

Package ‘dcmmodify’

October 13, 2022

Maintainer Mark van der Loo <mark.vanderloo@gmail.com>

License GPL-3

Title Modify Data Using Externally Defined Modification Rules

Type Package

LazyLoad yes

Description Data cleaning scripts typically contain a lot of 'if this change that' type of statements. Such statements are typically condensed expert knowledge. With this package, such 'data modifying rules' are taken out of the code and become in stead parameters to the work flow. This allows one to maintain, document, and reason about data modification rules as separate entities.

Version 0.1.9

Depends methods

URL <https://github.com/data-cleaning/dcmmodify>

BugReports <https://github.com/data-cleaning/dcmmodify/issues>

Imports yaml, validate (>= 0.1.5), settings, utils

Suggests tinytest, knitr, lumberjack, rmarkdown

VignetteBuilder knitr

Collate 'dplyr_verbs.R' 'guard.R' 'modifier.R' 'modify.R' 'validate.R'

RoxygenNote 7.1.1

NeedsCompilation no

Author Mark van der Loo [cre, aut],
Edwin de Jonge [aut]

Repository CRAN

Date/Publication 2021-09-24 09:40:02 UTC

R topics documented:

dcmmodify	2
modifier	2
modify	3
modify_so	3

Index**4**

dcmodify	<i>Data Modification By Modifying Rules</i>
----------	---

Description

Data often contain errors and missing data. Experts can often correct commonly occurring errors based on simple conditional rules. This package facilitates the expression, management, and application of such rules on data sets.

The general workflow in dcmodify follows the following pattern.

- Define or read a set of rules with `modifier`.
- `modify` data with the modification rules.
- Examine the results either graphically or by summary.

There are several convenience functions that allow one to define modification rules from the commandline, through a (freeform or yaml) file and to investigate and maintain the rules themselves. Please have a look at the introductory vignette

```
vignette("introduction",package="dcmodify")
```

modifier	<i>Create or read a set of data modification rules</i>
----------	--

Description

Create or read a set of data modification rules

Usage

```
modifier(..., .file)
```

Arguments

...	A comma-separated list of modification rules.
.file	A character vector of file locations.

Value

An object of class `modifier`.

Examples

```
m <- modifier( if (height < mean(height)) height <- 2*height
, if ( weight > mean(weight) ) weight <- weight/2 )
modify(women,m)
```

modify	<i>Modify a data set</i>
--------	--------------------------

Description

Modify a data set

Usage

```
modify(dat, x, ...)  
  
## S4 method for signature 'data.frame,modifier'  
modify(dat, x, ...)
```

Arguments

dat	An R object carrying data
x	An R object carrying modifying rules
...	Options.

Examples

```
m <- modifier( if (height < mean(height)) height <- 2*height  
, if ( weight > mean(weight) ) weight <- weight/2 )  
modify(women,m)
```

modify_so	<i>Shortcut to modify data</i>
-----------	--------------------------------

Description

Shortcut to modify data

Usage

```
modify_so(dat, ...)
```

Arguments

dat	A data.frame
...	A comma-separated list of modifying rules.

Index

dcmodify, 2
dcmodify-package (dcmodify), 2

modifier, 2, 2
modify, 2, 3
modify, data.frame, modifier-method
 (modify), 3
modify_so, 3