Package ‘dibble’

June 23, 2024

Type Package

Title Dimensional Data Frames

Version 0.3.0

Description Provides a 'dibble' that implements data cubes (derived from 'dimensional tibble'), and allows broadcasting by dimensional names.

License MIT + file LICENSE

Encoding UTF-8

Imports dplyr, memoise, pillar, purrr (>= 1.0.0), rlang, tibble, tidyr, tidyselect, vctrs

RoxygenNote 7.3.1

Suggests covr, testthat (>= 3.0.0)

Config/testthat/edition 3

Depends R (>= 4.4)

URL https://github.com/UchidaMizuki/dibble, https://uchidamizuki.github.io/dibble/

BugReports https://github.com/UchidaMizuki/dibble/issues

NeedsCompilation no

Author Mizuki Uchida [aut, cre]

Maintainer Mizuki Uchida <uchidamizuki@vivaldi.net>

Repository CRAN

Date/Publication 2024-06-23 06:30:01 UTC

Contents

---

apply *Apply functions over array margins*

Description

Applying a function to margins of a dibble or array, including a matrix.

Usage

```
apply(X, MARGIN, FUN, ...)

## Default S3 method:
apply(X, MARGIN, FUN, ..., simplify = TRUE)

## S3 method for class 'tbl_ddf'
apply(X, MARGIN, FUN, ...)

## S3 method for class 'ddf_col'
apply(X, MARGIN, FUN, ...)
```

Arguments

`X`	A dibble or array, including a matrix.
`MARGIN`	An integer or character vector giving the subscripts which the function will be applied over.
`FUN`	A function to be applied.
`...`	Optional arguments to FUN.
`simplify`	A logical indicating whether results should be simplified if possible.

Details

`apply()` overrides `base::apply()` to make it generic. The default method calls the base version.

Value

A dibble if X is a dibble. See `base::apply()` for the return value of the default method.

See Also

`base::apply()`.

Examples

```
x <- array(1:24, 2:4,
           list(axis1 = letters[1:2],
                axis2 = letters[1:3],
                axis3 = letters[1:4]))

apply(x, 2:3, sum)
apply(as_dibble(x), 2:3, sum)

apply(x, c("axis2", "axis3"), sum)
apply(as_dibble(x), c("axis2", "axis3"), sum)
```

`as_dibble` *Coerce an object to a dibble*

Description

`as_dibble()` turns an object into a dimensional data frame called a dibble.

Usage

```
as_dibble(x, ...)

## Default S3 method:
as_dibble(x, ...)

## S3 method for class 'rowwise_df'
as_dibble(x, ...)

## S3 method for class 'grouped_df'
as_dibble(x, ...)

## S3 method for class 'ddf_col'
as_dibble(x, ...)

## S3 method for class 'tbl_ddf'
as_dibble(x, ...)
```

Arguments

`x`	An object.
`...`	Unused, for extensibility.

Value

A dibble.

basic-matrices-arrays *Basic matrices and arrays*

Description

Create basic matrices and arrays.

Usage

```
eye(x, ...)

## Default S3 method:
eye(x, y = x, ...)

## S3 method for class 'matrix'
eye(x, ...)

## S3 method for class 'ddf_col'
eye(x, ...)

## S3 method for class 'tbl_ddf'
eye(x, ...)

ones(x, ...)

## Default S3 method:
ones(x, y = x, ...)

## S3 method for class 'array'
ones(x, ...)

## S3 method for class 'ddf_col'
ones(x, ...)

## S3 method for class 'tbl_ddf'
ones(x, ...)

zeros(x, ...)

## Default S3 method:
zeros(x, y = x, ...)

## S3 method for class 'array'
```

```
zeros(x, ...)

## S3 method for class 'ddf_col'
zeros(x, ...)

## S3 method for class 'tbl_ddf'
zeros(x, ...)
```

Arguments

`x`	An object.
`...`	Other arguments passed on to methods.
`y`	A scalar integer.

Details

These functions override base functions to make them generic. The default methods call the base versions.

Value

A dibble if x is a dibble. Otherwise, returns a matrix or an array.

`broadcast` *Broadcast to a new dimension*

Description

Broadcasts the dimension of the object to a new dimension.

Usage

```
broadcast(x, dim_names = NULL, ...)

## Default S3 method:
broadcast(x, dim_names = NULL, ...)

## S3 method for class 'ddf_col'
broadcast(x, dim_names, ...)

## S3 method for class 'tbl_ddf'
broadcast(x, dim_names, ...)
```

Arguments

`x`	A dibble, vector, or array.
`dim_names`	A character vector or list of dimension names.
`...`	Unused, for extensibility.

Details

Operations between dibbles are automatically broadcasted, but for safety reasons, warnings are issued. broadcast() can suppress the warnings if dim_names matches the dimension of x.

Value

A dibble.

Examples

```
x <- broadcast(1:2,
               list(axis1 = letters[1:2]))
y <- broadcast(1:3,
               list(axis2 = letters[1:3]))
broadcast(x * y, c("axis1", "axis2"))
```

---

diag *Matrix diagonals*

Description

Extract or replace the diagonal of a matrix, or construct a diagonal matrix.

Usage

```
diag(x, ...)

## Default S3 method:
diag(x = 1, nrow, ncol, names, ...)

## S3 method for class 'tbl_ddf'
diag(x, axes, ...)

## S3 method for class 'ddf_col'
diag(x, axes, ...)

diag(x, ...) <- value

## Default S3 replacement method:
diag(x, ...) <- value

## S3 replacement method for class 'tbl_ddf'
diag(x, ...) <- value

## S3 replacement method for class 'ddf_col'
diag(x, ...) <- value
```

Arguments

`x`	A dibble, matrix, vector or 1D array, or missing.
`...`	Unused, for extensibility.
`nrow, ncol`	Optional dimensions for the result when x is not a matrix.
`names`	(When x is a matrix) logical indicating if the resulting vector, the diagonal of x, should inherit names from dimnames(x) if available.
`axes`	A character vector of axes.
`value`	Replacement values.

Details

These functions override base functions to make them generic. The default methods call the base versions.

Value

A dibble if x is a dibble. See `base::diag()` for the return values of the default methods.

`dibble` *Build a dimensional data frame*

Description

`dibble()` constructs a dimensional data frame called a dibble.

Usage

```
dibble(..., .dim_names = NULL)
```

Arguments

`...`	A set of name-measure pairs.
`.dim_names`	A list of dimension names.

Details

Manipulation functions:

- mutate()
- rename()
- select() & relocate()
- slice()

Value

A dibble.

dibble_by *Constructs a dibble by one or more variables*

Description

dibble_by() constructs a dibble by one or more variables.

Usage

```
dibble_by(x, ..., .names_sep = NULL)
```

Arguments

x	A data frame or a dibble.
...	Variables.
.names_sep	Passed to tidyr::pack().

Value

A dibble.

extremes *Maxima and Minima*

Description

Returns the parallel maxima and minima of the input values.

Usage

```
pmax(..., na.rm = FALSE)

## Default S3 method:
pmax(..., na.rm = FALSE)

## S3 method for class 'ddf_col'
pmax(..., na.rm = FALSE)

## S3 method for class 'tbl_ddf'
pmax(..., na.rm = FALSE)

pmin(..., na.rm = FALSE)

## Default S3 method:
pmin(..., na.rm = FALSE)
```

```
## S3 method for class 'ddf_col'
pmin(..., na.rm = FALSE)

## S3 method for class 'tbl_ddf'
pmin(..., na.rm = FALSE)
```

Arguments

`...`	Dibbles, numeric or character arguments.
`na.rm`	a logical indicating whether missing values should be removed.

Details

These functions override base functions to make them generic. The default methods call the base versions.

Value

A dibble if `...` are dibbles. See `base::pmax()` and `base::pmin()` for the return value of the default method.

See Also

`base::pmax()`, `base::pmin()`.

`ifelse` *Conditional element selection*

Description

Selects elements from either `yes` or `no` depending on whether `test` is `TRUE` or `FALSE`.

Usage

```
ifelse(test, yes, no, ...)

## Default S3 method:
ifelse(test, yes, no, ...)

## S3 method for class 'tbl_ddf'
ifelse(test, yes, no, ...)

## S3 method for class 'ddf_col'
ifelse(test, yes, no, ...)
```

Arguments

`test`	An object which can be coerced to logical mode.
`yes`	Return values for true elements of test.
`no`	Return values for false elements of test.
`...`	Unused, for extensibility.

Details

`ifelse()` overrides `base::ifelse()` to make it generic. The default method calls the base version.

Value

A dibble if test is a dibble. See `base::ifelse()` for the return value of the default method.

See Also

`base::ifelse()`.

`is_dibble` *Test if the object is a dibble*

Description

Test if the object is a dibble

Usage

```
is_dibble(x)
```

Arguments

`x`	An object.

Value

A logical.

nrow-ncol *The number of rows/columns*

Description

nrow() and ncol() return the number of rows or columns present in x.

Usage

```
nrow(x, ...)

## Default S3 method:
nrow(x, ...)

## S3 method for class 'ddf_col'
nrow(x, ...)

## S3 method for class 'tbl_ddf'
nrow(x, ...)

ncol(x, ...)

## Default S3 method:
ncol(x, ...)

## S3 method for class 'ddf_col'
ncol(x, ...)

## S3 method for class 'tbl_ddf'
ncol(x, ...)
```

Arguments

x	An object.
...	Other arguments passed on to methods.

Details

These functions override base functions to make them generic. The default methods call the base versions.

Value

An integer or NULL.

row-colnames *Row and column names*

Description

Retrieve or set the row or column names of a matrix-like object.

Usage

```
rownames(x, ...)

## Default S3 method:
rownames(x, ...)

## S3 method for class 'ddf_col'
rownames(x, ...)

## S3 method for class 'tbl_ddf'
rownames(x, ...)

colnames(x, ...)

## Default S3 method:
colnames(x, ...)

## S3 method for class 'ddf_col'
colnames(x, ...)

## S3 method for class 'tbl_ddf'
colnames(x, ...)
```

Arguments

x	A matrix-like object.
...	Other arguments passed on to methods.

Details

These functions override base functions to make them generic. The default methods call the base versions.

Value

A list of row/column names.

Index