension, but this percentage is optimal.

Value

sv	Eigenvalues (lambda = "indicator") or singular values (lambda = "Burt", "adjusted" or "JCA")
lambda	Scaling method
inertia.e	Percentages of explained inertia
inertia.t	Total inertia
inertia.et	Total percentage of explained inertia with the nd-dimensional solution
levelnames	Names of the factor/level combinations, joined using ps
factors	A matrix containing the names of the factors and the names of the factor levels
levels.n	Number of levels in each factor
nd	User-specified dimensionality of the solution
nd.max	Maximum possible dimensionality of the solution
rownames	Row names
rowmass	Row masses
rowdist	Row chi-square distances to centroid
rowinertia	Row inertias
rowcoord	Row standard coordinates
rowpcoord	Row principal coordinates
rowctr	Row contributions
rowcor	Row squared correlations
colnames	Column names
colmass	Column masses

coldist	Column chi-square distances to centroid
colinertia	Column inertias
colcoord	Column standard coordinates
colpcoord	Column principal coordinates
colctr	column contributions
colcor	Column squared correlations
colsup	Indices of column supplementary points (of the Burt and Indicator matrix)
subsetcol	Indices of subset columns (subsetcat)
Burt	Burt matrix
Burt.upd	The updated Burt matrix (JCA only)
subinertia	Inertias of sub-matrices
JCA.iter	Vector of length two containing the number of iterations and the epsilon (JCA only)
indmat	Indicator matrix if reti was set to TRUE
call	Return of match.call

References

Nenadic, O. and Greenacre, M. (2007), Correspondence analysis in R, with two- and three-dimensional graphics: The ca package. *Journal of Statistical Software*, **20 (3)**, http://www.jstatsoft.org/v20/i03/

Nenadic, O. and Greenacre, M. (2007), Computation of Multiple Correspondence Analysis, with Code in R, in *Multiple Correspondence Analysis and Related Methods* (eds. M. Greenacre and J. Blasius), Boca Raton: Chapmann & Hall / CRC, pp. 523-551.

Greenacre, M.J. and Pardo, R. (2006), Subset correspondence analysis: visualizing relationships among a selected set of response categories from a questionnaire survey. *Sociological Methods and Research*, **35**, pp. 193-218.

See Also

eigen, plot.mjca, summary.mjca, print.mjca

Examples

```
data("wg93")
mjca(wg93[,1:4])

# table input
data(UCBAdmissions)
mjca(UCBAdmissions)
## Not run: plot(mjca(UCBAdmissions))

### Different approaches to multiple correspondence analysis:
# Multiple correspondence analysis based on the indicator matrix:
## Not run: mjca(wg93[,1:4], lambda = "indicator")
```

```
# Multiple correspondence analysis based on the Burt matrix:
## Not run: mjca(wg93[,1:4], lambda = "Burt")

# "Adjusted" multiple correspondence analysis (default setting):
## Not run: mjca(wg93[,1:4], lambda = "adjusted")

# Joint correspondence analysis:
## Not run: mjca(wg93[,1:4], lambda = "JCA")


### Subset analysis and supplementary variables:
# Subset analysis:
## Not run: mjca(wg93[,1:4], subsetcat = (1:20)[-seq(3,18,5)])

# Supplementary variables:
## Not run: mjca(wg93, supcol = 5:7)
```

`multilines` *Draw lines for groups distinguished by a factor*

Description

This is a convenience function for drawing a set of lines distinguished by the levels of a factor. It can be used to make more attractive plots than available via `plot.mjca`.

Usage

```
multilines(XY, group=NULL, which=1:nf, sort=1, type='l', col=palette(), lwd=1, ...)
```

Arguments

`XY`	A two-column data frame or matrix
`group`	A factor; a separate line is drawn for each level included in `which`
`which`	An integer vector used to select the factors for which lines are drawn. By default, all lines are drawn.
`sort`	Column of `XY` to sort upon before drawing the line for each group
`type`	Line type: `"l"` for line, `"b"` for line and points
`col`	A vector of colors to be used for the various lines, in the order of the levels in `group`; recycled as necessary.
`lwd`	A vector of line widths to be used for the various lines; recycled as necessary
`...`	Other graphic parameters passed to `lines`, e.g., `lty`

Value

none

Author(s)

Michael Friendly

See Also

lines

Examples

```
if (require(vcd)) {
  data(PreSex, package="vcd")
  presex.mca <- mjca(PreSex)
  res <- plot(presex.mca, labels=0, pch='.', cex.lab=1.2)
  coords <- data.frame(res$cols, presex.mca$factors)
  nlev <- rle(as.character(coords$factor))$lengths
  fact <- unique(as.character(coords$factor))

  cols <- c("blue", "red", "brown", "black")
  lwd <- c(2, 2, 2, 4)

  plot(Dim2 ~ Dim1, type='n', data=coords)
  points(coords[,1:2], pch=rep(16:19, nlev), col=rep(cols, nlev), cex=1.2)
  text(coords[,1:2], labels=coords$level, col=rep(cols, nlev), pos=3, cex=1.2, xpd=TRUE)

  multilines(coords[, c("Dim1", "Dim2")], group=coords$factor, col=cols, lwd=lwd)
}
```

---

pchlist *Listing the set of available symbols.*

Description

A plot of the available symbols for use with the option pch.

Usage

```
pchlist()
```

Details

This function generates a numbered list of the plotting symbols available for use in the functions plot.ca and plot3d.ca.

See Also

plot.ca, plot3d.ca

Examples

```
pchlist()
```

---

plot.ca *Plotting 2D maps in correspondence analysis*

Description

Graphical display of correspondence analysis results in two dimensions

Usage

```
## S3 method for class 'ca'
plot(x, dim = c(1,2), map = "symmetric", what = c("all", "all"),
               mass = c(FALSE, FALSE), contrib = c("none", "none"),
               col = c("blue", "red"),
               pch = c(16, 21, 17, 24),
               labels = c(2, 2),
               arrows = c(FALSE, FALSE),
               lines = c(FALSE, FALSE),
               lwd=1,
               xlab = "_auto_", ylab = "_auto_",
               col.lab = c("blue", "red"), ...)
```

Arguments

x	Simple correspondence analysis object returned by ca
dim	Numerical vector of length 2 indicating the dimensions to plot on horizontal and vertical axes respectively; default is first dimension horizontal and second dimension vertical.
map	Character string specifying the map type. Allowed options include "symmetric" (default) "rowprincipal" "colprincipal" "symbiplot" "rowgab" "colgab" "rowgreen" "colgreen"
what	Vector of two character strings specifying the contents of the plot. First entry sets the rows and the second entry the columns. Allowed values are "all" (all available points, default) "active" (only active points are displayed) "passive" (only supplementary points are displayed) "none" (no points are displayed) The status (active or supplementary) of rows and columns is set in ca using the options suprow and supcol.

mass	Vector of two logicals specifying if the mass should be represented by the area of the point symbols (first entry for rows, second one for columns)
contrib	Vector of two character strings specifying if contributions (relative or absolute) should be represented by different colour intensities. Available options are "none" (contributions are not indicated in the plot). "absolute" (absolute contributions are indicated by colour intensities). "relative" (relative contributions are indicated by colour intensities). If set to "absolute" or "relative", points with zero contribution are displayed in white. The higher the contribution of a point, the closer the corresponding colour to the one specified by the col option.
col	Vector of length 2 specifying the colours of row and column point symbols, by default blue for rows and red for columns. Colours can be entered in hexadecimal (e.g. "#FF0000"), rgb (e.g. rgb(1,0,0)) values or by R-name (e.g. "red").
pch	Vector of length 4 giving the type of points to be used for row active and supplementary, column active and supplementary points. See pchlist for a list of symbols.
labels	Vector of length two specifying if the plot should contain symbols only (0), labels only (1) or both symbols and labels (2). Setting labels to 2 results in the symbols being plotted at the coordinates and the labels with an offset.
arrows	Vector of two logicals specifying if the plot should contain points (FALSE, default) or arrows (TRUE). First value sets the rows and the second value sets the columns.
lines	Vector of two logicals specifying if the plot should join the points with lines (FALSE, default) or arrows (TRUE). First value sets the rows and the second value sets the columns.
lwd	Line width for arrows and lines
xlab, ylab	Labels for horizontal and vertical axes. The default, "_auto_" means that the function auto-generates a label of the form Dimension X (xx.xx %
col.lab	Vector of length 2 specifying the colours of row and column point labels
...	Further arguments passed to plot and points.

Details

The function plot.ca makes a two-dimensional map of the object created by ca with respect to two selected dimensions. By default the scaling option of the map is "symmetric", that is the so-called *symmetric map*. In this map both the row and column points are scaled to have inertias (weighted variances) equal to the principal inertia (eigenvalue or squared singular value) along the principal axes, that is both rows and columns are in pricipal coordinates. Other options are as follows:

- -"rowprincipal" or "colprincipal" - these are the so-called *asymmetric maps*, with either rows in principal coordinates and columns in standard coordinates, or vice versa (also known as row-metric-preserving or column-metric-preserving respectively). These maps are biplots;

- -"symbiplot" - this scales both rows and columns to have variances equal to the singular values (square roots of eigenvalues), which gives a symmetric biplot but does not preserve row or column metrics;

- -"rowgab" or "colgab" - these are asymmetric maps (see above) with rows (respectively, columns) in principal coordinates and columns (respectively, rows) in standard coordinates multiplied by the mass of the corresponding point. These are also biplots and were proposed by Gabriel & Odoroff (1990);
- -"rowgreen" or "colgreen" - these are similar to "rowgab" and "colgab" except that the points in standard coordinates are multiplied by the square root of the corresponding masses, giving reconstructions of the standardized residuals.

This function has options for sizing and shading the points. If the option mass is TRUE for a set of points, the size of the point symbol is proportional to the relative frequency (mass) of each point. If the option contrib is "absolute" or "relative" for a set of points, the colour intensity of the point symbol is proportional to the absolute contribution of the points to the planar display or, respectively, the quality of representation of the points in the display. To globally resize all the points (and text labels), use par("cex"=) before the plot.

Value

In addition to the side effect of producing the plot, the function invisibly returns the coordinates of the plotted points, a list of two components, with names rows and cols. These can be used to further annotate the plot using base R plotting functions.

References

Gabriel, K.R. and Odoroff, C. (1990). Biplots in biomedical research. *Statistics in Medicine*, **9**, pp. 469-485.
Greenacre, M.J. (1993) *Correspondence Analysis in Practice*. London: Academic Press.
Greenacre, M.J. (1993) Biplots in correspondence Analysis, *Journal of Applied Statistics*, **20**, pp. 251 - 269.

See Also

ca, summary.ca, print.ca, plot3d.ca, pchlist

Examples

```
data("smoke")

# A two-dimensional map with standard settings
plot(ca(smoke))

# Mass for rows and columns represented by the size of the point symbols
plot(ca(smoke), mass = c(TRUE, TRUE))

# Displaying the column profiles only with masses represented by size of point
# symbols and relative contributions by colour intensity.
# Since the arguments are recycled it is sufficient to give only one argument
# for mass and contrib.
data("author")
plot(ca(author), what = c("none", "all"), mass = TRUE, contrib = "relative")
```

plot.mjca *Plotting 2D maps in multiple and joint correspondence analysis*

Description

Graphical display of multiple and joint correspondence analysis results in two dimensions

Usage

```
## S3 method for class 'mjca'
plot(x, dim = c(1,2), map = "symmetric", centroids = FALSE,
     what = c("none", "all"), mass = c(FALSE, FALSE),
     contrib = c("none", "none"), col = c("#000000", "#FF0000"),
     pch = c(16, 1, 17, 24),
     labels = c(2, 2), collabels = c("both", "level", "factor"),
     arrows = c(FALSE, FALSE), xlab = "_auto_", ylab = "_auto_", ...)
```

Arguments

x	Multiple or joint correspondence analysis object returned by mjca
dim	Numerical vector of length 2 indicating the dimensions to plot on horizontal and vertical axes respectively; default is first dimension horizontal and second dimension vertical.
map	Character string specifying the map type. Allowed options include "symmetric" (default) "rowprincipal" "colprincipal" "symbiplot" "rowgab" "colgab" "rowgreen" "colgreen"
centroids	Logical indicating if column centroids should be added to the plot
what	Vector of two character strings specifying the contents of the plot. First entry sets the rows and the second entry the columns. Allowed values are "all" (all available points, default) "active" (only active points are displayed) "passive" (only supplementary points are displayed) "none" (no points are displayed) The status (active or supplementary) of columns is set in mjca using the option supcol.
mass	Vector of two logicals specifying if the mass should be represented by the area of the point symbols (first entry for rows, second one for columns)

contrib	Vector of two character strings specifying if contributions (relative or absolute) should be represented by different colour intensities. Available options are "none" (contributions are not indicated in the plot). "absolute" (absolute contributions are indicated by colour intensities). "relative" (relative contributions are indicated by colour intensities). If set to "absolute" or "relative", points with zero contribution are displayed in white. The higher the contribution of a point, the closer the corresponding colour to the one specified by the col option.
col	Vector of length 2 specifying the colours of row and column point symbols, by default black for rows and red for columns. Colours can be entered in hexadecimal (e.g. "#FF0000"), rgb (e.g. rgb(1,0,0)) values or by R-name (e.g. "red").
pch	Vector of length 4 giving the type of points to be used for row active and supplementary, column active and supplementary points. See pchlist for a list of symbols.
labels	Vector of length two specifying if the plot should contain symbols only (0), labels only (1) or both symbols and labels (2). Setting labels to 2 results in the symbols being plotted at the coordinates and the labels with an offset.
collabels	Determines the format used for column labels, when the columns are labeled in the plot. "both" uses the factor names and level value, in the form "factor:level" "level" uses the factor level value only "factor" uses the factor name only
arrows	Vector of two logicals specifying if the plot should contain points (FALSE, default) or arrows (TRUE). First value sets the rows and the second value sets the columns.
xlab, ylab	Labels for horizontal and vertical axes. The default, "_auto_" means that the function auto-generates a label of the form Dimension X (xx.xx %)
...	Further arguments passed to plot and points.

Details

The function plot.mjca makes a two-dimensional map of the object created by mjca with respect to two selected dimensions. By default the scaling option of the map is "symmetric", that is the so-called *symmetric map*. In this map both the row and column points are scaled to have inertias (weighted variances) equal to the principal inertia (eigenvalue) along the principal axes, that is both rows and columns are in pricipal coordinates. Other options are as follows:

- -"rowprincipal" or "colprincipal" - these are the so-called *asymmetric maps*, with either rows in principal coordinates and columns in standard coordinates, or vice versa (also known as row-metric-preserving or column-metric-preserving respectively). These maps are biplots;
- -"symbiplot" - this scales both rows and columns to have variances equal to the singular values (square roots of eigenvalues), which gives a symmetric biplot but does not preserve row or column metrics;
- -"rowgab" or "colgab" - these are asymmetric maps (see above) with rows (respectively, columns) in principal coordinates and columns (respectively, rows) in standard coordinates

multiplied by the mass of the corresponding point. These are also biplots and were proposed by Gabriel & Odoroff (1990);

- -"rowgreen" or "colgreen" - these are similar to "rowgab" and "colgab" except that the points in standard coordinates are multiplied by the square root of the corresponding masses, giving reconstructions of the standardized residuals.

This function has options for sizing and shading the points. If the option mass is TRUE for a set of points, the size of the point symbol is proportional to the relative frequency (mass) of each point. If the option contrib is "absolute" or "relative" for a set of points, the colour intensity of the point symbol is proportional to the absolute contribution of the points to the planar display or, respectively, the quality of representation of the points in the display. To globally resize all the points (and text labels), use par("cex"=) before the plot.

Value

In addition to the side effect of producing the plot, the function invisibly returns the coordinates of the plotted points, a list of two components, with names rows and cols. These can be used to further annotate the plot using base R plotting functions.

References

Gabriel, K.R. and Odoroff, C. (1990). Biplots in biomedical research. *Statistics in Medicine*, **9**, pp. 469-485.
Greenacre, M.J. (1993) *Correspondence Analysis in Practice*. London: Academic Press.
Greenacre, M.J. (1993) Biplots in correspondence Analysis, *Journal of Applied Statistics*, **20**, pp. 251 - 269.

See Also

mjca, summary.mjca, print.mjca, pchlist

Examples

```
data("wg93")

# A two-dimensional map with standard settings
plot(mjca(wg93[,1:4]))
```

---

plot3d.ca *Plotting 3D maps in correspondence analysis*

Description

Graphical display of correspondence analysis in three dimensions

Usage

```
## S3 method for class 'ca'
plot3d(x, dim = c(1, 2, 3), map = "symmetric", what = c("all", "all"),
             contrib = c("none", "none"), col = c("#6666FF","#FF6666"),
             labcol = c("#0000FF", "#FF0000"), pch = c(16, 1, 18, 9),
             labels = c(2, 2), sf = 0.00001, arrows  = c(FALSE, FALSE),
             axiscol = "#333333", axislcol = "#333333",
 laboffset = list(x = 0, y = 0.075, z = 0.05), ...)
```

Arguments

x	Simple correspondence analysis object returned by ca
dim	Numerical vector of length 2 indicating the dimensions to plot
map	Character string specifying the map type. Allowed options include "symmetric" (default) "rowprincipal" "colprincipal" "symbiplot" "rowgab" "colgab" "rowgreen" "colgreen"
what	Vector of two character strings specifying the contents of the plot. First entry sets the rows and the second entry the columns. Allowed values are "none" (no points are displayed) "active" (only active points are displayed, default) "supplementary" (only supplementary points are displayed) "all" (all available points) The status (active or supplementary) is set in ca.
contrib	Vector of two character strings specifying if contributions (relative or absolute) should be indicated by different colour intensities. Available options are "none" (contributions are not indicated in the plot). "absolute" (absolute contributions are indicated by colour intensities). "relative" (relative conrributions are indicated by colour intensities). If set to "absolute" or "relative", points with zero contribution are displayed in white. The higher the contribution of a point, the closer the corresponding colour to the one specified by the col option.
col	Vector of length 2 specifying the colours of row and column profiles. Colours can be entered in hexadecimal (e.g. "#FF0000"), rgb (e.g. rgb(1,0,0)) values or by R-name (e.g. "red").
labcol	Vector of length 2 specifying the colours of row and column labels.
pch	Vector of length 2 giving the type of points to be used for rows and columns.
labels	Vector of length two specifying if the plot should contain symbols only (0), labels only (1) or both symbols and labels (2). Setting labels to 2 results in the symbols being plotted at the coordinates and the labels with an offset.
sf	A scaling factor for the volume of the 3d primitives.

arrows	Vector of two logicals specifying if the plot should contain points (FALSE, default) or arrows (TRUE). First value sets the rows and the second value sets the columns.
axiscol	Colour of the axis line.
axislcol	Colour of the axis labels.
laboffset	List with 3 slots specifying the label offset in x, y, and z direction.
...	Further arguments passed to the rgl functions.

See Also

ca

---

print.ca *Printing ca objects*

Description

Printing method for correspondence analysis objects

Usage

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

Arguments

x	Simple correspondence analysis object returned by ca
...	Further arguments are ignored

Details

The function print.ca gives the basic statistics of the ca object. First the eigenvalues (that is, principal inertias) and their percentages with respect to total inertia are printed. Then for the rows and columns respectively, the following are printed: the masses, chi-square distances of the points to the centroid (i.e., centroid of the active points), point inertias (for active points only) and principal coordinates on the first nd dimensions requested (default = 2 dimensions). The function summary.ca gives more detailed results about the inertia contributions of each point on each principal axis.
For supplementary points, masses and inertias are not applicable.

See Also

ca

Examples

```
data("smoke")
print(ca(smoke))
```

print.mjca *Printing mjca objects*

Description

Printing method for multiple and joint correspondence analysis objects

Usage

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

Arguments

x	Multiple or joint correspondence analysis object returned by mjca
...	Further arguments are ignored

Details

The function print.mjca gives the basic statistics of the mjca object. First the eigenvalues (that is, principal inertias) and their percentages with respect to total inertia are printed. Then for the rows and columns respectively, the following are printed: the masses, chi-square distances of the points to the centroid (i.e., centroid of the active points), point inertias (for active points only) and principal coordinates on the first nd dimensions requested (default = 2 dimensions). The function summary.mjca gives more detailed results about the inertia contributions of each point on each principal axis.
For supplementary points, masses and inertias are not applicable.

See Also

mjca

Examples

```
data("wg93")
print(mjca(wg93[,1:4]))
# equivalent to:
mjca(wg93[,1:4])
```

print.summary.ca *Printing summeries of ca objects*

Description

Printing method for summaries of correspondence analysis objects

Usage

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

Arguments

x	Summary of a simple correspondence analysis object returned by summary.ca
...	Further arguments are ignored

See Also

ca, summary.ca

print.summary.mjca *Printing summeries of mjca objects*

Description

Printing method for summaries of multiple and joint correspondence analysis objects

Usage

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

Arguments

x	summary of a multiple or joint correspondence analysis object returned by summary.mjca
...	Further arguments are ignored

See Also

mjca, summary.mjca

smoke *Smoke dataset*

Description

Artificial dataset in Greenacre (1984)

Usage

```
data(smoke)
```

Format

Table containing 5 rows (staff group) and 4 columns (smoking categories), giving the frequencies of smoking categories in each staff group in a fictional organization.

References

Greenacre, M.J. (1984). *Theory and Applications of Correspondence Analysis*. London: Academic Press.

summary.ca *Summarizing simple correspondence analysis*

Description

Printed output summarizing the results of ca, including a scree-plot of the principal inertias and row and column contributions.

Usage

```
## S3 method for class 'ca'
summary(object, scree = TRUE, rows=TRUE, columns=TRUE, ...)
```

Arguments

object	Simple correspondence analysis object returned by ca.
scree	Logical flag specifying if a scree-plot should be included in the output.
rows	Logical: should row contribution summaries be included?
columns	Logical: should column contribution summaries be included?
...	Further arguments (ignored)

Details

The function summary.ca gives the detailed numerical results of the ca function. All the eigenvalues (principal inertias) are listed, their percentages with respect to total inertia, and a bar chart (also known as a scree plot). Then for the set of rows and columns a table of results is given in a standard format, where quantities are either multiplied by 1000 or expressed in permills (thousandths): the mass of each point (x1000), the quality of display in the solution subspace of nd dimensions, the inertia of the point (in permills of the total inertia), and then for each dimension of the solution the principal coordinate (x1000), the (relative) contribution COR of the principal axis to the point inertia (x1000) and the (absolute) contribution CTR of the point to the inertia of the axis (in permills of the principal inertia).
For supplementary points, masses, inertias and absolute contributions (CTR) are not applicable, but the relative contributions (COR) are valid as well as their sum over the set of chosen nd dimensions (QLT).

Examples

```
data("smoke")
summary(ca(smoke))
```

---

summary.mjca *Summarizing multiple and joint correspondence analysis*

Description

Textual output summarizing the results of mjca, including a scree-plot of the principal inertias and row and column contributions.

Usage

```
## S3 method for class 'mjca'
summary(object, scree = TRUE, rows = FALSE, columns = TRUE, ...)
```

Arguments

object	Multiple or joint correspondence analysis object returned by mjca.
scree	Logical flag specifying if a scree-plot should be included in the output.
rows	Logical specifying whether the results for the rows should be included in the output (default = FALSE).
columns	Logical specifying whether the results for the columns should be included in the output (default = TRUE).
...	Further arguments (ignored)

Details

The function summary.mjca gives the detailed numerical results of the mjca function. All the eigenvalues (principal inertias) are listed, their percentages with respect to total inertia, and a bar chart (also known as a scree plot). Then for the set of rows and columns a table of results is given in a standard format, where quantities are either multiplied by 1000 or expressed in permills (thousandths): the mass of each point (x1000), the quality of display in the solution subspace of nd dimensions, the inertia of the point (in permills of the total inertia), and then for each dimension of the solution the principal coordinate (x1000), the (relative) contribution COR of the principal axis to the point inertia (x1000) and the (absolute) contribution CTR of the point to the inertia of the axis (in permills of the principal inertia).
For supplementary points, masses, inertias and absolute contributions (CTR) are not applicable, but the relative contributions (COR) are valid as well as their sum over the set of chosen nd dimensions (QLT).

Examples

```
data("wg93")
summary(mjca(wg93[,1:4]))
```

---

wg93 — *International Social Survey Program on Environment 1993 - western German sample*

Description

This data frame contains records of four questions on attitude towards science with responses on a five-point scale (1=agree strongly to 5=disagree strongly) and three demographic variables (sex, age and education).

Usage

```
data(wg93)
```

Format

Data frame (871x7).

Source

ISSP (1993). International Social Survey Program: Environment. http://www.issp.org

Index