Package ‘mtb’

February 5, 2022

Type Package

Title Toolkit for Assisting Kitchen and Garden Projects

Version 0.1.6

Description The purpose of this package is to share a collection of functions the author wrote during weekends for managing kitchen and garden tasks, e.g. making plant growth charts or Thanksgiving kitchen schedule charts, etc.

Functions might include but not limited to:
1. aiding summarizing time related data;
2. generating axis transformation from data; and
3. aiding Markdown (with html output) and Shiny file editing.

License AGPL (>= 3)

Encoding UTF-8

Depends R (>= 3.6)

Imports htmltools (>= 0.4.0), ggplot2 (>= 3.3.0), scales (>= 1.0.0), labeling (>= 0.3)

Suggests rmarkdown (>= 1.16), knitr, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

URL https://github.com/yh202109/mtb

RooxygenNote 7.1.2

Language en-US

NeedsCompilation no

Author Y Hsu [aut, cre]

Maintainer Y Hsu <yh202109@gmail.com>

Repository CRAN

Date/Publication 2022-02-05 00:50:05 UTC
R topics documented:

add_colored_box ........................................... 2
add_colored_str ........................................... 3
color_set_palette .......................................... 4
color_test_palette ......................................... 5
time_plot_event ............................................. 6
time_plot_interval ......................................... 7
trans_composition ......................................... 8
trans_loglinear ............................................ 9

Index 11

---

add_colored_box  Add a box with specified color in an R Markdown file.

Description

add_colored_box returns a box component generated by htmltools with specified color and styles.

Usage

add_colored_box(
  type = "blue-default",
  label = "",
  info = "place details here using info option",
  bgcolor = NULL,
  width = 0.5,
  halign = "c",
  top = FALSE
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>One of:</td>
</tr>
<tr>
<td>• NULL for no default color or label</td>
<td></td>
</tr>
<tr>
<td>• 'blue-default' for a steel-blue box</td>
<td></td>
</tr>
<tr>
<td>• 'gray-info' for a gray box</td>
<td></td>
</tr>
<tr>
<td>• 'blue-info' for a blue box</td>
<td></td>
</tr>
<tr>
<td>• 'green-remainder' for a green box</td>
<td></td>
</tr>
<tr>
<td>• 'yellow-warning' for a yellow box</td>
<td></td>
</tr>
<tr>
<td>• 'red-stop' for a red box</td>
<td></td>
</tr>
</tbody>
</table>

label | One of: |
• NULL for no label if type is NULL or using label set by type |
• A string shown on the top of box |
add_colored_str

- info: A string including the main message of the box.
- bgcolor: NA or a length 3 vector with integer elements between 0 to 255.
- width: NA or a number between 0.25 to 0.95.
- halign: One of:
  - NA for center aligned
  - 'c' for center aligned
  - 'r' for right aligned
- top: One of:
  - NA
  - FALSE for inline
  - TRUE for top-of-page

Examples

```r
add_colored_box( type='blue-default', info='the document include information regarding...'
)
```

---

add_colored_str

Add a string with specified color or background color.

Description

add_colored_str returns a string component generated by htmltools with specified color and styles.

Usage

```r
add_colored_str(
  text = "",  
  color = c(51, 122, 183), 
  alpha = 255, 
  bgcolor = NULL, 
  bgalpha = 51, 
  fontsize = 1, 
  bold = FALSE, 
  it = FALSE 
)
```

Arguments

- text: A string. default="".
- color: One of
  - a color name, e.g. 'red'.
  - a HEX color string, e.g. '#000000' or '#000000FF'.
• an RGB vector for the color of text

alpha
An integer between 1 and 255 for text alpha. default=255.

bgcolor
One of
• a color name, e.g. 'red'.
• a HEX color string, e.g. '#000000' or '#000000FF'.
• an RGB vector for the color of text

bgalpha
An integer between 1 and 255 for background alpha. default=51.

fontsize
A real number between 0.5 and 5.0 for font size. default=1.

bold
A logical value for bold fonts. default=FALSE.

it
A Boolean value for italic fonts. default=FALSE.

Value
a formatted string

Examples

add_colored_str("warning: read this message carefully.", color = c(255, 0, 0))

color_set_palette Generate a color vector

description
Create a list of colors for a data vector by a list major colors.

Usage
color_set_palette(
  vect = c(),
  vectn = c(),
  cols = c("blue", "cyan", "darkorange"),
  black = "",
  gray9 = ""
)

Arguments

vect
A vector for groups.

vectn
An integer vector with length 0 or with the same length of vect for order of elements in vect. default=c()

cols
One of
• A color names vector
• An RGB triplet vector
• A HEX vector

black A level in vect that should be assigned to black color. default=""
gray9 A level in vect that should be assigned to gray9 color. default=""

Value
a named vector

Examples
color_set_palette( c('apple', 'orange', 'lime', 'apple'), c(2,1,3,2), 'red', 'blue')

color_test_palette( setNames(c(1,2,3,4), c('apple','orange','avocado','lime')))
Description

Create a plot for events with labels

Usage

getTime_plot_event(dt, xlab = "Time", anchor = TRUE, compact = FALSE)

Arguments

dt a data.frame with the following columns
- id for ID of each group
- idn for order of ID
- start for starting time
- end for ending time with arrow head
- label for labeling the starting time
- labelend for labeling the ending time of an interval
- type for event type as one of p (point), i (interval), b (box)
- color a string for event color

xlab A string for the x-axis title

anchor A Boolean value for the vertical lines linking start to the x-axis

compact A Boolean value for reducing the vertical spacing when applicable

Value

a plot

Examples

library(ggplot2)
dt = data.frame( id=paste("member", c(rep(c(1,2,3),each=3),3),sep=""),
idn=c(rep(1,3),rep(-1,3),rep(2,4)),
start=1800*c(0,1,2, 0.5, 1.2, 3, 1,2,3,4),
end=1800*c(2,NA,3, 2, 6, NA, 2.5,3, 3.5),
label=c(paste("event-",seq(1,10),sep="")),
labelend=c("","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","","

xlab = "Time", anchor = TRUE, compact = FALSE)

getTime_plot_event( dt )
time_plot_interval

Plot periods of events

Description
Create a plot for event periods by ID

Usage
time_plot_interval(
  dt,
  xlab = "DateTime",
  ylab = "ID",
  legend_title = "Group",
  arrow_wt = 1,
  arrow_color = "black"
)

Arguments
dt a data.frame with the following columns
• id for ID of each interval
• idn for order of ID
• start for starting time
• end for ending time with arrow head
• label for labeling the starting time
xlab A string for the label of X-axis
ylab A string for the label of Y-axis
legend_title A string for the title of legend
arrow_wt An integer for the weight of arrow
arrow_color A string for the color of arrow

Value
a plot

Examples
library(ggplot2)
dt = data.frame( id=c('ID01','ID12','ID3'), idn=c(1,3,2), start=1800*c(0,1,2), end=1800*c(2,-1,3), label=c('A','B','C'))
time_plot_interval( dt, xlab='Time', ylab='ID', legend_title='Group', arrow_wt=3, arrow_color='gray')
trans_composition

Transformation for continuous data with a finite number of distinct values

Description

trans_composition() derives a transformation from a numerical vector with a smaller number (ideally < 30) of distinct values. The return can be used with function ggplot::scale_x_continuous() or ggplot::scale_y_continuous() to create a desired axis.

Usage

trans_composition(x = NULL, nb = 30, brk = NA, dab = NA, dgrd = NA, dgrd2 = NA)

Arguments

x A numerical vector used in a plot as (typically) x
nb An integer for the maximum number of breaks. Default=30
brk One of
  • A numerical value within range(x). All values after the value will be spaced equally
  • NA or a numerical value that is greater than or equal to max(x). All values will be plotted in the original scale
  • A numerical value that is smaller than or equal to max(x). All values will be plotted in equal space

dab One of
  • NA for a value calculated automatically
  • A number for the distance after brk

dgrd One of
  • NA for a value calculated automatically
  • A number for the minimum space between major grids

dgrd2 One of
  • NA for a value calculated automatically
  • A number for the minimum space between major grids

Value

A transformation function
Examples

```r
library(ggplot2)
pdt=data.frame(x=rep(c(0.5, 1, 10, 11, 12, 100, 1000), each=5))
pdt$y=pdt$x+rnorm(length(pdt$x))
t=trans_composition(pdt$x, brk=50, dab=3)
ggplot(pdt, aes(x=x, y=y))+geom_point()+scale_x_continuous(trans=t)
```

---

trans_loglinear Transformation for continuous data with a finite number of distinct values

Description

trans_loglinear() derives a log transformation from a numerical vector with a smaller number (ideally < 30) of distinct values. The return can be used with function `ggplot::scale_x_continuous()` or `ggplot::scale_y_continuous()` to create a desired axis.

Usage

```r
trans_loglinear(x = NULL, nb = 30, int = NA, scale = NA, mindist = 0.03)
```

Arguments

- `x` A numerical vector used in a plot as (typically) `x`
- `nb` An integer for the maximum number of breaks. Default=30
- `int` One of
  - NA for a value calculated automatically
  - A real number (>0) for the shift before log transform
- `scale` One of
  - NA for a value calculated automatically
  - A real number (>0) for the scale before log transform
- `mindist` One of
  - NA for a default value set to 0.03
  - A real number between 0 and 0.2 for the minimum distance ratio between major ticks

Value

A transformation function
Examples

```r
library(ggplot2)
pdt=data.frame(x=rep(c(0.5, 1, 10, 11, 12, 100, 1000), each=5))
pdt$y=pdt$x+rnorm(length(pdt$x))
t=trans_loglinear(pdt$x)
ggplot(pdt, aes(x=x, y=y))+geom_point()+scale_x_continuous(trans=t)
```
Index

add_colored_box, 2
add_colored_str, 3

color_set_palette, 4
color_test_palette, 5

time_plot_event, 6
time_plot_interval, 7
trans_composition, 8
trans_loglinear, 9