

# Package ‘dimensio’

September 18, 2021

**Title** Multivariate Data Analysis

**Version** 0.2.2

**Maintainer** Nicolas Frerebeau

<nicolas.frerebeau@u-bordeaux-montaigne.fr>

**Description** Simple Principal Components Analysis (PCA) and Correspondence Analysis (CA) based on the Singular Value Decomposition (SVD). This package provides S4 classes and methods to compute, extract, summarize and visualize results of multivariate data analysis. It also includes methods for partial bootstrap validation described in Greenacre (1984) <isbn: 978-0-12-299050-2> and Lebart et al. (2006) <isbn: 978-2-10-049616-7>.

**License** GPL (>= 3)

**URL** <https://packages.tesselle.org/dimensio/>,  
<https://github.com/tesselle/dimensio>

**BugReports** <https://github.com/tesselle/dimensio/issues>

**Depends** R (>= 3.3)

**Imports** ggplot2, methods, rlang

**Suggests** covr, folio, FactoMineR, ggrepel, khroma, testthat (>= 3.0.0), vdiffr (>= 1.0.0)

**Config/testthat/edition** 3

**Encoding** UTF-8

**RoxygenNote** 7.1.1

**Collate** 'AllClasses.R' 'AllGenerics.R' 'bootstrap.R' 'ca.R'  
'dimensio-package.R' 'helpers.R' 'jackknife.R' 'joint.R'  
'mutators.R' 'pca.R' 'plot.R' 'show.R' 'subset.R' 'summary.R'  
'svd.R' 'zzz.R'

**NeedsCompilation** no

**Author** Nicolas Frerebeau [aut, cre] (<<https://orcid.org/0000-0001-5759-4944>>),  
Jean-Baptiste Fourvel [ctb] (<<https://orcid.org/0000-0002-1061-4642>>),  
Brice Lebrun [ctb] (<<https://orcid.org/0000-0001-7503-8685>>)

**Repository** CRAN

**Date/Publication** 2021-09-18 12:40:02 UTC

## R topics documented:

bootstrap . . . . .	2
ca . . . . .	5
CA-class . . . . .	7
jackknife . . . . .	8
MultivariateAnalysis . . . . .	9
MultivariateBootstrap . . . . .	9
MultivariateResults . . . . .	10
MultivariateSummary . . . . .	11
mutator . . . . .	11
pca . . . . .	14
PCA-class . . . . .	16
plot_contributions . . . . .	17
plot_coordinates . . . . .	18
plot_eigenvalues . . . . .	21
predict . . . . .	22
subset . . . . .	23
summary . . . . .	25

<b>Index</b>	<b>26</b>
--------------	-----------

---

bootstrap	<i>Partial Bootstrap Analysis</i>
-----------	-----------------------------------

---

### Description

Checks analysis with partial bootstrap resampling.

### Usage

```
bootstrap(object, ...)

## S4 method for signature 'numeric'
bootstrap(object, do, n, ...)

## S4 method for signature 'integer'
bootstrap(object, do, n, ...)

## S4 method for signature 'BootstrapVector'
summary(
  object,
  level = 0.95,
  type = c("student", "normal"),
```

```

    probs = c(0.25, 0.75),
    na.rm = FALSE,
    ...
)

## S4 method for signature 'CA'
bootstrap(object, n = 30)

## S4 method for signature 'PCA'
bootstrap(object, n = 30)

```

### Arguments

object	A <a href="#">numeric</a> or an <a href="#">integer</a> vector or a <a href="#">CA</a> or <a href="#">PCA</a> object (see below).
...	Currently not used.
do	A <a href="#">function</a> that takes object as an argument and returns a single numeric value.
n	A non-negative <a href="#">integer</a> giving the number of bootstrap replications.
level	A length-one <a href="#">numeric</a> vector giving the confidence level. Must be a single number between 0 and 1. If NULL, no confidence interval are computed.
type	A <a href="#">character</a> string giving the type of confidence interval to be returned. It must be one "student" (default) or "normal". Any unambiguous substring can be given. Only used if level is not NULL.
probs	A <a href="#">numeric</a> vector of probabilities with values in [0, 1] (see <a href="#">stats::quantile()</a> ). If NULL, quantiles are not computed.
na.rm	A <a href="#">logical</a> scalar: should missing values be removed from object before the sample statistics are computed?

### Value

If object is a [numeric](#) or an [integer](#) vector, `bootstrap()` returns a `BootstrapVector` object (i.e. a numeric vector of the n bootstrap values of do).

If object is a [CA](#) or a [PCA](#) object, `bootstrap()` returns a `BootstrapCA` or a `BootstrapPCA` object.

`summary()` returns a numeric vector with the following elements:

min Minimum value.

mean Mean value.

max Maximum value.

lower Lower bound of the confidence interval.

upper Upper bound of the confidence interval.

Q\* Sample quantile to \* probability.

### Methods (by class)

- `numeric`: Samples randomly from the elements of object with replacement.
- `integer`: Samples observations from a multinomial distribution.

**Author(s)**

N. Frerebeau

**References**

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

Lebart, L., Piron, M. and Morineau, A. *Statistique exploratoire multidimensionnelle: visualisation et inférence en fouille de données*. Paris: Dunod, 2006.

**See Also**

Other resampling methods: [jackknife\(\)](#)

**Examples**

```
library(ggrepel)

## Random samples from x with replacement
x <- rnorm(20) # numeric
boot <- bootstrap(x, do = mean, n = 100) # Sample mean
summary(boot)

## Sample observations from a multinomial distribution
x <- sample(1:100, 100, TRUE) # integer
boot <- bootstrap(x, do = median, n = 100)
summary(boot)

## Partial bootstrap on CA
## Data from Lebart et al. 2006, p. 170-172
color <- data.frame(
  brun = c(68, 15, 5, 20),
  chatain = c(119, 54, 29, 84),
  roux = c(26, 14, 14, 17),
  blond = c(7, 10, 16, 94),
  row.names = c("marron", "noisette", "vert", "bleu")
)

## Compute correspondence analysis
X <- ca(color)

## Plot results
plot(X) +
  ggrepel::geom_label_repel()

## Bootstrap (30 replicates)
Y <- bootstrap(X, n = 30)

## Get replicated coordinates
get_replications(Y, margin = 1)
```

```

get_replications(Y, margin = 2)

## Plot with ellipses
plot_rows(Y) +
  ggplot2::stat_ellipse()

plot_columns(Y) +
  ggplot2::stat_ellipse()

## Partial bootstrap on PCA
## Compute principal components analysis
data(iris)
X <- pca(iris)

## Plot results
plot_columns(X) +
  ggrepel::geom_label_repel()

## Bootstrap (30 replicates)
Y <- bootstrap(X, n = 30)

## Plot with ellipses
plot_columns(Y) +
  ggplot2::stat_ellipse()

```

---

ca

*Correspondence Analysis*


---

## Description

Computes a simple correspondence analysis based on the singular value decomposition.

## Usage

```

ca(object, ...)

## S4 method for signature 'data.frame'
ca(object, rank = NULL, sup_row = NULL, sup_col = NULL)

## S4 method for signature 'matrix'
ca(object, rank = NULL, sup_row = NULL, sup_col = NULL)

```

## Arguments

object	A $m \times p$ numeric <a href="#">matrix</a> or a <a href="#">data.frame</a> .
...	Currently not used.
rank	An <a href="#">integer</a> value specifying the maximal number of components to be kept in the results. If NULL (the default), $\min(m, p) - 1$ components will be returned.

sup\_row        A [numeric](#) or [logical](#) vector specifying the indices of the supplementary rows.  
sup\_col        A [numeric](#) or [logical](#) vector specifying the indices of the supplementary columns.

### Value

A [CA](#) object.

### Author(s)

N. Frerebeau

### References

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

Lebart, L., Piron, M. and Morineau, A. *Statistique exploratoire multidimensionnelle: visualisation et inférence en fouille de données*. Paris: Dunod, 2006.

### See Also

[get\\_\\*\(\)](#), [stats::predict\(\)](#), [svd\(\)](#)  
Other multivariate analysis: [pca\(\)](#), [predict\(\)](#)

### Examples

```
## Load data
data("zuni", package = "folio")

## The chi square of independence between the two variables
stats::chisq.test(zuni)

## Compute correspondence analysis
X <- ca(zuni)

## Get row coordinates
get_coordinates(X, margin = 1)

## Get column coordinates
get_coordinates(X, margin = 2)

## Get row distances to centroid
get_distances(X, margin = 1)

## Get row inertias
get_inertia(X, margin = 1)

## Get row contributions
get_contributions(X, margin = 1)

## Get eigenvalues
get_eigenvalues(X)
```

---

CA-class

*CA Results*

---

### Description

An S4 class to store the results of a simple correspondence analysis.

### Author(s)

N. Frerebeau

### See Also

Other class: [MultivariateAnalysis](#), [MultivariateBootstrap](#), [MultivariateResults](#), [MultivariateSummary](#), [PCA-class](#)

### Examples

```
## Load data
data("zuni", package = "folio")

## The chi square of independence between the two variables
stats::chisq.test(zuni)

## Compute correspondence analysis
X <- ca(zuni)

## Get row coordinates
get_coordinates(X, margin = 1)

## Get column coordinates
get_coordinates(X, margin = 2)

## Get row distances to centroid
get_distances(X, margin = 1)

## Get row inertias
get_inertia(X, margin = 1)

## Get row contributions
get_contributions(X, margin = 1)

## Get eigenvalues
get_eigenvalues(X)
```

---

jackknife

*Jackknife Estimation*

---

## Description

Jackknife Estimation

## Usage

```
jackknife(object, ...)  
  
## S4 method for signature 'numeric'  
jackknife(object, do, ...)  
  
## S4 method for signature 'JackknifeVector'  
summary(object, ...)
```

## Arguments

object	A <a href="#">numeric</a> vector.
...	Extra arguments passed to do.
do	A <a href="#">function</a> that takes object as an argument and returns a single numeric value.

## Value

jackknife() returns a JackknifeVector object (i.e. a numeric vector of the n leave-one-out values of do).

summary() returns a named numeric vector with the following elements:

mean The jackknife estimate of mean of do.  
bias The jackknife estimate of bias of do.  
error The jackknife estimate of standard error of do.

## Author(s)

N. Frerebeau

## See Also

Other resampling methods: [bootstrap\(\)](#)

## Examples

```
## Jackknife  
x <- rnorm(20)  
jack <- jackknife(x, do = mean) # Sample mean  
summary(jack)
```



---

MultivariateAnalysis *Output of Multivariate Data Analysis*

---

**Description**

A virtual S4 class to store the output of a multivariate data analysis.

**Slots**

data A [numeric matrix](#).

dimension An [integer](#) giving the dimension of the solution.

singular\_values A [numeric](#) vector giving the singular values.

rows A [MultivariateResults](#) object.

columns A [MultivariateResults](#) object.

**Subset**

In the code snippets below, x is a [MultivariateAnalysis](#) object.

x[[i]] Extracts information from a slot selected by subscript i. i is a length-one [character](#) vector.

**Author(s)**

N. Frerebeau

**See Also**

Other class: [CA-class](#), [MultivariateBootstrap](#), [MultivariateResults](#), [MultivariateSummary](#), [PCA-class](#)

---

MultivariateBootstrap *Output of Bootstrap Replications*

---

**Description**

A virtual S4 class to store the output of a bootstrap analysis.

**Slots**

replications An [integer](#) giving the number of bootstrap replications.

**Author(s)**

N. Frerebeau

**See Also**

Other class: [CA-class](#), [MultivariateAnalysis](#), [MultivariateResults](#), [MultivariateSummary](#), [PCA-class](#)

---

MultivariateResults    *Multivariate Data Analysis Results*

---

**Description**

An S4 class to store the results of a multivariate data analysis.

**Slots**

names A [character](#) vector specifying the row names.

principal A [numeric matrix](#) giving the principal coordinates.

standard A [numeric matrix](#) giving the standard coordinates.

contributions A [numeric matrix](#) giving the contributions to the definition of the dimensions.

cosine A [numeric matrix](#) giving the  $\cos^2$  values.

distances A [numeric](#) vector giving the distances to centroid.

weights A [numeric](#) vector giving the masses/weights.

supplement A [logical](#) vector specifying the supplementary points.

order An [integer](#) vector giving the original indices of the data (computation moves all supplementary points at the end of the results).

groups A [character](#) vector specifying the class for each observation.

**Author(s)**

N. Frerebeau

**See Also**

Other class: [CA-class](#), [MultivariateAnalysis](#), [MultivariateBootstrap](#), [MultivariateSummary](#), [PCA-class](#)

---

MultivariateSummary      *Summary of Multivariate Data Analysis*

---

**Description**

A virtual S4 class to store the summary of a multivariate data analysis.

**Slots**

data A [numeric matrix](#).

eigenvalues A [numeric matrix](#).

results A [numeric matrix](#).

supplement A [logical](#) vector specifying the supplementary points.

margin An [integer](#).

**Author(s)**

N. Frerebeau

**See Also**

Other class: [CA-class](#), [MultivariateAnalysis](#), [MultivariateBootstrap](#), [MultivariateResults](#), [PCA-class](#)

---

mutator                      *Get Results*

---

**Description**

Getters to retrieve parts of an object.

**Usage**

`get_coordinates(x, ...)`

`get_replications(x, ...)`

`get_contributions(x, ...)`

`get_correlations(x, ...)`

`get_cos2(x, ...)`

`get_data(x, ...)`

```
get_distances(x, ...)  
  
get_eigenvalues(x)  
  
get_inertia(x, ...)  
  
get_variance(x, ...)  
  
## S4 method for signature 'MultivariateAnalysis'  
dim(x)  
  
## S4 method for signature 'MultivariateAnalysis'  
rownames(x)  
  
## S4 method for signature 'MultivariateAnalysis'  
colnames(x)  
  
## S4 method for signature 'MultivariateAnalysis'  
dimnames(x)  
  
## S4 method for signature 'MultivariateAnalysis'  
get_contributions(x, margin = 1)  
  
## S4 method for signature 'MultivariateAnalysis'  
get_coordinates(x, margin = 1, sup_name = ".sup")  
  
## S4 method for signature 'MultivariateBootstrap'  
get_replications(x, margin = 1)  
  
## S4 method for signature 'BootstrapPCA'  
get_replications(x)  
  
## S4 method for signature 'PCA'  
get_correlations(x, sup_name = ".sup")  
  
## S4 method for signature 'MultivariateAnalysis'  
get_cos2(x, margin = 1, sup_name = ".sup")  
  
## S4 method for signature 'MultivariateAnalysis'  
get_data(x)  
  
## S4 method for signature 'MultivariateAnalysis'  
get_distances(x, margin = 1)  
  
## S4 method for signature 'MultivariateAnalysis'  
get_eigenvalues(x)
```

```
## S4 method for signature 'MultivariateAnalysis'
get_inertia(x, margin = 1)

## S4 method for signature 'MultivariateAnalysis'
get_variance(x, digits = 2)

## S4 method for signature 'PCA'
loadings(x)
```

### Arguments

<code>x</code>	An object from which to get element(s) (a <a href="#">CA</a> or <a href="#">PCA</a> object).
<code>...</code>	Currently not used.
<code>margin</code>	A length-one <a href="#">numeric</a> vector giving the subscript which the data will be returned: 1 indicates individuals/rows (the default), 2 indicates variables/columns.
<code>sup_name</code>	A <a href="#">character</a> string specifying the name of the column to create for supplementary points attribution (see below).
<code>digits</code>	An <a href="#">integer</a> indicating the number of decimal places to be used.

### Details

`get_data()` returns a `data.frame` of original data.

`get_contributions()` returns a `data.frame` of contributions to the definition of the principal dimensions.

`get_coordinates()` returns a `data.frame` of coordinates. An extra column (named after `sup_name`) is added specifying whether an observation is a supplementary point or not.

`get_replications()` returns an array of replicated coordinates (see [bootstrap\(\)](#)).

`get_correlations()` returns a `data.frame` of correlations between variables and dimensions (PCA). An extra column (named after `sup_name`) is added specifying whether an observation is a supplementary point or not.

`get_cos2()` returns a `data.frame` of  $\cos^2$  values (i.e. quality of the representation of the points on the factor map). An extra column (named after `sup_name`) is added specifying whether an observation is a supplementary point or not.

`get_eigenvalues()` returns a `data.frame` with the following columns: `eigenvalues`, `variance` (percentage of variance) and `cumulative` (cumulative percentage of variance).

`get_variance()` returns a numeric vector giving the percentage of explained variance of each dimension.

`loadings()` returns variable loadings (i.e. the coefficients of the linear combination of the original variables). `loadings()` is only implemented for consistency with `[stats][stats::loadings]`.

### Value

`get_*`() returns a [numeric](#) vector or a `data.frame`.

`loadings()` returns of a [matrix](#) of class `stats::loadings`.

**Author(s)**

N. Frerebeau

**See Also**Other mutator: [subset\(\)](#)

---

pca

*Principal Components Analysis*

---

**Description**

Computes a principal components analysis based on the singular value decomposition.

**Usage**

```
pca(object, ...)  
  
## S4 method for signature 'data.frame'  
pca(  
  object,  
  center = TRUE,  
  scale = TRUE,  
  rank = NULL,  
  sup_row = NULL,  
  sup_col = NULL,  
  weight_row = NULL,  
  weight_col = NULL  
)  
  
## S4 method for signature 'matrix'  
pca(  
  object,  
  center = TRUE,  
  scale = TRUE,  
  rank = NULL,  
  sup_row = NULL,  
  sup_col = NULL,  
  weight_row = NULL,  
  weight_col = NULL  
)
```

**Arguments**

object	A $m \times p$ numeric <a href="#">matrix</a> or a <a href="#">data.frame</a> .
...	Currently not used.

center	A <a href="#">logical</a> scalar: should the variables be shifted to be zero centered?
scale	A <a href="#">logical</a> scalar: should the variables be scaled to unit variance?
rank	An <a href="#">integer</a> value specifying the maximal number of components to be kept in the results. If NULL (the default), $p - 1$ components will be returned.
sup_row	A <a href="#">numeric</a> or <a href="#">logical</a> vector specifying the indices of the supplementary rows (individuals).
sup_col	A <a href="#">numeric</a> or <a href="#">logical</a> vector specifying the indices of the supplementary columns (variables).
weight_row	A <a href="#">numeric</a> vector specifying the active row (individual) weights. If NULL (the default), no weights are used.
weight_col	A <a href="#">numeric</a> vector specifying the active column (variable) weights. If NULL (the default), no weights are used.

**Value**

A [PCA](#) object.

**Author(s)**

N. Frerebeau

**References**

Lebart, L., Piron, M. and Morineau, A. *Statistique exploratoire multidimensionnelle: visualisation et inférence en fouille de données*. Paris: Dunod, 2006.

**See Also**

[get\\_\\*\(\)](#), [stats::predict\(\)](#), [svd\(\)](#)

Other multivariate analysis: [ca\(\)](#), [predict\(\)](#)

**Examples**

```
## Load data
data("compiegne", package = "folio")

## Compute principal components analysis
X <- pca(compiegne, scale = TRUE, sup_col = 7:10)

## Get row coordinates
get_coordinates(X, margin = 1)

## Get column coordinates
get_coordinates(X, margin = 2)

## Get row contributions
get_contributions(X, margin = 1)

## Get correlations between variables and dimensions
```

```
get_correlations(X)

## Get eigenvalues
get_eigenvalues(X)
```

---

PCA-class

*PCA Results*

---

## Description

An S4 class to store the results of a principal components analysis.

## Slots

`center` A **numeric** vector giving the column mean of the initial dataset (active individuals only).  
`scale` A **numeric** vector giving the column standard deviations of the initial dataset (active individuals only).

## Author(s)

N. Frerebeau

## See Also

Other class: [CA-class](#), [MultivariateAnalysis](#), [MultivariateBootstrap](#), [MultivariateResults](#), [MultivariateSummary](#)

## Examples

```
## Load data
data("compiegne", package = "folio")

## Compute principal components analysis
X <- pca(compiegne, scale = TRUE, sup_col = 7:10)

## Get row coordinates
get_coordinates(X, margin = 1)

## Get column coordinates
get_coordinates(X, margin = 2)

## Get row contributions
get_contributions(X, margin = 1)

## Get correlations between variables and dimensions
get_correlations(X)

## Get eigenvalues
get_eigenvalues(X)
```



---

plot\_contributions      *Visualize Contributions and cos<sup>2</sup>*

---

### Description

Plots contributions histogram and  $\cos^2$  scatterplot.

### Usage

```
plot_contributions(object, ...)  
  
plot_cos2(object, ...)  
  
## S4 method for signature 'MultivariateAnalysis'  
plot_contributions(  
  object,  
  margin = 2,  
  axes = 1,  
  sort = TRUE,  
  decreasing = TRUE,  
  limit = 10,  
  fill = "grey30",  
  border = "grey10"  
)  
  
## S4 method for signature 'MultivariateAnalysis'  
plot_cos2(  
  object,  
  margin = 2,  
  axes = c(1, 2),  
  active = TRUE,  
  sup = TRUE,  
  sort = TRUE,  
  decreasing = TRUE,  
  limit = 10,  
  fill = "grey30",  
  border = "grey10"  
)
```

### Arguments

object	A <a href="#">CA</a> or <a href="#">PCA</a> object.
...	Currently not used.
margin	A length-one <a href="#">numeric</a> vector giving the subscript which the data will be returned: 1 indicates individuals/rows (the default), 2 indicates variables/columns.
axes	A length-one <a href="#">numeric</a> vector giving the dimensions to be plotted.

sort	A <a href="#">logical</a> scalar: should the data be sorted?
decreasing	A <a href="#">logical</a> scalar: should the sort order be decreasing? Only used if sort is TRUE.
limit	An <a href="#">integer</a> specifying the number of top elements to be displayed.
fill, border	A <a href="#">character</a> string specifying the bars infilling and border colors.
active	A <a href="#">logical</a> scalar: should the active observations be plotted?
sup	A <a href="#">logical</a> scalar: should the supplementary observations be plotted?

**Author(s)**

N. Frerebeau

**See Also**

[ggplot2::ggplot\(\)](#)

Other plot: [plot\\_coordinates](#), [plot\\_eigenvalues](#)

**Examples**

```
## Load data
data("zuni", package = "folio")

## Compute correspondence analysis
X <- ca(zuni)

## Plot observations
plot(X)

## Screeplot
plot_variance(X)
```

---

plot\_coordinates      *Visualize Factor Map*

---

**Description**

Plots factor map.

**Usage**

```
plot_rows(object, ...)

plot_columns(object, ...)

## S4 method for signature 'MultivariateAnalysis'
plot_rows(
  object,
```

```
axes = c(1, 2),
active = TRUE,
sup = TRUE,
highlight = NULL,
group = NULL
)

## S4 method for signature 'CA,missing'
plot(
  x,
  margin = c(1, 2),
  axes = c(1, 2),
  active = TRUE,
  sup = TRUE,
  highlight = NULL,
  group = NULL
)

## S4 method for signature 'CA'
plot_columns(
  object,
  axes = c(1, 2),
  active = TRUE,
  sup = TRUE,
  highlight = NULL,
  group = NULL
)

## S4 method for signature 'PCA,missing'
plot(
  x,
  margin = 1,
  axes = c(1, 2),
  active = TRUE,
  sup = TRUE,
  highlight = NULL,
  group = NULL
)

## S4 method for signature 'PCA'
plot_columns(
  object,
  axes = c(1, 2),
  active = TRUE,
  sup = TRUE,
  highlight = NULL,
  group = NULL
)
```

```
## S4 method for signature 'BootstrapPCA'
plot_columns(
  object,
  axes = c(1, 2),
  active = TRUE,
  sup = TRUE,
  highlight = NULL,
  group = NULL
)
```

### Arguments

object, x	A <a href="#">CA</a> or <a href="#">PCA</a> object.
...	Currently not used.
axes	A length-two <a href="#">numeric</a> vector giving the dimensions to be plotted.
active	A <a href="#">logical</a> scalar: should the active observations be plotted?
sup	A <a href="#">logical</a> scalar: should the supplementary observations be plotted?
highlight	A <a href="#">character</a> string giving XXX. It must be one of "coordinates", "contributions" or "cos2". Any unambiguous substring can be given. If NULL (the default), no highlighting is applied.
group	A vector of categories specifying the categorical variable from which to color the individuals (only used if <code>highlight</code> is NULL; see below).
margin	A length-one <a href="#">numeric</a> vector giving the subscript which the data will be returned: 1 indicates individuals/rows (the default), 2 indicates variables/columns, <code>c(1, 2)</code> indicates rows and columns (CA).

### Aesthetic

Point shapes and line types are set whether an observation is a row/individual or a column/variable and is active or supplementary.

Colors are set according to `highlight` and `group`:

- If `highlight` is not NULL, the color gradient will vary according to the value of the selected parameter.
- If `group` is a numeric vector, the color gradient and size will vary by the value of `group`.
- If `group` is not a numeric vector, the colors will be mapped to the levels of `group`.
- If both are NULL (the default), then the same rule as for shapes is used.

### Author(s)

N. Frerebeau

### See Also

[ggplot2::ggplot\(\)](#)

Other plot: [plot\\_contributions\(\)](#), [plot\\_eigenvalues](#)

**Examples**

```
## Load data
data("zuni", package = "folio")

## Compute correspondence analysis
X <- ca(zuni)

## Plot observations
plot(X)

## Screeplot
plot_variance(X)
```

---

plot_eigenvalues	<i>Visualize Eigenvalues</i>
------------------	------------------------------

---

**Description**

Plot eigenvalues or variances histogram.

**Usage**

```
plot_variance(object, ...)
```

```
## S4 method for signature 'MultivariateAnalysis'
plot_variance(
  object,
  variance = TRUE,
  cumulative = TRUE,
  fill = "grey30",
  border = "grey10",
  color = "red"
)
```

**Arguments**

object	A <a href="#">CA</a> or <a href="#">PCA</a> object.
...	Currently not used.
variance	A <a href="#">logical</a> scalar: should the percentages of variance be plotted instead of the eigenvalues?
cumulative	A <a href="#">logical</a> scalar: should the cumulative percentages of variance be plotted?
fill, border	A <a href="#">character</a> string specifying the bars infilling and border colors.
color	A <a href="#">character</a> string specifying the line color.

**Author(s)**

N. Frerebeau

**See Also**

[ggplot2::ggplot\(\)](#)

Other plot: [plot\\_contributions\(\)](#), [plot\\_coordinates](#)

**Examples**

```
## Load data
data("zuni", package = "folio")

## Compute correspondence analysis
X <- ca(zuni)

## Plot observations
plot(X)

## Screeplot
plot_variance(X)
```

---

predict

*Predict New Coordinates*

---

**Description**

Predict the projection of new individuals/rows or variables/columns.

**Usage**

```
## S4 method for signature 'CA'
predict(object, newdata, margin = 1)

## S4 method for signature 'PCA'
predict(object, newdata, margin = 1)
```

**Arguments**

object	A <a href="#">CA</a> or <a href="#">PCA</a> object.
newdata	An object of supplementary points coercible to a <a href="#">matrix</a> for which to compute principal coordinates.
margin	A length-one <a href="#">numeric</a> vector giving the subscript which the data will be predicted: 1 indicates individuals/rows (the default), 2 indicates variables/columns.

**Value**

A [data.frame](#) of coordinates.

**Author(s)**

N. Frerebeau

**See Also**

Other multivariate analysis: [ca\(\)](#), [pca\(\)](#)

**Examples**

```
## Create a matrix
A <- matrix(data = sample(1:10, 100, TRUE), nrow = 10, ncol = 10)

## Compute correspondence analysis
X <- ca(A, sup_row = 8:10, sup_col = 7:10)

## Predict new row coordinates
Y <- matrix(data = sample(1:10, 120, TRUE), nrow = 20, ncol = 6)
predict(X, Y, margin = 1)

## Predict new column coordinates
Z <- matrix(data = sample(1:10, 140, TRUE), nrow = 7, ncol = 20)
predict(X, Z, margin = 2)
```

---

subset

*Extract Parts of an Object*

---

**Description**

Operators acting on objects to extract parts.

**Usage**

```
## S4 method for signature 'CA,ANY,missing'
x[[i]]

## S4 method for signature 'PCA,ANY,missing'
x[[i]]
```

**Arguments**

x	An object from which to extract element(s) or in which to replace element(s).
i	A <a href="#">character</a> string specifying elements to extract. Any unambiguous substring can be given (see details).

**Details**

If i is "data", returns a list with the following elements:

data A [numeric](#) matrix of raw data.

mean A [numeric](#) vector giving the variables means (PCA).

sd A [numeric](#) vector giving the variables standard deviations (PCA).

If `i` is "rows", returns a list with the following elements:

`coord` A **numeric** matrix of rows/individuals coordinates.

`cos2` A **numeric** matrix of rows/individuals squared cosine.

`masses` A **numeric** vector giving the rows masses/individual weights.

`sup` A **logical** vector specifying whether a point is a supplementary observation or not.

If `i` is "columns", returns a list with the following elements:

`coord` A **numeric** matrix of columns/variables coordinates.

`cor` A **numeric** matrix of correlation between variables and the dimensions (PCA).

`cos2` A **numeric** matrix of columns/variables squared cosine.

`masses` A **numeric** vector giving the columns masses/variable weights.

`sup` A **logical** vector specifying whether a point is a supplementary observation or not.

If `i` is "eigenvalues", returns a **numeric** vector of eigenvalues.

### Value

A **list**.

### Author(s)

N. Frerebeau

### See Also

Other mutator: [mutator](#)

### Examples

```
## Load data
data("mississippi", package = "folio")

## Compute principal components analysis
X <- pca(mississippi, scale = TRUE, sup_row = 8:10, sup_col = 7:10)

## Get results for the individuals
X[["individuals"]]

## Compute correspondence analysis
Y <- ca(mississippi, sup_row = 8:10, sup_col = 7:10)

## Get results for the rows
Y[["rows"]]
```



**Description**

Provides a summary of the results of a multivariate data analysis.

**Usage**

```
## S4 method for signature 'CA'  
summary(object, margin = 1, active = TRUE, sup = TRUE, rank = 3)  
  
## S4 method for signature 'PCA'  
summary(object, margin = 1, active = TRUE, sup = TRUE, rank = 3)
```

**Arguments**

object	A <a href="#">CA</a> or <a href="#">PCA</a> object.
margin	A length-one <a href="#">numeric</a> vector giving the subscript which the data will be summarized: 1 indicates individuals/rows (the default), 2 indicates variables/columns.
active	A <a href="#">logical</a> scalar: should the active observations be summarized?
sup	A <a href="#">logical</a> scalar: should the supplementary observations be summarized?
rank	An <a href="#">integer</a> value specifying the maximal number of components to be kept in the results.

**Author(s)**

N. Frerebeau

**Examples**

```
## Load data  
data("zuni", package = "folio")  
  
## Compute correspondence analysis  
X <- ca(zuni, rank = 5, sup_row = 1:50)  
  
## Rows summary  
summary(X, margin = 1)  
  
## Columns summary  
summary(X, margin = 2)
```

# Index

- \* **class**
  - CA-class, 7
  - MultivariateAnalysis, 9
  - MultivariateBootstrap, 9
  - MultivariateResults, 10
  - MultivariateSummary, 11
  - PCA-class, 16
- \* **multivariate analysis**
  - ca, 5
  - pca, 14
  - predict, 22
- \* **mutator**
  - mutator, 11
  - subset, 23
- \* **plot**
  - plot\_contributions, 17
  - plot\_coordinates, 18
  - plot\_eigenvalues, 21
- \* **resampling methods**
  - bootstrap, 2
  - jackknife, 8
- \* **summary**
  - summary, 25
  - .BootstrapCA (CA-class), 7
  - .BootstrapPCA (PCA-class), 16
  - .CA (CA-class), 7
  - .MultivariateAnalysis
    - (MultivariateAnalysis), 9
  - .MultivariateBootstrap
    - (MultivariateBootstrap), 9
  - .MultivariateResults
    - (MultivariateResults), 10
  - .MultivariateSummary
    - (MultivariateSummary), 11
  - .PCA (PCA-class), 16
  - .SummaryCA (MultivariateSummary), 11
  - .SummaryPCA (MultivariateSummary), 11
  - [[, CA, ANY, missing-method (subset), 23
  - [[, PCA, ANY, missing-method (subset), 23
  - bootstrap, 2, 8
  - bootstrap(), 13
  - bootstrap, CA-method (bootstrap), 2
  - bootstrap, integer-method (bootstrap), 2
  - bootstrap, numeric-method (bootstrap), 2
  - bootstrap, PCA-method (bootstrap), 2
  - bootstrap-method (bootstrap), 2
  - BootstrapCA, 3
  - BootstrapCA-class (CA-class), 7
  - BootstrapPCA, 3
  - BootstrapPCA-class (PCA-class), 16
  - CA, 3, 6, 13, 17, 20–22, 25
  - ca, 5, 15, 23
  - ca, data.frame-method (ca), 5
  - ca, matrix-method (ca), 5
  - CA-class, 7
  - ca-method (ca), 5
  - character, 3, 9, 10, 13, 18, 20, 21, 23
  - colnames, MultivariateAnalysis-method (mutator), 11
  - data.frame, 5, 13, 14, 22
  - dim, MultivariateAnalysis-method (mutator), 11
  - dimnames, MultivariateAnalysis-method (mutator), 11
  - function, 3, 8
  - get (mutator), 11
  - get\_\*(), 6, 15
  - get\_contributions (mutator), 11
  - get\_contributions, MultivariateAnalysis-method (mutator), 11
  - get\_contributions-method (mutator), 11
  - get\_coordinates (mutator), 11
  - get\_coordinates, MultivariateAnalysis-method (mutator), 11
  - get\_coordinates-method (mutator), 11

- get\_correlations (mutator), 11
- get\_correlations,PCA-method (mutator), 11
- get\_correlations-method (mutator), 11
- get\_cos2 (mutator), 11
- get\_cos2,MultivariateAnalysis-method (mutator), 11
- get\_cos2-method (mutator), 11
- get\_data (mutator), 11
- get\_data,MultivariateAnalysis-method (mutator), 11
- get\_data-method (mutator), 11
- get\_distances (mutator), 11
- get\_distances,MultivariateAnalysis-method (mutator), 11
- get\_distances-method (mutator), 11
- get\_eigenvalues (mutator), 11
- get\_eigenvalues,MultivariateAnalysis-method (mutator), 11
- get\_eigenvalues-method (mutator), 11
- get\_inertia (mutator), 11
- get\_inertia,MultivariateAnalysis-method (mutator), 11
- get\_inertia-method (mutator), 11
- get\_replications (mutator), 11
- get\_replications,BootstrapPCA-method (mutator), 11
- get\_replications,MultivariateBootstrap-method (mutator), 11
- get\_replications-method (mutator), 11
- get\_variance (mutator), 11
- get\_variance,MultivariateAnalysis-method (mutator), 11
- get\_variance-method (mutator), 11
- ggplot2::ggplot(), 18, 20, 22
- integer, 3, 5, 9–11, 13, 15, 18, 25
- jackknife, 4, 8
- jackknife,numeric-method (jackknife), 8
- jackknife-method (jackknife), 8
- list, 24
- loadings,PCA-method (mutator), 11
- logical, 3, 6, 10, 11, 15, 18, 20, 21, 24, 25
- matrix, 5, 9–11, 13, 14, 22
- MultivariateAnalysis, 7, 9, 10, 11, 16
- MultivariateAnalysis-class (MultivariateAnalysis), 9
- MultivariateBootstrap, 7, 9, 9, 10, 11, 16
- MultivariateBootstrap-class (MultivariateBootstrap), 9
- MultivariateResults, 7, 9, 10, 10, 11, 16
- MultivariateResults-class (MultivariateResults), 10
- MultivariateSummary, 7, 9, 10, 11, 16
- MultivariateSummary-class (MultivariateSummary), 11
- mutator, 11, 24
- numeric, 3, 6, 8–11, 13, 15–17, 20, 22–25
- PCA, 3, 13, 15, 17, 20–22, 25
- pca, 6, 14, 23
- pca,data.frame-method (pca), 14
- pca,matrix-method (pca), 14
- PCA-class, 16
- pca-method (pca), 14
- plot,CA,missing-method (plot\_coordinates), 18
- plot,PCA,missing-method (plot\_coordinates), 18
- plot\_columns (plot\_coordinates), 18
- plot\_columns,BootstrapPCA-method (plot\_coordinates), 18
- plot\_columns,CA-method (plot\_coordinates), 18
- plot\_columns,PCA-method (plot\_coordinates), 18
- plot\_columns-method (plot\_coordinates), 18
- plot\_contributions, 17, 20, 22
- plot\_contributions,MultivariateAnalysis-method (plot\_contributions), 17
- plot\_contributions-method (plot\_contributions), 17
- plot\_coordinates, 18, 18, 22
- plot\_cos2 (plot\_contributions), 17
- plot\_cos2,MultivariateAnalysis-method (plot\_contributions), 17
- plot\_cos2-method (plot\_contributions), 17
- plot\_eigenvalues, 18, 20, 21
- plot\_rows (plot\_coordinates), 18
- plot\_rows,MultivariateAnalysis-method (plot\_coordinates), 18
- plot\_rows-method (plot\_coordinates), 18
- plot\_variance (plot\_eigenvalues), 21

plot\_variance, MultivariateAnalysis-method  
    (plot\_eigenvalues), 21

plot\_variance-method  
    (plot\_eigenvalues), 21

predict, 6, 15, 22

predict, CA-method (predict), 22

predict, PCA-method (predict), 22

rownames, MultivariateAnalysis-method  
    (mutator), 11

set (mutator), 11

stats::loadings, 13

stats::predict(), 6, 15

stats::quantile(), 3

subset, 14, 23

summary, 25

summary, BootstrapVector-method  
    (bootstrap), 2

summary, CA-method (summary), 25

summary, JackknifeVector-method  
    (jackknife), 8

summary, PCA-method (summary), 25

SummaryCA-class (MultivariateSummary),  
    11

SummaryPCA-class (MultivariateSummary),  
    11

svd(), 6, 15