Package 'colourvalues'

April 11, 2023

Type Package

Title Assigns Colours to Values

Version 0.3.9

Date 2023-04-09

Description Maps one of the viridis colour palettes, or a user-specified palette to values. Viridis colour maps are created by Stéfan van der Walt and Nathaniel Smith, and were set as the default palette for the 'Python' 'Matplotlib' library <https://matplotlib.org/>.
Other palettes available in this library have been derived from 'RColorBrewer' <https://CRAN.R-project.org/package=RColorBrewer> and 'colorspace' <https://CRAN.R-project.org/package=colorspace> packages.

License GPL-3

URL https://symbolixau.github.io/colourvalues/

BugReports https://github.com/SymbolixAU/colourvalues/issues

Encoding UTF-8

Depends R (>= 3.3.0)

SystemRequirements C++17

LinkingTo BH (>= 1.81.0), Rcpp (>= 1.0.10)

Imports graphics, Rcpp (>= 1.0.10)

RoxygenNote 7.2.3

Suggests covr, microbenchmark, scales, testthat, viridisLite

NeedsCompilation yes

Author David Cooley [aut, cre]

Maintainer David Cooley <dcooley@symbolix.com.au>

Repository CRAN

Date/Publication 2023-04-11 01:20:13 UTC

R topics documented:

`blue2green` *Blue2green*

Description

Data Frame of the blue2green palette

Usage

```
blue2green()
```

`blue2red` *Blue2red*

Description

Data Frame of the blue2red palette

Usage

```
blue2red()
```

`blue2yellow` *Blue2yellow*

Description

Data Frame of the blue2yellow palette

Usage

```
blue2yellow()
```

blues *Blues*

Description

Data Frame of the blues palette

Usage

```
blues()
```

brbg *Brbg*

Description

Data Frame of the brbg palette

Usage

```
brbg()
```

bugn *Bugn*

Description

Data Frame of the bugn palette

Usage

```
bugn()
```

bupu *Bupu*

Description

Data Frame of the bupu palette

Usage

```
bupu()
```

cividis *Cividis*

Description

Data frame of the cividis palette

Usage

```
cividis()
```

cm *Cm*

Description

Data Frame of the cm palette

Usage

```
cm()
```

colour_palettes *Colour Palettes*

Description

List the available colour palettes.

Usage

```
colour_palettes(colours = NULL)

color_palettes(colours = NULL)
```

Arguments

colours	vector of source colour palettes to return, one or many of "viridis","rcolorbrewer","grdevices","colorspace NULL will reutrn all palettes.

Details

The palettes avaialble in `colourvalues` have been derived from those avaialble in the libraries

- viridis
- RColorBrewer
- grDevices
- colorspaces
- colorRamp

Examples

```
colour_palettes()
colour_palettes( "viridis" )
colour_palettes( colours = c("rcolorbrewer","grdevices") )
```

`colour_values` *Colour Values*

Description

maps colours to values, returning a vector of hex strings

Usage

```
colour_values(
  x,
  palette = "viridis",
  alpha = 255,
  na_colour = "#808080FF",
  include_alpha = TRUE,
  summary = FALSE,
  n_summaries = 0,
  format = TRUE,
  digits = 2
)

color_values(
  x,
  palette = "viridis",
  alpha = 255,
  na_colour = "#808080FF",
  include_alpha = TRUE,
  summary = FALSE,
  n_summaries = 0,
  format = TRUE,
```

```
  digits = 2
)

## S3 method for class 'character'
colour_values_to_hex(
  x,
  palette,
  alpha,
  na_colour,
  include_alpha,
  summary,
  n_summaries,
  format,
  digits
)

## S3 method for class 'logical'
colour_values_to_hex(
  x,
  palette,
  alpha,
  na_colour,
  include_alpha,
  summary,
  n_summaries,
  format,
  digits
)

## S3 method for class 'factor'
colour_values_to_hex(
  x,
  palette,
  alpha,
  na_colour,
  include_alpha,
  summary,
  n_summaries,
  format,
  digits
)

## S3 method for class 'Date'
colour_values_to_hex(
  x,
  palette,
  alpha,
  na_colour,
```

```
  include_alpha,
  summary,
  n_summaries,
  format,
  digits
)

## S3 method for class 'POSIXct'
colour_values_to_hex(
  x,
  palette,
  alpha,
  na_colour,
  include_alpha,
  summary,
  n_summaries,
  format,
  digits
)

## S3 method for class 'POSIXlt'
colour_values_to_hex(
  x,
  palette,
  alpha,
  na_colour,
  include_alpha,
  summary,
  n_summaries,
  format,
  digits
)
```

Arguments

x	vector of values to map to a colour
palette	colour palette. See details and examples
alpha	optional. Single value in [0,255] applied to all colours, or a decimal in [0, 1) (to indicate a percentage, noting 1 is excluded), or a vector of numeric values the same length as x. The numeric vector will be scaled into the range [0,255]. If a matrix palette is supplied this argument is ignored.
na_colour	hex string colour to use for NA values in the form #RRGGBBAA.
include_alpha	logical indicating if the returned hex or matrix should include the alpha values. Defaults to TRUE.
summary	logical indicating if a summary of the colours should be returned as well as the full colour mapping. This will be the unique elements of x mapped to the colour.

n_summaries	positive integer. If supplied a summary colour palette will be returned in a list, containing n_summaries equally spaced values of x in the range [min(x),max(x)], and their associated colours. If a non-numeric x is used this value is ignored
format	logical indicating if the summary values should be formatted.
digits	number of decimal places to show in the summary

Details

The palette can either be

- String - use colour_palettes() to view available palettes
- Matrix - At least 5 rows, and 3 (or 4) columns representing the red, green and blue (and alpha) values

The matrix palette requires 5 rows because the colours are interpolated using a cubic b-spline. This method requires 5 values.

See Also

colour_values_rgb

Examples

```
## in-built palettes
colour_values(x = 1:5) ## default is "viridis"
colour_values(x = 1:5, palette = "inferno")
colour_values(x = 1:5, palette = "plasma")
colour_values(x = 1:5, palette = "magma")
colour_values(x = 1:5, palette = "cividis")
colour_values(x = 1:5, palette = "rainbow")

## matrix palette
n <- 100
m <- grDevices::colorRamp(c("red", "green"))( (1:n)/n )
df <- data.frame(a = 10, x = 1:n)
df$col <- colour_values(df$x, palette = m)
barplot(height = df$a, col = df$col, border = NA, space = 0)

## with an alpha column on the palette
n <- 100
m <- grDevices::colorRamp(c("red", "green"))( (1:n)/n )
m <- cbind(m, seq(0, 255, length.out = 100))
df <- data.frame(a = 10, x = 1:n)
df$col <- colour_values(df$x, palette = m)
barplot(height = df$a, col = df$col, border = NA, space = 0)

## single alpha value for all colours
df <- data.frame(a = 10, x = 1:255)
df$col <- colour_values(df$x, alpha = 50)
barplot(height = df$a, col = df$col, border = NA, space = 0)
```

```
## vector of alpha values
df <- data.frame(a = 10, x = 1:300, y = rep(c(1:50, 50:1), 3) )
df$col <- colour_values(df$x, alpha = df$y)
barplot(height = df$a, col = df$col, border = NA, space = 0)

## returning a summary palette
colour_values(-10:10, n_summaries = 5)

colour_values(x = runif(20, 0, 1), n_summaries = 3, digits = 2)
colour_values(x = runif(20, 0, 1), n_summaries = 3, digits = 10)

## Formatting output
## default is TRUE
colour_values(
  x = seq(as.Date("2023-01-01"), as.Date("2023-01-31"), by = 1)
  , n_summaries = 5
)
colour_values(
  x = seq(as.Date("2023-01-01"), as.Date("2023-01-31"), by = 1)
  , n_summaries = 5
  , format = FALSE
)
```

`colour_values_rgb` *Colour Values RGB*

Description

Maps colours to values, returning a matrix of RGB(A) values

Usage

```
colour_values_rgb(
  x,
  palette = "viridis",
  alpha = 255,
  na_colour = "#808080FF",
  include_alpha = TRUE,
  summary = FALSE,
  n_summaries = 0,
  format = TRUE,
  digits = 2
)

color_values_rgb(
  x,
  palette = "viridis",
```

```
  alpha = 255,
  na_colour = "#808080FF",
  include_alpha = TRUE,
  summary = FALSE,
  n_summaries = 0,
  format = TRUE,
  digits = 2
)

## S3 method for class 'character'
colour_values_to_rgb(
  x,
  palette,
  alpha,
  na_colour,
  include_alpha,
  summary,
  n_summaries,
  format,
  digits
)

## S3 method for class 'logical'
colour_values_to_rgb(
  x,
  palette,
  alpha,
  na_colour,
  include_alpha,
  summary,
  n_summaries,
  format,
  digits
)

## S3 method for class 'factor'
colour_values_to_rgb(
  x,
  palette,
  alpha,
  na_colour,
  include_alpha,
  summary,
  n_summaries,
  format,
  digits
)
```

```
## S3 method for class 'Date'
colour_values_to_rgb(
  x,
  palette,
  alpha,
  na_colour,
  include_alpha,
  summary,
  n_summaries,
  format,
  digits
)

## S3 method for class 'POSIXct'
colour_values_to_rgb(
  x,
  palette,
  alpha,
  na_colour,
  include_alpha,
  summary,
  n_summaries,
  format,
  digits
)

## S3 method for class 'POSIXlt'
colour_values_to_rgb(
  x,
  palette,
  alpha,
  na_colour,
  include_alpha,
  summary,
  n_summaries,
  format,
  digits
)
```

Arguments

x	vector of values to map to a colour
palette	colour palette. See details and examples
alpha	optional. Single value in [0,255] applied to all colours, or a decimal in [0, 1) (to indicate a percentage, noting 1 is excluded), or a vector of numeric values the same length as x. The numeric vector will be scaled into the range [0,255]. If a matrix palette is supplied this argument is ignored.
na_colour	hex string colour to use for NA values in the form #RRGGBBAA.

include_alpha	logical indicating if the returned hex or matrix should include the alpha values. Defaults to TRUE.
summary	logical indicating if a summary of the colours should be returned as well as the full colour mapping. This will be the unique elements of x mapped to the colour.
n_summaries	positive integer. If supplied a summary colour palette will be returned in a list, containing n_summaries equally spaced values of x in the range [min(x),max(x)], and their associated colours. If a non-numeric x is used this value is ignored
format	logical indicating if the summary values should be formatted.
digits	number of decimal places to show in the summary

Details

The palette can either be

- String - use colour_palettes() to view available palettes
- Matrix - At least 5 rows, and 3 (or 4) columns representing the red, green and blue (and alpha) values

The matrix palette requires 5 rows because the colours are interpolated using a cubic b-spline. This method requires 5 values.

See Also

colour_values

Examples

```
colour_values_rgb(1:5)
colour_values_rgb(1:5, include_alpha = FALSE)
colour_values_rgb(-25:25, n_summaries = 5)
```

---

convert_colour *Convert Colour*

Description

Converts colours between RRGGBBAA and hex strings, in both directions.

Usage

```
convert_colour(x)

convert_colours(x)

convert_color(x)

convert_colors(x)
```

Arguments

x	character vector of hex strings, or numeric matrix of RRGGBBAA values

Details

If a combination of hex strings with and without alpha values are supplied, those without are assumed to have an alpha value of FF and will be returned in the RRGGBBAA matrix

Examples

```
convert_colour(c("#FFAA00"))
convert_colour(c("#FFAA00","#FF00AAFF"))

convert_colour(matrix(c(255,170,0),ncol = 3))
convert_colour(matrix(c(255,170,0,255),ncol = 4))
```

---

cyan2yellow *Cyan2yellow*

Description

Data Frame of the cyan2yellow palette

Usage

```
cyan2yellow()
```

---

diverge_hcl *Diverge_hcl*

Description

Data Frame of the diverge_hcl palette

Usage

```
diverge_hcl()
```

diverge_hsv *Diverge_hsv*

Description

Data Frame of the diverge_hsv palette

Usage

```
diverge_hsv()
```

get_palette *Get Palette*

Description

retrieves one of the available palettes

Usage

```
get_palette(palette, rgb = TRUE)
```

Arguments

palette	one of the available paletes. See colour_palettes
rgb	logical indicating if the palette should be returned as an RGB matrix TRUE, or a vector of hex strings FALSE

Value

3 column matrix if rgb = TRUE, otherwise a 256-length vector.

Examples

```
get_palette( "viridis" )
get_palette( "rainbow" )
```

gnbu *Gnbu*

Description

Data Frame of the gnbu palette

Usage

```
gnbu()
```

green2red *Green2red*

Description

Data Frame of the green2red palette

Usage

```
green2red()
```

greens *Greens*

Description

Data Frame of the greens palette

Usage

```
greens()
```

greys *Greys*

Description

Data Frame of the greys palette

Usage

```
greys()
```

heat *Heat*

Description

Data Frame of the heat palette

Usage

```
heat()
```

heat_hcl *Heat_hcl*

Description

Data Frame of the heat_hcl palette

Usage

```
heat_hcl()
```

inferno *Inferno*

Description

Data frame of the inferno palette

Usage

```
inferno()
```

magenta2green *Magenta2green*

Description

Data Frame of the magenta2green palette

Usage

```
magenta2green()
```

magma *Magma*

Description

Data frame of the magma palette

Usage

```
magma()
```

matlab_like *Matlab_like*

Description

Data Frame of the matlab_like palette

Usage

```
matlab_like()
```

matlab_like2 *Matlab_like2*

Description

Data Frame of the matlab_like2 palette

Usage

```
matlab_like2()
```

oranges *Oranges*

Description

Data Frame of the oranges palette

Usage

```
oranges()
```

orrd *Orrd*

Description

Data Frame of the orrd palette

Usage

```
orrd()
```

piyg *Piyg*

Description

Data Frame of the piyg palette

Usage

```
piyg()
```

plasma *Plasma*

Description

Data frame of the plasma palette

Usage

```
plasma()
```

prgn *Prgn*

Description

Data Frame of the prgn palette

Usage

```
prgn()
```

pubu *Pubu*

Description

Data Frame of the pubu palette

Usage

```
pubu()
```

pubugn *Pubugn*

Description

Data Frame of the pubugn palette

Usage

```
pubugn()
```

puor *Puor*

Description

Data Frame of the puor palette

Usage

```
puor()
```

purd *Purd*

Description

Data Frame of the purd palette

Usage

```
purd()
```

purples *Purples*

Description

Data Frame of the purples palette

Usage

```
purples()
```

rainbow *Rainbow*

Description

Data Frame of the rainbow palette

Usage

```
rainbow()
```

rainbow_hcl *Rainbow_hcl*

Description

Data Frame of the rainbow_hcl palette

Usage

```
rainbow_hcl()
```

rdbu *Rdbu*

Description

Data Frame of the rdbu palette

Usage

```
rdbu()
```

rdgy *Rdgy*

Description

Data Frame of the rdgy palette

Usage

```
rdgy()
```

rdpu *Rdpu*

Description

Data Frame of the rdpu palette

Usage

```
rdpu()
```

rdylbu *Rdylbu*

Description

Data Frame of the rdylbu palette

Usage

```
rdylbu()
```

rdylgn *Rdylgn*

Description

Data Frame of the rdylgn palette

Usage

```
rdylgn()
```

reds *Reds*

Description

Data Frame of the reds palette

Usage

```
reds()
```

sequential_hcl *Sequential_hcl*

Description

Data Frame of the sequential_hcl palette

Usage

```
sequential_hcl()
```

show_colours *Show Colours*

Description

Plots all the selected colours. See colour_palettes for avaialble colours.

Usage

```
show_colours(colours = colour_palettes())
```

Arguments

colours	vector of colour palettes

Examples

```
## view all the colour palettes
show_colours()

## view a selection of colour palettes
show_colours( colours = colour_palettes( c("viridis", "grdevices") ) )
```

spectral *Spectral*

Description

Data Frame of the spectral palette

Usage

```
spectral()
```

terrain *Terrain*

Description

Data frame of the terrain palette

Usage

```
terrain()
```

terrain_hcl *Terrain_hcl*

Description

Data Frame of the terrain_hcl palette

Usage

```
terrain_hcl()
```

topo *Topo*

Description

Data Frame of the topo palette

Usage

```
topo()
```

viridis *Viridis*

Description

Data frame of the viridis palette

Usage

```
viridis()
```

ygobb *Ygobb*

Description

Data Frame of the ygobb palette

Usage

```
ygobb()
```

ylgn *Ylgn*

Description

Data Frame of the ylgn palette

Usage

```
ylgn()
```

ylgnbu *Ylgnbu*

Description

Data Frame of the ylgnbu palette

Usage

```
ylgnbu()
```

`ylorbr` *Ylorbr*

Description

Data Frame of the ylorbr palette

Usage

```
ylorbr()
```

`ylorrd` *Ylorrd*

Description

Data Frame of the ylorrd palette

Usage

```
ylorrd()
```

Index