Package ‘MatrixGenerics’

May 18, 2024

Title S4 Generic Summary Statistic Functions that Operate on Matrix-Like Objects

Description S4 generic functions modeled after the ‘matrixStats’ API for alternative matrix implementations. Packages with alternative matrix implementation can depend on this package and implement the generic functions that are defined here for a useful set of row and column summary statistics. Other package developers can import this package and handle a different matrix implementations without worrying about incompatibilities.

biocViews Infrastructure, Software

URL https://bioconductor.org/packages/MatrixGenerics

BugReports https://github.com/Bioconductor/MatrixGenerics/issues

Version 1.16.0

License Artistic-2.0

Encoding UTF-8

Depends matrixStats (>= 1.0.0)

Imports methods

Suggests Matrix, sparseMatrixStats, SparseArray, DelayedArray, DelayedMatrixStats, SummarizedExperiment, testthat (>= 2.1.0)

RoxygenNote 7.3.1

Roxygen list(markdown = TRUE, old_usage = TRUE)

Collate 'MatrixGenerics-package.R' 'rowAlls.R' 'rowAnyNAs.R'
  'rowAnys.R' 'rowAvgsPerColSet.R' 'rowCollapse.R' 'rowCounts.R'
  'rowCummmaxs.R' 'rowCummmins.R' 'rowCumpords.R' 'rowCumsums.R'
  'rowDiffs.R' 'rowIQRDiffs.R' 'rowIQRs.R' 'rowLogSumExps.R'
  'rowMadDiffs.R' 'rowMads.R' 'rowMaxs.R' 'rowMeans.R'
  'rowMeans2.R' 'rowMedians.R' 'rowMins.R' 'rowOrderStats.R'
  'rowProds.R' 'rowQuantiles.R' 'rowRanges.R' 'rowRanks.R'
  'rowSdDiffs.R' 'rowSds.R' 'rowSums.R' 'rowSums2.R'
  'rowTabulates.R' 'rowVarDiffs.R' 'rowVars.R'
  'rowWeightedMads.R' 'rowWeightedMeans.R' 'rowWeightedMedians.R'
  'rowWeightedSds.R' 'rowWeightedVars.R'
git_url https://git.bioconductor.org/packages/MatrixGenerics

Contents

MatrixGenerics-package .................................................. 3
internal-helpers ............................................................ 3
rowAlls ................................................................. 3
rowAnyNAs .............................................................. 5
rowAnyys ............................................................... 6
rowAvgsPerColSet ...................................................... 8
rowCollapse ............................................................. 10
rowCounts ............................................................... 11
rowCummaxs ............................................................ 13
rowCummins ............................................................ 14
rowCumprods ........................................................... 16
rowCumsums ............................................................ 17
rowDiffs ................................................................. 19
rowIQRDiffs ............................................................. 20
rowIQRs ................................................................. 22
rowLogSumExps ........................................................ 23
rowMadDiffs ............................................................ 25
rowMads ................................................................. 26
rowMaxs ................................................................. 28
rowMeans ............................................................... 30
rowMeans2 .............................................................. 31
rowMedians ............................................................. 32
rowMins ................................................................. 34
rowOrderStats .......................................................... 35
rowProds ............................................................... 37
rowQuantiles ........................................................... 38
rowRanges .............................................................. 40
rowRanks ............................................................... 42
rowSdDiffs ............................................................. 44
rowSds ................................................................. 45
rowSums ............................................................... 47
rowSums2 .............................................................. 48
MatrixGenerics-package

The MatrixGenerics package

Description

The MatrixGenerics package defines S4 generic summary statistic functions that operate on matrix-Like objects.

internal-helpers

Internal helpers

Description

Not for end users

Usage

normarg_center(center, n, what)

rowAlls

Check if all elements in a row (column) of a matrix-like object are equal to a value

Description

Check if all elements in a row (column) of a matrix-like object are equal to a value.
rowAlls(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowAlls(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowAlls(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, ..., useNames = TRUE)

colAlls(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colAlls(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colAlls(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, ..., useNames = TRUE)

Arguments

x        An NxK matrix-like object.
rows, cols A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
value    The value to search for.
na.rm    If TRUE, missing values (NA or NaN) are omitted from the calculations.
...      Additional arguments passed to specific methods.
useNames If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.
dim.     An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowAlls / matrixStats::colAlls.

Value

Returns a logical vector of length N (K).
See Also

- `matrixStats::rowAlls()` and `matrixStats::colAlls()` which are used when the input is a matrix or numeric vector.
- For checks if any element is equal to a value, see `rowAnys()`.
- `base::all()`.

Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowAlls(mat)
colAlls(mat)
```

---

**rowAnyNAs**

Check if any elements in a row (column) of a matrix-like object is missing

### Description

Check if any elements in a row (column) of a matrix-like object is missing.

### Usage

```r
rowAnyNAs(x, rows = NULL, cols = NULL, ..., useNames = TRUE)
```

```r
## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowAnyNAs(x, rows = NULL, cols = NULL, ..., useNames = TRUE)
```

```r
## S4 method for signature 'ANY'
rowAnyNAs(x, rows = NULL, cols = NULL, ..., useNames = TRUE)
```

```r
colAnyNAs(x, rows = NULL, cols = NULL, ..., useNames = TRUE)
```

```r
## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colAnyNAs(x, rows = NULL, cols = NULL, ..., useNames = TRUE)
```

```r
## S4 method for signature 'ANY'
colAnyNAs(x, rows = NULL, cols = NULL, ..., useNames = TRUE)
```
Arguments

- **x**: An NxK matrix-like object.
- **rows, cols**: A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
- **...**: Additional arguments passed to specific methods.
- **useNames**: If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowAnyNAs / matrixStats::colAnyNAs.

Value

Returns a logical vector of length N (K).

See Also

- matrixStats::rowAnyNAs() and matrixStats::colAnyNAs() which are used when the input is a matrix or numeric vector.
- For checks if any element is equal to a value, see rowAnys().
- base::is.na() and base::any().

Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)

rowAnyNAs(mat)
colAnyNAs(mat)
```

---

| rowAnys | Check if any elements in a row (column) of a matrix-like object is equal to a value |

Description

Check if any elements in a row (column) of a matrix-like object is equal to a value.
Usage

rowAnys(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowAnys(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowAnys(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, ..., useNames = TRUE)

colAnys(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colAnys(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colAnys(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, ..., useNames = TRUE)

Arguments

- **x**: An N x K matrix-like object.
- **rows, cols**: A vector indicating the subset of rows (and/or columns) to operate over. If **NULL**, no subsetting is done.
- **value**: The value to search for.
- **na.rm**: If **TRUE**, missing values (**NA** or **NaN**) are omitted from the calculations.
- **...**: Additional arguments passed to specific methods.
- **useNames**: If **TRUE** (default), names attributes of result are set. Else if **FALSE**, no naming support is done.
- **dim.**: An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.

Details

The S4 methods for x of type **matrix, array, table, or numeric** call `matrixStats::rowAnys` / `matrixStats::colAnys`.

Value

Returns a logical vector of length N (K).
rowAvgsPerColSet

See Also

- matrixStats::rowAnys() and matrixStats::colAnys() which are used when the input is a matrix or numeric vector.
- For checks if all elements are equal to a value, see rowAlls().
- base::any().

Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowAnys(mat)
colAnys(mat)
```

Description

Calculates for each row (column) a summary statistic for equally sized subsets of columns (rows).

Usage

```r
rowAvgsPerColSet(X, W = NULL, rows = NULL, S, FUN = rowMeans, ..., na.rm = NA, tFUN = FALSE)
```

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowAvgsPerColSet(X, W = NULL, rows = NULL, S, FUN = rowMeans, ..., na.rm = NA, tFUN = FALSE)

## S4 method for signature 'ANY'
rowAvgsPerColSet(X, W = NULL, rows = NULL, S, FUN = rowMeans, ..., na.rm = NA, tFUN = FALSE)

colAvgsPerRowSet(X, W = NULL, cols = NULL, S, FUN = colMeans, ..., na.rm = NA, tFUN = FALSE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colAvgsPerRowSet(X, W = NULL, cols = NULL, S, FUN = colMeans, ..., na.rm = NA, tFUN = FALSE)

## S4 method for signature 'ANY'
colAvgsPerRowSet(X, W = NULL, cols = NULL, S, FUN = colMeans, ..., na.rm = NA, tFUN = FALSE)
rowAvgsPerColSet

\[ \text{rowAvgsPerColSet}(X, W = \text{NULL}, \text{cols} = \text{NULL}, S, \]
\[ \text{FUN} = \text{colMeans}, \ldots, \text{na.rm} = \text{NA}, \text{tFUN} = \text{FALSE}) \]

Arguments

- **X**: An \( N \times M \) matrix-like object.
- **W**: An optional numeric \( N \times M \) matrix of weights.
- **rows, cols**: A vector indicating the subset (and/or columns) to operate over. If \( \text{NULL} \), no subsetting is done.
- **S**: An integer \( K \times J \) matrix that specifying the \( J \) subsets. Each column hold \( K \) column (row) indices for the corresponding subset. The range of values is \([1, M]\) (\([1, N]\)).
- **FUN**: A row-by-row (column-by-column) summary statistic function. It is applied to each column (row) subset of \( X \) that is specified by \( S \).
- **...**: Additional arguments passed to \( \text{FUN}() \).
- **na.rm** (logical) Argument passed to \( \text{FUN}() \) as \( \text{na.rm} = \text{na.rm} \). If \( \text{NA} \) (default), then \( \text{na.rm} = \text{TRUE} \) is used if \( X \) or \( S \) holds missing values, otherwise \( \text{na.rm} = \text{FALSE} \).
- **tFUN**: If \( \text{TRUE} \), \( X \) is transposed before it is passed to \( \text{FUN} \).

Details

The S4 methods for \( x \) of type \text{matrix, array, table,} or \text{numeric} call \text{matrixStats::rowAvgsPerColSet} \text{/ matrixStats::colAvgsPerRowSet}.

Value

Returns a numeric \( J \times N \) (\( M \times J \)) matrix.

See Also

- \text{matrixStats::rowAvgsPerColSet()} and \text{matrixStats::colAvgsPerRowSet()} which are used when the input is a matrix or numeric vector.

Examples

```r
mat <- matrix(rnorm(20), nrow = 5, ncol = 4)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0
print(mat)

S <- matrix(1:ncol(mat), ncol = 2)
print(S)

rowAvgsPerColSet(mat, S = S, FUN = rowMeans)
rowAvgsPerColSet(mat, S = S, FUN = rowVars)
```
rowCollapse  

Extract one cell from each row (column) of a matrix-like object

Description

Extract one cell from each row (column) of a matrix-like object.

Usage

rowCollapse(x, idxs, rows = NULL, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowCollapse(x, idxs,
  rows = NULL, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowCollapse(x, idxs, rows = NULL, ..., useNames = TRUE)

colCollapse(x, idxs, cols = NULL, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colCollapse(x, idxs,
  cols = NULL, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colCollapse(x, idxs, cols = NULL, ..., useNames = TRUE)

Arguments

x        An NxK matrix-like object.
idxs     An index vector with the position to extract. It is recycled to match the number
         of rows (column)
rows, cols A vector indicating the subset of rows (and/or columns) to operate over. If
           NULL, no subsetting is done.
...      Additional arguments passed to specific methods.
useNames If TRUE (default), names attributes of result are set. Else if FALSE, no naming
         support is done.
dim.     An integer vector of length two specifying the dimension of x, essential when
         x is a numeric vector. Note, that this is not a generic argument and not all
         methods need provide it.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowCollapse
/matrixStats::colCollapse.
rowCounts

Value
Returns a numeric vector of length N (K).

See Also
- matrixStats::rowCollapse() and matrixStats::colCollapse() which are used when the input is a matrix or numeric vector.

Examples
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)

rowCollapse(mat, idxs = 2)
rowCollapse(mat, idxs = c(1, 1, 2, 3, 2))

colCollapse(mat, idxs = 4)

rowCounts

Count how often an element in a row (column) of a matrix-like object is equal to a value

Description
Count how often an element in a row (column) of a matrix-like object is equal to a value.

Usage
rowCounts(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, ...,
useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowCounts(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE, dim. = dim(x), ...,
useNames = TRUE)

## S4 method for signature 'ANY'
rowCounts(x, rows = NULL, cols = NULL, value = TRUE,
na.rm = FALSE, ...,
useNames = TRUE)

colCounts(x, rows = NULL, cols = NULL, value = TRUE, na.rm = FALSE,
...,
useNames = TRUE)
## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colCounts(x, rows = NULL,
cols = NULL, value = TRUE, na.rm = FALSE, dim. = dim(x), ...
useNames = TRUE)

## S4 method for signature 'ANY'
colCounts(x, rows = NULL, cols = NULL, value = TRUE,
na.rm = FALSE, ..., useNames = TRUE)

### Arguments

- **x**: An N\times K matrix-like object.
- **rows, cols**: A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
- **value**: The value to search for.
- **na.rm**: If TRUE, missing values (NA or NaN) are omitted from the calculations.
- **...**: Additional arguments passed to specific methods.
- **useNames**: If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.
- **dim.**: An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.

### Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowCounts / matrixStats::colCounts.

### Value

Returns a integer vector of length N (K).

### See Also

- matrixStats::rowCounts() and matrixStats::colCounts() which are used when the input is a matrix or numeric vector.
- For checks if any element is equal to a value, see rowAnys(). To check if all elements are equal, see rowAlls().

### Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <= 0

print(mat)
```
rowCummaxs

Calculates the cumulative maxima for each row (column) of a matrix-like object

Description

Calculates the cumulative maxima for each row (column) of a matrix-like object.

Usage

rowCummaxs(x, rows = NULL, cols = NULL, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowCummaxs(x, rows = NULL, cols = NULL, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowCummaxs(x, rows = NULL, cols = NULL, ..., useNames = TRUE)

colCummaxs(x, rows = NULL, cols = NULL, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colCummaxs(x, rows = NULL, cols = NULL, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colCummaxs(x, rows = NULL, cols = NULL, ..., useNames = TRUE)

Arguments

x          An NxK matrix-like object.
rows, cols  A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
...         Additional arguments passed to specific methods.
useNames    If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.
dim.        An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.
rowCummins

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowCummaxs /matrixStats::colCummaxs.

Value

Returns a numeric matrix with the same dimensions as x.

See Also

- matrixStats::rowCummaxs() and matrixStats::colCummaxs() which are used when the input is a matrix or numeric vector.
- For single maximum estimates, see rowMaxs().
- base::cummax().

Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowCummaxs(mat)
colCummaxs(mat)
```

rowCummins

Calculates the cumulative minima for each row (column) of a matrix-like object

Description

Calculates the cumulative minima for each row (column) of a matrix-like object.

Usage

rowCummins(x, rows = NULL, cols = NULL, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowCummins(x, rows = NULL, cols = NULL, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowCummins(x, rows = NULL, cols = NULL, ..., useNames = TRUE)
colCummins(x, rows = NULL, cols = NULL, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

colCummins(x, rows = NULL, cols = NULL, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'

colCummins(x, rows = NULL, cols = NULL, ..., useNames = TRUE)

Arguments

x
An NxK matrix-like object.

rows, cols
A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.

...
Additional arguments passed to specific methods.

useNames
If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

dim.
An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowCummins / matrixStats::colCummins.

Value

Returns a numeric matrix with the same dimensions as x.

See Also

- matrixStats::rowCummins() and matrixStats::colCummins() which are used when the input is a matrix or numeric vector.
- For single minimum estimates, see rowMins().
- base::cummin().

Examples

mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)

rowCummins(mat)
colCummins(mat)
Description

Calculates the cumulative product for each row (column) of a matrix-like object.

Usage

rowCumprods(x, rows = NULL, cols = NULL, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowCumprods(x, rows = NULL, 
  cols = NULL, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowCumprods(x, rows = NULL, cols = NULL, ..., 
  useNames = TRUE)

colCumprods(x, rows = NULL, cols = NULL, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colCumprods(x, rows = NULL, 
  cols = NULL, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colCumprods(x, rows = NULL, cols = NULL, ..., 
  useNames = TRUE)

Arguments

x An NxK matrix-like object.
rows, cols A vector indicating the subset of rows (and/or columns) to operate over. If
  NULL, no subsetting is done.
... Additional arguments passed to specific methods.
useNames If TRUE (default), names attributes of result are set. Else if FALSE, no naming
  support is done.
dim. An integer vector of length two specifying the dimension of x, essential when
  x is a numeric vector. Note, that this is not a generic argument and not all
  methods need provide it.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowCumprods
  /matrixStats::colCumprods.
rowCumsums

**Value**

Returns a numeric matrix with the same dimensions as x.

**See Also**

- matrixStats::rowCumprods() and matrixStats::colCumprods() which are used when the input is a matrix or numeric vector.
- base::cumprod().

**Examples**

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowCumprods(mat)
colCumprods(mat)
```

**Description**

Calculates the cumulative sum for each row (column) of a matrix-like object.

**Usage**

```r
rowCumsums(x, rows = NULL, cols = NULL, ..., useNames = TRUE)
```

```r
## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowCumsums(x, rows = NULL, cols = NULL, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowCumsums(x, rows = NULL, cols = NULL, ..., useNames = TRUE)
```

```r
colCumsums(x, rows = NULL, cols = NULL, ..., useNames = TRUE)
```

```r
## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colCumsums(x, rows = NULL, cols = NULL, dim. = dim(x), ..., useNames = TRUE)
```

# S4 method for signature 'ANY'
colCumsums(x, rows = NULL, cols = NULL, ...,
    useNames = TRUE)

**Arguments**

- **x** An NxK matrix-like object.
- **rows, cols** A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
- **...** Additional arguments passed to specific methods.
- **useNames** If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.
- **dim.** An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.

**Details**

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowCumsums / matrixStats::colCumsums.

**Value**

Returns a numeric matrix with the same dimensions as x.

**See Also**

- matrixStats::rowCumsums() and matrixStats::colCumsums() which are used when the input is a matrix or numeric vector.
- base::cumsum().

**Examples**

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowCumsums(mat)
colCumsums(mat)
```
rownDiffs

Calculates the difference between each element of a row (column) of a matrix-like object

Description

Calculates the difference between each element of a row (column) of a matrix-like object.

Usage

rownDiffs(x, rows = NULL, cols = NULL, lag = 1L, differences = 1L, 
useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rownDiffs(x, rows = NULL, 
cols = NULL, lag = 1L, differences = 1L, dim. = dim(x), 
useNames = TRUE)

## S4 method for signature 'ANY'
rownDiffs(x, rows = NULL, cols = NULL, lag = 1L, 
differences = 1L, ..., useNames = TRUE)

colDiffs(x, rows = NULL, cols = NULL, lag = 1L, 
differences = 1L, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colDiffs(x, rows = NULL, 
cols = NULL, lag = 1L, differences = 1L, dim. = dim(x), 
useNames = TRUE)

## S4 method for signature 'ANY'
colDiffs(x, rows = NULL, cols = NULL, lag = 1L, 
differences = 1L, ..., useNames = TRUE)

Arguments

- **x**: An NxK matrix-like object.
- **rows, cols**: A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
- **lag**: An integer specifying the lag.
- **differences**: An integer specifying the order of difference.
- **...**: Additional arguments passed to specific methods.
- **useNames**: If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.
- **dim.**: An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.
Details

The S4 methods for `x` of type `matrix`, `array`, `table`, or `numeric` call `matrixStats::rowDiffs / matrixStats::colDiffs`.

Value

Returns a numeric matrix with one column (row) less than `x`: \( N x (K - 1) \) or \( (N - 1) x K \).

See Also

- `matrixStats::rowDiffs()` and `matrixStats::colDiffs()` which are used when the input is a matrix or numeric vector.
- `base:::diff()`.

Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowDiffs(mat)
colDiffs(mat)
```

---

**rowIQRDiffs**

Calculates the interquartile range of the difference between each element of a row (column) of a matrix-like object

Description

Calculates the interquartile range of the difference between each element of a row (column) of a matrix-like object.

Usage

```r
rowIQRDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L,
            trim = 0, ..., useNames = TRUE)
```

```r
## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowIQRDiffs(x, rows = NULL,
            cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ...,
            useNames = TRUE)

## S4 method for signature 'ANY'
rowIQRDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE,
```
Arguments

- **x**: AnNxK matrix-like object.
- **rows, cols**: A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
- **na.rm**: If TRUE, missing values (NA or NaN) are omitted from the calculations.
- **diff**: An integer specifying the order of difference.
- **trim**: A double in [0,1/2] specifying the fraction of observations to be trimmed from each end of (sorted) x before estimation.
- **...**: Additional arguments passed to specific methods.
- **useNames**: If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowIQRDiffs / matrixStats::colIQRDiffs.

Value

Returns a numeric vector of length N (K).

See Also

- matrixStats::rowIQRDiffs() and matrixStats::colIQRDiffs() which are used when the input is a matrix or numeric vector.
- For the direct interquartile range see also rowIQRs.

Examples

```R
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0
```
rowIQRs(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowIQRs(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowIQRs(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

colIQRs(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colIQRs(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'ANY'
colIQRs(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

Arguments

x                An NxK matrix-like object.
rows, cols       A vector indicating the subset of rows (and/or columns) to operate over. If
                 NULL, no subsetting is done.
na.rm            If TRUE, missing values (NA or NaN) are omitted from the calculations.
...              Additional arguments passed to specific methods.
useNames         If TRUE (default), names attributes of result are set. Else if FALSE, no naming
                 support is done.
rowLogSumExps

Details
The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowIQRs / matrixStats::colIQRs.

Value
Returns a numeric vector of length N (K).

See Also
- matrixStats::rowIQRs() and matrixStats::colIQRs() which are used when the input is a matrix or numeric vector.
- For a non-robust analog, see rowSds(). For a more robust version see rowMads().
- stats::IQR().

Examples
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowIQRs(mat)
colIQRs(mat)

rowLogSumExps

Description
Accurately calculates the logarithm of the sum of exponentials for each row (column) of a matrix-like object.

Usage
rowLogSumExps(lx, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowLogSumExps(lx, rows = NULL, cols = NULL, na.rm = FALSE, dim. = dim(lx), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowLogSumExps(lx, rows = NULL, cols = NULL,
   na.rm = FALSE, ..., useNames = TRUE)

colLogSumExps(lx, rows = NULL, cols = NULL, na.rm = FALSE, ...
   useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colLogSumExps(lx,
   rows = NULL, cols = NULL, na.rm = FALSE, dim. = dim(lx), ...
   useNames = TRUE)

## S4 method for signature 'ANY'
colLogSumExps(lx, rows = NULL, cols = NULL,
   na.rm = FALSE, ..., useNames = TRUE)

Arguments

lx          An NxK matrix-like object. Typically lx are \log(x) values.
rows, cols  A vector indicating the subset (and/or columns) to operate over. If NULL, no subsetting is done.
na.rm       If TRUE, missing values (NA or NaN) are omitted from the calculations.
...          Additional arguments passed to specific methods.
useNames    If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.
dim.        An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowLogSumExps / matrixStats::colLogSumExps.

Value

Returns a numeric vector of length N (K).

See Also

- matrixStats::rowLogSumExps() and matrixStats::colLogSumExps() which are used when the input is a matrix or numeric vector.
- rowSums2()

Examples

mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
**rowMadDiffs**

Calculates the mean absolute deviation of the difference between each element of a row (column) of a matrix-like object.

### Description

Calculates the mean absolute deviation of the difference between each element of a row (column) of a matrix-like object.

### Usage

```r
rowMadDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
```

#### S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

```r
rowMadDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
```

#### S4 method for signature 'ANY'

```r
rowMadDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
```

```r
colMadDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
```

#### S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

```r
colMadDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
```

#### S4 method for signature 'ANY'

```r
colMadDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
```

### Arguments

- **x**: An NxK matrix-like object.
- **rows, cols**: A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
na.rm  If TRUE, missing values (NA or NaN) are omitted from the calculations.
diff   An integer specifying the order of difference.
trim   A double in [0,1/2] specifying the fraction of observations to be trimmed from each end of (sorted) x before estimation.
...    Additional arguments passed to specific methods.
useNames If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowMadDiffs / matrixStats::colMadDiffs.

Value

Returns a numeric vector of length N (K).

See Also

• matrixStats::rowMadDiffs() and matrixStats::colMadDiffs() which are used when the input is a matrix or numeric vector.

Examples

mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowMadDiffs(mat)
colMadDiffs(mat)

rowMads                                                                 Calculates the median absolute deviation for each row (column) of a matrix-like object

Description

Calculates the median absolute deviation for each row (column) of a matrix-like object.
rowMads

Usage

rowMads(x, rows = NULL, cols = NULL, center = NULL, constant = 1.4826, 
        na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowMads(x, rows = NULL, 
        cols = NULL, center = NULL, constant = 1.4826, na.rm = FALSE, 
        dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowMads(x, rows = NULL, cols = NULL, center = NULL, 
        constant = 1.4826, na.rm = FALSE, ..., useNames = TRUE)

colMads(x, rows = NULL, cols = NULL, center = NULL, constant = 1.4826, 
        na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colMads(x, rows = NULL, 
        cols = NULL, center = NULL, constant = 1.4826, na.rm = FALSE, 
        dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colMads(x, rows = NULL, cols = NULL, center = NULL, 
        constant = 1.4826, na.rm = FALSE, ..., useNames = TRUE)

Arguments

x        An NxK matrix-like object.
rows, cols A vector indicating the subset of rows (and/or columns) to operate over. If
            NULL, no subsetting is done.
center    (optional) the center, defaults to the row means
constant  A scale factor. See stats::mad() for details.
na.rm     If TRUE, missing values (NA or NaN) are omitted from the calculations.
...       Additional arguments passed to specific methods.
useNames  If TRUE (default), names attributes of result are set. Else if FALSE, no naming
          support is done.
dim.      An integer vector of length two specifying the dimension of x, essential when
          x is a numeric vector. Note, that this is not a generic argument and not all
          methods need provide it.

details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowMads /
matrixStats::colMads.
Value

Returns a numeric vector of length N (K).

See Also

- matrixStats::rowMads() and matrixStats::colMads() which are used when the input is a matrix or numeric vector.
- For mean estimates, see rowMeans2() and rowMeans().
- For non-robust standard deviation estimates, see rowSds().

Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowMads(mat)
colMads(mat)
```

---

**rowMaxs**

*Calculates the maximum for each row (column) of a matrix-like object*

Description

Calculates the maximum for each row (column) of a matrix-like object.

Usage

```r
rowMaxs(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)
```

```
## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowMaxs(x, rows = NULL, cols = NULL, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowMaxs(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

colMaxs(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colMaxs(x, rows = NULL,
```

```r
```
rowMaxs

```r
cols = NULL, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colMaxs(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., 
useNames = TRUE)

Arguments

x
An N x K matrix-like object.

rows, cols
A vector indicating the subset of rows (and/or columns) to operate over. If
NULL, no subsetting is done.

na.rm
If TRUE, missing values (NA or NaN) are omitted from the calculations.

...
Additional arguments passed to specific methods.

useNames
If TRUE (default), names attributes of result are set. Else if FALSE, no naming
support is done.

dim.
An integer vector of length two specifying the dimension of x, essential when
x is a numeric vector. Note, that this is not a generic argument and not all
methods need provide it.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowMaxs/
matrixStats::colMaxs.

Value

Returns a numeric vector of length N (K).

See Also

• matrixStats::rowMaxs() and matrixStats::colMaxs() which are used when the input is
a matrix or numeric vector.
• For min estimates, see rowMins().

Examples

mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)

rowMaxs(mat)
colMaxs(mat)
```
Calculates the mean for each row (column) of a matrix-like object

**Description**

Calculates the mean for each row (column) of a matrix-like object.

**Usage**

```r
rowMeans(x, na.rm = FALSE, dims = 1, ...)
```

```r
colMeans(x, na.rm = FALSE, dims = 1, ...)
```

**Arguments**

- **x**: An NxK matrix-like object, a numeric data frame, or an array-like object of two or more dimensions.
- **na.rm**: If TRUE, missing values (NA or NaN) are omitted from the calculations.
- **dims**: A single integer indicating which dimensions are regarded as rows or columns to mean over. For `rowMeans`, the mean is over dimensions `dims+1, ...`; for `colMeans` it is over dimensions `1:dims`.
- **...**: Additional arguments passed to specific methods.

**Details**

This man page documents the `rowMeans` and `colMeans` S4 generic functions defined in the `Matrix-Generics` package. See `?base::colMeans` for the default methods (defined in the `base` package).

**Value**

Returns a numeric vector of length N (K).

**See Also**

- `base::colMeans` for the default `rowMeans` and `colMeans` methods.
- `Matrix::colMeans` in the `Matrix` package for `rowMeans` and `colMeans` methods defined for `CsparseMatrix` derivatives (e.g. `dgCMatrix` objects).

**Examples**

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0
print(mat)
```
rowMeans2

rowMeans(mat)
colMeans(mat)

---

rowMeans2

*Calculates the mean for each row (column) of a matrix-like object*

Description

Calculates the mean for each row (column) of a matrix-like object.

Usage

```r
rowMeans2(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowMeans2(x, rows = NULL, cols = NULL, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowMeans2(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

colMeans2(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colMeans2(x, rows = NULL, cols = NULL, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colMeans2(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)
```

Arguments

- **x**: An NxK matrix-like object.
- **rows, cols**: A vector indicating the subset of rows (and/or columns) to operate over. If `NULL`, no subsetting is done.
- **na.rm**: If `TRUE`, missing values (NA or NaN) are omitted from the calculations.
- **...**: Additional arguments passed to specific methods.
- **useNames**: If `TRUE` (default), names attributes of result are set. Else if `FALSE`, no naming support is done.
- **dim.**: An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.
The S4 methods for \( x \) of type \texttt{matrix}, \texttt{array}, \texttt{table}, or \texttt{numeric} call \texttt{matrixStats::rowMeans2} / \texttt{matrixStats::colMeans2}.

**Value**

Returns a numeric vector of length \( N(K) \).

**See Also**

- \texttt{matrixStats::rowMeans2()} and \texttt{matrixStats::colMeans2()} which are used when the input is a matrix or numeric vector.
- See also \texttt{rowMeans()} for the corresponding function in base R.
- For variance estimates, see \texttt{rowVars()}.
- See also the base R version \texttt{base::rowMeans()}.

**Examples**

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowMeans2(mat)
colMeans2(mat)
```

---

**rowMedians**

*Calculates the median for each row (column) of a matrix-like object*

**Description**

Calculates the median for each row (column) of a matrix-like object.

**Usage**

```r
rowMedians(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)
```

```r
## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowMedians(x, rows = NULL, cols = NULL, na.rm = FALSE, useNames = TRUE)

## S4 method for signature 'ANY'
rowMedians(x, rows = NULL, cols = NULL, na.rm = FALSE, useNames = TRUE)
```
\texttt{colMedians(x, rows = NULL, cols = NULL, na.rm = FALSE, ...,
useNames = TRUE)}

\texttt{## S4 method for signature 'matrix\_OR\_array\_OR\_table\_OR\_numeric'\n\texttt{colMedians(x, rows = NULL, \cols = NULL, na.rm = FALSE, \dim. = dim(x), ... \useNames = TRUE)\n}}

\texttt{\texttt{## S4 method for signature 'ANY'\ncolMedians(x, rows = NULL, cols = NULL, na.rm = FALSE, \... \useNames = TRUE)\n}}

\textbf{Arguments}

\begin{itemize}
  \item \texttt{x} \hspace{2cm} An N\texttimes{}K matrix-like object.
  \item \texttt{rows, cols} \hspace{1cm} A \texttt{vector} indicating the subset of rows (and/or columns) to operate over. If \texttt{NULL}, no subsetting is done.
  \item \texttt{na.rm} \hspace{1cm} If \texttt{TRUE}, missing values (\texttt{NA} or \texttt{NaN}) are omitted from the calculations.
  \item \texttt{...} \hspace{1cm} Additional arguments passed to specific methods.
  \item \texttt{useNames} \hspace{1cm} If \texttt{TRUE} (default), names attributes of result are set. Else if \texttt{FALSE}, no naming support is done.
  \item \texttt{dim.} \hspace{1cm} An \texttt{integer vector} of length two specifying the dimension of \texttt{x}, essential when \texttt{x} is a \texttt{numeric} vector. Note, that this is not a generic argument and not all methods need provide it.
\end{itemize}

\textbf{Details}

The S4 methods for \texttt{x} of type \texttt{matrix, array, table, or numeric} call \texttt{matrixStats::rowMedians} / \texttt{matrixStats::colMedians}.

\textbf{Value}

Returns a \texttt{numeric vector} of length N (K).

\textbf{See Also}

- \texttt{matrixStats::rowMedians()} and \texttt{matrixStats::colMedians()} which are used when the input is a matrix or numeric vector.
- For mean estimates, see \texttt{rowMeans2()} and \texttt{rowMeans()}.

\textbf{Examples}

\begin{verbatim}
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
\end{verbatim}
rowMins(mat)
colMedians(mat)

**Description**
Calculates the minimum for each row (column) of a matrix-like object.

**Usage**

```r
rowMins(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)
```

## S4 method for signature 'matrix.OR.array.OR.table.OR.numeric'
rowMins(x, rows = NULL, cols = NULL, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowMins(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

colMins(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix.OR.array.OR.table.OR.numeric'
colMins(x, rows = NULL, cols = NULL, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colMins(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

**Arguments**

- **x**: An N×K matrix-like object.
- **rows, cols**: A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
- **na.rm**: If TRUE, missing values (NA or NaN) are omitted from the calculations.
- **...**: Additional arguments passed to specific methods.
- **useNames**: If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.
- **dim.**: An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.
rowOrderStats

Details
The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowMins / matrixStats::colMins.

Value
Returns a numeric vector of length N (K).

See Also
- matrixStats::rowMins() and matrixStats::colMins() which are used when the input is a matrix or numeric vector.
- For max estimates, see rowMaxs().

Examples
```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowMins(mat)
colMins(mat)
```

Description
Calculates an order statistic for each row (column) of a matrix-like object.

Usage
```r
rowOrderStats(x, rows = NULL, cols = NULL, which, ..., useNames = TRUE)
```

```r
## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowOrderStats(x, rows = NULL, cols = NULL, which, dim. = dim(x), ..., useNames = TRUE)
```

```r
## S4 method for signature 'ANY'
rowOrderStats(x, rows = NULL, cols = NULL, which, ..., useNames = TRUE)
```

```r
colOrderStats(x, rows = NULL, cols = NULL, which, ..., useNames = TRUE)
```
## S4 method for signature 'matrix/OR_array/OR_table/OR_numeric'
colOrderStats(x, rows = NULL, 
cols = NULL, which, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colOrderStats(x, rows = NULL, cols = NULL, which, ..., 
useNames = TRUE)

### Arguments

- **x**
  - An NxK matrix-like object.

- **rows, cols**
  - A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.

- **which**
  - An integer index in [1,K] ([1,N]) indicating which order statistic to be returned

- **...**
  - Additional arguments passed to specific methods.

- **useNames**
  - If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

- **dim.**
  - An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.

### Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowOrderStats
/ matrixStats::colOrderStats.

### Value

Returns a numeric vector of length N (K).

### See Also

- matrixStats::rowOrderStats() and matrixStats::colOrderStats() which are used when the input is a matrix or numeric vector.

### Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- 2
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)

rowOrderStats(mat, which = 1)
colOrderStats(mat, which = 3)
```
rowProds

Calculates the product for each row (column) of a matrix-like object

Description

Calculates the product for each row (column) of a matrix-like object.

Usage

rowProds(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowProds(x, rows = NULL, cols = NULL, na.rm = FALSE, method = c("direct", "expSumLog"), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowProds(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

colProds(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colProds(x, rows = NULL, cols = NULL, na.rm = FALSE, method = c("direct", "expSumLog"), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colProds(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

Arguments

x

An NxK matrix-like object.

rows, cols

A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.

na.rm

If TRUE, missing values (NA or NaN) are omitted from the calculations.

...

Additional arguments passed to specific methods.

useNames

If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

method

A character vector of length one that specifies the how the product is calculated. Note, that this is not a generic argument and not all implementation have to provide it.
The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowProds / matrixStats::colProds.

Returns a numeric vector of length N (K).

- matrixStats::rowProds() and matrixStats::colProds() which are used when the input is a matrix or numeric vector.
- For sums across rows (columns), see rowSums2() (colSums2())
- base::prod().

Examples
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0
print(mat)
rowProds(mat)
colProds(mat)

Calculates quantiles for each row (column) of a matrix-like object

rowQuantiles(x, rows = NULL, cols = NULL, probs = seq(from = 0, to = 1, by = 0.25), na.rm = FALSE, type = 7L, ..., useNames = TRUE, drop = TRUE)

rowQuantiles(x, rows = NULL, cols = NULL, probs = seq(from = 0, to = 1, by = 0.25), na.rm = FALSE, type = 7L, ..., useNames = TRUE, drop = TRUE)

rowQuantiles(x, rows = NULL, cols = NULL, probs = seq(from = 0, to = 1, by = 0.25), na.rm = FALSE, type = 7L, ..., useNames = TRUE, drop = TRUE)

rowQuantiles(x, rows = NULL, cols = NULL,
probs = seq(from = 0, to = 1, by = 0.25), na.rm = FALSE, type = 7L, ...
useNames = TRUE, drop = TRUE)

colQuantiles(x, rows = NULL, cols = NULL, probs = seq(from = 0, to = 1,
by = 0.25), na.rm = FALSE, type = 7L, ..., useNames = TRUE, drop = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colQuantiles(x, rows = NULL,
cols = NULL, probs = seq(from = 0, to = 1, by = 0.25), na.rm = FALSE,
type = 7L, ..., useNames = TRUE, drop = TRUE)

## S4 method for signature 'ANY'
colQuantiles(x, rows = NULL, cols = NULL,
probs = seq(from = 0, to = 1, by = 0.25), na.rm = FALSE, type = 7L,
..., useNames = TRUE, drop = TRUE)

Arguments

x An NxK matrix-like object.
rows, cols A vector indicating the subset of rows (and/or columns) to operate over. If
NULL, no subsetting is done.
probs A numeric vector of J probabilities in [0, 1].
na.rm If TRUE, missing values (NA or NaN) are omitted from the calculations.
type An integer specifying the type of estimator. See stats::quantile(). for more
details.
... Additional arguments passed to specific methods.
useNames If TRUE (default), names attributes of result are set. Else if FALSE, no naming
support is done.
drop If TRUE a vector is returned if J == 1.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowQuantiles
/matrixStats::colQuantiles.

Value

a numeric NxJ (KxJ) matrix, where N (K) is the number of rows (columns) for which the J values
are calculated.

See Also

- matrixStats::rowQuantiles() and matrixStats::colQuantiles() which are used when
the input is a matrix or numeric vector.
- stats::quantile
Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowQuantiles(mat)
colQuantiles(mat)
```

---

**rowRanges**

*Calculates the minimum and maximum for each row (column) of a matrix-like object*

**Description**

Calculates the minimum and maximum for each row (column) of a matrix-like object.

**Usage**

```r
rowRanges(x, ...) ~
## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowRanges(x, rows = NULL,
    cols = NULL, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowRanges(x, ...)

colRanges(x, rows = NULL, cols = NULL, na.rm = FALSE, ...,
    useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colRanges(x, rows = NULL,
    cols = NULL, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colRanges(x, rows = NULL, cols = NULL, na.rm = FALSE,
    ..., useNames = TRUE)
```

**Arguments**

- `x` An NxK matrix-like object.
- `...` Additional arguments passed to specific methods.
- `rows, cols` A vector indicating the subset of rows (and/or columns) to operate over. If `NULL`, no subsetting is done.
rowRanges

na.rm If TRUE, missing values (NA or NaN) are omitted from the calculations.
dim. An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.
useNames If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowRanges / matrixStats::colRanges.

Value

a numeric N\times2 (K\times2) matrix, where N (K) is the number of rows (columns) for which the ranges are calculated.

Note

Unfortunately for the argument list of the rowRanges() generic function we cannot follow the scheme used for the other row/column matrix summarization generic functions. This is because we need to be compatible with the historic rowRanges() getter for RangedSummarizedExperiment objects. See ?SummarizedExperiment::rowRanges.

See Also

• matrixStats::rowRanges() and matrixStats::colRanges() which are used when the input is a matrix or numeric vector.
• For max estimates, see rowMaxs().
• For min estimates, see rowMins().
• base::range().

Examples

mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)

rowRanges(mat)
colRanges(mat)
rowRanks

Calculates the rank of the elements for each row (column) of a matrix-like object.

Description

Calculates the rank of the elements for each row (column) of a matrix-like object.

Usage

rowRanks(x, rows = NULL, cols = NULL, ties.method = c("max", "average"),
          ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowRanks(x, rows = NULL,
          cols = NULL, ties.method = c("max", "average", "first", "last", "random",
                                      "max", "min", "dense"), dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowRanks(x, rows = NULL, cols = NULL,
          ties.method = c("max", "average"), ..., useNames = TRUE)

colRanks(x, rows = NULL, cols = NULL, ties.method = c("max", "average"),
          ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colRanks(x, rows = NULL,
          cols = NULL, ties.method = c("max", "average", "first", "last", "random",
                                      "max", "min", "dense"), dim. = dim(x), preserveShape = FALSE, ...,
          useNames = TRUE)

## S4 method for signature 'ANY'
colRanks(x, rows = NULL, cols = NULL,
          ties.method = c("max", "average"), ..., useNames = TRUE)

Arguments

x      An NxK matrix-like object.
rows, cols A vector indicating the subset of rows (and/or columns) to operate over. If
          NULL, no subsetting is done.
ties.method A character string specifying how ties are treated. Note that the default specifies
          fewer options than the original matrixStats package.
...      Additional arguments passed to specific methods.
useNames If TRUE (default), names attributes of result are set. Else if FALSE, no naming
          support is done.
rowRanks

**dim.** An integer vector of length two specifying the dimension of \( x \), essential when \( x \) is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.

**preserveShape** If TRUE the output matrix has the same shape as the input \( x \). Note, that this is not a generic argument and not all implementation of this function have to provide it.

**Details**

The S4 methods for \( x \) of type matrix, array, table, or numeric call matrixStats::rowRanks / matrixStats::colRanks.

The matrixStats::rowRanks() function can handle a lot of different values for the ties.method argument. Users of the generic function should however only rely on max and average because the other ones are not guaranteed to be implemented:

- **max** for values with identical values the maximum rank is returned
- **average** for values with identical values the average of the ranks they cover is returned. Note, that in this case the return value is of type numeric.

**Value**

A matrix of type integer is returned unless ties.method = "average". It has dimensions' \( N \times J \) matrix, where \( N \) (\( K \)) is the number of rows (columns) of the input \( x \).

**See Also**

- matrixStats::rowRanks() and matrixStats::colRanks() which are used when the input is a matrix or numeric vector.
- base::rank

**Examples**

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0
print(mat)
rowRanks(mat)
colRanks(mat)
```
Calculates the standard deviation of the difference between each element of a row (column) of a matrix-like object.

### Description

Calculates the standard deviation of the difference between each element of a row (column) of a matrix-like object.

### Usage

```r
rowSdDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
```

```r
# S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowSdDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
```

```r
# S4 method for signature 'ANY'
rowSdDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
```

```r
colSdDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
```

```r
# S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colSdDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
```

```r
# S4 method for signature 'ANY'
colSdDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
```

### Arguments

- `x`: An NxK matrix-like object.
- `rows`, `cols`: A vector indicating the subset of rows (and/or columns) to operate over. If `NULL`, no subsetting is done.
- `na.rm`: If `TRUE`, missing values (NA or NaN) are omitted from the calculations.
- `diff`: An integer specifying the order of difference.
- `trim`: A double in [0,1/2] specifying the fraction of observations to be trimmed from each end of (sorted) x before estimation.
- `...`: Additional arguments passed to specific methods.
- `useNames`: If `TRUE` (default), names attributes of result are set. Else if `FALSE`, no naming support is done.
**rowSds**

**Details**

The S4 methods for x of type `matrix, array, table, or numeric` call `matrixStats::rowSdDiffs` / `matrixStats::colSdDiffs`.

**Value**

Returns a numeric vector of length N (K).

**See Also**

- `matrixStats::rowSdDiffs()` and `matrixStats::colSdDiffs()` which are used when the input is a matrix or numeric vector.
- for the direct standard deviation see `rowSds()`.

**Examples**

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowSdDiffs(mat)
colSdDiffs(mat)
```

---

**rowSds**  
*Calculates the standard deviation for each row (column) of a matrix-like object*

**Description**

Calculates the standard deviation for each row (column) of a matrix-like object.

**Usage**

```r
rowSds(x, rows = NULL, cols = NULL, na.rm = FALSE, center = NULL, ..., 
useNames = TRUE)
```

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

```r
rowSds(x, rows = NULL, 
cols = NULL, na.rm = FALSE, center = NULL, dim. = dim(x), ..., 
useNames = TRUE)
```

## S4 method for signature 'ANY'

```r
rowSds(x, rows = NULL, cols = NULL, na.rm = FALSE, 
center = NULL, ..., useNames = TRUE)
```
colSds(x, rows = NULL, cols = NULL, na.rm = FALSE, center = NULL, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colSds(x, rows = NULL,
    cols = NULL, na.rm = FALSE, center = NULL, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
colSds(x, rows = NULL, cols = NULL, na.rm = FALSE,
    center = NULL, ..., useNames = TRUE)

Arguments

- **x**: An NxK matrix-like object.
- **rows, cols**: A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
- **na.rm**: If TRUE, missing values (NA or NaN) are omitted from the calculations.
- **center**: (optional) the center, defaults to the row means
- **...**: Additional arguments passed to specific methods.
- **useNames**: If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.
- **dim.**: An integer vector of length two specifying the dimension of x, essential when x is a numeric vector. Note, that this is not a generic argument and not all methods need provide it.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowSds / matrixStats::colSds.

Value

Returns a numeric vector of length N (K).

See Also

- matrixStats::rowSds() and matrixStats::colSds() which are used when the input is a matrix or numeric vector.
- For mean estimates, see rowMeans2() and rowMeans().
- For variance estimates, see rowVars().
Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)

rowSds(mat)
colSds(mat)
```

Description

Calculates the sum for each row (column) of a matrix-like object.

Usage

```r
rowSums(x, na.rm = FALSE, dims = 1, ...)
colSums(x, na.rm = FALSE, dims = 1, ...)
```

Arguments

- **x**: An NxK matrix-like object, a numeric data frame, or an array-like object of two or more dimensions.
- **na.rm**: If `TRUE`, missing values (NA or NaN) are omitted from the calculations.
- **dims**: A single integer indicating which dimensions are regarded as rows or columns to sum over. For `rowSums`, the sum is over dimensions `dims+1, ...`; for `colSums` it is over dimensions `1:dims`.
- **...**: Additional arguments passed to specific methods.

Details

This man page documents the `rowSums` and `colSums` *S4 generic functions* defined in the `Matrix-Generics` package. See `?base::colSums` for the default methods (defined in the `base` package).

Value

Returns a numeric vector of length N (K).

See Also

- `base::colSums` for the default `rowSums` and `colSums` methods.
- `Matrix::colSums` in the `Matrix` package for `rowSums` and `colSums` methods defined for `CsparseMatrix` derivatives (e.g. `dgCMatrix` objects).
Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowSums(mat)
colSums(mat)
```

---

**rowSums2**

*Calculates the sum for each row (column) of a matrix-like object*

Description

Calculates the sum for each row (column) of a matrix-like object.

Usage

```r
rowSums2(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowSums2(x, rows = NULL, cols = NULL, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowSums2(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

colSums2(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

colSums2(x, rows = NULL, cols = NULL, na.rm = FALSE, dim. = dim(x), ..., useNames = TRUE)

colSums2(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)
```

Arguments

- **x**
  - An NxK matrix-like object.
- **rows, cols**
  - A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
Tabulates the values in a matrix-like object by row (column)

Description

Tabulates the values in a matrix-like object by row (column).
Usage

rowTabulates(x, rows = NULL, cols = NULL, values = NULL, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowTabulates(x, rows = NULL, cols = NULL, values = NULL, ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowTabulates(x, rows = NULL, cols = NULL, values = NULL, ..., useNames = TRUE)

colTabulates(x, rows = NULL, cols = NULL, values = NULL, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colTabulates(x, rows = NULL, cols = NULL, values = NULL, ..., useNames = TRUE)

## S4 method for signature 'ANY'
colTabulates(x, rows = NULL, cols = NULL, values = NULL, ..., useNames = TRUE)

Arguments

x An NxK matrix-like object.
rows, cols A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
values the values to search for.
... Additional arguments passed to specific methods.
useNames If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowTabulates / matrixStats::colTabulates.

Value

a numeric NxJ (KxJ) matrix, where N (K) is the number of rows (columns) for which the J values are calculated.

See Also

• matrixStats::rowTabulates() and matrixStats::colTabulates() which are used when the input is a matrix or numeric vector.
• base::table()
Examples

mat <- matrix(rpois(15, lambda = 3), nrow = 5, ncol = 3)
mat[2, 1] <- NA_integer_
mat[3, 3] <- 0L
mat[4, 1] <- 0L
print(mat)
rowTabulates(mat)
colTabulates(mat)
rowTabulates(mat, values = 0)
colTabulates(mat, values = 0)

---

rowVarDiffs

Calculates the variance of the difference between each element of a row (column) of a matrix-like object.

Description

Calculates the variance of the difference between each element of a row (column) of a matrix-like object.

Usage

rowVarDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowVarDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowVarDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)

colVarDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colVarDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)

## S4 method for signature 'ANY'
colVarDiffs(x, rows = NULL, cols = NULL, na.rm = FALSE, diff = 1L, trim = 0, ..., useNames = TRUE)
Arguments

- **x**: An NxK matrix-like object.
- **rows, cols**: A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
- **na.rm**: If TRUE, missing values (NA or NaN) are omitted from the calculations.
- **diff**: An integer specifying the order of difference.
- **trim**: A double in [0,1/2] specifying the fraction of observations to be trimmed from each end of (sorted) x before estimation.
- **...**: Additional arguments passed to specific methods.
- **useNames**: If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowVarDiffs / matrixStats::colVarDiffs.

Value

Returns a numeric vector of length N (K).

See Also

- matrixStats::rowVarDiffs() and matrixStats::colVarDiffs() which are used when the input is a matrix or numeric vector.
- for the direct variance see rowVars().

Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
m[2, 1] <- NA
m[3, 3] <- Inf
m[4, 1] <- 0

print(m)

rowVarDiffs(m)
colVarDiffs(m)
```
rowVars

Calculates the variance for each row (column) of a matrix-like object

Description

Calculates the variance for each row (column) of a matrix-like object.

Usage

rowVars(x, rows = NULL, cols = NULL, na.rm = FALSE, center = NULL, ...,
         useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowVars(x, rows = NULL, cols = NULL, na.rm = FALSE, center = NULL, dim. = dim(x), ...,
         useNames = TRUE)

## S4 method for signature 'ANY'
rowVars(x, rows = NULL, cols = NULL, na.rm = FALSE, center = NULL, ...,
         useNames = TRUE)

colVars(x, rows = NULL, cols = NULL, na.rm = FALSE, center = NULL, ...,
         useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colVars(x, rows = NULL, cols = NULL, na.rm = FALSE, center = NULL, dim. = dim(x), ...,
         useNames = TRUE)

## S4 method for signature 'ANY'
colVars(x, rows = NULL, cols = NULL, na.rm = FALSE, center = NULL, ...,
         useNames = TRUE)

Arguments

x
An NxK matrix-like object.

rows, cols
A vector indicating the subset of rows (and/or columns) to operate over. If
NULL, no subsetting is done.

na.rm
If TRUE, missing values (NA or NaN) are omitted from the calculations.

center
(optional) the center, defaults to the row means.

... Additional arguments passed to specific methods.

useNames
If TRUE (default), names attributes of result are set. Else if FALSE, no naming
support is done.

dim.
An integer vector of length two specifying the dimension of x, essential when
x is a numeric vector. Note, that this is not a generic argument and not all
methods need provide it.
Details

The S4 methods for `x` of type `matrix`, `array`, `table`, or `numeric` call `matrixStats::rowVars` / `matrixStats::colVars`.

Value

Returns a numeric vector of length N (K).

See Also

- `matrixStats::rowVars()` and `matrixStats::colVars()` which are used when the input is a matrix or numeric vector.
- For mean estimates, see `rowMeans2()` and `rowMeans()`.
- For standard deviation estimates, see `rowSds()`.
- `stats::var()`.

Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowVars(mat)
colVars(mat)
```

---

`rowWeightedMads`  
Calculates the weighted median absolute deviation for each row (column) of a matrix-like object

Description

Calculates the weighted median absolute deviation for each row (column) of a matrix-like object.

Usage

```r
rowWeightedMads(x, w = NULL, rows = NULL, cols = NULL, na.rm = FALSE, constant = 1.4826, center = NULL, ..., useNames = TRUE)
```

```r
## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowWeightedMads(x, w = NULL, rows = NULL, cols = NULL, na.rm = FALSE, constant = 1.4826, center = NULL, ..., useNames = TRUE)
```

```r
## S4 method for signature 'ANY'
```

rowWeightedMads(x, w = NULL, rows = NULL, cols = NULL, 
    na.rm = FALSE, constant = 1.4826, center = NULL, ..., 
    useNames = TRUE)

colWeightedMads(x, w = NULL, rows = NULL, cols = NULL, na.rm = FALSE, 
    constant = 1.4826, center = NULL, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

colWeightedMads(x, w = NULL, 
    rows = NULL, cols = NULL, na.rm = FALSE, constant = 1.4826, 
    center = NULL, ..., useNames = TRUE)

## S4 method for signature 'ANY'

colWeightedMads(x, w = NULL, rows = NULL, cols = NULL, 
    na.rm = FALSE, constant = 1.4826, center = NULL, ..., 
    useNames = TRUE)

Arguments

x
An NxK matrix-like object.

w
A numeric vector of length K (N) that specifies by how much each element is weighted.

rows, cols
A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.

na.rm
If TRUE, missing values (NA or NaN) are omitted from the calculations.

constant
A scale factor. See stats::mad() for details.

center
(optional) the center, defaults to the row means

...
Additional arguments passed to specific methods.

useNames
If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowWeightedMads 
/matrixStats::colWeightedMads.

Value

Returns a numeric vector of length N (K).

See Also

• matrixStats::rowWeightedMads() and matrixStats::colWeightedMads() which are used when the input is a matrix or numeric vector.

• See also rowMads for the corresponding unweighted function.
Examples

mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
w <- rnorm(n = 5, mean = 3)
rowWeightedMads(mat, w = w[1:3])
colWeightedMads(mat, w = w)

rowWeightedMeans

Calculates the weighted mean for each row (column) of a matrix-like object

Description

Calculates the weighted mean for each row (column) of a matrix-like object.

Usage

rowWeightedMeans(x, w = NULL, rows = NULL, cols = NULL, na.rm = FALSE, 
... , useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowWeightedMeans(x, w = NULL, 
   rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowWeightedMeans(x, w = NULL, rows = NULL, cols = NULL, 
   na.rm = FALSE, ..., useNames = TRUE)

colWeightedMeans(x, w = NULL, rows = NULL, cols = NULL, na.rm = FALSE, 
... , useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colWeightedMeans(x, w = NULL, 
   rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'ANY'
colWeightedMeans(x, w = NULL, rows = NULL, cols = NULL, 
   na.rm = FALSE, ..., useNames = TRUE)

Arguments

x An NxK matrix-like object.
rowWeightedMedians

**w**  
A numeric vector of length K (N) that specifies by how much each element is weighted.

**rows, cols**  
A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.

**na.rm**  
If TRUE, missing values (NA or NaN) are omitted from the calculations.

**...**  
Additional arguments passed to specific methods.

**useNames**  
If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

**Details**

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowWeightedMeans / matrixStats::colWeightedMeans.

**Value**

Returns a numeric vector of length N (K).

**See Also**

- matrixStats::rowWeightedMeans() and matrixStats::colWeightedMeans() which are used when the input is a matrix or numeric vector.
- See also rowMeans2 for the corresponding unweighted function.

**Examples**

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
w <- rnorm(n = 5, mean = 3)
rowWeightedMeans(mat, w = w[1:3])
colWeightedMeans(mat, w = w)
```

---

**rowWeightedMedians**  
Calculates the weighted median for each row (column) of a matrix-like object

**Description**

Calculates the weighted median for each row (column) of a matrix-like object.
Usage

rowWeightedMedians(x, w = NULL, rows = NULL, cols = NULL, 
na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowWeightedMedians(x, 
    w = NULL, rows = NULL, cols = NULL, na.rm = FALSE, ..., 
    useNames = TRUE)

## S4 method for signature 'ANY'
rowWeightedMedians(x, w = NULL, rows = NULL, cols = NULL, 
na.rm = FALSE, ..., useNames = TRUE)

colWeightedMedians(x, w = NULL, rows = NULL, cols = NULL, 
na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colWeightedMedians(x, 
    w = NULL, rows = NULL, cols = NULL, na.rm = FALSE, ..., 
    useNames = TRUE)

## S4 method for signature 'ANY'
colWeightedMedians(x, w = NULL, rows = NULL, cols = NULL, 
na.rm = FALSE, ..., useNames = TRUE)

Arguments

- **x**: An NxK matrix-like object.
- **w**: A numeric vector of length K (N) that specifies by how much each element is weighted.
- **rows, cols**: A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
- **na.rm**: If TRUE, missing values (NA or NaN) are omitted from the calculations.
- **...**: Additional arguments passed to specific methods.
- **useNames**: If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowWeightedMedians / matrixStats::colWeightedMedians.

Value

Returns a numeric vector of length N (K).
rowWeightedSds

See Also

- matrixStats::rowWeightedMedians() and matrixStats::colWeightedMedians() which are used when the input is a matrix or numeric vector.
- See also rowMedians for the corresponding unweighted function.

Examples

mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
w <- rnorm(n = 5, mean = 3)
rowWeightedMedians(mat, w = w[1:3])
colWeightedMedians(mat, w = w)

rowWeightedSds

Calculates the weighted standard deviation for each row (column) of a matrix-like object

Description

Calculates the weighted standard deviation for each row (column) of a matrix-like object.

Usage

rowWeightedSds(x, w = NULL, rows = NULL, cols = NULL, na.rm = FALSE, ...,
               useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowWeightedSds(x, w = NULL,
               rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'ANY'
rowWeightedSds(x, w = NULL, rows = NULL, cols = NULL,
               na.rm = FALSE, ..., useNames = TRUE)

colWeightedSds(x, w = NULL, rows = NULL, cols = NULL, na.rm = FALSE, ...
               useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colWeightedSds(x, w = NULL,
               rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

## S4 method for signature 'ANY'
colWeightedSds(x, w = NULL, rows = NULL, cols = NULL,
               na.rm = FALSE, ..., useNames = TRUE)
Arguments

x  An NxK matrix-like object.

w  A numeric vector of length K (N) that specifies by how much each element is weighted.

rows, cols A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.

na.rm  If TRUE, missing values (NA or NaN) are omitted from the calculations.

...  Additional arguments passed to specific methods.

useNames  If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowWeightedSds / matrixStats::colWeightedSds.

Value

Returns a numeric vector of length N (K).

See Also

- matrixStats::rowWeightedSds() and matrixStats::colWeightedSds() which are used when the input is a matrix or numeric vector.
- See also rowSds for the corresponding unweighted function.

Examples

mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
w <- rnorm(n = 5, mean = 3)
rowWeightedSds(mat, w = w[1:3])
colWeightedSds(mat, w = w)

---

rowWeightedVars  Calculates the weighted variance for each row (column) of a matrix-like object

Description

Calculates the weighted variance for each row (column) of a matrix-like object.
rowWeightedVars

Usage

rowWeightedVars(x, w = NULL, rows = NULL, cols = NULL, na.rm = FALSE, 
... , useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
rowWeightedVars(x, w = NULL, 
rows = NULL, cols = NULL, na.rm = FALSE, ... , useNames = TRUE)

## S4 method for signature 'ANY'
rowWeightedVars(x, w = NULL, rows = NULL, cols = NULL, 
na.rm = FALSE, ... , useNames = TRUE)

colWeightedVars(x, w = NULL, rows = NULL, cols = NULL, 
na.rm = FALSE, ... , useNames = TRUE)

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'
colWeightedVars(x, w = NULL, 
rows = NULL, cols = NULL, na.rm = FALSE, ... , useNames = TRUE)

## S4 method for signature 'ANY'
colWeightedVars(x, w = NULL, rows = NULL, cols = NULL, 
na.rm = FALSE, ... , useNames = TRUE)

Arguments

x        An NxK matrix-like object.
w        A numeric vector of length K (N) that specifies by how much each element is weighted.
rows,cols A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
na.rm    If TRUE, missing values (NA or NaN) are omitted from the calculations.
...      Additional arguments passed to specific methods.
useNames If TRUE (default), names attributes of result are set. Else if FALSE, no naming support is done.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowWeightedVars / matrixStats::colWeightedVars.

Value

Returns a numeric vector of length N (K).

See Also

- matrixStats::rowWeightedVars() and matrixStats::colWeightedVars() which are used when the input is a matrix or numeric vector.
• See also rowVars for the corresponding unweighted function.

Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
w <- rnorm(n = 5, mean = 3)
rowWeightedVars(mat, w = w[1:3])
colWeightedVars(mat, w = w)
```
Index

* algebra
  rowMeans, 30
  rowSums, 47
* arith
  rowMeans, 30
  rowSums, 47
* array
  rowAvgsPerColSet, 8
  rowCollapse, 10
  rowCummaxs, 13
  rowCummins, 14
  rowCumprods, 16
  rowCumsums, 17
  rowDiffs, 19
  rowIQRDiffs, 20
  rowIQRs, 22
  rowLogSumExps, 23
  rowMadDiffs, 25
  rowMads, 26
  rowMaxs, 28
  rowMeans, 30
  rowMeans2, 31
  rowMedians, 32
  rowMins, 34
  rowOrderStats, 35
  rowProds, 37
  rowQuantiles, 38
  rowRanges, 40
  rowRanks, 42
  rowSdDiffs, 44
  rowSds, 45
  rowSums, 47
  rowSums2, 48
  rowTabulates, 49
  rowVarDiffs, 51
  rowVars, 53
  rowWeightedMads, 54
  rowWeightedMeans, 56
  rowWeightedMedians, 57
  rowWeightedSds, 59
  rowWeightedVars, 60
* internal
  internal-helpers, 3
* iteration
  rowAvgsPerColSet, 8
  rowCollapse, 10
  rowCummaxs, 13
  rowCummins, 14
  rowCumprods, 16
  rowCumsums, 17
  rowDiffs, 19
  rowIQRDiffs, 20
  rowIQRs, 22
  rowLogSumExps, 23
  rowMadDiffs, 25
  rowMads, 26
  rowMaxs, 28
  rowMeans, 30
  rowMeans2, 31
  rowMedians, 32
  rowMins, 34
  rowOrderStats, 35
  rowProds, 37
  rowQuantiles, 38
  rowRanges, 40
  rowRanks, 42
  rowSdDiffs, 44
  rowSds, 45
  rowSums, 47
  rowSums2, 48
  rowTabulates, 49
  rowVarDiffs, 51
  rowVars, 53
  rowWeightedMads, 54
  rowWeightedMeans, 56
  rowWeightedMedians, 57
  rowWeightedSds, 59
  rowWeightedVars, 60
<table>
<thead>
<tr>
<th>* robust</th>
</tr>
</thead>
<tbody>
<tr>
<td>rowAvgsPerColSet, 8</td>
</tr>
<tr>
<td>rowCollapse, 10</td>
</tr>
<tr>
<td>rowCummmaxs, 13</td>
</tr>
<tr>
<td>rowCummins, 14</td>
</tr>
<tr>
<td>rowCumprods, 16</td>
</tr>
<tr>
<td>rowCumsums, 17</td>
</tr>
<tr>
<td>rowDiffs, 19</td>
</tr>
<tr>
<td>rowIQRDiffs, 20</td>
</tr>
<tr>
<td>rowIQRs, 22</td>
</tr>
<tr>
<td>rowLogSumExps, 23</td>
</tr>
<tr>
<td>rowMadDiffs, 25</td>
</tr>
<tr>
<td>rowMads, 26</td>
</tr>
<tr>
<td>rowMaxs, 28</td>
</tr>
<tr>
<td>rowMeans, 30</td>
</tr>
<tr>
<td>rowMeans2, 31</td>
</tr>
<tr>
<td>rowMedians, 32</td>
</tr>
<tr>
<td>rowMins, 34</td>
</tr>
<tr>
<td>rowOrderStats, 35</td>
</tr>
<tr>
<td>rowProds, 37</td>
</tr>
<tr>
<td>rowSdDiffs, 44</td>
</tr>
<tr>
<td>rowSds, 45</td>
</tr>
<tr>
<td>rowSums, 47</td>
</tr>
<tr>
<td>rowSums2, 48</td>
</tr>
<tr>
<td>rowTabulates, 49</td>
</tr>
<tr>
<td>rowVarDiffs, 51</td>
</tr>
<tr>
<td>rowVars, 53</td>
</tr>
<tr>
<td>rowWeightedMads, 54</td>
</tr>
<tr>
<td>rowWeightedMeans, 56</td>
</tr>
<tr>
<td>rowWeightedMedians, 57</td>
</tr>
<tr>
<td>rowWeightedSds, 59</td>
</tr>
<tr>
<td>rowWeightedVars, 60</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>* utilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>internal-helpers, 3</td>
</tr>
</tbody>
</table>

| all, 5 |
| any, 6, 8 |

| array, 4, 6, 7, 9, 10, 12, 14–16, 18, 20, 21, 23, 24, 26, 27, 29, 32, 33, 35, 36, 38, 39, 41, 43, 45, 46, 49, 50, 52, 54, 55, 57, 58, 60, 61 |

| base::rank, 43 |

<table>
<thead>
<tr>
<th>class:matrix_OR_array_OR_table_OR_numeric</th>
</tr>
</thead>
<tbody>
<tr>
<td>(MatrixGenerics-package), 3</td>
</tr>
<tr>
<td>colAlls, 4, 5</td>
</tr>
<tr>
<td>colAlls (rowAlls), 3</td>
</tr>
<tr>
<td>colAlls,ANY-method (rowAlls), 3</td>
</tr>
<tr>
<td>colAlls,matrix_OR_array_OR_table_OR_numeric-method (rowAlls), 3</td>
</tr>
<tr>
<td>colAnyNAs, 6</td>
</tr>
<tr>
<td>colAnyNAs (rowAnyNAs), 5</td>
</tr>
<tr>
<td>colAnyNAs,ANY-method (rowAnyNAs), 5</td>
</tr>
<tr>
<td>colAnyNAs,matrix_OR_array_OR_table_OR_numeric-method (rowAnyNAs), 5</td>
</tr>
<tr>
<td>colAnys, 7, 8</td>
</tr>
<tr>
<td>colAnys (rowAnys), 6</td>
</tr>
<tr>
<td>colAnys,ANY-method (rowAnys), 6</td>
</tr>
<tr>
<td>colAnys,matrix_OR_array_OR_table_OR_numeric-method (rowAnys), 6</td>
</tr>
</tbody>
</table>
colAvgsPerRowSet, 9
colAvgsPerRowSet (rowAvgsPerColSet), 8
colAvgsPerRowSet, ANY-method
  (rowAvgsPerColSet), 8
colAvgsPerRowSet, matrix_OR_array_OR_table_OR_numeric-method
  (rowAvgsPerColSet), 8
colCollapse, 10, 11
colCollapse (rowCollapse), 10
colCollapse, ANY-method (rowCollapse), 10
colCollapse, matrix_OR_array_OR_table_OR_numeric-method
  (rowCollapse), 10
colCounts, 12
colCounts (rowCounts), 11
colCounts, ANY-method (rowCounts), 11
colCounts, matrix_OR_array_OR_table_OR_numeric-method
  (rowCounts), 11
colCummaxs, 14
colCummaxs (rowCummaxs), 13
colCummaxs, ANY-method (rowCummaxs), 13
colCummaxs, matrix_OR_array_OR_table_OR_numeric-method
  (rowCummaxs), 13
colCummins, 15
colCummins (rowCummins), 14
colCummins, ANY-method (rowCummins), 14
colCummins, matrix_OR_array_OR_table_OR_numeric-method
  (rowCummins), 14
colCumprods, 16, 17
colCumprods (rowCumprods), 16
colCumprods, ANY-method (rowCumprods), 16
colCumprods, matrix_OR_array_OR_table_OR_numeric-method
  (rowCumprods), 16
colCumsums, 18
colCumsums (rowCumsums), 17
colCumsums, ANY-method (rowCumsums), 17
colCumsums, matrix_OR_array_OR_table_OR_numeric-method
  (rowCumsums), 17
colDiffs, 20
colDiffs (rowDiffs), 19
colDiffs, ANY-method (rowDiffs), 19
colDiffs, matrix_OR_array_OR_table_OR_numeric-method
  (rowDiffs), 19
colIQRDiffs, 21
colIQRDiffs (rowIQRDiffs), 20
colIQRDiffs, ANY-method (rowIQRDiffs), 20
colIQRDiffs, matrix_OR_array_OR_table_OR_numeric-method
  (rowIQRDiffs), 20
colIQRs, 23
colIQRs (rowIQRs), 22
colProds (rowProds), 37
colProds, ANY-method (rowProds), 37
colProds, matrix_or_array_or_table_or_numeric-method (rowProds), 37
colQuantiles, 39
colQuantiles (rowQuantiles), 38
colQuantiles, ANY-method (rowQuantiles), 38
colQuantiles, matrix_or_array_or_table_or_numeric-method (rowQuantiles), 38
colRanks, 43
colRanks (rowRanks), 42
colRanks, ANY-method (rowRanks), 42
colRanks, matrix_or_array_or_table_or_numeric-method (rowRanks), 42
colSdDiffs, 45
colSdDiffs (rowSdDiffs), 44
colSdDiffs, ANY-method (rowSdDiffs), 44
colSdDiffs, matrix_or_array_or_table_or_numeric-method (rowSdDiffs), 44
colSds, 46
colSds (rowSds), 45
colSds, ANY-method (rowSds), 45
colSds, matrix_or_array_or_table_or_numeric-method (rowSds), 45
colSums, 47
colSums (rowSums), 47
colSums2, 49
colSums2 (rowSums2), 48
colSums2, ANY-method (rowSums2), 48
colSums2, matrix_or_array_or_table_or_numeric-method (rowSums2), 48
colTabulates, 50
colTabulates (rowTabulates), 49
colTabulates, ANY-method (rowTabulates), 49
colTabulates, matrix_or_array_or_table_or_numeric-method (rowTabulates), 49
colVarDiffs, 52
colVarDiffs (rowVarDiffs), 51
colVarDiffs, ANY-method (rowVarDiffs), 51
colVarDiffs, matrix_or_array_or_table_or_numeric-method (rowVarDiffs), 51

FALSE, 4, 6, 7, 10, 12, 13, 15, 16, 18, 19, 21, 22, 24, 26, 27, 29, 31, 33, 34, 36, 37, 39, 41, 42, 44, 46, 49, 50, 52, 53, 55, 57, 58, 60, 61
<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>integer, ANY-method (rowAlls, ANY-method)</td>
<td>67</td>
</tr>
<tr>
<td>rowAvgsPerColSet, matrix_OR_array_OR_table_OR_numeric-method (rowAvgsPerColSet)</td>
<td>8</td>
</tr>
<tr>
<td>rowCumprods, ANY-method (rowCumprods, ANY-method)</td>
<td>16</td>
</tr>
<tr>
<td>rowCummins, ANY-method (rowCummins, ANY-method)</td>
<td>14</td>
</tr>
<tr>
<td>rowCumsums, ANY-method (rowCumsums, ANY-method)</td>
<td>17</td>
</tr>
<tr>
<td>rowDiffs, ANY-method (rowDiffs, ANY-method)</td>
<td>19</td>
</tr>
<tr>
<td>rowLogSumExps, ANY-method (rowLogSumExps, ANY-method)</td>
<td>23</td>
</tr>
<tr>
<td>rowIQRs, ANY-method (rowIQRs, ANY-method)</td>
<td>22</td>
</tr>
<tr>
<td>rowIQRDiffs, ANY-method (rowIQRDiffs, ANY-method)</td>
<td>20</td>
</tr>
<tr>
<td>rowCounts, ANY-method (rowCounts, ANY-method)</td>
<td>11</td>
</tr>
<tr>
<td>rowCounts, matrix_OR_array_OR_table_OR_numeric-method (rowCounts, ANY-method)</td>
<td>10</td>
</tr>
<tr>
<td>rowCollapse, ANY-method (rowCollapse, ANY-method)</td>
<td>10</td>
</tr>
<tr>
<td>rowCummaxs, ANY-method (rowCummaxs, ANY-method)</td>
<td>13</td>
</tr>
<tr>
<td>rowCummins, matrix_OR_array_OR_table_OR_numeric-method (rowCummins, ANY-method)</td>
<td>14</td>
</tr>
<tr>
<td>rowCumprods, matrix_OR_array_OR_table_OR_numeric-method (rowCumprods, ANY-method)</td>
<td>16</td>
</tr>
<tr>
<td>rowCumsums, matrix_OR_array_OR_table_OR_numeric-method (rowCumsums, ANY-method)</td>
<td>17</td>
</tr>
<tr>
<td>rowIQRs, matrix_OR_array_OR_table_OR_numeric-method (rowIQRs, ANY-method)</td>
<td>22</td>
</tr>
<tr>
<td>rowIQRDiffs, matrix_OR_array_OR_table_OR_numeric-method (rowIQRDiffs, ANY-method)</td>
<td>20</td>
</tr>
<tr>
<td>rowIQRs, matrix_OR_array_OR_table_OR_numeric-method (rowIQRs, ANY-method)</td>
<td>22</td>
</tr>
<tr>
<td>rowLogSumExps, matrix_OR_array_OR_table_OR_numeric-method (rowLogSumExps, ANY-method)</td>
<td>23</td>
</tr>
<tr>
<td>rowLogSumExps, ANY-method (rowLogSumExps, ANY-method)</td>
<td>23</td>
</tr>
<tr>
<td>rowAnys, ANY-method (rowAnys, ANY-method)</td>
<td>6</td>
</tr>
<tr>
<td>rowAvgsPerColSet, ANY-method (rowAvgsPerColSet)</td>
<td>8</td>
</tr>
<tr>
<td>rowAvgsPerColSet, matrix_OR_array_OR_table_OR_numeric-method (rowAvgsPerColSet)</td>
<td>8</td>
</tr>
<tr>
<td>rowCollapse, ANY-method (rowCollapse)</td>
<td>10</td>
</tr>
<tr>
<td>rowCollapse, matrix_OR_array_OR_table_OR_numeric-method (rowCollapse)</td>
<td>10</td>
</tr>
<tr>
<td>rowCounts, ANY-method (rowCounts)</td>
<td>11</td>
</tr>
<tr>
<td>rowCounts, matrix_OR_array_OR_table_OR_numeric-method (rowCounts)</td>
<td>11</td>
</tr>
<tr>
<td>rowCummaxs, ANY-method (rowCummaxs)</td>
<td>13</td>
</tr>
<tr>
<td>rowCummaxs, matrix_OR_array_OR_table_OR_numeric-method (rowCummaxs)</td>
<td>13</td>
</tr>
<tr>
<td>rowCummins, ANY-method (rowCummins)</td>
<td>14</td>
</tr>
<tr>
<td>rowCummins, matrix_OR_array_OR_table_OR_numeric-method (rowCummins)</td>
<td>14</td>
</tr>
<tr>
<td>rowCumprods, ANY-method (rowCumprods)</td>
<td>16</td>
</tr>
<tr>
<td>rowCumprods, matrix_OR_array_OR_table_OR_numeric-method (rowCumprods)</td>
<td>16</td>
</tr>
<tr>
<td>rowCumsums, ANY-method (rowCumsums)</td>
<td>17</td>
</tr>
<tr>
<td>rowCumsums, matrix_OR_array_OR_table_OR_numeric-method (rowCumsums)</td>
<td>17</td>
</tr>
<tr>
<td>rowDiffs, ANY-method (rowDiffs)</td>
<td>19</td>
</tr>
<tr>
<td>rowDiffs, matrix_OR_array_OR_table_OR_numeric-method (rowDiffs)</td>
<td>19</td>
</tr>
<tr>
<td>rowIQRDiffs, ANY-method (rowIQRDiffs)</td>
<td>20</td>
</tr>
<tr>
<td>rowIQRDiffs, matrix_OR_array_OR_table_OR_numeric-method (rowIQRDiffs)</td>
<td>20</td>
</tr>
<tr>
<td>rowIQRs, ANY-method (rowIQRs)</td>
<td>21</td>
</tr>
<tr>
<td>rowIQRs, matrix_OR_array_OR_table_OR_numeric-method (rowIQRs)</td>
<td>22</td>
</tr>
<tr>
<td>rowLogSumExps, ANY-method (rowLogSumExps)</td>
<td>23</td>
</tr>
<tr>
<td>rowLogSumExps, matrix_OR_array_OR_table_OR_numeric-method (rowLogSumExps)</td>
<td>23</td>
</tr>
<tr>
<td>rowAnys, ANY-method (rowAnys)</td>
<td>6</td>
</tr>
<tr>
<td>rowAvgsPerColSet, ANY-method (rowAvgsPerColSet)</td>
<td>8</td>
</tr>
<tr>
<td>rowAvgsPerColSet, matrix_OR_array_OR_table_OR_numeric-method (rowAvgsPerColSet)</td>
<td>8</td>
</tr>
<tr>
<td>rowCollapse, ANY-method (rowCollapse)</td>
<td>10</td>
</tr>
<tr>
<td>rowCollapse, matrix_OR_array_OR_table_OR_numeric-method (rowCollapse)</td>
<td>10</td>
</tr>
<tr>
<td>rowCounts, ANY-method (rowCounts)</td>
<td>11</td>
</tr>
<tr>
<td>rowCounts, matrix_OR_array_OR_table_OR_numeric-method (rowCounts)</td>
<td>11</td>
</tr>
<tr>
<td>rowCummaxs, ANY-method (rowCummaxs)</td>
<td>13</td>
</tr>
<tr>
<td>rowCummaxs, matrix_OR_array_OR_table_OR_numeric-method (rowCummaxs)</td>
<td>13</td>
</tr>
<tr>
<td>rowCummins, ANY-method (rowCummins)</td>
<td>14</td>
</tr>
<tr>
<td>rowCummins, matrix_OR_array_OR_table_OR_numeric-method (rowCummins)</td>
<td>14</td>
</tr>
<tr>
<td>rowCumprods, ANY-method (rowCumprods)</td>
<td>16</td>
</tr>
<tr>
<td>rowCumprods, matrix_OR_array_OR_table_OR_numeric-method (rowCumprods)</td>
<td>16</td>
</tr>
<tr>
<td>rowCumsums, ANY-method (rowCumsums)</td>
<td>17</td>
</tr>
<tr>
<td>rowCumsums, matrix_OR_array_OR_table_OR_numeric-method (rowCumsums)</td>
<td>17</td>
</tr>
<tr>
<td>rowDiffs, ANY-method (rowDiffs)</td>
<td>19</td>
</tr>
<tr>
<td>rowDiffs, matrix_OR_array_OR_table_OR_numeric-method (rowDiffs)</td>
<td>19</td>
</tr>
<tr>
<td>rowIQRDiffs, ANY-method (rowIQRDiffs)</td>
<td>20</td>
</tr>
<tr>
<td>rowIQRDiffs, matrix_OR_array_OR_table_OR_numeric-method (rowIQRDiffs)</td>
<td>20</td>
</tr>
<tr>
<td>rowIQRs, ANY-method (rowIQRs)</td>
<td>21</td>
</tr>
<tr>
<td>rowIQRs, matrix_OR_array_OR_table_OR_numeric-method (rowIQRs)</td>
<td>22</td>
</tr>
<tr>
<td>rowLogSumExps, ANY-method (rowLogSumExps)</td>
<td>23</td>
</tr>
<tr>
<td>rowLogSumExps, matrix_OR_array_OR_table_OR_numeric-method (rowLogSumExps)</td>
<td>23</td>
</tr>
</tbody>
</table>
rowMadDiffs, ANY-method (rowMadDiffs) 25
(rowMadDiffs, matrix OR array OR table OR numeric-method (rowMadDiffs)) 25
rowMads, 26, 27, 28, 55
rowMads, ANY-method (rowMads) 26
(rowMads, matrix OR array OR table OR numeric-method (rowMads)) 26
rowMaxs, 14, 28, 29, 35, 41
rowMaxs, ANY-method (rowMaxs) 28
(rowMaxs, matrix OR array OR table OR numeric-method (rowMaxs)) 28
rowMeans, 28, 30, 32, 33, 46, 49, 54
rowMeans2, 28, 31, 32, 33, 46, 49, 54, 57
rowMeans2, ANY-method (rowMeans2) 31
(rowMeans2, matrix OR array OR table OR numeric-method (rowMeans2)) 31
rowMedians, 32, 33, 39
rowMedians, ANY-method (rowMedians) 32
(rowMedians, matrix OR array OR table OR numeric-method (rowMedians)) 32
rowMins, 15, 29, 34, 35, 41
rowMins, ANY-method (rowMins) 34
(rowMins, matrix OR array OR table OR numeric-method (rowMins)) 34
rowOrderStats, 35, 36
rowOrderStats, ANY-method (rowOrderStats) 35
(rowOrderStats, matrix OR array OR table OR numeric-method (rowOrderStats)) 35
rowProds, 37, 38
rowProds, ANY-method (rowProds) 37
(rowProds, matrix OR array OR table OR numeric-method (rowProds)) 37
rowQuantiles, 38, 39
rowQuantiles, ANY-method (rowQuantiles) 38
(rowQuantiles, matrix OR array OR table OR numeric-method (rowQuantiles)) 38
rowRanges, 40, 41
rowRanges, ANY-method (rowRanges) 40
(rowRanges, matrix OR array OR table OR numeric-method (rowRanges)) 40
rowRanks, 42, 43
rowRanks, ANY-method (rowRanks) 42
(rowRanks, matrix OR array OR table OR numeric-method (rowRanks)) 42
rowSds, 23, 28, 45, 46, 54, 60
rowSds, ANY-method (rowSds) 45
(rowSds, matrix OR array OR table OR numeric-method (rowSds)) 54
(rowSds, ANY-method (rowSds)) 45
(rowSds, matrix OR array OR table OR numeric-method (rowSds)) 45
rowSums, 47
rowSums2, 38, 48, 49
(rowSums2, ANY-method (rowSums2)) 48
(rowSums2, matrix OR array OR table OR numeric-method (rowSums2)) 48
rowTabulates, 49, 50
(rowTabulates, ANY-method (rowTabulates)) 49
(rowTabulates, matrix OR array OR table OR numeric-method (rowTabulates)) 49
rowWeightedMads, 54, 55
rowWeightedMads, ANY-method (rowWeightedMads) 54
(rowWeightedMads, matrix OR array OR table OR numeric-method (rowWeightedMads)) 54
rowWeightedMeans, 56, 57
(rowWeightedMeans, ANY-method (rowWeightedMeans)) 56
(rowWeightedMeans, matrix OR array OR table OR numeric-method (rowWeightedMeans)) 56
rowWeightedSds, 59, 60
(rowWeightedSds, ANY-method (rowWeightedSds)) 59
(rowWeightedSds, matrix OR array OR table OR numeric-method (rowWeightedSds)) 59
INDEX

rowWeightedSds, matrix OR array OR table OR numeric-method
  (rowWeightedSds), 59
rowWeightedVars, 60, 61
rowWeightedVars, ANY-method
  (rowWeightedVars), 60
rowWeightedVars, matrix OR array OR table OR numeric-method
  (rowWeightedVars), 60

stats::quantile, 39
sum, 49

table, 4, 6, 7, 9, 10, 12, 14–16, 18, 20, 21, 23,
  24, 26, 27, 29, 32, 33, 35, 36, 38, 39,
  41, 43, 45, 46, 49, 50, 52, 54, 55, 57,
  58, 60, 61
TRUE, 4, 6, 7, 10, 12, 13, 15, 16, 18, 19, 21, 22,
  24, 26, 27, 29–31, 33, 34, 36, 37, 39,
  41, 42, 44, 46, 47, 49, 50, 52, 53, 55,
  57, 58, 60, 61

var, 54
vector, 4, 6, 7, 9–13, 15, 16, 18, 19, 21–50,
  52–55, 57, 58, 60, 61