

Package ‘pla’

October 10, 2021

Type Package

Title Principle Loading Analysis

Version 0.1.0

Description This package provides principle loading analysis, a method for reducing the number of features within a data set.

License MIT

Imports methods, Rdpack

RdMacros Rdpack

Encoding UTF-8

LazyData true

URL <https://github.com/Ronho/pla>

BugReports <https://github.com/Ronho/pla/issues>

RoxygenNote 7.1.1.9001

Collate 'block.R'

'cor.R'

'explained-variance.R'

'get-blocks.R'

'pla-package.R'

'thresholding.R'

'scale.R'

'utils.R'

'pla.R'

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

R topics documented:

Block-class	2
pla	2
pla.drop_blocks	3
pla.keep_blocks	4
print.pla	5
show,Block-method	6
str,Block-method	6
Index	7

Block-class	<i>Block</i>
-------------	--------------

Description

Class used within the package to keep the structure and information about the generated blocks.

Slots

`features` a vector of numeric which contains the indices of the block.

`explained_varaiance` a numeric which contains the variance explained of the blocks variables based on the whole data set.

`is_valid` a logical which indicates if the block structure is valid.

pla	<i>Principle Loading Analysis This function performs principle loading analysis on a data set.</i>
-----	----------------------------------------------------------------------------------------------------

Description

Principle Loading Analysis

This function performs principle loading analysis on a data set.

Usage

```
pla(
  x,
  cor = FALSE,
  scaled_ev = FALSE,
  thresholds = 0.33,
  threshold_mode = "cutoff",
  expvar = "approx",
  check = "rnc",
  ...
)
```

Arguments

<code>x</code>	a numeric matrix or data frame which provides the data for the principal loading analysis.
<code>cor</code>	a logical value indicating whether the calculation should use the correlation or the covariance matrix. The default is set to "TRUE" indicating the use of the correlation matrix.
<code>scaled_ev</code>	a logical value indicating whether the eigenvectors should be scaled. The default is set to "FALSE".
<code>thresholds</code>	a numeric or list of numeric used to determine "small" values inside the eigenvectors. If multiple values are given, a list of pla results will be returned. The default is set to 0.33.

threshold_mode	a character string indicating how the threshold is determined and used. "cutoff" indicates that the threshold value is used as a general maximum for all elements. "percentage" indicates that the cutoff value is determined by the maximum element of each vector multiplied with the threshold value. The default is set to "cutoff".
expvar	a character string indicating the method used for calculating the explained variance. "approx" indicates the use of an approximation. "exact" indicates the exact calculation. The default is set to "approx".
check	a character string indicating if only rows or rows and columns are used. "rows" checks if the rows fulfill the required structure. "rnc" checks if rows and columns fulfill the required structure. The default is set to "rnc".
...	further arguments passed to or from other methods.

Value

single or list of pla class containing the following attributes:

x	a numeric matrix or data frame which equals the input of 'x'.
c	a numeric matrix or data frame which is the covariance or correlation matrix based on the input of 'cov'.
loadings	a matrix of variable loadings (i.e., a matrix whose columns contain the eigenvectors).
threshold	a numeric value which equals the input of thresholds at the corresponding position.
threshold_mode	a character string which equals the input of threshold_mode.
blocks	a list of blocks which are identified through the principle loading analysis.

See <https://arxiv.org/pdf/2007.05215.pdf>, <https://arxiv.org/pdf/2102.09912.pdf> for more information.

Examples

```
data <- data.frame(
  a = c(1:3),
  b = c(4:6),
  c = c(7:9)
)
pla(data)
```

pla.drop_blocks	<i>Drop Blocks Used to remove each variable from the original data set which is part of any of the blocks according to the passed indices.</i>
-----------------	------------------------------------------------------------------------------------------------------------------------------------------------

Description

Drop Blocks

Used to remove each variable from the original data set which is part of any of the blocks according to the passed indices.

Usage

```
pla.drop_blocks(object, blocks, ...)
```

Arguments

object	a pla object.
blocks	a list of numeric values indicating the indices of the blocks that should be removed.
...	further arguments passed to or from other methods.

Value

list of the following attributes:

x	a numeric matrix or data frame which equals the input data for the pla object without any feature that is not part of the blocks that should be removed.
cc_matrix	a numeric matrix or data frame which contains either the conditional covariance or correlation matrix.

Examples

```
data <- data.frame(
  a = c(1:3),
  b = c(4:6),
  c = c(7:9),
  d = c(10:12)
)
obj <- pla(data)
data <- pla.drop_blocks(obj, c(1))
```

pla.keep_blocks	<i>Keep Blocks Used to only keep each variable of the original data set which is part of any of the blocks according to the passed indices.</i>
-----------------	-------------------------------------------------------------------------------------------------------------------------------------------------

Description

Keep Blocks

Used to only keep each variable of the original data set which is part of any of the blocks according to the passed indices.

Usage

```
pla.keep_blocks(object, blocks, ...)
```

Arguments

object	a pla object.
blocks	a list of numeric values indicating the indices of the blocks that should be kept.
...	further arguments passed to or from other methods.

Value

list of the following attributes:

x	a numeric matrix or data frame which equals the input data for the pla object without any feature that is not part of the blocks that should be kept.
cc_matrix	a numeric matrix or data frame which contains either the conditional covariance or correlation matrix.

Examples

```
data <- data.frame(
  a = c(1:3),
  b = c(4:6),
  c = c(7:9),
  d = c(10:12)
)
obj <- pla(data)
data <- pla.keep_blocks(obj, c(1))
```

print.pla	<i>Print Function for pla S3 Prints the blocks, threshold, threshold_mode and the loadings.</i>
-----------	-------------------------------------------------------------------------------------------------

Description

Print Function for pla S3

Prints the blocks, threshold, threshold_mode and the loadings.

Usage

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

Arguments

x	a pla object.
...	further arguments passed to or from other methods.

Examples

```
data <- data.frame(
  a = c(1:3),
  b = c(4:6),
  c = c(7:9)
)
obj <- pla(data)
print(obj)
```

show,Block-method	<i>Block - Show Prints the blocks structure.</i>
-------------------	--------------------------------------------------

Description

Block - Show
Prints the blocks structure.

Usage

```
## S4 method for signature 'Block'
show(object)
```

Arguments

object block.

Examples

```
block <- new("Block", features = c(2, 5), explained_variance = 0.03)
print(block)
```

str,Block-method	<i>Block - str Generic function to create a string out of the blocks structure.</i>
------------------	-------------------------------------------------------------------------------------

Description

Block - str
Generic function to create a string out of the blocks structure.

Usage

```
## S4 method for signature 'Block'
str(object)
```

Arguments

object block.

Examples

```
block <- new("Block", features = c(2, 5), explained_variance = 0.03)
str(block)
```

Index

Block-class, [2](#)

pla, [2](#)

pla.drop_blocks, [3](#)

pla.keep_blocks, [4](#)

print.pla, [5](#)

show, Block-method, [6](#)

str, Block-method, [6](#)