Package: dformula (via r-universe)

November 1, 2024

Type Package

Version 1.0

Date 2023-12-01

Title Data Manipulation using Formula

Description A tool for manipulating data using the generic formula. A single formula allows to easily add, replace and remove variables before running the analysis.

Depends R (>= 3.5.0)

Imports utils, stats, formula.tools(>= 1.7.1)

Suggests knitr, rmarkdown

VignetteBuilder knitr

License GPL (>= 2)

URL https://serafinialessio.github.io/dformula/

BugReports https://github.com/serafinialessio/dformula/issues

NeedsCompilation no

Encoding UTF-8

LazyData true

Repository https://serafinialessio.r-universe.dev

RemoteUrl https://github.com/serafinialessio/dformula

RemoteRef HEAD

RemoteSha 3151214f1d812cc41e647e3a0c1b6dd7ef69ca50

Contents

add *Add variables*

Description

Add new variables by mutating the input variables using a formula.

Usage

```
add(from, formula, as = NULL,
    position = c("right", "left"),
    na.remove = FALSE, logic_convert = TRUE,...)
```

Arguments

`from`	a data.frame object with variables
`formula`	a formula indicating the operation to create new varibles. Look at the detail section for explanantion.
`as`	a character vector with names of new variables.
`position`	if the new varaibles are positioned at the begining (`right`) or at the left (`left`) of the data in input.
`na.remove`	a logical value indicating whether NA values should be removed.
`logic_convert`	logical value indicating if the new logical varaible are convertet to `0` or `1`
`...`	further arguments

Details

The formula is composed of two part:

`~ new_variables`

the right-hand are the new varaible to add starting from the existing varaibles, using the `I()` function.

For example:

`~ I(log(column_names1)) + I(column_names2/100)`

the `column_names1` and `log(column_names1)` are added to the data.

If `na.remove` is set ti `TRUE`, new variables are created, added to the dataset in input and then the observation with missing are removed.

Value

Returns a data.frame object with the original and the new varaibles.

Author(s)

Alessio Serafini

Examples

```
data("airquality")
dt <- airquality

head(add(from = dt, formula =   ~ log(Ozone)))
head(add(from = dt, formula =   ~ log(Ozone) +  log(Wind)))
head(add(from = dt, formula =   ~ log(Ozone), as = "Ozone_1"))


head(add(from = dt, formula =  Ozone + Wind ~ log()))
head(add(from = dt, formula =  ~ log()))
head(add(from = dt, formula =  .~ log(), position = "left"))

head(add(from = dt, formula =  .~ log(), na.remove = TRUE))

head(add(from = dt, formula =   ~ I((Ozone>5))))
head(add(from = dt, formula =   ~ I((Ozone>5)), logic_convert = FALSE ))

head(add(from = dt, formula = Ozone + Wind ~ C(Ozone-Ozone)))
head(add(from = dt, formula =  ~ C(log(Ozone))))
head(add(from = dt, formula =  ~ C(5)))
head(add(from = dt, formula = Ozone + Wind ~ C(Ozone-Ozone)))
head(add(from = dt, formula =  Ozone + Wind ~ C(log(Ozone))))



foo <- function(x, a = 100){return(x-x + a)}

head(add(from = dt, formula =  Ozone + Month~ I(foo(a = 100))))
head(add(from = dt, formula =  Ozone + Month~ foo()))
head(add(from = dt, formula =  ~ I(foo(Ozone, a = 100))))
```

`population_data` *World population*

Description

World population and countries are

Usage

```
data("population_data")
```

Format

A data frame with 159 observations on the following 3 variables.

`Country` a character vector with countries names

`Population` a numeric vector with population

`Area` a numeric vector with area of the counties

Source

https://www.worldometers.info

Examples

```
data(population_data)
str(population_data)
```

`remove` *Remove a subset*

Description

Selects the row and the varaibles to remove by specifing a condition using a formula.

Usage

```
remove(from, formula = .~., na.remove = FALSE, ...)
```

Arguments

`from`	a data.frame object with variables
`formula`	a formula indicating the operation to create new varibles. Look at the detail section for explanantion.
`na.remove`	a logical value indicating whether NA values should be removed.
`...`	further arguments

Details

The formula is composed of two part:

`column_names ~ rows_conditions`

the left-hand side are the names of the column to remove, and the right-hand the operation to remove the rows, using the `I()` function.

For example:

`column_names1 + column_names2 ~ I(column_names1 == "a") + I(column_names2 > 4)`

first the row are selected to be removed if the observation in the `column_names1` are equal to `a` and if the observation in the `column_names2` are biggers than `4`, then the `column_names1` and `column_names2` are removed and the other varaibles are returned.

If `na.remove` is set to `TRUE`, after the subsetting the observations with missing are removed.

Value

Returns a data.frame object without the selected elements.

Author(s)

Alessio Serafini

Examples

```
data("airquality")
dt <- airquality

head(remove(from = dt, formula = .~ I(Ozone > 10)))
head(remove(from = dt, formula = .~ I(Ozone > 10), na.remove = TRUE))
head(remove(from = dt, formula = Ozone ~ .))

head(remove(from = dt, formula = Ozone~ I(Ozone > 10)))
head(remove(from = dt, formula = Ozone + Wind~ I(Ozone > 10)))

head(remove(from = dt, formula = Ozone + . ~ I(Ozone > 10)))
head(remove(from = dt, formula = Ozone + NULL ~ I(Ozone > 10)))
```

---

rename *Rename variables*

Description

Rename variables using formulas

Usage

```
rename(from, formula, ...)
```

Arguments

from	a data.frame object with variables
formula	a formula indicating the operation to create new varibles. Look at the detail section for explanantion.
...	further arguments

Details

The formula is composed of two part:

column_names ~ new_variables_name

the left-hand side select the columns to change the names, and the right-hand the new names of the selected columns

For example:

column_names1 + column_names2 ~ new_variables_name1 + new_variables_name2

the name of the column 1 and the name of the column 2 are changed in new_variables_name1 and new_variables_name2

Value

The original data.frame with changed column names

Author(s)

Alessio Serafini

Examples

```
data("airquality")
dt <- airquality

head(rename(from = dt, Ozone ~ Ozone1))
head(rename(from = dt, Ozone + Wind ~ Ozone_new + Wind_new))
```

---

select *Select a subset*

Description

Selects the row and the varaibles by specifing a condition using a formula.

Usage

```
select(from, formula = .~., as = NULL, na.remove = FALSE, na.return = FALSE,...)
```

Arguments

from	a data.frame object with variables
formula	a formula indicating the operation to create new varibles. Look at the detail section for explanantion
as	a character vector with names of new variables.
na.remove	a logical value indicating whether NA values should be removed
na.return	a logical value indicating whether only the observation with NA values should be shown
...	further arguments

Details

The formula is composed of two part:

`column_names ~ row_conditions`

the left-hand side are the names of the column to select, and the right-hand the operations to select the rows, using the `I()` function.

For example:

`column_names1 + column_names2 ~ I(column_names1 == "a") + I(column_names2 > 4)`

first the rows are selected if the observation in the column_names1 are equal to a and if the observation in the column_names2 are biggers than 4, then the column_names1 and column_names2 are returned.

If na.remove is set to TRUE, after the subsetting the observations with missing are removed.

Value

Returns a data.frame object containing the selected elements.

Author(s)

Alessio Serafini

Examples

```
data("airquality")
dt <- airquality

## Selects columns and filter rows

select(from = dt, formula = .~ I(Ozone > 10 & Wind > 10))
select(from = dt, formula = Ozone ~ I(Wind > 10))
select(from = dt, formula = Ozone + Wind~ I(Ozone > 10))

## All rows and filter columns

select(from = dt, formula = Ozone ~ .)
select(from = dt, formula = Ozone + Wind ~ NULL)
```

---

transform *Transform varibles*

Description

Mutate input variables using a formula.

Usage

```
transform(from, formula, as = NULL,
          na.remove = FALSE, logic_convert = TRUE, ...)
```

Arguments

from	a data.frame object with variables
formula	a formula indicating the operation to create new varibles. Look at the detail section for explanantion.
as	a character vector with names of new variables.
na.remove	a logical value indicating whether NA values should be removed.
logic_convert	logical value indicating if the new logical varaible are converted to 0 or 1
...	further arguments

Details

The formula is composed of two part:

column_names ~ trasformed_variables

the left-hand side are the names of the column to transform, and the right-hand the operations applied to the selected columns, using the I() function.

For example:

column_names1 + column_names2 ~ I(log(column_names1)) + I(column_names2/100)

the column_names1 is mutated in log(column_names1) and column_names2 is divided by 100.

If na.remove is set to TRUE, variables are mutaded, and then the observation with missing are removed.

Value

Returns the original data.frame object with mutaded varaibles.

Author(s)

Alessio Serafini

Examples

```
data("airquality")
dt <- airquality

head(transform(from = dt, Ozone ~ I(Ozone-Ozone)))
head(transform(from = dt, Ozone ~ log(Ozone)))
head(transform(from = dt, Ozone ~ I(Ozone>5)))
head(transform(from = dt, Ozone ~ I(Ozone>5), logic_convert = TRUE))


head(transform(from = dt,  ~ log()))
head(transform(from = dt, . ~ log()))
head(transform(from = dt, NULL ~ log()))

head(transform(from = dt, Ozone + Day ~ log()))
head(transform(from = dt, Ozone + Day ~ log(Ozone/100) + exp(Day)))
head(transform(from = dt, Ozone ~ log()))
```

```
head(transform(from = dt,Ozone + Wind ~ C(log(1))))
head(transform(from = dt,Ozone + Wind ~ log(Ozone) + C(10)))


head(transform(from = dt, Ozone + Wind~ C(log(Ozone))))


foo <- function(x, a = 100){return(x-x + a)}
head(transform(from = dt, Ozone + Wind ~ foo(a = 100)))
head(transform(from = dt, . ~ foo(a = 100)))

head(transform(from = dt, Ozone + Wind ~ log(log(1))))
```

Index