Package ‘foodwebWrapper’

December 20, 2023

Version 1.1.0

Date 2023-12-20

Title Enhanced Wrapper to Show Which Functions Call What

Maintainer Barry Zeeberg <barryz2013@gmail.com>

Depends R (>= 4.2.0)

LazyData true

Imports mvbutils, utils, tibble, dplyr, R2HTML, textshaping, magrittr, tidyverse, stringr

Description Enhances the functionality of the mvbutils::foodweb() program. The matrix-format output of the original program contains identical row names and column names, each name representing a retrieved function. This format is enhanced by using the find_funs() program [see Sebastian (2017) <https://sebastiansauer.github.io/finds_funs/>] to concatenate the package name to the function name. Each package is assigned a unique color, that is used to color code the text naming the packages and the functions. This color coding is extended to the entries of value ``1" within the matrix, indicating the pattern of ancestor and descendent functions.

License GPL (>= 2)

Encoding UTF-8

VignetteBuilder knitr

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

RoxygenNote 7.2.3

Config/testthat/edition 3

NeedsCompilation no

Author Barry Zeeberg [aut, cre]

Repository CRAN

Date/Publication 2023-12-20 15:20:02 UTC

R topics documented:

`addStyle` *addStyle*

Description

insert tags into HTML code to implement rotating table text

Usage

```
addStyle(x, m, colorMap, pawn)
```

Arguments

`x`	character vector containing HTML code
`m`	character matrix containing table that is represented in x
`colorMap`	character array of colors
`pawn`	Boolean if TRUE use chess symbols rather than rectangles in html table

Value

returns modified HTML code

Examples

```
if(interactive()){
load("data/x_x2.RData")
load("data/x_m5.RData")
load("data/x_colorMap.RData")
y<-addStyle(x_x2,x_m5,x_colorMap,pawn=TRUE)
}
```

`attachedFunctions` *attachedFunctions*

Description

print a list of attached packages and their functions for the user to select from

Usage

```
attachedFunctions(verbose)
```

Arguments

`verbose`	Boolean if TRUE output several user messages

Value

returns a list whose components are

- l list of user-selected packages and corresponding functions
- where character vector of selected packages

Examples

```
if(interactive()){
attachedFunctions(verbose=TRUE)
}
```

`attachedFunctionsBatch` *attachedFunctionsBatch*

Description

same as attachedFunctions() but no user interaction needed

Usage

```
attachedFunctionsBatch(packs)
```

Arguments

`packs`	list of character strings containing the names of packages package name is like "pack", not like "package:pack"

Value

returns a list whose components are

- l list of user-selected packages and corresponding functions
- where character vector of selected packages

Examples

```
if(interactive()){
attachedFunctionsBatch(c("SherlockHolmes","textBoxPlacement"))
}
```

`attachedPackages` *attachedPackages*

Description

print a list of attached packages for the user to select from

Usage

```
attachedPackages()
```

Value

returns a character vector of selected packages

Examples

```
if(interactive()){
attachedPackages()
}
```

`colorTag` *colorTag*

Description

add tag to color function name in column 2, based on package in column 1

Usage

```
colorTag(v1, v2, nc, x, colorMap, pawn)
```

Arguments

`v1`	character vector first column of m (excluding first 2 entries of m)
`v2`	character vector second column of m (excluding first 2 entries of m)
`nc`	integer number of columns of m
`x`	return value of replaceRotTag()
`colorMap`	character array of colors
`pawn`	Boolean if TRUE use chess symbols rather than rectangles in html table

Details

v1 is first column of m (excluding first 2 entries of m) v2 is second column of m (excluding first 2 entries of m)

Value

returns

concatPackFunc2 *concatPackFunc2*

Description

match the package names with the function names

Usage

```
concatPackFunc2(m, v)
```

Arguments

`m` character matrix return value component $funmat of foodweb()

`v` character vector of package names returned by find_funz()

Value

returns augmented character matrix m

Examples

```
if(interactive()){
load("data/x_x.RData")
load("data/x_v.RData")
m<-concatPackFunc2(x_x$funmat,x_v)
}
```

consolidate *consolidate*

Description

create a permutation list of package names for re-ordering rows and columns of matrix m, in decreasing order of function counts per package

Usage

```
consolidate(v)
```

Arguments

`v` character vector of package names component of return value of find_funz()

Value

returns a list whose components are character vector for permuting order of m

Examples

```
if(interactive()){
load("data/x_v.RData")
l<-consolidate(x_v)
}
```

find_funz *find_funz*

Description

determine in which R package a function ‘resides’

Usage

```
find_funz(packs, rfuns)
```

Arguments

packs	list of character strings containing the names of the packages
rfuns	list of character strings containing the names of functions in packs to which the result is to be restricted

Value

returns vector of character strings, names are functions and values are packages

Examples

```
if(interactive()){
load("data/x_packages.RData")
load("data/x_funs.RData")
find_funz(packs=x_packages,rfuns=x_funs)
}
```

foodwebWrapper *foodwebWrapper*

Description

wrapper for the function foodweb() concatenate the R package name for each retrieved R function

Usage

```
foodwebWrapper(
  where = character(0),
  ofile = "~/foodwebWrapper.html",
  zeros = TRUE,
  pawn = FALSE,
  verbose = TRUE
)
```

Arguments

`where`	position(s) on search path, or an environment, or a list of environments
`ofile`	character string containing path name for output file
`zeros`	Boolean if TRUE delete rows and cols that contain all 0's
`pawn`	Boolean if TRUE use chess symbols rather than rectangles in html table
`verbose`	Boolean if TRUE output several user messages

Details

if where is missing, then the user is presented with the option of choosing from a list of attached packages

Value

foodweb returns an object of (S3) class foodweb. This has three components:

- funmat a matrix of 0s and 1s showing what (row) calls what (column). The dimnames are the function names.
- x shows the x-axis location of the centre of each function's name in the display, in par("usr") units
- level shows the y-axis location of the centre of each function's name in the display, in par("usr") units. For small numbers of functions, this will be an integer; for larger numbers, there will some adjustment around the nearest integer

Examples

```
if(interactive()){
load("data/x_packages.RData")
ofile<-sprintf("%s/foodwebWrapper.html",tempdir())
foodwebWrapper(ofile=ofile)
foodwebWrapper(where=x_packages,ofile=ofile)
}
```

`funs_examples` *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(funs_examples)
```

`mapFunctionsColors` *mapFunctionsColors*

Description

map functions to color coding

Usage

```
mapFunctionsColors(row1, col1, colors)
```

Arguments

`row1`	character vector containing names of packages
`col1`	character vector containing names of packages
`colors`	character vector containing names of colors

Value

returns a character vector mapping colors to package names

Examples

```
if(interactive()){
colors<-c("darkmagenta","darkolivegreen","darkorange3","brown4","red","blue")
load("data/x_m3.RData")
colorMap<-mapFunctionsColors(x_m3[1,c(-1,-2)],x_m3[c(-1,-2),1],colors)
}
```

rearrangeM *rearrangeM*

Description

rearrange the order of rows or columns of matrix based on entries in a vector

Usage

```
rearrangeM(m, v2)
```

Arguments

m	character matrix return value of concatPackFunc2()
v2	list whose components are package names for permuting order of m, return value of consolidate()

Value

returns rearranged version of m

Examples

```
if(interactive()){
load("data/x_m.RData")
load("data/x_v2.RData")
m2<-rearrangeM(x_m,x_v2)
}
```

removeZeroRowsCols *removeZeroRowsCols*

Description

delete rows and cols of matrix m that contain all "0"s

Usage

```
removeZeroRowsCols(m)
```

Arguments

m	character matrix whose entries are either "0" or "1"

Value

returns an altered version of character matrix m with removed rows and columns

Examples

```
if(interactive()){
load("data/x_m2.RData")
m3<-removeZeroRowsCols(x_m2)
}
```

replaceRotTag *replaceRotTag*

Description

insert html tags for rotating text

Usage

```
replaceRotTag(x, l, dims)
```

Arguments

`x`	return value of readLines(), HTML code containing data table
`l`	return values of spanTag()
`dims`	return value of dim()

Value

returns modified version of HTML code containing data table

Examples

```
if(interactive()){
load("data/x_x.RData")
load("data/x_l.RData")
load("data/x_m3.RData")
x<-replaceRotTag(x_x,x_l,dim(x_m3))
}
```

rotStyle *rotStyle*

Description

add html style definition for rotation

Usage

```
rotStyle()
```

Value

returns character string containing html style definition for rotation

Examples

```
r<-rotStyle()
```

rotTag *rotTag*

Description

add html tag to rotate function name

Usage

```
rotTag(v1, v2, colorMap)
```

Arguments

`v1`	character vector containing first row of matrix m (excluding first 2 entries of m)
`v2`	character vector containing second row of matrix m (excluding first 2 entries of m)
`colorMap`	character array of colors

Details

see https://stackoverflow.com/questions/47261100/how-to-rotate-text-90-degrees-inline also need to increase height of row to accommodate rotated text see https://resultuniversity.com/html/html-table-width-height#:~:text=To%20set%20the%20height%20of%20a%20specific%20row%20in%20an,property%20in%20pixels%

Value

returns character vector containing inserted html tags

Examples

```
if(interactive()){
load("data/x_m5.RData")
load("data/x_colorMap.RData")
rt<-rotTag(x_m5[1,c(-1,-2)],x_m5[2,c(-1,-2)],x_colorMap)
}
```

`spanTag` *spanTag*

Description

Add html tag for package name to span multiple columns. Also insert hyperlink to CRAN package and function documentation.

Usage

```
spanTag(v, direction = "COLSPAN", colorMap)
```

Arguments

`v`	character vector representing first row of m (excluding first 2 entries of m)
`direction`	character string COLSPAN or ROWSPAN
`colorMap`	character array of colors

Details

see https://www.pierobon.org/html/span.htm#:~:text=Cells%20within%20HTML%20tables%20can,span%20more%20than

Value

returns a list whose components are

- u return value of unique(v)
- tab return value of table(v)
- v2 character vector modified version of v containing html span tags

Examples

```
if(interactive()){
load("data/x_m5.RData")
load("data/x_colorMap.RData")
l<-spanTag(x_m5[1,c(-1,-2)],"COLSPAN",x_colorMap)
}
```

x_colorMap *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_colorMap)
```

x_examples *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_examples)
```

x_f *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_f)
```

x_funs *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_funs)
```

`x_m` *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_m)
```

`x_m2` *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_m2)
```

`x_m3` *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_m3)
```

`x_m4` *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_m4)
```

x_m5 *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_m5)
```

x_packages *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_packages)
```

x_v *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_v)
```

x_v2 *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_v2)
```

`x_where` *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_where)
```

`x_x` *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_x)
```

`x_x2` *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_x2)
```

`x_x3` *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_x3)
```

x_y *foodwebWrapper data sets*

Description

foodwebWrapper data sets

Usage

```
data(x_y)
```

Index