Package: booklet (via r-universe)

June 1, 2026

Type Package

Title Multivariate Exploratory Data Analysis

Version 1.0.1

Description Exploratory data analysis methods to summarize, visualize and describe datasets. The main principal component methods are available, those with the largest potential in terms of applications: principal component analysis (PCA) when variables are quantitative, correspondence analysis (CA) when variables are categorical, Multiple Factor Analysis (MFA) when variables are structured in groups.

License MIT + file LICENSE

URL https://github.com/alexym1/booklet, https://alexym1.github.io/booklet/

BugReports https://github.com/alexym1/booklet/issues

Depends R (>= 4.1.0)

Suggests covr, devtools, factoextra, FactoMineR, knitr, pak, renv, testthat

VignetteBuilder knitr

Encoding UTF-8

LazyData false

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Repository https://alexym1.r-universe.dev

Date/Publication 2025-09-02 18:26:27 UTC

RemoteUrl https://github.com/alexym1/booklet

RemoteRef HEAD

RemoteSha c2d2fdd81e597acb738471a9bc5403b9703a9d3b

Contents

`ca_col_contrib` *Compute col contributions*

Description

Return col contributions for each correspondence component

Usage

```
ca_col_contrib(col_coords, X, eigs)
```

Arguments

`col_coords`	col coordinates
`X`	standardized matrix
`eigs`	eigs computed by `ca_weighted_eigen`

Value

A dataframe of col contributions.

Examples

```
library(booklet)

X_scaled <- mtcars[, c(2, 8:11)] |>
  ca_standardize()

eigs <- X_scaled |>
  ca_weighted_eigen()

eigs |>
  ca_col_coords() |>
  ca_col_contrib(X_scaled, eigs) |>
  head()
```

`ca_col_coords` *Compute col coordinates*

Description

Return Correspondence component for columns

Usage

```
ca_col_coords(eigs)

ca_col_sup_coords(X_sup, eigs)
```

Arguments

`eigs`	eigs computed by `ca_weighted_eigen`
`X_sup`	Supplementary dataset

Value

A dataframe of col coordinates.

Examples

```
library(booklet)

mtcars[, c(2, 8:11)] |>
  ca_standardize() |>
  ca_weighted_eigen() |>
  ca_col_coords() |>
  head()
```

ca_col_cos2 *Compute col squared cosines*

Description

Return col squared cosines for each correspondence component

Usage

```
ca_col_cos2(col_coords, X)

ca_col_sup_cos2(col_coords, X_sup, X)
```

Arguments

col_coords	col coordinates
X	active dataset
X_sup	supplementary dataset

Value

A dataframe of col squared cosines.

Examples

```
library(booklet)

X_scaled <- mtcars[, c(2, 8:11)] |>
  ca_standardize()

X_scaled |>
  ca_weighted_eigen() |>
  ca_col_coords() |>
  ca_col_cos2(X_scaled) |>
  head()
```

ca_col_inertia *Compute col inertia*

Description

Return col inertia for each correspondence component

Usage

```
ca_col_inertia(X)
```

Arguments

X	standardized matrix

Value

A dataframe of col inertia.

Examples

```
library(booklet)

mtcars[, c(2, 8:11)] |>
  ca_standardize() |>
  ca_col_inertia()
```

---

ca_row_contrib *Compute row contributions*

Description

Return row contributions for each correspondence component

Usage

```
ca_row_contrib(row_coords, X, eigs)
```

Arguments

row_coords	row coordinates
X	standardized matrix
eigs	eigs computed by ca_weighted_eigen

Value

A dataframe of row contributions.

Examples

```
library(booklet)

X_scaled <- mtcars[, c(2, 8:11)] |>
  ca_standardize()

eigs <- X_scaled |>
  ca_weighted_eigen()

eigs |>
  ca_row_coords() |>
  ca_row_contrib(X_scaled, eigs) |>
  head()
```

ca_row_coords *Compute row coordinates*

Description

Return Correspondence component for individuals

Usage

```
ca_row_coords(eigs)

ca_row_sup_coords(X_sup, eigs)
```

Arguments

eigs	eigs computed by ca_weighted_eigen
X_sup	Supplementary dataset

Value

A dataframe of row coordinates.

Examples

```
library(booklet)

mtcars[, c(2, 8:11)] |>
  ca_standardize() |>
  ca_weighted_eigen() |>
  ca_row_coords() |>
  head()
```

ca_row_cos2 *Compute row squared cosines*

Description

Return row squared cosines for each correspondence component

Usage

```
ca_row_cos2(row_coords, X)

ca_row_sup_cos2(row_coords, X_sup, X)
```

Arguments

row_coords	row coordinates
X	Active standardized matrix
X_sup	Supplementary standardized matrix

Value

A dataframe of row squared cosines.

Examples

```
library(booklet)

X_scaled <- mtcars[, c(2, 8:11)] |>
  ca_standardize()

X_scaled |>
  ca_weighted_eigen() |>
  ca_row_coords() |>
  ca_row_cos2(X_scaled) |>
  head()
```

---

ca_row_inertia *Compute row inertia*

Description

Return row inertia for each correspondence component

Usage

```
ca_row_inertia(X)
```

Arguments

X	standardized matrix

Value

A dataframe of row inertia.

Examples

```
library(booklet)

mtcars[, c(2, 8:11)] |>
  ca_standardize() |>
  ca_row_inertia()
```

ca_standardize *Data standardization for CA*

Description

Perform data standardization for multivariate exploratory data analysis.

Usage

```
ca_standardize(X, weighted_row = rep(1, nrow(X)))

ca_standardize_sup(X, type = c("row", "col"), weighted_row = rep(1, nrow(X)))
```

Arguments

X	Active or supplementary datasets
weighted_row	row weights
type	standardization for supplementary rows or cols

Value

A dataframe of the same size as X.

Examples

```
library(booklet)

mtcars[, c(2, 8:11)] |>
  ca_standardize() |>
  head()
```

ca_weighted_eigen *Compute eigenvalues and eigenvectors for CA*

Description

Return eigenvalues and eigenvectors of a matrix

Usage

```
ca_weighted_eigen(X)
```

Arguments

X	X_active

Value

A list containing results of Single Value Decomposition (SVD).

Examples

```
library(booklet)

mtcars[, c(2, 8:11)] |>
  ca_standardize() |>
  ca_weighted_eigen() |>
  head()
```

`facto_ca` *Perform CA with FactoMineR's style*

Description

Return CA results with FactoMineR's style

Usage

```
facto_ca(X, ncp = 5, row_sup = NULL, col_sup = NULL, weighted_row = NULL)
```

Arguments

`X`	a data frame with n rows (individuals) and p columns (numeric variables)
`ncp`	an integer, the number of components to keep (value set by default)
`row_sup`	a vector indicating the indexes of the supplementary rows
`col_sup`	a vector indicating the indexes of the supplementary cols
`weighted_row`	row weights

Value

A list containing results of FactoMineR's correspondence analysis (CA).

Examples

```
library(booklet)
res <- facto_ca(X = mtcars[, c(2, 8:11)], ncp = 2)
```

facto_mfa *Perform MFA with FactoMineR's style*

Description

Return MFA results with FactoMineR's style

Usage

```
facto_mfa(X, groups, ncp = 2)
```

Arguments

X	a data frame with n rows (individuals) and p columns (numeric variables)
groups	a vector indicating the group of each variable
ncp	an integer, the number of components to keep (value set by default)

Value

A list containing results of FactoMineR's multiple factor analysis (MFA).

Examples

```
library(booklet)

res <- facto_mfa(X = iris[, -c(5)], groups = c(2, 2), ncp = 2)
```

facto_pca *Perform PCA with FactoMineR's style*

Description

Return PCA results with FactoMineR's style

Usage

```
facto_pca(
  X,
  ncp = 5,
  scale.unit = TRUE,
  ind_sup = NULL,
  quanti_sup = NULL,
  weighted_col = NULL
)
```

Arguments

X	a data frame with n rows (individuals) and p columns (numeric variables)
ncp	an integer, the number of components to keep (value set by default)
scale.unit	a boolean, if TRUE (value set by default) then data are scaled to unit variance
ind_sup	a vector indicating the indexes of the supplementary individuals
quanti_sup	a vector indicating the indexes of the quantitative supplementary variables
weighted_col	column weights

Value

A list containing results of FactoMineR's principal components analysis (PCA).

Examples

```
library(booklet)

res <- facto_pca(iris[, -5], ncp = 2, ind_sup = 1, quanti_sup = 1)
```

---

pca_eigen *Compute eigenvalues and eigenvectors*

Description

Return eigenvalues and eigenvectors of a matrix

Usage

```
pca_eigen(X)

pca_weighted_eigen(
  X,
  weighted_row = rep(1, nrow(X))/nrow(X),
  weighted_col = rep(1, ncol(X))
)
```

Arguments

X	X_active
weighted_row	row weights
weighted_col	column weights

Details

Standardization depends on what you need to perform factor analysis. We implemented two types:

- pca_weighted_eigen: This is the default method in FactoMineR to compute eigvalues, eigvectors and U matrix.
- pca_eigen: This is the standard method to compute eigenvalues, eigenvectors.

Value

A list containing results of Single Value Decomposition (SVD).

Examples

```
library(booklet)

iris[, -5] |>
  pca_standardize_norm() |>
  pca_eigen()
```

`pca_ind_contrib` *Compute individual contributions*

Description

Return individual contributions for each principal component

Usage

```
pca_ind_contrib(
  ind_coords,
  eigs,
  weighted_row = rep(1, nrow(ind_coords))/nrow(ind_coords)
)
```

Arguments

`ind_coords`	individual coordinates
`eigs`	eigs computed by `pca_eigen` or `pca_weighted_eigen`
`weighted_row`	row weights

Details

If you want to compute the contributions of the individuals to the principal components, you have to change the weighted_col argument to rep(1, nrow(ind_cos2)).

Value

A dataframe of individual contributions.

Examples

```
library(booklet)

eigs <- iris[, -5] |>
  pca_standardize_norm() |>
  pca_weighted_eigen()

eigs |>
  pca_ind_coords() |>
  pca_ind_contrib(eigs) |>
  head()
```

`pca_ind_coords` *Compute coordinates for individuals*

Description

Return principal component for individuals

Usage

```
pca_ind_coords(eigs)
```

Arguments

`eigs`	eigs computed by `pca_eigen` or `pca_weighted_eigen`

Value

A dataframe of individual coordinates.

Examples

```
library(booklet)

iris[, -5] |>
  pca_standardize_norm() |>
  pca_weighted_eigen() |>
  pca_ind_coords() |>
  head()
```

pca_ind_cos2 *Compute individual squared cosines*

Description

Return individual squared cosines for each principal component

Usage

```
pca_ind_cos2(ind_coords, weighted_col = rep(1, ncol(ind_coords)))
```

Arguments

ind_coords	individual coordinates
weighted_col	column weights

Value

A dataframe of individual squared cosines.

Examples

```
library(booklet)

iris[, -5] |>
  pca_standardize_norm() |>
  pca_weighted_eigen() |>
  pca_ind_coords() |>
  pca_ind_cos2() |>
  head()
```

pca_standardize_norm *Data standardization for PCA*

Description

Perform data standardization for multivariate exploratory data analysis.

Usage

```
pca_standardize_norm(X, center = TRUE, scale = TRUE)

pca_standardize(X, scale = TRUE, weighted_row = rep(1, nrow(X))/nrow(X))
```

Arguments

X	matrix
center	centering by the mean
scale	scaling by the standard deviation
weighted_row	row weights

Details

Standardization depends on what you need to perform factor analysis. Two methods are implemented:

- standardize: standardization is performed by centering the data matrix and dividing by the square root of the sum of squares of the weights. This is the same method used in FactoMineR::PCA().
- standardize_norm: standardization is performed by centering and scaling the data matrix. (X - μ) / S, where μ is the mean and S is the standard deviation.

Value

A dataframe of the same size as X.

Examples

```
library(booklet)

iris[, -5] |>
  pca_standardize_norm() |>
  head()
```

---

pca_var_contrib *Compute variable contributions*

Description

Return variable contributions

Usage

```
pca_var_contrib(var_cos2, eigs, weighted_col = rep(1, ncol(var_cos2)))
```

Arguments

var_cos2	variable coordinates
eigs	eigs computed by pca_eigen or pca_weighted_eigen
weighted_col	column weights

Value

A dataframe of variable contributions.

Examples

```
library(booklet)

eigs <- iris[, -5] |>
  pca_standardize_norm() |>
  pca_weighted_eigen()

eigs |>
  pca_var_coords() |>
  pca_var_cos2() |>
  pca_var_contrib(eigs) |>
  head()
```

---

pca_var_coords *Compute variable coordinates*

Description

Return variable coordinates

Usage

```
pca_var_coords(eigs)
```

Arguments

eigs	eigs computed by pca_eigen or pca_weighted_eigen

Value

A dataframe of variable coordinates.

Examples

```
library(booklet)

iris[, -5] |>
  pca_standardize_norm() |>
  pca_weighted_eigen() |>
  pca_var_coords() |>
  head()
```

pca_var_cor *Compute variable correlation*

Description

Return variable correlation

Usage

```
pca_var_cor(eigs)
```

Arguments

eigs	eigs computed by pca_eigen or pca_weighted_eigen

Value

A dataframe of variable correlation.

Examples

```
library(booklet)

iris[, -5] |>
  pca_standardize_norm() |>
  pca_weighted_eigen() |>
  pca_var_cor() |>
  head()
```

pca_var_cos2 *Compute variable squared cosines*

Description

Return variable squared cosines

Usage

```
pca_var_cos2(var_coords)
```

Arguments

var_coords	variable coordinates

Value

A dataframe of variable squared consines.

Examples

```
library(booklet)

iris[, -5] |>
  pca_standardize_norm() |>
  pca_weighted_eigen() |>
  pca_var_coords() |>
  pca_var_cos2() |>
  head()
```

Index