Package 'diagonals'

August 29, 2016

Title Block Diagonal Extraction or Replacement

Version 0.4.0

Description Several tools for handling block-matrix diagonals and similar constructs are implemented. Block-diagonal matrices can be extracted or removed using two small functions implemented here. In addition, non-square matrices are supported. Block diagonal matrices occur when two dimensions of a data set are combined along one edge of a matrix. For example, trade-flow data in the 'decompr' and 'gvc' packages have each country-industry combination occur along both edges of the matrix.

Depends R (>= 2.10)

License GPL-3

LazyData true

URL http://qua.st/diagonals, https://github.com/bquast/diagonals

BugReports https://github.com/bquast/diagonals/issues

Suggests testthat, knitr

VignetteBuilder knitr

NeedsCompilation no

Author Bastiaan Quast [aut, cre]

Maintainer Bastiaan Quast <bquast@gmail.com>

Repository CRAN

Date/Publication 2015-10-05 12:58:14

R topics documented:

diagonals *diagonals*

Description

diagonals

Author(s)

Bastiaan Quast <bquast@gmail.com>

See Also

http://qua.st/diagonals

fatdiag *Fat Matrix Diagonals*

Description

Fat Matrix Diagonals

fatdiag set

Usage

```
fatdiag(x = 1, steps = NULL, size = NULL, nrow = NULL, ncol = NULL)

fatdiag(x, steps = NULL, size = NULL, on_diagonal = TRUE) <- value
```

Arguments

x	a matrix where the dimensions are integer multiples of size or integer dividors of steps
steps	the required number of steps (block matrices) across the diagonal
size	the width or height of the matrix being dropped over the diagonal of matrix x
nrow	the number of rows
ncol	the number of columns
on_diagonal	should the operation be apply to the elements on the fat diagonal.
value	replacement value

Details

Either steps or size is expected to be provided.

Functions

- `fatdiag<-`: the set version of fatdiag

Examples

```
fatdiag(12, steps=3)

( m <- matrix(111, nrow=6, ncol=9) )
fatdiag(m, steps=3) <- 5

fatdiag(m, steps=3)

fatdiag(12, size=4)

fatdiag(12, size=c(3,4) )
```

`matricise` *Matricise*

Description

Matricise

Usage

```
matricise(x, row_dim = c(NULL, 3, 4), col_dim = c(NULL, 3, 4))
```

Arguments

`x`	a higher-order array (length(dim(x)) >= 3)
`row_dim`	the input array dimension which should be added to the row dimension of the output matrix, the value has to be 3 or 4.
`col_dim`	the input array dimension which should be added to the column dimension of the output matrix, the value has to be 3 or 4.

Value

a matrix (length(dim(x)) == 2)

`split_vector` *Split Vector*

Description

Split Vector

Usage

```
split_vector(x, steps = NULL, size = NULL, replacement = 0)
```

Arguments

`x`	a numeric or character vector
`steps`	the number of steps
`size`	the size of the step
`replacement`	value to be inserted on the diagonal, by default this is zero (0).

Details

Either steps or size is expected to be provided.

Index