

Package ‘bpca’

June 12, 2018

Type Package

Title Biplot of Multivariate Data Based on Principal Components Analysis

Version 1.3-0

Date 2018-06-07

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

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 2018-06-11 22:04:46 UTC

R topics documented:

bpca-package	2
bpca	6
dt.tools	11
gabriel1971	13
gge2003	14
latex.bpca	15
marina	21
ontario	22
plot	22
print	33
qbpc	34

summary.bpca	37
summary.latex.bpca	38
var.rbf	40
var.rdf	41

Index	45
--------------	-----------

bpca-package	<i>Biplot of Multivariate Data Based on Principal Components Analysis</i>
--------------	---

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. *Questiío*, 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

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

<code>var.rb</code>	A logical value. If TRUE, all correlation coefficients for all variables (<i>under the biplot projection</i>) will be computed.
<code>var.rd</code>	A logical value. If TRUE, the diagnostic of the representation of variables projected by the biplot will be computed. If <code>var.rd</code> is TRUE the <code>var.rb</code> parameter must be also TRUE.
<code>limit</code>	A vector giving the percentual limit to define poor representation of variables.
<code>...</code>	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

$${}_n Y_p$$

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

$$svd({}_n Y_p)$$

- 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,

$$s^2 = \text{diag}(d)$$

$$n = n_objects(Y)$$

it is possible an approximation of Y :

$${}_n Y_p \approx Y_m = g.h'$$

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

$${}_n Y_p \approx Y_m$$

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

$${}_n Y_p$$

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 regenerate, or computes, the inverse of the svd decomposition of the given data

$${}_n Y_p$$

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:

<code>call</code>	The call used.
<code>eigenvalues</code>	A vector of the eigenvalues.
<code>eigenvectors</code>	A vector of the eigenvectors.
<code>numer</code>	A vector of the number of eigenvalues considered in the reduction.
<code>importance</code>	A matrix with the <i>general</i> and <i>partial</i> variation explained by the reduction.
<code>coord</code>	A list with the coordinates of the two components: objects and variables.
<code>var.rb</code>	A matrix of all correlation coefficients for all variables under the biplot projection.
<code>var.rd</code>	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. *Qüestió*, 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 plotting 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)

```

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))), 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" "electricity" ...
..$ : 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 didactic 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)
```

Description

This function returns a LaTeX table of the bpca objects.

Usage

```

latex.bpca(x,
           round=4,
           where='!tbp',
           caption=NULL,
           label=NULL,
           size='',
           v.retained='Variance retained (%)',
           v.partial='Partial',
           v.accumulated='Accumulated',
           eigenvalues='Eigenvalues',
           eigenvectors='Eigenvectors',
           algtable='\\flushleft',
           alg1='l',
           alg2='l',
           algnumbers='r',
           algheader='c',
           algsubheader='c',
           pc.label='PC',
           bpca.label=NULL,
           ft.variable='',
           ft.components='',
           hline1='\\hline',
           hline2='\\hline',
           footnote='')

```

Arguments

x	A object of the class <code>bpca</code> or a list of the <code>bpca</code> objects.
round	The numbers of decimal places to be used. The default is '4'.
where	Specifies the location to which the floating body can move. The default is '!tbp'. Possible values are the tabular environment same, e.g., 'h', 'b' and 'p'.
caption	Is a legend of table. The default is NULL.
label	A text string representing a symbolic label for the table for referencing in the LaTeX '\label' and '\ref' commands. 'label' is useful in a Rnoweb document only if caption is also provided.
size	Specifies the font size of the table. The default is empty. The possible values are the same size used for letters in latex, e.g., '\\scriptsize', '\\large' and '\\LARGE'.
v.retained	The name of the variance retained to be labeled in the table. The default is 'Variance retained'.
v.partial	The name of the variance partial to be labeled in the table. The default is 'Partial'.
v.accumulated	The name of the variance accumulated to be labeled in the table. The default is 'Accumalated'.

eigenvalues	The name of eigenvalues to be labeled in the table. The default is 'Eigenvalues'.
eigenvectors	The name of eigenvectors to be labeled in the table. The default is 'Eigenvectors'.
algtable	Specifies the alignment of the table on page. The default is '\\flushleft'. Possible values are: '\\flushleft', '\\centering' and '\\flushright'.
alg1	Specifies the alignment of the first column. The default is 'l'. Possible values are the tabular environment same, e.g., 'l', 'r' and 'c'.
alg2	Specifies the alignment of the second column. The default is 'l'. Possible values are the tabular environment same, e.g., 'l', 'r' and 'c'.
algnumbers	Specifies the alignment of the numbers column. The default is 'r'. Possible values are the tabular environment same, e.g., 'l', 'r' and 'c'.
algheader	Specifies the alignment of the name main column header. The default is 'c'. Possible values are the tabular environment same, e.g., 'l', 'r' and 'c'.
algsubheader	Specifies the alignment of the name of the bpca objects. It only works where there is a list of the bpca objects. The default is 'c'. Possible values are the tabular environment same, e.g., 'l', 'r' and 'c'.
pc.label	The name of principal components to be labeled in the table. The default is 'PC'.
bpca.label	The labels of the bpca objects. Must be used only with a list of the bpca objects.
ft.variable	'bold' or 'italic' variable names? The default is empty.
ft.components	'bold' or 'italic' components names? The default is empty.
hline1	The line type of the table top. The default is '\\hline'.
hline2	The line type of the lower table header. The default is '\\hline'.
footnote	To insert footer in the table. The default is empty.

Details

It is necessary to declare the latex packages: 'threeparttable', 'multirow', 'array' and 'colortbl' in the preamble of the Rnoweb file to make available all the resources of the function latex.bpca.

The function latex.bpca was developed to make the life easier for those who make latex tables with the results of the bpca function.

Some people could ask: why do not use the function latex of the package Hmisc or xtable of the xtable? Both latex and xtable functions are a bit complex and with many parameters for construct an adequate bpca table. It is not always intuitive to all users, discouraging some people to use the latex or xtable function for constructing bpca tables, making the preparation of the bpca analysis reports data laborious.

The function latex.bpca is extremely easy to use. Obviously, the function provides a formatting pre-set according to what the authors think ideal for presentation in articles, reports, among others. If the user is not satisfied with the formatting provided by this function, the functions latex and xtable provides arguments that allow formatting of tables according to user need.

Value

An object of the class latex.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](#), [latex](#), [xtable](#)

Examples

```
#Example 1: The simplest possible
bp <- bpca(iris[-5],
          d=1:2)

latex.bpca(bp)

## Not run:
#Example 2: with caption
bp2 <- bpca(gabriel1971)

latex.bpca(bp2,
           caption='Biplot gabriel1971')

# Example 3: Almost worked
latex.bpca(bp,
           caption='Biplot da base de dados iris',
           v.partial='Parcial',
           v.accumulated='Acumulada',
           eigenvalues='Autovalores',
           eigenvectors='Autovetores',
           v.retained='Var. retida')

# Example 4: Changing the column first alignment
bp3 <- bpca(gge2003,
           d=1:3)

latex.bpca(bp3,
           caption='Biplot gge2003. Changing the column first alignment',
           alg1='r')

# Example 5: Changing the column second alignment
latex.bpca(bp3,
           caption='Biplot gge2003. Changing the column second alignment',
           alg2='>{\raggedright}p{0.1cm}')
```

```

# Example 6: Changing the column alignment with numbers
latex.bpca(bp3,
  caption='Biplot gge2003. Changing the column alignment with numbers',
  alg1='r',
  alg2='r',
  algnumbers='>{\raggedleft}p{2.2cm}')

# Example 7: Changing the header alignment
latex.bpca(bp3,
  caption='Biplot gge2003. Changing the header alignment',
  algheader='r')

# Example 8: Changing the table alignment
latex.bpca(bp3,
  caption='Biplot gge2003. Changing the table alignment',
  algtable='\\flushright')

# Example 9: I want only two places decimal
latex.bpca(bp3,
  round=2,
  caption='Biplot gge2003',
  algnumbers='>{\centering}p{2.2cm}',
  pc.label='Principal Component-')

# Example 10: With bold in the header, subheader and variables
latex.bpca(bp3,
  round=2,
  caption='Biplot gge2003',
  eigenvalues='\\textbf{Eigenvalues}',
  eigenvectors='\\textbf{Eigenvectors}',
  v.retained='\\textbf{Variance retained (%)}',
  v.partial='\\textbf{Partial}',
  v.accumulated='\\textbf{Accumulated}',
  ft.variable='bold',
  ft.components='bold')

# Example 11: The font of table is very large
latex.bpca(bp3,
  round=2,
  caption='Biplot gge2003',
  pc.label='\\textbf{Principal Component-}',
  size='\\tiny')

# Example 12: Italic in the variables names
latex.bpca(bp2,
  round=2,
  caption='Biplot gabriel1971',
  pc.label='\\textbf{Principal Component-}',
  algnumbers='>{\centering}p{2.5cm}',
  ft.variable='italic')

```

```
# Example 13: With footnote???
latex.bpca(bp2,
           round=2,
           caption='Biplot gabriel1971',
           footnote='\\scriptsize $^1$ Example with footnote')

# Example 14: Over a footer
latex.bpca(bp2,
           round=2,
           size='\\tiny',
           caption='Biplot gabriel1971',
           footnote='\\scriptsize $^1$ Example one.\\newline
\\item $^2$ Example two.')

# Example 15: With others principal components
bp4 <- bpca(gabriel1971,
            d=2:4)

latex.bpca(bp4,
           round=2,
           caption='Biplot gabriel1971')

# Example 16: More than one bpca objects
data(marina)

y_2007 <- subset(marina,
                 year==2007)

y_2008 <- subset(marina,
                 year==2008)

y_2009 <- subset(marina,
                 year==2009)

bp_2007 <- bpca(y_2007[, -c(1:2)],
               d=1:3)

bp_2008 <- bpca(y_2008[, -c(1:2)],
               d=1:2)

bp_2009 <- bpca(y_2009[, -c(1:2)],
               d=1:2)

latex.bpca(list(bp_2007,
                bp_2008),
           round=2,
           caption='Biplot Marina',
           size='\\scriptsize')

# Example 17: With two lines in the table
latex.bpca(list(bp_2007,
                bp_2008,
                bp_2009),
```

```

round=4,
caption='Biplot Marina',
pc.label=c('2007', '2008', '2009'),
size='\\scriptsize',
hline1='\\hline \\hline',
hline2='\\hline \\hline',
algnumbers='>{\\raggedleft}p{1.3cm}',
footnote='Note: F - Movie; D - Doc.; DH - Doc. dir. by men; DF - Doc. dir. by women. ')

## End(Not run)

```

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 'qbpc'
plot(x,
      xlab='Index',
      ylab='r',
      pch=c(1,8),
      col=c(4,2), ...)

```

Arguments

x	A bpca. 2d, bpca. 3d or qbpc 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' - discriminativeness vs. representativeness; 'ms' - mean vs. stability; 'ro' - rank objects with ref. to an "ideal" variable; 'rv' - rank variables with ref. to an "ideal" object.

<code>c.color</code>	The color of circles.
<code>c.lwd</code>	The lwd of circles.
<code>c.number</code>	The number of circles.
<code>c.radio</code>	The radio of circles.
<code>obj.id</code>	An object(s) number(s) or name(s). It is used with reprojections to identify the object(s) when the type option is 'eo' or 'co'. The default is '1' for 'eo' and 1:2 for 'co'.
<code>var.id</code>	A variable number or name. It is used with reprojections to identify the variable when the 'type' option is 'ev'.
<code>base.color</code>	The color for the base lines. It is used only with reprojections.
<code>base.lty</code>	The 'lty' for the base lines. It is used only with reprojections.
<code>proj.color</code>	The color for the projections lines. It is used only with reprojections.
<code>proj.lty</code>	The 'lty' for the projections lines. It is used only with reprojections.
<code>a.color</code>	The color for the arrow. It is used only with reprojections.
<code>a.lty</code>	The 'lty' for the arrow. It is used only with reprojections.
<code>a.lwd</code>	The 'lwd' for the arrow. It is used only with reprojections.
<code>a.length</code>	The 'length' for the arrow. It is used only with reprojections.
<code>rgl.use</code>	A logical value. If TRUE the 3d scatter will be done with the rgl environment, in another way the scatterplot3d will be used.
<code>ref.lines</code>	A logical value. If TRUE, the reference lines will be added to the biplot.
<code>ref.color</code>	Line color for reference lines.
<code>ref.lty</code>	Line type of the reference lines.
<code>clear3d</code>	A logical value. It clears (or not) a 3d biplot before making a new one. Used only if <code>rgl.use=TRUE</code> .
<code>simple.axes</code>	A logical value to draw simple axes. Used only if <code>rgl.use=TRUE</code> .
<code>aspect</code>	A vector of the apparent ratios of the 'x', 'y', and 'z' axes of the bounding box. Used only if <code>rgl.use=TRUE</code> .
<code>var.factor</code>	Factor of expansion/reduction of length lines of the variables.
<code>var.color</code>	A vector of colors for the variables representation.
<code>var.lty</code>	Line type for the variables. Used only if <code>rgl.use=FALSE</code> .
<code>var.pch</code>	A vector of plotting symbols or characters for the variables. If too short, the values are recycled. Used only if <code>rgl.use=FALSE</code> .
<code>var.pos</code>	Position of labels for the variables.
<code>var.cex</code>	Character expansion for the variables.
<code>var.offset</code>	The distance (in character widths) which separates the label from identified points of variables. Negative values are allowed. Used only if <code>rgl.use=FALSE</code> .
<code>obj.factor</code>	Factor of expansion/reduction of length lines of the objects.
<code>obj.color</code>	A vector of colors for the objects representation.
<code>obj.pch</code>	A vector of plotting symbols or characters for objects. If too short, the values are recycled. Used only if <code>rgl.use=FALSE</code> .

<code>obj.pos</code>	Position of labels for objects.
<code>obj.cex</code>	Character expansion for objects.
<code>obj.offset</code>	The distance (in character widths) which separates the label from identified points of objects. Negative values are allowed. Used only if <code>rgl.use=FALSE</code> .
<code>obj.names</code>	A logical value to represent objects as spheres or points.
<code>obj.identify</code>	A logical value. If TRUE, it is possible, in interactive way, to identify the objects with the mouse.
<code>obj.labels</code>	A vector of labels for objects.
<code>box</code>	A logical value to whether to draw a box. Used only if <code>'rgl.use=TRUE'</code> .
<code>angle</code>	Angle between 'x' and 'y' axis (Attention: result depends on scaling). For $180 < angle < 360$ the returned functions <code>xyz.convert</code> and <code>points3d</code> will not work properly. Used only if <code>rgl.use=FALSE</code> .
<code>pch</code>	A vector of plotting symbols or characters.
<code>col</code>	A vector of colors.
<code>xlab</code>	A label for the 'x' axis.
<code>ylab</code>	A label for the 'y' axis.
<code>zlab</code>	A label for the 'z' axis (bPCA.3d only).
<code>xlim</code>	The 'x' limits of the plot.
<code>ylim</code>	The 'y' limits of the plot.
<code>zlim</code>	The 'z' limits of the plot (bPCA.3d only).
<code>...</code>	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.
- 'dynamic' - 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. *Qüestió*, 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 plotting 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

Print Method for latex.bpca Objects

Description

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

Usage

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

Arguments

`x` A given object of the class `latex.bpca`.
`...` 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

[latex.bpca](#)

Examples

```
##
## Example 1
## one bpca objects
##

bp <- bpca(gabriel1971)

lbp <- latex.bpca(bp)

lbp

##
```

```
## Example 2
## two bpca objects
##
y_2007 <- subset(marina,
                 year==2007)

y_2008 <- subset(marina,
                 year==2008)

bp_2007 <- bpca(y_2007[, -c(1:2)],
                d=1:3)

bp_2008 <- bpca(y_2008[, -c(1:2)],
                d=1:2)

lbp <- latex.bpca(list(bp_2007,
                       bp_2008))

lbp
```

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  
# SQRTE  
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 formatted 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](#), [latex.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)
```

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

Description

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

Usage

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

Arguments

object A given object of the class `latex.bpca`.
... 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](#), [latex.bpca](#)

Examples

```
##  
## Example 1  
## bpca - 2d  
##  
# bpca  
bp <- bpca(gabriel1971)  
  
lbp <- latex.bpca(bp)  
summary(lbp)  
  
##  
## Example 2  
## bpca - 3d  
##  
lbp <- latex.bpca(bpca(gabriel1971,  
                      d=1:3))  
  
summary(lbp)  
  
##  
## Example 3  
## two bpca objects  
##  
y_2007 <- subset(marina,  
                  year == 2007)  
  
y_2008 <- subset(marina,  
                  year == 2008)  
  
bp_2007 <- bpca(y_2007[, -c(1:2)],  
                d=1:3)  
  
bp_2008 <- bpca(y_2008[, -c(1:2)],  
                d=1:2)
```

```
lbp <- latex.bpca(list(bp_2007,  
                      bp_2008))  
  
summary(lbp)
```

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)
```

Index

- *Topic **bpca**
 - summary.bpca, 37
 - summary.latex.bpca, 38
 - *Topic **datasets**
 - gabriel1971, 13
 - gge2003, 14
 - marina, 21
 - ontario, 22
 - *Topic **latex.bpca**
 - summary.latex.bpca, 38
 - *Topic **movie**
 - marina, 21
 - *Topic **multivariate, table, latex, bpca**
 - latex.bpca, 15
 - *Topic **multivariate**
 - bpca, 6
 - bpca-package, 2
 - dt.tools, 11
 - plot, 22
 - print, 33
 - qbpca, 34
 - summary.bpca, 37
 - var.rbf, 40
 - var.rdf, 41
 - *Topic **package**
 - bpca-package, 2
 - *Topic **summary**
 - summary.bpca, 37
 - summary.latex.bpca, 38
- biplot.prcomp, 9, 27
bpca, 6, 12, 16–18, 34, 35, 38–40, 42
bpca-package, 2
- dt.tools, 11
- gabriel1971, 13
gge2003, 14
- latex, 17, 18
- latex.bpca, 15, 33, 38, 39
- marina, 21
- ontario, 22
- plot, 22
print, 33
- qbpca, 34
- summary.bpca, 37
summary.latex.bpca, 38
- var.rbf, 40
var.rdf, 41
- xtable, 17, 18