Package 'trimmer'

December 18, 2019

Title Trim an Object

Version 0.8.1

Description A lightweight toolkit to reduce the size of a list object. The object is minimized by recursively removing elements from the object one-by-one. The process is constrained by a reference function call specified by the user, where the target object is given as an argument. The procedure will not allow elements to be removed from the object, that will cause results from the function call to diverge from the function call with the original object.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Suggests testthat (>= 2.1.0), knitr, rmarkdown

Imports data.table, crayon, cli, pryr

RoxygenNote 6.1.1

VignetteBuilder knitr

NeedsCompilation no

Author Lars Kjeldgaard [aut, cre]

Maintainer Lars Kjeldgaard <lars_kjeldgaard@hotmail.com>

Repository CRAN

Date/Publication 2019-12-18 22:30:10 UTC

R topics documented:

adjust_candidates *Adjust Data Table with Candidate Elements for Elimination*

Description

Adjusts positions of all candidates for elimination in data.table after removing a candidate (due to the fact, that the positions may shift).

Usage

```
adjust_candidates(cand, cand_top_idx)
```

Arguments

cand	data.table with candidates for elimination given by their position indices.
cand_top_idx	numeric position index of candidate to be removed.

Value

data.table candidates after any adjustments to position indices of candidates.

convert_idx_to_name *Convert Numbered Index to Named Index of List Element*

Description

Convert Numbered Index to Named Index of List Element

Usage

```
convert_idx_to_name(vec, obj)
```

Arguments

vec	numeric numeric index of list element.
obj	list

Value

character named index of list element.

Examples

```
d <- list(a = list(b = list(c = 3, d = 5), e = c(2,4)))
num_idx <- c(1,1,2)
convert_idx_to_name(num_idx, d)
```

fix_undefined_global_vars

Fix til at undgå R CMD check notes for "no visible binding for global variable"

Description

Dette script gør det muligt at referere til kolonner i data frames ved hjælp af Non Standard Evaluation (NSE) i databehandlingspakker som data.table og dplyr, uden at dette medfører R CMD check notes angående "no visible binding for global variable". Navnene på de variable, der refereres til ved hjælp af NSE, skal blot angives i en vektor til funktionen globalVariables() nedenfor.

Usage

```
fix_undefined_global_vars()
```

Details

Dette er den anbefalede løsning fra CRAN.

get_results_for_object

Compute Results From Function Call with Object as Argument

Description

Compute Results From Function Call with Object as Argument

Usage

```
get_results_for_object(obj, obj_arg_name, fun, ...,
  tolerate_warnings = TRUE)
```

Arguments

obj	list R object to be trimmed. _MUST_ inherit from the 'list' class.
obj_arg_name	character what is the name of the parameter, that 'obj' must be set to, when invoking 'fun'. Defaults to NULL, in which case the function assumes, that the 'obj' matches the first parameter of 'fun'.
fun	function function that must return the same results, when invoked with 'obj' both before and after trimming.
...	other (named) arguments for 'fun'.
tolerate_warnings	logical tolerate warnings (=TRUE) Or not (=FALSE) from function call results?

Value

results from function call.

`pf_obj_size` *Convert Size in Bytes to Print Friendly String*

Description

Convert Size in Bytes to Print Friendly String

Usage

```
pf_obj_size(x, digits = 2)
```

Arguments

`x`	`numeric` object size in digits.
`digits`	`numeric` number of digits you want.

Value

`character` priend friendly string.

Examples

```
pf_obj_size(10)
pf_obj_size(1010)
pf_obj_size(2e06)
```

`trim` *Trim an R Object*

Description

Trims an R object whilst presuming the results of a given function call, where the R object is given as an argument. One popular example could be trimming an R model object whilst presuming the results of the `predict` function on a sample of data.

Usage

```
trim(obj, obj_arg_name = NULL, fun = predict, size_target = 0,
  tolerate_warnings = FALSE, verbose = TRUE, dont_touch = list(),
  ...)
```

Arguments

obj	list R object to be trimmed. _MUST_ inherit from the 'list' class.
obj_arg_name	character what is the name of the parameter, that 'obj' must be set to, when invoking 'fun'. Defaults to NULL, in which case the function assumes, that the 'obj' matches the first parameter of 'fun'.
fun	function function that must return the same results, when invoked with 'obj' both before and after trimming.
size_target	numeric desired maximum size in _MegaBytes_ of object after trimming has been conducted. When this size is achieved, the trimming stops. Defaults to 0, in which case trimming continues, until no further trimming can be done without breaking results from 'fun'.
tolerate_warnings	logical tolerate warnings (=TRUE) Or not (=FALSE) from function call results?
verbose	logical print messages?
dont_touch	list list with name indices of elements, that must not be removed from object by trimming procedure.
...	other (named) arguments for 'fun'.

Examples

```
# get training data for predictive model.
trn <- datasets::mtcars

# estimate model.
mdl <- lm(mpg ~ ., data = trn)
trim(obj = mdl, obj_arg_name = "object", fun = predict, newdata = trn)
trim(obj = mdl, obj_arg_name = "object", fun = predict, newdata = trn,
dont_touch = list(c("model"), c("qr","tol")))
```

Index