

Package ‘colourvalues’

July 22, 2025

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

Contents

blue2green	3
blue2red	3
blue2yellow	3
blues	4
brbg	4
bugn	4
bupu	4
cividis	5
cm	5
colour_palettes	5
colour_values	6
colour_values_rgb	10
convert_colour	13
cyan2yellow	14
diverge_hcl	14
diverge_hsv	15
get_palette	15
gnbu	16
green2red	16
greens	16
greys	16
heat	17
heat_hcl	17
inferno	17
magenta2green	17
magma	18
matlab_like	18
matlab_like2	18
oranges	18
orrd	19
piyg	19
plasma	19
prgn	19
pubu	20
pubugn	20
puor	20
purd	20
purples	21
rainbow	21
rainbow_hcl	21
rdbu	21
rdgy	22
rdpu	22
rdylbu	22
rdylgn	22
reds	23

sequential_hcl	23
show_colours	23
spectral	24
terrain	24
terrain_hcl	24
topo	24
viridis	25
ygobb	25
ylgn	25
ylgnbu	25
ylorbr	26
ylorrd	26

Index

27

blue2green	<i>Blue2green</i>
------------	-------------------

Description

Data Frame of the blue2green palette

Usage

blue2green()

blue2red	<i>Blue2red</i>
----------	-----------------

Description

Data Frame of the blue2red palette

Usage

blue2red()

blue2yellow	<i>Blue2yellow</i>
-------------	--------------------

Description

Data Frame of the blue2yellow palette

Usage

blue2yellow()

blues	<i>Blues</i>
-------	--------------

Description

Data Frame of the blues palette

Usage

blues()

brbg	<i>Brbg</i>
------	-------------

Description

Data Frame of the brbg palette

Usage

brbg()

bugn	<i>Bugn</i>
------	-------------

Description

Data Frame of the bugn palette

Usage

bugn()

bupu	<i>Bupu</i>
------	-------------

Description

Data Frame of the bupu palette

Usage

bupu()

cividis	<i>Cividis</i>
---------	----------------

Description

Data frame of the cividis palette

Usage

cividis()

cm	<i>Cm</i>
----	-----------

Description

Data Frame of the cm palette

Usage

cm()

colour_palettes	<i>Colour Palettes</i>
-----------------	------------------------

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 available in colourvalues have been derived from those available in the libraries

- viridis
- RColorBrewer
- grDevices
- colorspace
- colorRamp

Examples

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

colour_values	<i>Colour Values</i>
---------------	----------------------

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

<code>x</code>	vector of values to map to a colour
<code>palette</code>	colour palette. See details and examples
<code>alpha</code>	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 <code>x</code> . The numeric vector will be scaled into the range [0,255]. If a matrix palette is supplied this argument is ignored.
<code>na_colour</code>	hex string colour to use for NA values in the form #RRGGBBAA.
<code>include_alpha</code>	logical indicating if the returned hex or matrix should include the alpha values. Defaults to TRUE.
<code>summary</code>	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 <code>x</code> 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	<i>Colour Values RGB</i>
-------------------	--------------------------

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

<code>x</code>	vector of values to map to a colour
<code>palette</code>	colour palette. See details and examples
<code>alpha</code>	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 <code>x</code> . The numeric vector will be scaled into the range [0,255]. If a matrix palette is supplied this argument is ignored.
<code>na_colour</code>	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	<i>Cyan2yellow</i>
-------------	--------------------

Description

Data Frame of the cyan2yellow palette

Usage

```
cyan2yellow()
```

diverge_hcl	<i>Diverge_hcl</i>
-------------	--------------------

Description

Data Frame of the diverge_hcl palette

Usage

```
diverge_hcl()
```

diverge_hsv	<i>Diverge_hsv</i>
-------------	--------------------

Description

Data Frame of the diverge_hsv palette

Usage

```
diverge_hsv()
```

get_palette	<i>Get Palette</i>
-------------	--------------------

Description

retrieves one of the available palettes

Usage

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

Arguments

palette	one of the available palettes. 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	<i>Gnbu</i>
------	-------------

Description

Data Frame of the gnbu palette

Usage

gnbu()

green2red	<i>Green2red</i>
-----------	------------------

Description

Data Frame of the green2red palette

Usage

green2red()

greens	<i>Greens</i>
--------	---------------

Description

Data Frame of the greens palette

Usage

greens()

greys	<i>Greys</i>
-------	--------------

Description

Data Frame of the greys palette

Usage

greys()

heat	<i>Heat</i>
------	-------------

Description

Data Frame of the heat palette

Usage

heat()

heat_hcl	<i>Heat_hcl</i>
----------	-----------------

Description

Data Frame of the heat_hcl palette

Usage

heat_hcl()

inferno	<i>Inferno</i>
---------	----------------

Description

Data frame of the inferno palette

Usage

inferno()

magenta2green	<i>Magenta2green</i>
---------------	----------------------

Description

Data Frame of the magenta2green palette

Usage

magenta2green()

magma	<i>Magma</i>
-------	--------------

Description

Data frame of the magma palette

Usage

magma()

matlab_like	<i>Matlab_like</i>
-------------	--------------------

Description

Data Frame of the matlab_like palette

Usage

matlab_like()

matlab_like2	<i>Matlab_like2</i>
--------------	---------------------

Description

Data Frame of the matlab_like2 palette

Usage

matlab_like2()

oranges	<i>Oranges</i>
---------	----------------

Description

Data Frame of the oranges palette

Usage

oranges()

orrd	<i>Orrd</i>
------	-------------

Description

Data Frame of the orrd palette

Usage

orrd()

piyg	<i>Piyg</i>
------	-------------

Description

Data Frame of the piyg palette

Usage

piyg()

plasma	<i>Plasma</i>
--------	---------------

Description

Data frame of the plasma palette

Usage

plasma()

prgn	<i>Prgn</i>
------	-------------

Description

Data Frame of the prgn palette

Usage

prgn()

pubu	<i>Pubu</i>
------	-------------

Description

Data Frame of the pubu palette

Usage

pubu()

pubugn	<i>Pubugn</i>
--------	---------------

Description

Data Frame of the pubugn palette

Usage

pubugn()

puor	<i>Puor</i>
------	-------------

Description

Data Frame of the puor palette

Usage

puor()

purd	<i>Purd</i>
------	-------------

Description

Data Frame of the purd palette

Usage

purd()

purples	<i>Purples</i>
---------	----------------

Description

Data Frame of the purples palette

Usage

```
purples()
```

rainbow	<i>Rainbow</i>
---------	----------------

Description

Data Frame of the rainbow palette

Usage

```
rainbow()
```

rainbow_hcl	<i>Rainbow_hcl</i>
-------------	--------------------

Description

Data Frame of the rainbow_hcl palette

Usage

```
rainbow_hcl()
```

rdbu	<i>Rdbu</i>
------	-------------

Description

Data Frame of the rdbu palette

Usage

```
rdbu()
```

rdgy	<i>Rdgy</i>
------	-------------

Description

Data Frame of the rdgy palette

Usage

rdgy()

rdpu	<i>Rdpu</i>
------	-------------

Description

Data Frame of the rdpu palette

Usage

rdpu()

rdylbu	<i>Rdylbu</i>
--------	---------------

Description

Data Frame of the rdylbu palette

Usage

rdylbu()

rdylgn	<i>Rdylgn</i>
--------	---------------

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 available 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	<i>Spectral</i>
----------	-----------------

Description

Data Frame of the spectral palette

Usage

spectral()

terrain	<i>Terrain</i>
---------	----------------

Description

Data frame of the terrain palette

Usage

terrain()

terrain_hcl	<i>Terrain_hcl</i>
-------------	--------------------

Description

Data Frame of the terrain_hcl palette

Usage

terrain_hcl()

topo	<i>Topo</i>
------	-------------

Description

Data Frame of the topo palette

Usage

topo()

viridis	<i>Viridis</i>
---------	----------------

Description

Data frame of the viridis palette

Usage

viridis()

ygobb	<i>Ygobb</i>
-------	--------------

Description

Data Frame of the ygobb palette

Usage

ygobb()

ylgn	<i>Ylgn</i>
------	-------------

Description

Data Frame of the ylgn palette

Usage

ylgn()

ylgnbu	<i>Ylgnbu</i>
--------	---------------

Description

Data Frame of the ylgnbu palette

Usage

ylgnbu()

<code>ylorbr</code>	<i>Ylorbr</i>
---------------------	---------------

Description

Data Frame of the ylorbr palette

Usage

`ylorbr()`

<code>ylorrd</code>	<i>Ylorrd</i>
---------------------	---------------

Description

Data Frame of the ylorrd palette

Usage

`ylorrd()`

Index

blue2green, 3
blue2red, 3
blue2yellow, 3
blues, 4
brbg, 4
bugn, 4
bupu, 4

cividis, 5
cm, 5
color_palettes (colour_palettes), 5
color_values (colour_values), 6
color_values_rgb (colour_values_rgb), 10
colour_palettes, 5, 15, 23
colour_values, 6
colour_values_rgb, 10
colour_values_to_hex.character
 (colour_values), 6
colour_values_to_hex.Date
 (colour_values), 6
colour_values_to_hex.factor
 (colour_values), 6
colour_values_to_hex.logical
 (colour_values), 6
colour_values_to_hex.POSIXct
 (colour_values), 6
colour_values_to_hex.POSIXlt
 (colour_values), 6
colour_values_to_rgb.character
 (colour_values_rgb), 10
colour_values_to_rgb.Date
 (colour_values_rgb), 10
colour_values_to_rgb.factor
 (colour_values_rgb), 10
colour_values_to_rgb.logical
 (colour_values_rgb), 10
colour_values_to_rgb.POSIXct
 (colour_values_rgb), 10
colour_values_to_rgb.POSIXlt
 (colour_values_rgb), 10

convert_color (convert_colour), 13
convert_colors (convert_colour), 13
convert_colour, 13
convert_colours (convert_colour), 13
cyan2yellow, 14

diverge_hcl, 14
diverge_hsv, 15

get_palette, 15
gnbu, 16
green2red, 16
greens, 16
greys, 16

heat, 17
heat_hcl, 17

inferno, 17

magenta2green, 17
magma, 18
matlab_like, 18
matlab_like2, 18

oranges, 18
orrd, 19

piyg, 19
plasma, 19
prgn, 19
pubu, 20
pubugn, 20
puor, 20
purd, 20
purples, 21

rainbow, 21
rainbow_hcl, 21
rdbu, 21
rdgy, 22

rdpu, [22](#)
rdylbu, [22](#)
rdylgn, [22](#)
reds, [23](#)

sequential_hcl, [23](#)
show_colours, [23](#)
spectral, [24](#)

terrain, [24](#)
terrain_hcl, [24](#)
topo, [24](#)

viridis, [25](#)

ygobb, [25](#)
ylgn, [25](#)
ylgnbu, [25](#)
ylorbr, [26](#)
ylorrd, [26](#)