Package ‘biplotEZ’

November 28, 2023

Title EZ-to-Use Biplots
Version 1.2.0
Description Provides users with an EZ-to-use platform for representing data with biplots. Currently principal component analysis (PCA) and canonical variate analysis (CVA) biplots are included. This is accompanied by various formatting options for the samples and axes. Alpha-bags and concentration ellipses are included for visual enhancements and interpretation. For an extensive discussion on the topic, see Gower, J.C., Lubbe, S. and le Roux, N.J. (2011, ISBN: 978-0-470-01255-0) Understanding Biplots. Wiley: Chichester.
License MIT + file LICENSE
Encoding UTF-8
RoxygenNote 7.2.3
Imports ggplot2, ggrepel, grDevices, grid, withr
Suggests knitr, rmarkdown
VignetteBuilder knitr
NeedsCompilation no
Author Sugnet Lubbe [aut, cre] (<https://orcid.org/0000-0003-2762-9944>), Niël le Roux [aut] (<https://orcid.org/0000-0002-1791-746X>), Johané Nienkemper-Swanepoel [aut] (<https://orcid.org/0000-0001-6086-8272>), Raeesa Ganey [aut] (<https://orcid.org/0009-0008-6973-0999>), Carel van der Merwe [aut] (<https://orcid.org/0000-0003-0676-8240>)
Maintainer Sugnet Lubbe <muvisu@sun.ac.za>
Repository CRAN
Date/Publication 2023-11-28 10:30:02 UTC

R topics documented:

alpha.bags .............................................................. 2
axes ........................................................................... 3
biplot ......................................................................... 5
Description

This function produces α-bags, which is a useful graphical summary of the scatter plot. The alpha-bag refers to a contour which contains α% of the observations.

Usage

alpha.bags(bp, alpha = 0.95, which = NULL, col = bp$sample$col[which], lty = 1, lwd = 1, max = 2500, trace = TRUE)

Arguments

bp an object of class biplot.
alpha value between 0 and 1 to determine coverage of bag (α), with default 0.95.
which the selection of groups or classes to be fitted with α-bags.
col vector of colours for the α-bags. Multiple α bags for one group will be displayed in the same colour.
lty vector of line types for the α-bags. The same line type will be used per value of α.
lwd vector of line widths for the α-bags. The same line width will be used per value of α.
max maximum number of samples to include in α-bag calculations, with default 2500. If more samples are in the group, a random sample of size max is taken for the computations.
trace logical, indicating progress of computation.
Value

A list with the following components is available:

- `alpha.bags` list of coordinates for the \( \alpha \)-bags for each group.
- `col` vector of colours for the \( \alpha \)-bags.
- `lty` vector of line types for the \( \alpha \)-bags.
- `lwd` vector of line widths for the \( \alpha \)-bags.

References


Examples

```r
biplot (iris[,1:4]) |> PCA(group.aes=iris[,5]) |> alpha.bags(alpha=0.95) |> plot()
biplot (iris[,1:4],group.aes=iris[,5]) |> PCA() |> alpha.bags(alpha=0.95) |> plot()
```

axes

Aesthetics for biplot axes

Description

This function allows formatting changes to axes.

Usage

```r
axes(bp, X.names=colnames(bp$X), which = 1:bp$p, col = grey(0.7),
    lwd = 1, lty = 1, label.dir = "Orthog", label.col = col, label.cex = 0.75,
    label.line = 0.1, ticks = 5, tick.col = col, tick.size = 1, tick.label = TRUE,
    tick.label.col = tick.col, tick.label.cex = 0.6, tick.label.side = "left",
    tick.label.offset = 0.5, tick.label.pos = 1, predict.col = col, predict.lwd = lwd,
    predict.lty = lty, ax.names = X.names, orthogx = 0, orthogy = 0)
```

Arguments

- `bp` an object of class `biplot`.
- `X.names` refers to the column names of bp to specify which axes to label.
- `which` vector of columns to be displayed in the biplot, with default `1:bp$p`.
- `col` axis colour, with default `grey(0.7)`.
- `lwd` axis line width, with default `1`.
- `lty` axis line type, with default `1`.
- `label.dir` direction of axis label, with default `Orthog`.
- `label.col` axis label colour, with default `col`.
axes

label.cex axis label expansion, with default 0.75.
label.line axis label written on which margin line, with default 0.1.
ticks number of tick marks per axis, with default 5.
tick.col tick mark colour, with default col.
tick.size tick mark size, with default 1.
tick.label logical, whether axes should be labelled or not, with default TRUE.
tick.label.col tick mark label colour, with default tick.col.
tick.label.cex tick mark label expansion, with default 0.6.
tick.label.side side of the tick mark label, with default left.
tick.label.offset tick mark label offset, with default 0.5.
tick.label.pos side of the tick mark label, with default below.
predict.col predicted samples colour, with default col.
predict.lwd predicted samples line width, with default lwd.
predict.lty predicted samples line type, with default lty.
ax.names vector of size p containing user defined names for the variables.
orthogx the horizontal translation, with default 0.
orthogy the vertical translation with default 0.

Value

A list with the following components is available:

which vector of the columns displayed as axes.
col vector of axis colours.
lwd vector of axis line widths.
lty vector of axis line types.
label.dir direction of the axis labels.
label.col vector of axis label colours.
label.cex vector of axis labels expansions.
label.line vector of axis label margin lines from axes.
ticks vector representing the number of tick marks per axis.
tick.col vector of tick mark colours.
tick.size vector of tick mark sizes.
tick.label vector of logical values indicating whether axes are labelled.
tick.label.col vector of tick mark label colours.
tick.label.cex vector of tick mark label expansions.
tick.label.side vector of the side of tick mark labels.
biplot

Examples

```r
biplot(iris[,1:4]) |> PCA() |> axes(col="purple") |> plot()
biplot(iris[,1:4]) |> PCA() |> samples(col="purple",pch=15) |> axes() |> plot()
```

Description

This function produces a list of elements to be used when producing a biplot, which provides a useful data analysis tool and allows the visual appraisal of the structure of large data matrices. Biplots are the multivariate analogue of scatter plots. They approximate the multivariate distribution of a sample in a few dimensions and they superimpose on this display representations of the variables on which the samples are measured.

Usage

```r
biplot(data, classes = NULL, group.aes = NULL, center = TRUE, scaled = FALSE, Title = NULL)
```

Arguments

data

a dataframe or matrix containing all variables the user wants to analyse.

classes

vector identifying class membership.

group.aes

vector identifying groups for aesthetic formatting.

center

logical, indicating whether data should be column centered, with default TRUE.

scaled

logical, indicating whether data should be standardized to unit column variances, with default FALSE.

Title

title of the biplot to be rendered, enter text in " ".

Value

A list with the following components is available:

- \( X \) matrix of the centered and scaled numeric variables.
- \( \text{Xcat} \) data frame of the categorical variables.
- \( \text{raw.X} \) original data.
- \( \text{center} \) TRUE or FALSE, whether \( X \) is centered.
- \( \text{scaled} \) TRUE or FALSE, whether \( X \) is scaled.
- \( \text{means} \) vector of means for each numeric variable.
- \( \text{sd} \) vector of standard deviations for each numeric variable.
- \( \text{group.aes} \) vector of category levels for the grouping variable. This is to be used for colour, pch and cex specifications.
- \( \text{Title} \) title of the biplot to be rendered

References


Examples

```r
biplot(data = iris)
# create a PCA biplot
biplot(data = iris) |> PCA() |> plot()
```

---

CVA  

**Canonical Variate Analysis (CVA) method**

Description

This function produces a list of elements to be used for CVA biplot construction.

Usage

```r
CVA(bp, dim.biplot = c(2, 1, 3), e.vects = 1:ncol(bp$X),
    classes=bp$classes, weightedCVA = "weighted", show.class.means = TRUE)
```
Arguments

- **bp**: object of class `biplot` obtained from preceding function `biplot()`.
- **dim.biplot**: dimension of the biplot. Only values 1, 2 and 3 are accepted, with default 2.
- **e.vects**: which eigenvectors (canonical variates) to extract, with default `1:dim.biplot`.
- **classes**: vector of the same length as the number of rows in the data matrix with the class indicator for the samples.
- **weightedCVA**: the default is "weighted", specifying a weighted CVA to be performed. Other possible values are "unweightedI" and "unweightedCent".
- **show.class.means**: logical, indicating whether to plot the class means on the biplot.

Value

Object of class CVA with the following elements:

- **X**: matrix of the centered and scaled numeric variables.
- **Xcat**: matrix of the categorical variables.
- **raw.X**: original data.
- **na.action**: vector of observations that have been removed.
- **center**: TRUE or FALSE, whether X is centred.
- **scaled**: TRUE or FALSE, whether X is scaled.
- **means**: mean of each numerical variable.
- **sd**: standard deviation of each numerical variable.
- **n**: number of observations.
- **p**: number of variables.
- **group.aes**: vector of the same length as the number of rows in the data matrix with differentiated aesthetics for samples.
- **g.names**: descriptive name to be used for group labels.
- **g**: number of groups.
- **Title**: title of the biplot to be rendered.
- **Gmat**: indicator matrix for class membership.
- **Xmeans**: matrix of class means.
- **Z**: matrix with each row containing the details of the point to be plotted (i.e. coordinates).
- **Zmeans**: matrix of canonical means.
- **Lmat**: matrix for transformation to the canonical space.
- **eigenvalues**: vector of eigenvalues of the two-sided eigenvalue problem.
- **Cmat**: Centring matrix based on different choices of weighting. For "weighted", `Cmat` is a diagonal matrix with the class sizes, for "unweightedI", `Cmat` is an indicator matrix and for "unweightedCent", `Cmat` is the usual centring matrix.
- **Bmat**: Between class sums of squares and cross products matrix.
CVA.biplot

Wmat  Within class sums of squares and cross products matrix.
e.vects  vector indicating which canonical variates are plotted in the biplot.
ax.one.unit  one unit in the positive direction of each biplot axis.
class.means  logical value, indicating whether the class means should be plotted in the biplot.
Zmeans  matrix of the class mean coordinates to be plotted in the biplot.

Examples

biplot(iris[,1:4]) |> CVA(classes=iris[,5])
# create a CVA biplot
biplot(iris[,1:4]) |> CVA(classes=iris[,5]) |> plot()

CVA.biplot

CVA biplot

Description

Performs calculations for a CVA biplot.

Usage

## S3 method for class 'biplot'
CVA(
  bp,
  dim.biplot = c(2, 1, 3),
  e.vects = 1:ncol(bp$X),
  classes = bp$classes,
  weightedCVA = "weighted",
  show.class.means = TRUE
)

Arguments

bp  object of class biplot obtained from preceding function biplot().
dim.biplot  dimension of the biplot. Only values 1, 2 and 3 are accepted, with default 2.
e.vects  which eigenvectors (canonical variates) to extract, with default 1:dim.biplot.
classes  vector of the same length as the number of rows in the data matrix with the class indicator for the samples.
weightedCVA  the default is "weighted", specifying a weighted CVA to be performed. Other possible values are "unweightedI" and "unweightedCent".
show.class.means  logical, indicating whether to plot the class means on the biplot.

Value

an object of class CVA, inherits from class biplot.
Examples

biplot(iris[,1:4]) |> CVA(classes=iris[,5])

---

descriptions

Concentration ellipses ($\kappa$-ellipses)

Description

This function produces $\kappa$-ellipses, which is a useful geometrical description of the data points about the sample mean.

Usage

ellipses(bp, df=2, kappa = NULL, which = NULL, alpha = 0.95, col = bp$sample$col[which], lty = 1, lwd = 1, alpha.transparency = 0.5, trace = TRUE)

Arguments

bp an object of class biplot.
df degrees of freedom, with default 2.
kappa value to construct $\kappa$-ellipse (the value of $\kappa$).
which the selection of the group for ellipse construction.
alpha size of $\alpha$-bag, with default 0.95.
col colour of ellipse. Multiple $\kappa$-ellipse for one group will be displayed in the same colour.
lty line type of ellipse. The same line type will be used per value of $\kappa$.
lwd line width of ellipse. The same line width will be used per value of $\kappa$.
alpha.transparency level of opacity, with default 0.5.
trace logical, indicating progress of computation.

Value

A list with the following components is available:

conc.ellipses list of coordinates for the $\kappa$-ellipses for each group.
col vector of colours for the $\kappa$-ellipses.
lty vector of line types for the $\kappa$-ellipses.
lwd vector of line widths for the $\kappa$-ellipses.
alpha vector of $\alpha$ values.
References


Examples

biplot(iris[,1:4]) |> PCA(group.aes=iris[,5]) |> ellipses(kappa=2) |> plot()

fit.measures

Compute measures of fit for biplot.

Description

Compute measures of fit for biplot.

Usage

fit.measures(bp)

Arguments

bp an object of class biplot.

Value

An object of class biplot. The object is augmented with additional items, depending on the type of biplot object.

quality overall quality of fit.

adequacy adequacy of representation of variables.

For an object of class PCA:

axis.predictivity fit measure of each individual axis.

sample.predictivity fit measure for each individual sample.

For an object of class CVA:

axis.predictivity fit measure of each individual axis.

class.predictivity fit measure for each class mean.

within.class.axis.predictivity fit measure for each axis based on values expressed as deviations from their class means.

within.class.sample.predictivity fit measure for each sample expressed as deviation from its class mean.
Examples

```r
out <- biplot(iris[,1:4]) |> PCA() |> fit.measures()
summary(out)
```

**interpolate**

Interpolation of new samples

**Description**

Interpolation of new samples

**Usage**

```r
interpolate(bp, newdata)
```

**Arguments**

- `bp` an object of class `biplot` obtained from preceding function `biplot()`.
- `newdata` a new data set, similar in structure to the data set supplied to `biplot()` containing supplementary data points to be added on the biplot.

**Value**

Object of class PCA with the following elements:

- `X` matrix of the centered and scaled numeric variables.
- `Xcat` matrix of the categorical variables.
- `raw.X` original data.
- `na.action` vector of observations that have been removed.
- `center` TRUE or FALSE, whether X is centred.
- `scaled` TRUE or FALSE, whether X is scaled.
- `means` mean of each numerical variable.
- `sd` standard deviation of each numerical variable.
- `n` number of observations.
- `p` number of variables.
- `group.aes` vector of the same length as the number of rows in the data matrix for differentiated aesthetics for samples.
- `g.names` descriptive name to be used for group labels.
- `g` number of groups.
- `Title` title of the biplot to be rendered
- `Z` matrix with each row containing the details of the point to be plotted (i.e. coordinates).
**Lmat**
matrix for transformation to the principal components.

**e.vects**
vector indicating which principal components are plotted in the biplot.

**ax.one.unit**
one unit in the positive direction of each biplot axis.

**Xnew.raw**
newdata numerical variables.

**Xnew**
matrix of the centered and scaled new numeric variables.

**Xnew.cat**
matrix of the new categorical variables.

**Znew**
matrix of the coordinates of the newdata in the biplot.

---

**Examples**

```r
biplot(data = iris[1:145,]) |> PCA() |> interpolate(newdata = iris[146:150,]) |> plot()
```

---

**legend.type**

<table>
<thead>
<tr>
<th>Legend type</th>
</tr>
</thead>
<tbody>
<tr>
<td>legend.type</td>
</tr>
</tbody>
</table>

**Description**

This function enables the user to format the legend and make a required selection to display.

**Usage**

```r
legend.type(bp, samples = FALSE, means = FALSE, bags = FALSE, ellipses = FALSE, new = FALSE, ...)
```

**Arguments**

- **bp**
an object of class `biplot`.
- **samples**
logical, indicating whether legend should be printed for samples, with default `FALSE`.
- **means**
logical, indicating whether legend should be printed for means, with default `FALSE`.
- **bags**
logical, indicating whether legend should be printed for bags, with default `FALSE`.
- **ellipses**
logical, indicating whether legend should be printed for concentration ellipses, with default `FALSE`.
- **new**
logical, indicating whether the legend should appear in a new window, with default `FALSE`.
- **...**
additional arguments to be sent to `legend()`.
Value

A list with the following components is available:

- **samples**: TRUE or FALSE, whether legend for samples are provided.
- **means**: TRUE or FALSE, whether legend for class means are provided.
- **bags**: TRUE or FALSE, whether legend for alpha-bags are provided.
- **ellipses**: TRUE or FALSE, whether legend for kappa-ellipses are provided.
- **new**: TRUE or FALSE, whether legend appears on new plot.

Examples

```r
biplot(iris[,1:4], Title="Test biplot") |> PCA(group.aes = iris[,5]) |> 
legend.type(samples=TRUE) |> plot()
```

---

Aesthetics for biplot class / group means

**Description**

This function allows formatting changes to class means or group means.

**Usage**

```r
means(bp, which = NULL, col = NULL, pch = 1, cex = 1, label = FALSE, 
label.col = NULL, label.cex = 0.75, label.side = "bottom", label.offset = 0.5)
```

**Arguments**

- **bp**: an object of class `biplot`.
- **which**: vector of which means to display, with default `bp$g`.
- **col**: mean colour, with default to sample colour.
- **pch**: mean plotting character, with default `o`.
- **cex**: mean character expansion, with default 1.
- **label**: logical, whether means should be labelled or not, with default TRUE.
- **label.col**: vector of length `g` with the colour of the labels, defaulting to the colour of the means.
- **label.cex**: label text expansion, with default 0.75.
- **label.side**: side of the plotting character where label appears, with default bottom. Note that unlike the argument `pos` in `text()`, options are "bottom", "left", "top", "right" and not 1, 2, 3, 4.
- **label.offset**: offset of the label from the mean point. See `?text` for a detailed explanation of the argument `offset`.
Details

The number of classes or groups (defined by group.aes) is indicated as g. If an argument is not of length g, recycling is used.

Value

A list with the following components is available:

- `which`: which means to display.
- `col`: colour of the means.
- `pch`: plotting character of the means.
- `cex`: expansion of the plotting character of the means.
- `label`: logical, whether means should be labelled.
- `label.col`: colour of the label.
- `label.cex`: expansion of the label.
- `label.side`: side at which to plot the label of means.
- `label.offset`: offset of the label from the mean point.

Examples

```r
biplot(iris[,1:4]) |> PCA() |>
  means(col = "purple", pch = 15, cex = 2) |> plot()
```

newsamples

Aesthetics for supplementary (new) biplot samples

Description

This function allows formatting changes to new samples.

Usage

```r
newsamples (bp, col = "darkorange1", pch = 1, cex = 1, label = FALSE,
  label.col = NULL, label.cex = 0.75, label.side = "bottom", label.offset = 0.5,
  connected = FALSE, connect.col = "black", connect.lty=1, connect.lwd=1)
```

Arguments

- `bp`: an object of class biplot.
- `col`: new sample colour, with default darkorange1.
- `pch`: new sample plotting character, with default o.
- `cex`: new sample character expansion, with default 1.
- `label`: logical, whether samples should be labelled or not, with default FALSE.
newsamples

label.col vector of length number of new samples with the colour of the labels, defaulting to the colour of the sample points.

label.cex label text expansion, with default 0.75.

label.side side of the plotting character where label appears, with default bottom. Note that unlike the argument pos in text(), options are "bottom", "left", "top", "right" and not 1, 2, 3, 4.

label.offset offset of the label from the data point. See ?text for a detailed explanation of the argument offset.

classed logical, whether samples are connected in order of rows of data matrix, with default FALSE.

connect.col colour of the connecting line, with default black.

connect.lty line type of the connecting line, with default 1.

connect.lwd line width of the connecting line, with default 1.

Value

A list with the following components is available:

col colour of the samples.

pch plotting character of the samples.

cex expansion of the plotting character of the samples.

label TRUE or FALSE, whether samples should be labelled.

label.col colour of the label.

label.cex expansion of the label.

label.side side at which to plot the label of samples.

label.offset offset of the label from the data point.

connected TRUE or FALSE, whether samples should be connected in row order of X.

connect.col colour of the connecting line.

connect.lty line type of the connecting line.

connect.lwd line width of the connecting line.

Examples

biplot(data = iris[1:145,]) |> PCA() |> samples(col = "grey") |> interpolate(newdata = iris[146:150,]) |> newsamples(col = rainbow(6), pch=15) |> plot()
**Principal Component Analysis (PCA) method**

**Description**

This function produces a list of elements to be used for PCA biplot construction.

**Usage**

```r
PCA(bp, dim.biplot = c(2, 1, 3), e.vects = 1:ncol(bp$X),
group.aes = NULL, show.group.means = FALSE, correlation.biplot = FALSE)
```

**Arguments**

- `bp`: an object of class `biplot` obtained from preceding function `biplot()`.
- `dim.biplot`: dimension of the biplot. Only values 1, 2 and 3 are accepted, with default 2.
- `e.vects`: which eigenvectors (principal components) to extract, with default `1:dim.biplot`.
- `group.aes`: vector of the same length as the number of rows in the data matrix for differentiated aesthetics for samples.
- `show.group.means`: logical, indicating whether group means should be plotted in the biplot.
- `correlation.biplot`: logical, if `FALSE`, the distances between sample points are optimally approximated in the biplot. If `TRUE`, the correlations between variables are optimally approximated by the cosine of the angles between axes. Default is `FALSE`.

**Value**

Object of class `PCA` with the following elements:

- `X`: matrix of the centered and scaled numeric variables.
- `Xcat`: matrix of the categorical variables.
- `raw.X`: original data.
- `na.action`: vector of observations that have been removed.
- `center`: `TRUE` or `FALSE`, whether `X` is centred.
- `scaled`: `TRUE` or `FALSE`, whether `X` is scaled.
- `means`: mean of each numerical variable.
- `sd`: standard deviation of each numerical variable.
- `n`: number of observations.
- `p`: number of variables.
- `group.aes`: vector of the same length as the number of rows in the data matrix for differentiated aesthetics for samples.
- `g.names`: descriptive name to be used for group labels.
**PCA.biplot**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>g</code></td>
<td>number of groups.</td>
</tr>
<tr>
<td><code>Title</code></td>
<td>title of the biplot to be rendered</td>
</tr>
<tr>
<td><code>Z</code></td>
<td>matrix with each row containing the details of the point to be plotted (i.e. coordinates).</td>
</tr>
<tr>
<td><code>Lmat</code></td>
<td>matrix for transformation to the principal components.</td>
</tr>
<tr>
<td><code>eigenvalues</code></td>
<td>vector of eigenvalues of the covariance matrix of <code>X</code>.</td>
</tr>
<tr>
<td><code>e.vects</code></td>
<td>vector indicating which principal components are plotted in the biplot.</td>
</tr>
<tr>
<td><code>ax.one.unit</code></td>
<td>one unit in the positive direction of each biplot axis.</td>
</tr>
<tr>
<td><code>class.means</code></td>
<td>logical, indicating whether group means should be plotted in the biplot.</td>
</tr>
<tr>
<td><code>Zmeans</code></td>
<td>matrix of class mean coordinates to be plotted in the biplot.</td>
</tr>
</tbody>
</table>

**References**


**Examples**

```r
biplot(iris[,1:4]) |> PCA()  
# create a PCA biplot
biplot(data = iris) |> PCA() |> plot()
```

**Description**

Performs calculations for a PCA biplot.

**Usage**

```r
## S3 method for class 'biplot'
PCA(
  bp,
  dim.biplot = c(2, 1, 3),
  e.vects = 1:ncol(bp$X),
  group.aes = NULL,
  show.group.means = FALSE,
  correlation.biplot = FALSE
)
```
**plot.biplot**

**Generic Plotting function of objects of class biplot**

**Description**

Generic Plotting function of objects of class biplot

**Usage**

```r
## S3 method for class 'biplot'
plot(
  x,
  exp.factor = 1.2,
  axis.predictivity = NULL,
  sample.predictivity = NULL,
  ...
)
```

**Arguments**

- **bp**: an object of class biplot obtained from preceding function `biplot()`.
- **dim.biplot**: dimension of the biplot. Only values 1, 2 and 3 are accepted, with default 2.
- **e.vects**: which eigenvectors (principal components) to extract, with default `1:dim.biplot`.
- **group.aes**: vector of the same length as the number of rows in the data matrix for differentiated aesthetics for samples.
- **show.group.means**: logical, indicating whether group means should be plotted in the biplot.
- **correlation.biplot**: logical, if FALSE, the distances between sample points are optimally approximated in the biplot. If TRUE, the correlations between variables are optimally approximated by the cosine of the angles between axes. Default is FALSE.

**Value**

an object of class PCA, inherits from class biplot.

**Examples**

```r
biplot(iris[,1:4]) |> PCA()
# create a PCA biplot
biplot(data = iris) |> PCA() |> plot()
```
### Arguments

- **x**
  - An object of class `biplot`.

- **exp.factor**
  - Factor to expand plotting area beyond samples.

- **axis.predictivity**
  - Either logical or a numeric value between 0 and 1. If it is a numeric value, this value is used as threshold so that only axes with axis predictivity larger than the threshold is displayed. If `axis.predictivity = TRUE`, the axis color is 'diluted' in proportion with the axis predictivity.

- **sample.predictivity**
  - Either a logical or a numeric value between 0 and 1. If it is a numeric value, this value is used as threshold so that only samples with sample predictivity larger than the threshold is displayed. If `sample.predictivity = TRUE`, the sample size is shrinked in proportion with the sample predictivity.

- ...  
  - Additional arguments.

### Value

An object of class `biplot`.

### Examples

```r
biplot(iris[,1:4]) |> PCA() |> plot()
```

---

### Description

Generic print function of objects of class `biplot`.

### Usage

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

### Arguments

- **x**
  - An object of class `biplot`.

- ...  
  - Additional arguments.

### Value

No return value, called for side effects.

### Examples

```r
out <- biplot(iris[,1:4]) |> PCA() 
out
```
samples

Aesthetics for biplot samples

Description
This function allows formatting changes to samples.

Usage
samples (bp, which = 1:bp$g, col = ez.col, pch = 3, cex = 1, label = FALSE,
label.col = NULL, label.cex = 0.75, label.side = "bottom", label.offset = 0.5,
connected = FALSE, connect.col = "black", connect.lty = 1, connect.lwd = 1)

Arguments
bp an object of class biplot.
which vector of which groups of samples to display, with default bp$g.
col sample colour, with default blue.
pch sample plotting character, with default +.
cex sample character expansion, with default 1.
label logical, whether samples should be labelled or not, with default FALSE.
lable.col vector of length number of samples with the colour of the labels, defaulting to
the colour of the sample points.
lable.cex label text expansion, with default 0.75.
lable.side side of the plotting character where label appears, with default bottom. Note that
 unlike the argument pos in text(), options are "bottom", "left", "top", "right"
 and not 1, 2, 3, 4.
lable.offset offset of the label from the data point. See ?text for a detailed explanation of the
argument offset.
connected logical, whether samples are connected in order of rows of data matrix, with
default FALSE.
connect.col colour of the connecting line, with default black.
connect.lty line type of the connecting line, with default 1.
connect.lwd line width of the connecting line, with default 1.

Details
The arguments which, col, pch and cex are based on the specification of group.aes or classes. If
no groups are specified, a single colour, plotting character and / or character expansion is expected.
If g groups are specified, vectors of length g is expected, or values are recycled to length g.
The arguments label, label.cex, label.side and label.offset are based on the sample size n.
A single value will be recycled n times or a vector of length n is expected.
Value

A list with the following components is available:

- `which` which means to display.
- `col` colour of the samples.
- `pch` plotting character of the samples.
- `cex` expansion of the plotting character of the samples.
- `label` TRUE or FALSE, whether samples should be labelled.
- `label.col` colour of the label.
- `label.cex` expansion of the label.
- `label.side` side at which to plot the label of samples.
- `label.offset` offset of the label from the data point.
- `connected` TRUE or FALSE, whether samples should be connected in row order of X.
- `connect.col` colour of the connecting line.
- `connect.lty` line type of the connecting line.
- `connect.lwd` line width of the connecting line.

Examples

biplot(iris[,1:4]) |> PCA() |> samples(col="purple",pch=15) |> plot()

summary.biplot

Generic summary function of objects of class biplot

Description

Generic summary function of objects of class biplot

Usage

```r
## S3 method for class 'biplot'
summary(
  object,
  adequacy = TRUE,
  axis.predictivity = TRUE,
  sample.predictivity = TRUE,
  class.predictivity = TRUE,
  within.class.axis.predictivity = TRUE,
  within.class.sample.predictivity = TRUE,
  ...
)
```
summary.biplot

Arguments

object an object of class biplot.
adequacy logical, whether variable adequacies should be reported.
axis.predictivity logical, whether axis predictivities should be reported.
sample.predictivity logical, whether sample predictivities should be reported.
class.predictivity logical, whether class predictivities should be reported (only applicable to objects of class CVA).
within.class.axis.predictivity logical, whether within class axis predictivity should be reported (only applicable to objects of class CVA).
within.class.sample.predictivity logical, whether within class sample predictivity should be reported (only applicable to objects of class CVA).
...
additional arguments.

Value

no return value, called for side effects.

Examples

out <- biplot(iris[,1:4]) |> PCA() |> fit.measures()
summary(out)
Index

alpha.bags, 2
axes, 3

biplot, 5

CVA, 6
CVA.biplot, 8

ellipses, 9

fit.measures, 10

interpolate, 11

legend.type, 12

means, 13

newsamples, 14

PCA, 16
PCA.biplot, 17
plot.biplot, 18
print.biplot, 19

samples, 20
summary.biplot, 21