

Package: Dmisc (via r-universe)

October 21, 2024

Title Daniel miscellaneous functions

Version 0.3.11

Description The Dmisc package provides a collection of versatile R functions developed by Daniel E. de la Rosa. These are general-purpose tools, not tied to any specific domain, aiming to aid a wide range of tasks in data analysis and visualization.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

URL <https://drdsdaniel.github.io/Dmisc/>,
<https://github.com/drdsdaniel/Dmisc>

BugReports <https://github.com/drdsdaniel/Dmisc/issues>

Language en-US

RdMacros lifecycle

VignetteBuilder knitr

Imports conflicted, dplyr, glue, lifecycle, magrittr, odbc, rlang, stringr, tidyr

Suggests cli, DBI, DT, flextable, forecast, ggcorrplot, ggplot2, htmltools, janitor, keyring, knitr, lmtest, lubridate, patchwork, pins, purrr, renv, RPostgres, rsample, rvest, scales, sf, strucchange, testthat, tibble, tidyselect, tseries, vars, withr, xts, yardstick, zoo

Repository <https://adatar-do.r-universe.dev>

RemoteUrl <https://github.com/dnldelaros/Dmisc>

RemoteRef HEAD

RemoteSha c4e5beb8e392e779b3503f9e781382c97a22dc12

Contents

cut3	2
cut3_quantile	3
describe	3
pin_get_or_create	4
vars_to_date	5
%<...%	6
%...>%	7

Index	8
--------------	----------

cut3	<i>Convert numeric variables to weighted factors by group</i> [Experimental]
------	---

Description

This function converts a numeric column in a data frame to a factor variable, allowing for custom break points and grouping.

Usage

```
cut3(tbl, var_name, breaks, groups = NULL, bf_args = list(), .inf = FALSE, ...)
```

Arguments

tbl	data.frame : The data frame containing the data to be converted.
var_name	character : The name of the numeric variable to convert to a factor.
breaks	numeric : Break points defining factor levels. See See Also section for more details.
groups	character : The name of the variable for grouping data before conversion.
bf_args	list : Additional arguments to be passed to the break function.
.inf	logical : Whether to extend break points with -Inf and Inf.
...	Additional arguments passed to base::cut .

Value

A data frame identical to the input `tbl`, with `var_name` converted to a factor.

See Also

[base::cut](#), for the underlying cut function used. `vignette("cut3", package = "Dmisc")`, for examples and extended usage.

Examples

```
datos <- data.frame(edad = seq(1:100))
dplyr::count(cut3(datos, "edad", 5), edad)
```

cut3_quantile	<i>Cut3 by quantile</i> [Experimental]
---------------	---

Description

Cut3 by quantile **[Experimental]**

Usage

```
cut3_quantile(tbl, var_name, .labels = NULL, .groups = NULL, .inf = TRUE, ...)
```

Arguments

tbl	data.frame : Database connection or data.frame
var_name	character : variable name
.labels	list : labels for the breaks
.groups	character : name of a groups variable
.inf	logical : indicates if the breaks need to be extended by -Inf and Inf
...	argument passed to quantile

Value

same as tbl input with var_name converted to factor by quantiles

Examples

```
## Not run:  
datos <- data.frame(edad = seq(1:100))  
cut3_quantile(datos, "edad")  
  
## End(Not run)
```

describe	<i>Create a summary table for a data frame</i> [Experimental]
----------	--

Description

This function generates a summary table for a data frame, containing summary statistics for each variable in the data frame.

Usage

```
describe(data, digits = 4, t = TRUE, flextable = FALSE, ft_args = list(), ...)
```

Arguments

<code>data</code>	A data frame containing the data to be summarized
<code>digits</code>	The number of 0digits to display in the summary table (default is 4)
<code>t</code>	If TRUE, the table will be transposed (default is TRUE)
<code>flextable</code>	If TRUE, the table will be converted to a simple flextable (default is FALSE)
<code>ft_args</code>	A list of additional arguments to pass to the flextable function if flextable = TRUE (default is an empty list)
<code>...</code>	Other arguments to pass to the summary function

Value

A table containing summary statistics for each variable in the data frame

Examples

```
# Using summary
summary(cars)

# Using describe
describe(cars)
describe(cars, flextable = TRUE)
```

`pin_get_or_create` *Retrieve or Create a Pin*

Description

This function attempts to retrieve a specified pin from a board. If the pin doesn't exist or the data is different, it creates a new pin.

Usage

```
pin_get_or_create(.data, .board, .name, type = "csv", ...)
```

Arguments

<code>.data</code>	The data to be pinned if a new pin is to be created.
<code>.board</code>	The board where the pin is located or should be created.
<code>.name</code>	The name of the pin to be retrieved or created.
<code>type</code>	The type of data being pinned, defaults to 'csv'.
<code>...</code>	Additional arguments to be passed to <code>pins::pin_write</code> .

Value

The version of the pin that was retrieved or created.

Examples

```
## Not run:
  board <- pins::board_register_local()
  data <- mtcars
  pin_version <- pin_get_or_create(data, board, "mtcars_pin")

## End(Not run)
```

vars_to_date *Multiple variables to unique date variable* **[Experimental]**

Description

Multiple variables to unique date variable **[Experimental]**

Usage

```
vars_to_date(
  tbl,
  year = NULL,
  quarter = NULL,
  month = NULL,
  day = NULL,
  date = NULL,
  drop_vars = TRUE,
  clean_names = FALSE,
  date_format = "%d-%m-%y",
  origin = "1900-01-01",
  .round = c("end", "middle", "start")
)
```

Arguments

tbl	data.frame or tbl connection
year	year variable position or name
quarter	quarter variable position or name
month	month variable position or name
day	day variable position or name
date	a variable name or position containing a like date format
drop_vars	indicates if variables should be dropped
clean_names	indicates if all variable names should be cleaned
date_format	actual date format of variable in date argument
origin	base date for variable conversion to date
.round	indicates if the date should be rounded to the end, middle or start of the period

Value

tbl a new data.frame with the compute variable

Examples

```
tbl <- data.frame(
  year = rep("2021", 12),
  month = month.name,
  day = sample(1:3, 12, TRUE),
  value = sample(100:1000, 12, TRUE)
)
```

```
tbl
```

```
vars_to_date(tbl, year = 1, month = 2, day = 3)
```

and supports various frequencies and date formats

```
tbl <- data.frame(
  year = rep("2021", 12),
  quarter = sample(
    c(
      "Enero-Marzo",
      "Abril-Junio",
      "Julio-Septiembre",
      "Octubre-Diciembre"
    ),
    12,
    TRUE
  ),
  value = sample(100:1000, 12, TRUE)
)
```

```
tbl
```

```
vars_to_date(tbl, year = 1, quarter = 2)
```

%<...%

Unpack and assign values to specified names in an environment **[Experimental]**

Description

Unpack and assign values to specified names in an environment **[Experimental]**

Usage

```
names %<...% values
```

Arguments

names A list or character vector specifying the names of variables to be assigned.
 values A list or vector containing the values to be assigned to the names.

Value

NULL. The function performs assignments in the specified environment.

See Also

Additional arguments that can be passed to names include

- `.envir` The environment where the variables will be assigned. Defaults to the global environment.
- `.warn` Logical flag indicating whether to show a warning message. Defaults to TRUE.

Examples

```
## Not run:
c("x", "y") %<...% list(1, 2)

## End(Not run)
```

`%...>%` *Assign values to specified names in an environment and unpack the values (reverse of %<...%)*

Description

Assign values to specified names in an environment and unpack the values (reverse of %<...%)

Usage

```
values %...>% names
```

Arguments

values A list or vector containing the values to be assigned to the names.
 names A list or character vector specifying the names of variables to be assigned.

Value

NULL. The function performs assignments in the specified environment.

Examples

```
## Not run:
list(1, 2) %...>% c("x", "y")

## End(Not run)
```

Index

`%...>%`, 7

`%<...%`, 6

`base::cut`, 2

character, 2, 3

`cut3`, 2

`cut3_quantile`, 3

data.frame, 2, 3

`describe`, 3

list, 2, 3

logical, 2, 3

numeric, 2

`pin_get_or_create`, 4

`vars_to_date`, 5