Package ‘MatrixGenerics’

February 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  1.14.0

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  7.2.3

Roxygen  list(markdown = TRUE, old_usage = TRUE)

git_url  https://git.bioconductor.org/packages/MatrixGenerics

git_branch  RELEASE_3_18

git_last_commit  9a52271

git_last_commit_date  2023-10-24

Repository  Bioconductor 3.18

Date/Publication  2024-02-18

Author  Constantin Ahlmann-Eltze [aut]
        (https://orcid.org/0000-0002-3762-068X),
        Peter Hickey [aut, cre] (https://orcid.org/0000-0002-8153-6258),
        Hervé Pagès [aut]

Maintainer  Peter Hickey <peter.hickey@gmail.com>

R topics documented:

MatrixGenerics-package ........................................... 3
rowAlls .......................................................... 3
rowAnyNAs ....................................................... 5
rowAnys .......................................................... 6
rowAvgsPerColSet ................................................ 8
rowCollapse ...................................................... 9
rowCounts ......................................................... 11
rowCummaxs ....................................................... 12
rowCummins ....................................................... 14
rowCumprods ..................................................... 15
rowCumsums ....................................................... 17
rowDiffs ............................................................ 18
rowIQRDiffs ...................................................... 20
rowIQRs ........................................................... 21
rowLogSumExps .................................................. 23
rowMadDiffs ....................................................... 24
rowMads ........................................................... 26
rowMaxs ............................................................ 27
rowMeans .......................................................... 29
rowMeans2 ........................................................ 30
rowMedians ......................................................... 32
rowMins ............................................................ 33
rowOrderStats .................................................... 35
rowProds .......................................................... 36
rowQuantiles ...................................................... 38
rowRanges ......................................................... 39
rowRanks .......................................................... 41
rowSdDiffs ........................................................ 43
rowSds .............................................................. 45
rowSums ............................................................ 46
rowSums2 .......................................................... 47
rowTabulates ...................................................... 49
The MatrixGenerics package defines S4 generic summary statistic functions that operate on matrix-Like objects.

**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.

**Usage**

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

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

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

## S4 method for signature 'ANY'

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

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

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

```r
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

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

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

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

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

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

## 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

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

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

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

## S4 method for signature 'ANY'

```r
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)

```r
## 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 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::rowAnys` or `matrixStats::colAnys`.

### Value

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

### 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)
```
**rowAvgsPerColSet**

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

**Description**

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

**Usage**

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

## S4 method for signature 'matrix\_OR\_array\_OR\_table\_OR\_numeric'

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

## S4 method for signature 'ANY'

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

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

## S4 method for signature 'matrix\_OR\_array\_OR\_table\_OR\_numeric'

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

## S4 method for signature 'ANY'

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

**Arguments**

- **X**
  - An N\*M matrix-like object.
- **W**
  - An optional numeric N\*M matrix of weights.
- **rows, cols**
  - A vector indicating the subset (and/or columns) to operate over. If NULL, no subsetting is done.
- **S**
  - An integer K\*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 to each column (row) subset of X that is specified by S.
- **...**
  - Additional arguments passed to FUN.
- **na.rm**
  - (logical) Argument passed to FUN() as na.rm = na.rm. If NA (default), then na.rm = TRUE is used if X or S holds missing values, otherwise na.rm = FALSE.
- **tFUN**
  - If TRUE, X is transposed before it is passed to FUN.
Details

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

Value

Returns a numeric JxN (MxJ) matrix.

See Also

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

Examples

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)
## 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.

### 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

```r
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**

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

### S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

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

### S4 method for signature 'ANY'

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

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

### S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

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

### S4 method for signature 'ANY'

```r
colCounts(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::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)
rowCounts(mat)
colCounts(mat)
  rowCounts(mat, value = 0)
colCounts(mat, value = Inf, na.rm = TRUE)
```

Description

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

Usage

```r
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.

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

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

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

Returns a numeric matrix with the same dimensions as x.

• 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().

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.

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

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)
rowCumsums

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

Description

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

Usage

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

## 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)

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

## 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)
```

---

### Description

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

### Usage

```r
rowDiffs(x, rows = NULL, cols = NULL, lag = 1L, differences = 1L, ..., useNames = TRUE)
```

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

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

```r
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: \( N \times (K - 1) \) or \( (N - 1) \times K \).

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

rowIQRDiffs(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'
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, diff = 1L, trim = 0, ..., useNames = TRUE)

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

## S4 method for signature 'ANY'
colIQRDiffs(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::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
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0

print(mat)
rowIQRDiffs(mat)
colIQRDiffs(mat)

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

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

Usage
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.

**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**

```r
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

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

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

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

print(m)
rowLogSumExps(m)
colLogSumExps(m)
```

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)
```
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
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.

**Usage**

```r
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)
```

```r
colMads(x, rows = NULL, cols = NULL, center = NULL, constant = 1.4826, na.rm = FALSE, ..., 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.
rowMaxs

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

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

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.
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, 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 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.

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).
**rowMeans**

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, ...) 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**

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

**Examples**

```r
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)
```
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)
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)
```

```r
## 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)
```

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

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

```r
## 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)
```

```r
## 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.

Details

The S4 methods for x of type matrix, array, table, or numeric call matrixStats::rowMeans2/ matrixStats::colMeans2.

Value

Returns a numeric vector of length N (K).

See Also

- matrixStats::rowMeans2() and matrixStats::colMeans2() which are used when the input is a matrix or numeric vector.
- See also rowMeans() for the corresponding function in base R.
- For variance estimates, see rowVars().
- See also the base R version 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
rowMedians(x, rows = NULL, cols = NULL, na.rm = FALSE, ..., useNames = TRUE)

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

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

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

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

## S4 method for signature 'ANY'
colMedians(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.
Details

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

Value

Returns a numeric vector of length N (K).

See Also

- matrixStats::rowMedians() and matrixStats::colMedians() which are used when the input is a matrix or numeric vector.
- For mean estimates, see rowMeans2() and 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)
rowMedians(mat)
colMedians(mat)
```

rowMins

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

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)
```

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

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)
```
rowOrderStats  
**Calculates an order statistic for each row (column) of a matrix-like object**

---

**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)
```

---

**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`.
rowProds

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)
```

Description

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

Usage

```r
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"), ...,
```
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.

Details

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

Value

Returns a numeric vector of length N (K).

See Also

- 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

```r
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)
```
rowQuantiles

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

Description

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

Usage

rowQuantiles(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'
rowQuantiles(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'
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

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.
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 Nx2 (Kx2) 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.
- **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’ NxJ (KxJ) 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)
```

---

**rowSdDiffs**  
*C 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)
```
## 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.

### 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)
```
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)
```

```r
## S4 method for signature 'matrix.OR_array.OR_table.OR_numeric'
rowSds(x, rows = NULL, cols = NULL, na.rm = FALSE, center = NULL, dim. = dim(x), ..., useNames = TRUE)
```

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

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

```r
## 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)
```

```r
## S4 method for signature 'ANY'
colSds(x, rows = NULL, cols = NULL, na.rm = FALSE, center = NULL, ..., 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.
- **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 \texttt{matrix}, \texttt{array}, \texttt{table}, or \texttt{numeric} call \texttt{matrixStats::rowSds} / \texttt{matrixStats::colSds}.

Value

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

See Also

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

---

\textbf{Description}

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

\textbf{Usage}

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

\textbf{Arguments}

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

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

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, ...)
## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

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

## S4 method for signature 'ANY'

```r
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.

- **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::rowSums2 / matrixStats::colSums2`.

### Value

Returns a numeric vector of length N (K).

### See Also

- `matrixStats::rowSums2()` and `matrixStats::colSums2()` which are used when the input is a matrix or numeric vector.
- For mean estimates, see `rowMeans2()` and `rowMeans()`.
- `base::sum()`.

### Examples

```r
mat <- matrix(rnorm(15), nrow = 5, ncol = 3)
mat[2, 1] <- NA
mat[3, 3] <- Inf
mat[4, 1] <- 0
print(mat)
```
rowTabulates
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 $N \times J$ (or $K \times J$) 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

```r
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

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

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

\begin{itemize}
  \item \textbf{x} \hspace{1cm} An NxK matrix-like object.
  \item \textbf{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 \textbf{na.rm} \hspace{1cm} If \texttt{TRUE}, missing values (\texttt{NA} or \texttt{NaN}) are omitted from the calculations.
  \item \textbf{diff} \hspace{1cm} An integer specifying the order of difference.
  \item \textbf{trim} \hspace{1cm} A double in \([0,1/2]\) specifying the fraction of observations to be trimmed from each end of (sorted) \texttt{x} before estimation.
  \item \textbf{...} \hspace{1cm} Additional arguments passed to specific methods.
  \item \textbf{useNames} \hspace{1cm} If \texttt{TRUE} (default), names attributes of result are set. Else if \texttt{FALSE}, no naming support is done.
\end{itemize}

\textbf{Details}

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

\textbf{Value}

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

\textbf{See Also}

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

\textbf{Examples}

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

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.
rowWeightedMads

### 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)
```

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

```r
## 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.
rowWeightedMeans

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,
rowWeightedMeans

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.
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  \hspace{1cm} 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**

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

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

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

## S4 method for signature 'ANY'

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

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

## S4 method for signature 'matrix_OR_array_OR_table_OR_numeric'

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

## S4 method for signature 'ANY'

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

**Arguments**

- `x`  \hspace{1cm} An NxK matrix-like object.
- `w`  \hspace{1cm} A numeric vector of length K (N) that specifies by how much each element is weighted.
- `rows, cols`  \hspace{1cm} A vector indicating the subset of rows (and/or columns) to operate over. If NULL, no subsetting is done.
- `na.rm`  \hspace{1cm} If TRUE, missing values (NA or NaN) are omitted from the calculations.
- `...`  \hspace{1cm} Additional arguments passed to specific methods.
- `useNames`  \hspace{1cm} 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).

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

```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)
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

```r
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.

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, 29
  rowSums, 46
* arith
  rowMeans, 29
  rowSums, 46
* array
  rowAvgsPerColSet, 8
  rowCollapse, 9
  rowCummaxs, 12
  rowCummins, 14
  rowCumprods, 15
  rowCumsums, 17
  rowDiffs, 18
  rowIQRDiffs, 20
  rowIQRs, 21
  rowLogSumExps, 23
  rowMadDiffs, 24
  rowMads, 26
  rowMaxs, 27
  rowMeans, 29
  rowMeans2, 30
  rowMedians, 32
  rowMins, 33
  rowOrderStats, 35
  rowProds, 36
  rowQuantiles, 38
  rowRanges, 39
  rowRanks, 41
  rowSdDiffs, 43
  rowSds, 45
  rowSums, 46
  rowSums2, 47
  rowTabulates, 49
  rowVarDiffs, 50
  rowVars, 52
  rowWeightedMads, 53
  rowWeightedMeans, 55
  rowWeightedMedians, 57
  rowWeightedSds, 58
  rowWeightedVars, 60
* iteration
  rowAvgsPerColSet, 8
  rowCollapse, 9
  rowCummaxs, 12
  rowCummins, 14
  rowCumprods, 15
  rowCumsums, 17
  rowDiffs, 18
  rowIQRDiffs, 20
  rowIQRs, 21
  rowLogSumExps, 23
  rowMadDiffs, 24
  rowMads, 26
  rowMaxs, 27
  rowMeans, 29
  rowMeans2, 30
  rowMedians, 32
  rowMins, 33
  rowOrderStats, 35
  rowProds, 36
  rowQuantiles, 38
  rowRanges, 39
  rowRanks, 41
  rowSdDiffs, 43
  rowSds, 45
  rowSums, 46
  rowSums2, 47
  rowTabulates, 49
  rowVarDiffs, 50
  rowVars, 52
  rowWeightedMads, 53
  rowWeightedMeans, 55
  rowWeightedMedians, 57
  rowWeightedSds, 58
  rowWeightedVars, 60
* robust
  rowAvgsPerColSet, 8
logical, 4, 5, 7
mad, 26, 54
matrix, 4, 5, 7, 9, 10, 12, 13, 15–19, 21, 22, 24, 25, 27, 28, 31, 33–35, 37, 39, 40, 42, 44, 46, 48–51, 53, 54, 56, 58, 59, 61
matrix_OR_array_OR_table_OR_numeric (MatrixGenerics-package), 3
matrix_OR_array_OR_table_OR_numeric-class (MatrixGenerics-package), 3
MatrixGenerics-package, 3
NULL, 4, 5, 7, 8, 10, 11, 13, 14, 16, 17, 19, 20, 22, 23, 25, 26, 28, 31, 32, 34, 35, 37, 38, 40, 42, 44, 45, 48, 49, 51, 52, 54, 56, 57, 59, 60
prod, 37
quantile, 38
range, 41
RangedSummarizedExperiment, 41
rowAlls, 3, 4, 7, 12
rowAlls, ANY-method (rowAlls), 3
rowAlls, matrix_OR_array_OR_table_OR_numeric-method (rowAlls), 3
rowAnyNAs, 5, 5, 6
rowAnyNAs, ANY-method (rowAnyNAs), 5
rowAnyNAs, matrix_OR_array_OR_table_OR_numeric-method (rowAnyNAs), 5
rowAnys, 4, 6, 6, 7, 12
rowAnys, ANY-method (rowAnys), 6
rowAnys, matrix_OR_array_OR_table_OR_numeric-method (rowAnys), 6
rowAvgssPerColSet, 8, 9
rowAvgssPerColSet, ANY-method (rowAvgssPerColSet), 8
rowAvgssPerColSet, matrix_OR_array_OR_table_OR_numeric-method (rowAvgssPerColSet), 8
rowCollapse, 9, 10
rowCollapse, ANY-method (rowCollapse), 9
rowCollapse, matrix_OR_array_OR_table_OR_numeric-method (rowCollapse), 9
rowCounts, 11, 12
rowCounts, ANY-method (rowCounts), 11
rowCounts, matrix_OR_array_OR_table_OR_numeric-method (rowCounts), 11
rowCummaxs, 12, 13
rowCummaxs, ANY-method (rowCummaxs), 12
rowCummaxs, matrix_OR_array_OR_table_OR_numeric-method (rowCummaxs), 12
rowCummins, 14, 15
rowCummins, ANY-method (rowCummins), 14
rowCummins, matrix_OR_array_OR_table_OR_numeric-method (rowCummins), 14
rowCumprods, 15, 16
rowCumprods, ANY-method (rowCumprods), 15
rowCumprods, matrix_OR_array_OR_table_OR_numeric-method (rowCumprods), 15
rowCumsums, 17, 17, 18
rowCumsums, ANY-method (rowCumsums), 17
rowCumsums, matrix_OR_array_OR_table_OR_numeric-method (rowCumsums), 17
rowDiffs, 18, 19
rowDiffs, ANY-method (rowDiffs), 18
rowDiffs, matrix_OR_array_OR_table_OR_numeric-method (rowDiffs), 18
rowIQRDiffs, 20, 21
rowIQRDiffs, ANY-method (rowIQRDiffs), 20
rowIQRDiffs, matrix_OR_array_OR_table_OR_numeric-method (rowIQRDiffs), 20
rowIQRs, 21, 21, 22
rowIQRs, ANY-method (rowIQRs), 21
rowIQRs, matrix_OR_array_OR_table_OR_numeric-method (rowIQRs), 21
rowLogSumExps, 23, 24
rowLogSumExps, ANY-method (rowLogSumExps), 23
rowLogSumExps, matrix_OR_array_OR_table_OR_numeric-method (rowLogSumExps), 23
rowMads, 26, 27, 55
rowMads, ANY-method (rowMads), 26
rowMads, matrix_OR_array_OR_table_OR_numeric-method
rowMeans2, ANY-method (rowMeans2), 24
rowMaxs, ANY-method (rowMaxs), 27
rowMaxs, matrix_OR_array_OR_table_OR_numeric-method (rowMaxs), 27
rowMeans, 27, 29, 31, 33, 46, 48, 53
rowMeans2, 27, 30, 31, 33, 46, 48, 53, 56
rowMeans2, ANY-method (rowMeans2), 30
rowMeans2, matrix_OR_array_OR_table_OR_numeric-method (rowMeans2), 30
rowMedians, 32, 33, 58
rowMedians, ANY-method (rowMedians), 32
rowMedians, matrix_OR_array_OR_table_OR_numeric-method (rowMedians), 32
rowMins, 15, 29, 33, 41
rowMins, ANY-method (rowMins), 33
rowMins, matrix_OR_array_OR_table_OR_numeric-method (rowMins), 33
rowOrderStats, 35, 35, 36
rowOrderStats, ANY-method (rowOrderStats), 35
rowOrderStats, matrix_OR_array_OR_table_OR_numeric-method (rowOrderStats), 35
rowProds, 36, 37
rowProds, ANY-method (rowProds), 36
rowProds, matrix_OR_array_OR_table_OR_numeric-method (rowProds), 36
rowQuantiles, 38, 39
rowQuantiles, ANY-method (rowQuantiles), 38
rowQuantiles, matrix_OR_array_OR_table_OR_numeric-method (rowQuantiles), 38
rowRanges, 39, 40, 41
rowRanges, ANY-method (rowRanges), 39
rowRanges, matrix_OR_array_OR_table_OR_numeric-method (rowRanges), 39
rowRanks, 41, 42, 43
rowRanks, ANY-method (rowRanks), 41
rowRanks, matrix_OR_array_OR_table_OR_numeric-method (rowRanks), 41
rowSdDiffs, 43, 44
rowSdDiffs, ANY-method (rowSdDiffs), 43
rowSdDiffs, matrix_OR_array_OR_table_OR_numeric-method (rowSdDiffs), 43
rowSds, 22, 27, 45, 46, 53, 59
rowSds, ANY-method (rowSds), 44
rowSds, matrix_OR_array_OR_table_OR_numeric-method (rowSds), 45
(rowSds), 46
rowSums, 46
rowSums2, 37, 47, 48
rowVarDiffs, 50, 51
rowVarDiffs, ANY-method (rowVarDiffs), 50
(rowVarDiffs), 51
rowVars, 31, 46, 52, 53, 61
rowVars, ANY-method (rowVars), 52
rowVarDiffs, matrix_OR_array_OR_table_OR_numeric-method (rowVars), 52
rowWeightedMads, 53, 54, 55
rowWeightedMads, ANY-method (rowWeightedMads), 53
rowWeightedMads, matrix_OR_array_OR_table_OR_numeric-method (rowWeightedMads), 53
rowWeightedMeans, 55, 56
rowWeightedMeans, ANY-method (rowWeightedMeans), 55
rowWeightedMeans, matrix_OR_array_OR_table_OR_numeric-method (rowWeightedMeans), 55
rowWeightedMedians, 57, 58
rowWeightedMedians, ANY-method (rowWeightedMedians), 57
rowWeightedMedians, matrix_OR_array_OR_table_OR_numeric-method (rowWeightedMedians), 57
rowWeightedSds, 58, 59
rowWeightedSds, ANY-method (rowWeightedSds), 58
rowWeightedSds, matrix_OR_array_OR_table_OR_numeric-method (rowWeightedSds), 58
rowWeightedVars, 60, 61
rowWeightedVars, ANY-method (rowWeightedVars), 60
rowWeightedVars, matrix_OR_array_OR_table_OR_numeric-method (rowWeightedVars), 60
stats::quantile, 39

---

stats::quantile, 39
INDEX

table, 4, 5, 7, 9, 10, 12, 13, 15–17, 19, 21, 22, 24, 25, 27, 28, 31, 33–35, 37, 39, 40, 42, 44, 46, 48–51, 53, 54, 56, 58, 59, 61
var, 53