Package ‘bpca’

October 12, 2022

Type Package

Title Biplot of Multivariate Data Based on Principal Components Analysis

Version 1.3-4

Date 2021-03-22

Author José Cláudio Faria, Clarice Garcia Borges Demétrio and Ivan Bezerra Allaman

Maintainer Jose Claudio Faria `<joseclaudio.faria@gmail.com>`

Depends R (>= 2.6.0), scatterplot3d, rgl

Imports xtable

Description Implements biplot (2d and 3d) of multivariate data based on principal components analysis and diagnostic tools of the quality of the reduction.

License GPL (>= 2)

Encoding latin1

LazyLoad yes

LazyData yes

NeedsCompilation no

Repository CRAN

Date/Publication 2021-03-22 19:50:02 UTC

R topics documented:

`bpca-package` *Biplot of Multivariate Data Based on Principal Components Analysis*

Description

Implements biplot (2d and 3d) and diagnostic tools of the quality of the reduction.

Author(s)

José Cláudio Faria (<`joseclaudio.faria@gmail.com`>)
Clarice Garcia Borges Demétrio (<`clarice@esalq.usp.br`>)
and
Ivan Bezerra Allaman (<`ivanalaman@gmail.com`>)

References

Gabriel, K. R. (1971) The biplot graphical display of matrices with application to principal component analysis. *Biometrika* **58**, 453-467.

Galindo Vilardón, M. P. (1986) Una alternativa de representación simultánea: HJ-Biplot. *Qüestiió*, 10(1):13-23, 1986.

Johnson, R. A. and Wichern, D. W. (1988) *Applied multivariate statistical analysis*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 6 ed.

Gower, J.C. and Hand, D. J. (1996) *Biplots*. Chapman & Hall.

Yan, B. W. and Kang, M. S. (2003) *GGE biplot analysis: a graphical tool for breeders, geneticists, and agronomists*. CRC Press, New York, 288p.

Examples

```
##
## Grouping objects with different symbols and colors - 2d and 3d
##

dev.new(w=6, h=6)
oask <- devAskNewPage(dev.interactive(orNone=TRUE))

## Not run:
# 2d
plot(bpca(iris[-5]),
     var.factor=.3,
     var.cex=.7,
     obj.names=FALSE,
```

```
     obj.cex=1.5,
     obj.col=c('red', 'green3', 'blue')[unclass(iris$Species)],
     obj.pch=c('+', '*', '-')[unclass(iris$Species)])

# 3d static
plot(bpca(iris[-5],
          d=1:3),
     var.factor=.2,
     var.color=c('blue', 'red'),
     var.cex=1,
     obj.names=FALSE,
     obj.cex=1,
     obj.col=c('red', 'green3', 'blue')[unclass(iris$Species)],
     obj.pch=c('+', '*', '-')[unclass(iris$Species)])

# 3d dynamic
plot(bpca(iris[-5],
          method='hj',
          d=1:3),
     rgl.use=TRUE,
     var.col='brown',
     var.factor=.3,
     var.cex=1.2,
     obj.names=FALSE,
     obj.cex=.8,
     obj.col=c('red', 'green3', 'orange')[unclass(iris$Species)],
     simple.axes=FALSE,
     box=TRUE)

## End(Not run)

##
## New options plotting
##
plot(bpca(ontario))

# Labels for all objects
(obj.lab <- paste('g',
                  1:18,
                  sep=''))

# Giving obj.labels
plot(bpca(ontario),
    obj.labels=obj.lab)

# Evaluate an object (1 is the default)
plot(bpca(ontario),
     type='eo',
     obj.cex=1)

plot(bpca(ontario),
     type='eo',
     obj.id=7,
```

```
     obj.cex=1)

# Giving obj.labels
plot(bpca(ontario),
     type='eo',
     obj.labels=obj.lab,
     obj.id=7,
     obj.cex=1)

# The same as above
plot(bpca(ontario),
     type='eo',
     obj.labels=obj.lab,
     obj.id='g7',
     obj.cex=1)

# Evaluate a variable (1 is the default)
plot(bpca(ontario),
     type='ev',
     var.pos=2,
     var.cex=1)

plot(bpca(ontario),
     type='ev',
     var.id='E7',
     obj.labels=obj.lab,
     var.pos=1,
     var.cex=1)

# A complete plot
cl <- 1:3
plot(bpca(iris[-5]),
     type='ev',
     var.id=1,
     var.fac=.3,
     obj.names=FALSE,
     obj.col=cl[unclass(iris$Species)])

legend('topleft',
       legend=levels(iris$Species),
       text.col=cl,
       pch=19,
       col=cl,
       cex=.9,
       box.lty=0)

# Compare two objects (1 and 2 are the default)
plot(bpca(ontario),
     type='co')

plot(bpca(ontario),
     type='co',
     obj.labels=obj.lab)
```

```
plot(bpca(ontario),
     type='co',
     obj.labels=obj.lab,
     obj.id=13:14)

plot(bpca(ontario),
     type='co',
     obj.labels=obj.lab,
     obj.id=c('g7', 'g13'))

# Compare two variables
plot(bpca(ontario),
     type='cv')

# Which won where/what
plot(bpca(ontario),
     type='ww')

# Discrimitiveness vs. representativeness
plot(bpca(ontario),
     type='dv')

# Means vs. stability
plot(bpca(ontario),
     type='ms')

# Rank objects with ref. to the ideal variable
plot(bpca(ontario),
     type='ro')

# Rank variables with ref. to the ideal object
plot(bpca(ontario),
     type='rv')

## Not run:
plot(bpca(iris[-5]),
     type='eo',
     obj.id=42,
     obj.cex=1)

plot(bpca(iris[-5]),
     type='ev',
     var.id='Sepal.Width')

plot(bpca(iris[-5]),
     type='ev',
     var.id='Sepal.Width',
     var.factor=.3)

## End(Not run)

devAskNewPage(oask)
```

`bpca` *Biplot of Multivariate Data Based on Principal Components Analysis*

Description

Computes biplot reduction on `data.frame`, `matrix` or `prcomp` objects and returns a `bpca` object.

Usage

```
  bpca(x, ...)
  ## Default S3 method:
bpca(x,
     d=1:2,
     center=2,
     scale=TRUE,
     method=c('hj', 'sqrt', 'jk', 'gh'),
     iec=FALSE,
     var.rb=FALSE,
     var.rd=FALSE,
     limit=10, ...)
  ## S3 method for class 'prcomp'
bpca(x,
     d=1:2, ...)
```

Arguments

`x`	A `data.frame`, `matrix` or `prcomp` object.
`d`	A vector giving the first and last eigenvalue to be considered by the biplot reduction. It can be `d=1:3` or `d=c(1,3)` for 3d biplot. The default is `d=1:2`.
`center`	Numeric. The type of centering to be performed: ‘`0`’ - no centering ‘`1`’ - global-centered = `sweep(x, 1, mean(x))` ‘`2`’ - column-centered = `sweep(x, 2, apply(x, 2, mean))` ‘`3`’ - double-centered = `sweep(sweep(x, 1, apply(x, 1, mean)), 2, apply(x, 2, mean)) + mean(x)` The default is 2.
`scale`	Logical. A value indicating whether the variables should be scaled to have unit variance before the analysis takes place: `FALSE` - no scale; `TRUE` - scale.
`method`	A vector of character strings that indicates the method of factorization: ‘`hj`’ - ‘`HJ`’ (‘`simetric`’, Galindo Villardón (1986)); ‘`sqrt`’ - ‘`SQRT`’ (‘`squared root - simetric`’, Gabriel (1971)); ‘`jk`’ - ‘`JK`’ (‘`row metric preserving`’, Gabriel (1971)); ‘`gh`’ - ‘`GH`’ (‘`column metric preserving`’, Gabriel (1971)).

`iec`	Logical. If `TRUE` the matrix of eigenvalues, coordinates od objects and variables will be inverted. The default is `FALSE`.
`var.rb`	A logical value. If `TRUE`, all correlation coefficients for all variables (*under the biplot projection*) will be computed.
`var.rd`	A logical value. If `TRUE`, the diagnostic of the representation of variables projected by the biplot will be computed. If `var.rd` is `TRUE` the `var.rb` parameter must be also `TRUE`.
`limit`	A vector giving the percentual limit to define poor representation of variables.
`...`	Additional parameters. It is necessary to be S3 method.

Details

The biplot is a multivariate method for graphing row and column elements using a single plot (Gabriel, 1971).

The biplot of a matrix

$$_nY_p$$

projects on the same plot: *rows* (associated with *n* objects) and *columns* (associated with *p* variables), markers from data that forms a two-way table (`data.frame` or `matrix` object). The markers are computed from the singular value decomposition, `svd(Y)`, and subsequent factorization.

The *bi* refers to the kind of information contained in a data set disposed in a two-way table. If the data are a *tri*-dimensional array the method will be called *triplot* (not still contemplated in the bpca package).

The basic idea behind the biplot method was to add the information about the variables to the principal component graph (Johnson \& Wichern, 1988).

Considering the results of

$$svd(_nY_p)$$

- *d*: A vector containing the singular values of Y, of length `min(n, p)`;
- *u*: A matrix whose columns contain the left singular vectors of Y, present if ‘`nu > 0`’. Dimension ‘`c(n, nu)`’;
- *v*: A matrix whose columns contain the right singular vectors of Y, present if ‘`nv > 0`’. Dimension `c(p, nv)`.

and also,

$$s^2 = diag(d)$$

$$n = n\_objects(Y)$$

it is possible an approximation of Y:

$$_nY_p \approx Y_m = g.h'$$

in various ways. The methods of factorization computed by the `bpca` function are:

- *HJ - simetric*, Galindo Villardón (1986):

$$g = u * s^2$$

$$h = s^2 * v'$$

- *SQRT - squared root simetric*, Gabriel (1971):

$$g = u * \sqrt{s^2}$$

$$h = \sqrt{s^2} * v'$$

- *JK - row metric preserving*, Gabriel (1971):

$$g = u * s^2$$

$$h = v'$$

- *GH - column metric preserving*, Gabriel (1971):

$$g = \sqrt{n-1} * u$$

$$h = \frac{1}{\sqrt{n-1}} * s^2 * v'$$

Considering

$$_nY_p \approx Y_m$$

it is possible to deduce that if the rank (r) of the matrix

$$_nY_p$$

is bigger than ‘m’, the biplot representation of Y will be an approximation, and accurate only in the case of $r=m$.

Due to the need of different methods of factorization, if ‘`x`’ is a `prcomp` object, the method `bpca.prcomp` will go back from the `prcomp` function. In other words, it will regenerates, or computes, the inverse of the `svd` decomposition of the given data

$$_nY_p$$

After this, it will make a call to the method `bpca.default` with the adequate parameters.

The biplot is used with many multivariate methods to display relationships between objects, variables and the interrelationship between objects and variables (as prevalence, importance). There are many variations of biplots (see the references).

Value

The function `bpca` returns an object of class `bpca.2d` or `bpca.3d`. Both are `list` objects with the slots:

`call`	The call used.
`eigenvalues`	A vector of the eigenvalues.
`eigenvectors`	A vector of the eigenvectors.
`numer`	A vector of the number of eigenvalues considered in the reduction.
`importance`	A matrix with the *general* and *partial* variation explained by the reduction.
`coord`	A list with the coordinates of the two components: objects and variables.
`var.rb`	A matrix of all correlation coefficients for all variables under the biplot projection.
`var.rd`	A matrix of the diagnostic of the poor projection of variable correlations by the biplot reduction.

Author(s)

José Cláudio Faria (<`joseclaudio.faria@gmail.com`>)
Clarice Garcia Borges Demétrio (<`clarice@esalq.usp.br`>)
and
Ivan Bezerra Allaman (<`ivanalaman@gmail.com`>)

References

Gabriel, K. R. (1971) The biplot graphical display of matrices with application to principal component analysis. *Biometrika* **58**, 453-467.

Galindo Vilardón, M. P. (1986) Una alternativa de representación simultánea: HJ-Biplot. *Qüestiió*, 10(1):13-23, 1986.

Johnson, R. A. and Wichern, D. W. (1988) *Applied multivariate statistical analysis*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 6 ed.

Gower, J.C. and Hand, D. J. (1996) *Biplots*. Chapman & Hall.

Yan, B. W. and Kang, M. S. (2003) *GGE biplot analysis: a graphical tool for breeders, geneticists, and agronomists*. CRC Press, New York, 288p.

See Also

`biplot.prcomp`

Examples

```
##
## Example 1
## Computing and ploting a bpca object with 'graphics' package - 2d
##

bp <- bpca(gabriel1971)

dev.new(w=6, h=6)
oask <- devAskNewPage(dev.interactive(orNone=TRUE))
plot(bp,
     var.factor=2)

# Exploring the object 'bp' created by the function 'bpca'
class(bp)
names(bp)
str(bp)

summary(bp)
bp$call
bp$eigenval
bp$eigenvec
bp$numb
bp$import
bp$coord
bp$coord$obj
```

```
bp$coord$var
bp$var.rb
bp$var.rd

## Not run:
##
## Example 2
## Computing and plotting a bpca object with 'scatterplot3d' package - 3d
##

bp <- bpca(gabriel1971,
           d=2:4)

plot(bp,
     var.factor=3,
     xlim=c(-2,2),
     ylim=c(-2,2),
     zlim=c(-2,2))

# Exploring the object 'bp' created by the function 'bpca'
class(bp)
names(bp)
str(bp)

summary(bp)
bp$call
bp$eigenval
bp$eigenvec
bp$numb
bp$import
bp$coord
bp$coord$obj
bp$coord$var
bp$var.rb
bp$var.rd

##
## Example 3
## Computing and plotting a bpca object with 'rgl' package - 3d
##

plot(bpca(gabriel1971,
          d=1:3),
     rgl.use=TRUE,
     var.factor=2)

# Suggestion: Interact with the graphic with the mouse
# left button: press, maintain and movement it to interactive rotation;
# right button: press, maintain and movement it to interactive zoom.
# Enjoy it!

##
## Example 4
```

```
## Grouping objects with different symbols and colors - 2d and 3d
##

# 2d
plot(bpca(iris[-5]),
     var.factor=.3,
     var.cex=.7,
     obj.names=FALSE,
     obj.cex=1.5,
     obj.col=c('red', 'green3', 'blue')[unclass(iris$Species)],
     obj.pch=c('+', '*', '-')[unclass(iris$Species)])

# 3d static
plot(bpca(iris[-5],
          d=1:3),
     var.factor=.2,
     var.color=c('blue', 'red'),
     var.cex=1,
     obj.names=FALSE,
     obj.cex=1,
     obj.col=c('red', 'green3', 'blue')[unclass(iris$Species)],
     obj.pch=c('+', '*', '-')[unclass(iris$Species)])

# 3d dynamic
plot(bpca(iris[-5],
          method='hj',
          d=1:3),
     rgl.use=TRUE,
     var.col='brown',
     var.factor=.3,
     var.cex=1.2,
     obj.names=FALSE,
     obj.cex=.8,
     obj.col=c('red', 'green3', 'orange')[unclass(iris$Species)],
     simple.axes=FALSE,
     box=TRUE)

## End(Not run)

devAskNewPage(oask)
```

`dt.tools` *Data Tools for Multivariate*

Description

Calculates vector variable lengths, angles between vector variables and variable correlations from ‘data.frame’ or ‘matrix’ objects

Usage

```
dt.tools(x,
         center=2,
         scale=TRUE)
```

Arguments

`x`	A `data.frame` or `matrix` object.
`center`	Numeric. The type of centering to be performed: '`0`' - no centering; '`1`' - global-centered = `sweep(x, 1, mean(x));` '`2`' - column-centered = `sweep(x, 2, apply(x, 2, mean));` '`3`' - double-centered = `sweep(sweep(x, 1, apply(x, 1, mean)), 2, apply(x, 2, mean))` + `mean(x)`. The default is 2.
`scale`	Logical. A value indicating whether the variables should be scaled to have unit variance before the analysis takes place: `FALSE` - no scale; `TRUE` - scale.

Details

This function computes: vector variable lengths, angles between vector variables and variable correlations from `data.frame` or `matrix` objects.

If the data are centered (`center=2`), the correlations are the same as those obtained by the `cor` function.

Value

An `list` with the components:

`length`	A vector of the lengths.
`angle`	A matrix of the angles.
`r`	A matrix of the observed correlations.

Author(s)

José Cláudio Faria (<`joseclaudio.faria@gmail.com`>)
Clarice Garcia Borges Demétrio (<`clarice@esalq.usp.br`>)
and
Ivan Bezerra Allaman (<`ivanalaman@gmail.com`>)

References

Johnson, R. A. and Wichern, D. W. (1988) *Applied multivariate statistical analysis*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 6 ed.

See Also

bpca

Examples

```
##
## Computes: vector variable lengths, angles between vector variables and
## variable correlations from data.frame or matrix objects (n x p)
## n = rows (objects)
## p = columns (variables)
##

dt <- dt.tools(iris,
               2)  # No numeric columns are removed in dt.tools

# Exploring the object 'bp' created by the function 'var.tools'
class(dt)
names(dt)
str(dt)

dt$length
dt$angle
dt$r
dt

# Checking the determinations
(iris.tools <- round(dt.tools(iris,
                               center=2)$r,
                     5))

(iris.obsv  <- round(cor(iris[-5]),
                     5))

all(iris.tools == iris.obsv)
```

---

gabriel1971 *Percentages of households having various facilities and appliances in East Jerusalem Arab areas, by quarters of the town*

Description

Percentages of households having various facilities and appliances in East Jerusalem Arab areas, by quarters of the town. The average percentages in each quarter indicate the standard of living of that area and the average percentage of each facility or appliance its over-all prevalence.

Usage

```
data(gabriel1971)
```

Format

The format is:

```
num [1:8, 1:9] 98.2 78.8 14.4 86.2 32.9 73 4.6 29.2 97.2 81 ...
- attr(*, "dimnames")=List of 2
..$ : chr [1:8] "toilet" "kitchen" "bath" "eletricity" ...
..$ : chr [1:9] "CRISTIAN" "ARMENIAN" "JEWISH" "MOSLEM" ...
```

References

Gabriel, K. R. (1971) The biplot graphical display of matrices with application to principal component analysis. *Biometrika* **58**, 453-467.

Examples

```
##
## A simple example
##
data(gabriel1971)
bp <- bpca(gabriel1971)

dev.new(w=6, h=6)
plot(bp,
     var.factor=2)

# Exploring the object 'bp' created by the function 'bpca'
class(bp)
names(bp)
str(bp)

summary(bp)
bp$call
bp$eigenval
bp$eigenvec
bp$numb
bp$import
bp$coord
bp$coord$obj
bp$coord$var
bp$var.rb
bp$var.rd
```

`gge2003` *A didactic matrix of genotypes (rows) and environments (columns)*

Description

A didatic matrix of genotypes (rows) and environments (columns) proposed by Weikai Yan and Manjit S. Kang in GGE biplot analysis: a graphical tool for breeders, geneticists, and agronomists.

Usage

```
data(gge2003)
```

Format

The format is:

```
num [1:4, 1:3] 20 6 -10 8 -9 12 -6 -12 6 -15 ...
- attr(*, "dimnames")=List of 2
..$ : chr [1:4] "G1" "G2" "G3" "G4"
..$ : chr [1:3] "E1" "E2" "E3"
```

References

Yan, B. W. and Kang, M. S. (2003) *GGE biplot analysis : a graphical tool for breeders, geneticists, and agronomists*. CRC Press, New York, 288p.

Examples

```
##
## Example from YAN, W & KANG, M.S. GGE biplot analysis : a graphical tool
## for breeders, geneticists, and agronomists
##

data(gge2003)
bp <- bpca(t(gge2003), var.rb=TRUE)

as.dist(bp$var.rb)

dev.new(w=8, h=4)
op = par(no.readonly=TRUE)
par(mfrow=c(1,2))

plot(bpca(gge2003, var.pos=2),
     main='Columns as variables \n (var.pos=2)',
     var.col=1,
     obj.col=c(2:4, 2), obj.cex=.8)

plot(bpca(gge2003, var.pos=1),
     main='Rows as variables \n (var.pos=1)',
     var.col=1,
     obj.col=2:4, obj.cex=.8)

par(op)
```

`marina` *Films shown at five festivals in Brazil from 2007 to 2011*

Description

A data.frame containing the films shown at five festivals in Brazil from 2007 to 2011.

Usage

```
data(marina)
```

Format

The format is:

```
'data.frame': 25 obs. of  6 variables:
year: int  2011 2011 2011 2011 2011 2010 2010 2010 2010 2010 ...
regE: Factor w/ 5 levels "CO","N","NE",..: 1 5 4 3 2 1 5 4 3 2 ...
F   : int  84 55 63 44 25 40 54 37 49 27 ...
D   : int  26 13 19 16 7 9 14 11 19 6 ...
MD  : int  22 9 13 14 5 6 11 10 13 3 ...
WD  : int  4 4 6 2 2 3 3 1 6 3 ...
#
# Description
#
year: Year in which the film was shown
regE: Region where the film was shown
F   : Total number of films
D   : Number of documentaries
MD  : Documentary directed by men
WD  : Documentary directed by woman
```

Examples

```
data(marina)

marina
```

`ontario` *Ontario winter wheat (1993)*

Description

The sample data are yields from the 1993 Ontario winter wheat (Triticum aestivum L.) performance trials, in which 18 cultivars were tested at nine locations (Yan and Kang 2003)

Usage

```
data(ontario)
```

Format

A data frame with 18 observations on the following 10 variables.

Source

Yan W, Kang MS (2003). GGE Biplot Analysis: A Graphical Tool for Breeders, Geneticists, and Agronomists. CRC Press, Boca Raton, FL, USA.

Examples

```
data(ontario)

# 2d
plot(bpca(ontario,
          d=1:2))

# 3d
plot(bpca(ontario,
          d=1:3),
     rgl.use=TRUE)
```

`plot` *Biplot of Multivariate Data Based on Principal Components Analysis*

Description

These are methods for objects of class `bpca.2d`, `bpca.3d` and `qbpca`.

Usage

```
  ## S3 method for class 'bpca.2d'
plot(x,
     type=c('bp', 'eo', 'ev', 'co', 'cv', 'ww', 'dv', 'ms', 'ro', 'rv'),
     c.color='darkgray',
     c.lwd=1,
     c.number=5,
     c.radio=1,
     obj.id=1:2,
     var.id=1,
     base.color='red3',
     base.lty='dotted',
     proj.color='gray',
     proj.lty='dotted',
     a.color='blue',
```

```
     a.lty='solid',
     a.lwd=2,
     a.length=.1,
     ref.lines=TRUE,
     ref.color='navy',
     ref.lty='dotted',
     var.factor=1,
     var.color='red3',
     var.lty='solid',
     var.pch=20,
     var.pos=4,
     var.cex=.6,
     var.offset=.2,
     obj.factor=1,
     obj.color='black',
     obj.pch=20,
     obj.pos=4,
     obj.cex=.6,
     obj.offset=.2,
     obj.names=TRUE,
     obj.labels,
     obj.identify=FALSE,
     xlim,
     ylim,
     xlab,
     ylab, ...)

  ## S3 method for class 'bpca.3d'
plot(x,
     rgl.use=FALSE,
     ref.lines=TRUE,
     ref.color='navy',
     ref.lty=ifelse(rgl.use, NA, 'dotted'),
     clear3d=ifelse(rgl.use, TRUE, NULL),
     simple.axes=ifelse(rgl.use, TRUE, NULL),
     aspect=ifelse(rgl.use, c(1, 1, 1), NULL),
     var.factor=1,
     var.color='red3',
     var.lty=ifelse(rgl.use, NA, 'solid'),
     var.pch=ifelse(rgl.use, NULL, 20),
     var.pos=ifelse(rgl.use, 0, 4),
     var.cex=ifelse(rgl.use, .8, .6),
     var.offset=ifelse(rgl.use, NULL, .2),
     obj.color='black',
     obj.pch=ifelse(rgl.use, NULL, 20),
     obj.pos=ifelse(rgl.use, 0, 4),
     obj.cex=ifelse(rgl.use, .8, .6),
     obj.offset=ifelse(rgl.use, NULL, .2),
```

```
         obj.names=TRUE,
         obj.labels,
         obj.identify=FALSE,
         box=FALSE,
         angle=ifelse(rgl.use, NULL, 40),
         xlim,
         ylim,
         zlim,
         xlab,
         ylab,
         zlab, ...)

   ## S3 method for class 'qbpca'
 plot(x,
         xlab='Index',
         ylab='r',
         pch=c(1,8),
         col=c(4,2), ...)
```

Arguments

`x`	A `bpca.2d`, `bpca.3d` or `qbpca` object.
`type`	Type of biplot: ‘`bp`’ - conventional biplot; ‘`eo`’ - evaluate an object; ‘`ev`’ - evaluate a variable; ‘`co`’ - compare two objects; ‘`cv`’ - compare (relationship among) variables; ‘`ww`’ - which won where/what; ‘`dv`’ - discrimitiveness vs. representativeness; ‘`ms`’ - mean vs. stability; ‘`ro`’ - rank objects with ref. to an “ideal” variable; ‘`rv`’ - rank variables with ref. to an “ideal” object.
`c.color`	The color of circles.
`c.lwd`	The `lwd` of circles.
`c.number`	The number of circles.
`c.radio`	The radio of circles.
`obj.id`	An object(s) number(s) or name(s). It is used with reprojctions to identify the object(s) when the `type` option is ‘`'eo'`’ or ‘`'co'`’. The default is ‘`1`’ for ‘`'eo'`’ and `1:2` for ‘`'co'`’.
`var.id`	A variable number or name. It is used with reprojections to identify the variable when the ‘`type`’ option is `'ev'`.
`base.color`	The color for the base lines. It is used only with reprojections.
`base.lty`	The ‘`lty`’ for the base lines. It is used only with reprojections.
`proj.color`	The color for the projections lines. It is used only with reprojections.
`proj.lty`	The ‘`lty`’ for the projections lines. It is used only with reprojections.

`a.color`	The color for the arrow. It is used only with reprojections.
`a.lty`	The ‘`lty`’ for the arrow. It is used only with reprojections.
`a.lwd`	The ‘`lwd`’ for the arrow. It is used only with reprojections.
`a.length`	The ‘`length`’ for the arrow. It is used only with reprojections.
`rgl.use`	A logical value. If `TRUE` the 3d scatter will be done with the `rgl` environment, in another way the `scatterplot3d` will be used.
`ref.lines`	A logical value. If `TRUE`, the reference lines will be added to the biplot.
`ref.color`	Line color for reference lines.
`ref.lty`	Line type of the reference lines.
`clear3d`	A logical value. It clears (or not) a 3d biplot before making a new one. Used only if `rgl.use=TRUE`.
`simple.axes`	A logical value to draw simple axes. Used only if `rgl.use=TRUE`.
`aspect`	A vector of the apparent ratios of the ‘`x`’, ‘`y`’, and ‘`z`’ axes of the bounding box. Used only if `rgl.use=TRUE`.
`var.factor`	Factor of expansion/reduction of length lines of the variables.
`var.color`	A vector of colors for the variables representation.
`var.lty`	Line type for the variables. Used only if `rgl.use=FALSE`.
`var.pch`	A vector of plotting symbols or characters for the variables. If too short, the values are recycled. Used only if `rgl.use=FALSE`.
`var.pos`	Position of labels for the variables.
`var.cex`	Character expansion for the variables.
`var.offset`	The distance (in character widths) which separates the label from identified points of variables. Negative values are allowed. Used only if `rgl.use=FALSE`.
`obj.factor`	Factor of expansion/reduction of length lines of the objects.
`obj.color`	A vector of colors for the objects representation.
`obj.pch`	A vector of plotting symbols or characters for objects. If too short, the values are recycled. Used only if `rgl.use=FALSE`.
`obj.pos`	Position of labels for objects.
`obj.cex`	Character expansion for objects.
`obj.offset`	The distance (in character widths) which separates the label from identified points of objects. Negative values are allowed. Used only if `rgl.use=FALSE`.
`obj.names`	A logical value to represent objects as spheres or points.
`obj.identify`	A logical value. If `TRUE`, it is possible, in interative way, to identify the objects with the mouse.
`obj.labels`	A vector of labels for objects.
`box`	A logical value to whether to draw a box. Used only if ‘`rgl.use=TRUE`’.
`angle`	Angle between ‘`x`’ and ‘`y`’ axis (Attention: result depends on scaling). For $180 < angle < 360$ the returned functions xyz.convert and points3d will not work properly. Used only if `rgl.use=FALSE`.
`pch`	A vector of plotting symbols or characters.

`col`	A vector of colors.
`xlab`	A label for the ‘`x`’ axis.
`ylab`	A label for the ‘`y`’ axis.
`zlab`	A label for the ‘`z`’ axis (bpca.3d only).
`xlim`	The ‘`x`’ limits of the plot.
`ylim`	The ‘`y`’ limits of the plot.
`zlim`	The ‘`z`’ limits of the plot (bpca.3d only).
`...`	Other graphical parameters may also be passed as arguments to these functions.

Details

A biplot aims to represent both the observations and variables of a matrix of multivariate data on the same plot.

The methods `plot.bpca.2d` draw a 2d biplot (PC1 and PC2 on axis ‘`x`’ and ‘`y`’, respectively). It uses the traditional graphics system.

The methods `plot.bpca.3d` draw a 3d biplot (PC1, PC2 and PC3 on axis ‘`x`’, ‘`y`’ and ‘`z`’, respectively) in two ways:

- ‘`static`’ - It uses the package `scatterplot3d` based on traditional graphic system. Use the parameter ‘`rgl.use=FALSE`’ for it.
- ‘`dinamic`’ - It uses the package `rgl` a 3D real-time rendering device driver system for R. Use the parameter ‘`rgl.use=TRUE`’ for it.

The method `qb.pca` draws a scatterplot of the correlations observed (in the data) and projected (under the biplot) related to the variables. It uses the traditional graphics system. `qb.pca` is a simple (and graphical) measure of the quality of the biplot reduction associated to the variables.

Author(s)

José Cláudio Faria (<`joseclaudio.faria@gmail.com`>)
Clarice Garcia Borges Demétrio (<`clarice@esalq.usp.br`>)
and
Ivan Bezerra Allaman (<`ivanalaman@gmail.com`>)

References

Gabriel, K. R. (1971) The biplot graphical display of matrices with application to principal component analysis. *Biometrika* **58**, 453-467.

Galindo Vilardón, M. P. (1986) Una alternativa de representación simultánea: HJ-Biplot. *Qüestiió*, 10(1):13-23, 1986.

Johnson, R. A. and Wichern, D. W. (1988) *Applied multivariate statistical analysis*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 6 ed.

Gower, J.C. and Hand, D. J. (1996) *Biplots*. Chapman & Hall.

Yan, B. W. and Kang, M. S. (2003) *GGE biplot analysis: a graphical tool for breeders, geneticists, and agronomists*. CRC Press, New York, 288p.

See Also

`biplot.prcomp`

Examples

```
##
## Example 1
## Computing and ploting a bpca object with 'graphics' package - 2d
##

bp <- bpca(gabriel1971)

dev.new(w=6, h=6)
oask <- devAskNewPage(dev.interactive(orNone=TRUE))
plot(bp,
     var.factor=2)

# Additional graphical parameters (nonsense)
plot(bpca(gabriel1971,
          meth='sqrt'),
     main='gabriel1971 - sqrt',
     sub='The graphical parameters are working fine!',
     var.factor=2,
     var.cex=.6,
     var.col=rainbow(9),
     var.pch='v',
     obj.pch='o',
     obj.cex=.5,
     obj.col=rainbow(8),
     obj.pos=1,
     obj.offset=.5)

##
## Example 2
## Computing and plotting a bpca object with 'scatterplot3d' package - 3d
##

bp <- bpca(gabriel1971,
           d=1:3)

plot(bp,
     var.factor=3)

# Additional graphical parameters (nonsense)
plot(bpca(gabriel1971,
          d=1:3,
          meth='jk'),
     main='gabriel1971 - jk',
     sub='The graphical parameters are working fine!',
     var.factor=6,
     var.pch='+',
     var.cex=.6,
```

```
     var.col='green4',
     obj.pch='*',
     obj.cex=.8,
     obj.col=1:8,
     ref.lty='solid',
     ref.col='red',
     angle=70)

##
## Example 3
## Computing and plotting a bpca object with 'rgl' package - 3d
##

plot(bpca(gabriel1971,
          d=1:3),
     rgl.use=TRUE,
     var.factor=2)

# Suggestion: Interact with the graphic with the mouse
# left button: press, maintain and movement it to interactive rotation;
# right button: press, maintain and movement it to interactive zoom.
# Enjoy it!

## Not run:
##
## Example 4
## Grouping objects with different symbols and colors - 2d and 3d
##

# 2d
plot(bpca(iris[-5]),
     var.factor=.3,
     var.cex=.7,
     obj.names=FALSE,
     obj.cex=1.5,
     obj.col=c('red', 'green3', 'blue')[unclass(iris$Species)],
     obj.pch=c('+', '*', '-')[unclass(iris$Species)])

# 3d static
plot(bpca(iris[-5],
          d=1:3),
     var.factor=.2,
     var.color=c('blue', 'red'),
     var.cex=1,
     obj.names=FALSE,
     obj.cex=1,
     obj.col=c('red', 'green3', 'blue')[unclass(iris$Species)],
     obj.pch=c('+', '*', '-')[unclass(iris$Species)])

# 3d dynamic
plot(bpca(iris[-5],
          method='hj',
          d=1:3),
```

```
     rgl.use=TRUE,
     var.col='brown',
     var.factor=.3,
     var.cex=1.2,
     obj.names=FALSE,
     obj.cex=.8,
     obj.col=c('red', 'green3', 'orange')[unclass(iris$Species)],
     simple.axes=FALSE, box=TRUE)

##
## Example 5
## Computing and plotting a bpca object with 'obj.identify=TRUE' parameter - 2d
##

bp <- bpca(gabriel1971)

# Normal labels
if(interactive())
plot(bp,
     obj.names=FALSE,
     obj.identify=TRUE)

# Alternative labels
if(interactive())
plot(bp,
     obj.names=FALSE,
     obj.labels=c('toi', 'kit', 'bat', 'ele', 'wat', 'rad', 'tv', 'ref'),
     obj.identify=TRUE)

##
## Example 6
## Computing and plotting a bpca object with 'obj.identify=TRUE' parameter - 3d
##

bp <- bpca(gabriel1971,
           d=1:3)

# Normal labels
if(interactive())
plot(bp,
     obj.names=FALSE,
     obj.identify=TRUE)

# Alternative labels
if(interactive())
plot(bp,
     obj.names=FALSE,
     obj.labels=c('toi', 'kit', 'bat', 'ele', 'wat', 'rad', 'tv', 'ref'),
     obj.identify=TRUE)

##
## New options plotting
##
```

```
plot(bpca(ontario))

# Labels for all objects
(obj.lab <- paste('g',
                  1:18,
                  sep=''))

# Giving obj.labels
plot(bpca(ontario),
    obj.labels=obj.lab)

# Evaluate an object (1 is the default)
plot(bpca(ontario),
     type='eo',
     obj.cex=1)

plot(bpca(ontario),
     type='eo',
     obj.id=7,
     obj.cex=1)

# Giving obj.labels
plot(bpca(ontario),
     type='eo',
     obj.labels=obj.lab,
     obj.id=7,
     obj.cex=1)

# The same as above
plot(bpca(ontario),
     type='eo',
     obj.labels=obj.lab,
     obj.id='g7',
     obj.cex=1)

# Evaluate a variable (1 is the default)
plot(bpca(ontario),
     type='ev',
     var.pos=2,
     var.cex=1)

plot(bpca(ontario),
     type='ev',
     var.id='E7',
     obj.labels=obj.lab,
     var.pos=1,
     var.cex=1)

# A complete plot
cl <- 1:3
plot(bpca(iris[-5]),
     type='ev',
     var.id=1,
```

```
     var.fac=.3,
     obj.names=FALSE,
     obj.col=cl[unclass(iris$Species)])
legend('topleft',
       legend=levels(iris$Species),
       text.col=cl,
       pch=19,
       col=cl,
       cex=.9,
       box.lty=0)

# Compare two objects (1 and 2 are the default)
plot(bpca(ontario),
     type='co',
     c.radio=.4,
     c.color='blue',
     c.lwd=2)

plot(bpca(ontario),
     type='co',
     obj.labels=obj.lab,
     c.radio=.5,
     c.color='blue',
     c.lwd=2)

plot(bpca(ontario),
     type='co',
     obj.labels=obj.lab,
     obj.id=13:14)

plot(bpca(ontario),
     type='co',
     obj.labels=obj.lab,
     obj.id=c('g7',
              'g13'))

# Compare two variables
plot(bpca(ontario),
     type='cv',
     c.number=3,
     c.radio=1.5)

# Which won where/what
plot(bpca(ontario),
     type='ww')

# Discrimitiveness vs. representativeness
plot(bpca(ontario),
     type='dv')

plot(bpca(ontario),
     type='dv',
     c.number=4,
```

```
     c.radio=1)

# Means vs. stability
plot(bpca(ontario),
     type='ms')

plot(bpca(ontario),
     type='ms',
     c.number=3)

# Rank objects with ref. to the ideal variable
plot(bpca(ontario),
     type='ro')

plot(bpca(ontario),
     type='ro',
     c.number=6,
     c.radio=.5)

# Rank variables with ref. to the ideal object
plot(bpca(ontario),
     type='rv')

plot(bpca(ontario),
     type='rv',
     c.number=6,
     c.radio=.5)

plot(bpca(iris[-5]),
     type='eo',
     obj.id=42,
     obj.cex=1)

plot(bpca(iris[-5]),
     type='ev',
     var.id='Sepal.Width')

plot(bpca(iris[-5]),
     type='ev',
     var.id='Sepal.Width',
     var.factor=.3)

## End(Not run)

devAskNewPage(oask)
```

`print.xtable` *Print Method for xtable.bpca Objects*

Description

Returns (and prints) a summary list for `xtable.bpca` objects.

Usage

```
## S3 method for class 'xtable.bpca'
print(x,
      hline.after = getOption("xtable.hline.after", NULL),
      include.colnames = getOption("xtable.include.colnames", FALSE),
      add.to.row = getOption("xtable.add.to.row", NULL),
     sanitize.text.function = getOption("xtable.sanitize.text.function", NULL),
    sanitize.rownames.function = getOption("xtable.sanitize.rownames.function",
                                           sanitize.text.function),
    sanitize.colnames.function = getOption("xtable.sanitize.rownames.function",
                                           sanitize.text.function),...)
```

Arguments

`x`	A given object of the class `xtable` or `xtable.bpca`.
`hline.after`	When type="latex", a vector of numbers between -1 and nrow(x), inclusive, indicating the rows after which a horizontal line should appear. Default value is NULL which means draw a line before and after the columns names, draw a line before variables and at the end of the table.
`include.colnames`	If TRUE the columns names are printed. Default value is FALSE which means a column more elaborate was done.
`add.to.row`	A list of two components. The first component (which should be called 'pos') is a list that contains the position of rows on which extra commands should be added at the end. The second component (which should be called 'command') is a character vector of the same length as the first component, which contains the command that should be added at the end of the specified rows. Default value is NULL.
`sanitize.text.function`	All non-numeric entries (except row and column names) are sanitized in an attempt to remove characters which have special meaning for the output format. If sanitize.text.function is not NULL, it should be a function taking a character vector and returning one, and will be used for the sanitization instead of the default internal function. Default value is NULL.
`sanitize.rownames.function`	Like the sanitize.text.function, but applicable to row names. The default uses the sanitize.text.function.
`sanitize.colnames.function`	Like the sanitize.text.function, but applicable to column names. The default uses the sanitize.text.function.
`...`	Other arguments of the print.xtable function (see xtable package).

Author(s)

José Cláudio Faria (<`joseclaudio.faria@gmail.com`>)
Clarice Garcia Borges Demétrio (<`clarice@esalq.usp.br`>)
and
Ivan Bezerra Allaman (<`ivanalaman@gmail.com`>)

See Also

xtable.bpca; xtable

Examples

```
## Example 1: Principal labels in portuguese
library(xtable)

bp2 <- bpca(gabriel1971)
tbl <- xtable(bp2)
rownames(tbl) <- gsub('Eigenvectors','Autovetores',rownames(tbl))
rownames(tbl) <- c(rownames(tbl)[1:9],'Autovalores','Variância retida','Variância acumulada')
dimnames(tbl)[[2]] <- c('CP 1','CP 2')

print(tbl)

## Example 2: With bold in the column
tbl1 <- xtable(bp2)
bold <- function(x){
  paste('\textbf{',
        x,
        '}')
}

print(tbl1,
      sanitize.colnames.function = bold)

# Example 3: With italic in the rows
tbl2 <- xtable(bp2)
italic <- function(x){
  paste('& \textit{',
        x,
        '}')
} # It is necessary the character "&" to adapt the number of column of the table!

print(tbl2,
      sanitize.rownames.function = italic)
```

---

qbpca *Quality of the Representation of Variables by Biplot*

Description

This function returns an object of the class qbpca. It is a simple measure of the quality of biplot representation of the variables. The observed (in the data) and projected (under biplot reduction) correlations are computed.

Usage

```
qbpca(x,
      bpca)
```

Arguments

`x`	A `data.frame` or `matrix` object.
`bpca`	A object of the class `bpca`.

Details

This function binds the vectors of observed (from the matrix or data.frame) and projected (under biplot reduction) correlations for all variables.

Value

An object of class `qbpca` and `data.frame` with two columns:

`obs`	A vector of the observed correlations for all variables.
`var.rb`	A vector of the projected correlations for all variables determined under biplot reduction).

Author(s)

José Cláudio Faria (<`joseclaudio.faria@gmail.com`>)
Clarice Garcia Borges Demétrio (<`clarice@esalq.usp.br`>)
and
Ivan Bezerra Allaman (<`ivanalaman@gmail.com`>)

References

Johnson, R. A. and Wichern, D. W. (1988) *Applied multivariate statistical analysis*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 6 ed.

See Also

`bpca`

Examples

```
##
## Example 1
## Example of 'var.rb=TRUE' parameter as a measure of the quality of the biplot - 2d
##

oask <- devAskNewPage(dev.interactive(orNone=TRUE))

## Differences between methods of factorization
# SQRT
bp1 <- bpca(gabriel1971,
```

```
            meth='sqrt',
            var.rb=TRUE)

qbp1 <- qbpca(gabriel1971,
              bp1)

plot(qbp1,
     main='sqrt - 2d \n (poor)')


# JK
bp2 <- bpca(gabriel1971,
            meth='jk',
            var.rb=TRUE)

qbp2 <- qbpca(gabriel1971,
              bp2)

plot(qbp2,
     main='jk - 2d \n (very poor)')


# GH
bp3 <- bpca(gabriel1971,
            meth='gh',
            var.rb=TRUE)

qbp3 <- qbpca(gabriel1971,
              bp3)

plot(qbp3,
     main='gh - 2d \n (good)')


# HJ
bp4 <- bpca(gabriel1971,
            meth='hj',
            var.rb=TRUE)

qbp4 <- qbpca(gabriel1971,
             bp4)

plot(qbp4,
     main='hj - 2d \n (good)')

##
## Example 2
## Example of 'var.rb=TRUE' parameter as a measure of the quality of the biplot - 3d
##

## Differences between methods of factorization
# SQRT
bp1 <- bpca(gabriel1971,
```

```
            meth='sqrt',
            d=1:3,
            var.rb=TRUE)

qbp1 <- qbpca(gabriel1971,
              bp1)

plot(qbp1,
     main='sqrt - 3d \n (poor)')


# JK
bp2 <- bpca(gabriel1971,
            meth='jk',
            d=1:3,
            var.rb=TRUE)

qbp2 <- qbpca(gabriel1971,
             bp2)

plot(qbp2,
     main='jk - 3d \n (very poor)')


# GH
bp3 <- bpca(gabriel1971,
            meth='gh',
            d=1:3,
            var.rb=TRUE)

qbp3 <- qbpca(gabriel1971,
              bp3)

plot(qbp3,
     main='gh - 3d \n (whow!)')


# HJ
bp4 <- bpca(gabriel1971,
            meth='hj',
            d=1:3,
            var.rb=TRUE)

qbp4 <- qbpca(gabriel1971,
              bp4)

plot(qbp4,
     main='hj - 3d \n (whow!)')

devAskNewPage(oask)
```

`summary.bpca` *Summary Method for bpca Objects*

Description

Returns (and prints) a summary list for `bpca` (`bpca.2d` and `bpca.3d`) objects.

Usage

```
  ## S3 method for class 'bpca'
summary(object,
        presentation=FALSE, ...)
```

Arguments

`object`	A given object of the class `bpca`, (`bpca.2d` or `bpca.3d`).
`presentation`	Logic. If `TRUE` the summary of the class `bpca`, (`bpca.2d`, `bpca.3d` is showed well formated in the screen, else, return a list. The default is `FALSE`.
`...`	Potential further arguments (require by generic).

Author(s)

José Cláudio Faria (<`joseclaudio.faria@gmail.com`>)
Clarice Garcia Borges Demétrio (<`clarice@esalq.usp.br`>)
and
Ivan Bezerra Allaman (<`ivanalaman@gmail.com`>)

See Also

`bpca`, `xtable.bpca`

Examples

```
##
## Example 1
## bpca - 2d
##
# bpca
bp <- bpca(gabriel1971)
summary(bp)
summary(bp,
        presentation=TRUE)

##
## Example 2
## bpca - 3d
##
bp <- bpca(gabriel1971,
```

```
           d=1:3)

# bpca
sm <- summary(bp)
str(sm)
sm
summary(bp,
        presentation=TRUE)
```

`var.rbf` *Projected Correlations by bpca*

Description

Computes the matrix of graphical correlations represented by biplot for a matrix of variable coordinates.

Usage

```
var.rbf(x)
```

Arguments

`x`	A given object of the classes ‘`bpca.2d`’ and ‘`bpca.3d`’.

Value

A `matrix` of graphical correlations represented by biplot.

Note

This function is mainly for internal use in the **bpca** package, and may not remain available (unless we see a good reason).

Author(s)

José Cláudio Faria (<`joseclaudio.faria@gmail.com`>)
Clarice Garcia Borges Demétrio (<`clarice@esalq.usp.br`>)
and
Ivan Bezerra Allaman (<`ivanalaman@gmail.com`>)

See Also

`bpca`

Examples

```
##
## Particular use
##

bp1 <- bpca(gabriel1971)
bp1$var.rb # NA

# Computes the correlations of all variables under the biplot projection
(res <- var.rbf(bp1$coord$var))

##
## Common use
##

bp2 <- bpca(gabriel1971,
            var.rb=TRUE)

bp2$var.rb
```

`var.rdf` *Diagnostic of Projected Correlations*

Description

Computes the diagnostic of poor graphical correlations projected by biplot according to an arbitrary '`limit`'.

Usage

```
var.rdf(x,
        var.rb,
        limit)
```

Arguments

`x`	A given object of the classe `data.frame` or `matrix`.
`var.rb`	A given object of the class `matrix` with the projected correlations by biplot.
`limit`	A vector giving the percentual limit to define poor representation of variables.

Value

A `data.frame` of poor graphical correlations projected by biplot.

Note

This function is mainly for internal use in the **bpca** package, and may not remain available (unless we see a good reason).

Author(s)

José Cláudio Faria (<`joseclaudio.faria@gmail.com`>)
Clarice Garcia Borges Demétrio (<`clarice@esalq.usp.br`>)
and
Ivan Bezerra Allaman (<`ivanalaman@gmail.com`>)

See Also

`bpca`.

Examples

```
##
## Example 1
## Diagnostic of gabriel1971 dataset representation
##

oask <- devAskNewPage(dev.interactive(orNone=TRUE))

bp1 <- bpca(gabriel1971,
            meth='hj',
            var.rb=TRUE)

(res <- var.rdf(gabriel1971,
                bp1$var.rb,
                lim=3))
class(res)

##
## Example 2
## Diagnostic of gabriel1971 dataset representation with var.rd parameter
##

bp2 <- bpca(gabriel1971,
            meth='hj',
            var.rb=TRUE,
            var.rd=TRUE,
            limit=3)

plot(bp2,
     var.factor=2)

bp2$var.rd

bp2$eigenvectors

# Graphical visualization of the importance of the variables not contemplated
# in the reduction
plot(bpca(gabriel1971,
          meth='hj',
          d=3:4),
     main='hj',
```

```
     xlim=c(-1,1),
     ylim=c(-1,1))

# Interpretation:
# RUR followed by CRISTIAN contains information dimensions that
# wasn't contemplated by the biplot reduction (PC3).
# Between all, RUR followed by CRISTIAN, variables are the most poor represented
# by a 2d biplot.

## Not run:
##
## Example 3
## Diagnostic of iris dataset representation with var.rd parameter
##

bp3 <- bpca(iris[-5],
            var.rb=TRUE,
            var.rd=TRUE,
            limit=3)

plot(bp3,
     obj.col=c('red', 'green3', 'blue')[unclass(iris$Species)],
     var.factor=.3)

bp3$var.rd
bp3$eigenvectors

# Graphical diagnostic
plot(bpca(iris[-5],
          d=3:4),
     obj.col=c('red', 'green3', 'blue')[unclass(iris$Species)],
     obj.names=FALSE,
     var.factor=.6,
     xlim=c(-2,3),
     ylim=c(-1,1))

# Interpretation:
# Sepal.length followed by Petal.Width contains information in dimensions
# (PC3 - the PC3 is, essentially, a contrast among both) that wasn't fully
# contemplated by the biplot reduction (PC1 and PC2) .
# Therefore, between all variables, they have the most poor representation by a
# 2d biplot.

bp4 <- bpca(iris[-5],
            d=1:3,
            var.rb=TRUE,
            var.rd=TRUE,
            limit=2)

plot(bp4,
     obj.names=FALSE,
     obj.pch=c('+', '-', '*')[unclass(iris$Species)],
     obj.col=c('red', 'green3', 'blue')[unclass(iris$Species)],
```

```
     obj.cex=1,
     xlim=c(-5,5),
     ylim=c(-5,5),
     zlim=c(-5,5),
     var.factor=.5)

bp4$var.rd
bp4$eigenvectors

round(bp3$var.rb, 2)

round(cor(iris[-5]), 2)

# Good representation of all variables with a 3d biplot!

## End(Not run)

devAskNewPage(oask)
```

`xtable.bpca` *LaTeX table of the Biplot.*

Description

This function returns a LaTeX table of the bpca objects.

Usage

```
xtable.bpca(x,
            caption = NULL,
            label = NULL,
            align = NULL,
            digits = NULL,
            display = NULL,
            auto = FALSE, ...)
```

Arguments

`x`	A object of the class `bpca` objects.
`caption`	Character vector of length 1 or 2 containing the table's caption or title. If length is 2, the second item is the "short caption" used when LaTeX generates a "List of Tables". Set to NULL to suppress the caption. Default value is `NULL`.
`label`	Character vector of length 1 containing the LaTeX '`\label`' or HTML anchor. Set to `NULL` to suppress the label. Default value is `NULL`.
`align`	Character vector of length equal to the number of columns of the resulting table, indicating the alignment of the corresponding columns. Also, "|" may be used to produce vertical lines between columns in LaTeX tables, but these are effectively ignored when considering the required length of the supplied vector. If a

	character vector of length one is supplied, it is split as strsplit(align, "")[[1]] before processing. Since the row names are printed in the first column, the length of align is one greater than ncol(x) if x is a data.frame. Use "l", "r", and "c" to denote left, right, and center alignment, respectively. Use "p3cm" etc. for a LaTeX column of the specified width. For HTML output the "p" alignment is interpreted as "l", ignoring the width request. Default depends on the class of x.
`digits`	Numeric vector of length equal to one (in which case it will be replicated as necessary) or to the number of columns of the resulting `table` or `matrix` of the same size as the resulting table, indicating the number of digits to display in the corresponding columns. Since the row names are printed in the first column, the length of the vector digits or the number of columns of the matrix digits is one greater than ncol(x) if x is a `data.frame`. Default depends on the class of x. If values of digits are negative, the corresponding values of x are displayed in scientific format with abs(digits) digits.
`display`	Character vector of length equal to the number of columns of the resulting table, indicating the format for the corresponding columns. Since the row names are printed in the first column, the length of display is one greater than ncol(x) if x is a data.frame. These values are passed to the formatC function. Use "d" (for integers), "f", "e", "E", "g", "G", "fg" (for reals), or "s" (for strings). "f" gives numbers in the usual xxx.xxx format; "e" and "E" give n.ddde+nn or n.dddE+nn (scientific format); "g" and "G" put x[i] into scientific format only if it saves space to do so. "fg" uses fixed format as "f", but digits as number of significant digits. Note that this can lead to quite long result strings. Default depends on the class of x.
`auto`	Logical, indicating whether to apply automatic format when no value is passed to align, digits, or display. This ¿autoformat¿ (based on xalign, xdigits, and xdisplay) can be useful to quickly format a typical `matrix` or `data.frame`. Default value is `FALSE`.
`...`	Additional arguments. (Currently ignored.)

Details

This function extracts tabular information from x and returns an object of class "xtable.bpca", "xtable" or "data.frame".

It is necessary to declare the latex packages: ‘`multirow`’ in the preamble of the Rnoweb file to make available all the resources of the function `xtable.bpca`.

Value

An object of the class `xtable.bpca`.

Author(s)

José Cláudio Faria (<`joseclaudio.faria@gmail.com`>)
Clarice Garcia Borges Demétrio (<`clarice@esalq.usp.br`>)
and
Ivan Bezerra Allaman (<`ivanalaman@gmail.com`>)

References

Johnson, R. A. and Wichern, D. W. (1988) *Applied multivariate statistical analysis*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 6 ed.

See Also

`bpca`, `xtable`

Examples

```
## Example 1: The simplest possible
library(xtable)

bp <- bpca(iris[-5],
           d=1:3)

xtable(bp)

## Example 2: With caption and label
bp2 <- bpca(gabriel1971)

xtable(bp2,
       caption='Biplot gabriel1971',
       label='example_2')
```

Index