Package: gghsci (via r-universe)

September 18, 2024

Type Package

Title HSCI ggplot2 theme

Version 1.2.2

Author Eetu Mäkelä

Maintainer Eetu Mäkelä <eetu.makela@helsinki.fi>

Description ggplot2 theme used by the Human Sciences -- Computing Interaction research group at the University of Helsinki.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports ggplot2, magrittr, scales, dplyr, grDevices, rlang

Suggests testthat

RoxygenNote 7.2.3

Repository https://hsci-r.r-universe.dev

RemoteUrl https://github.com/hsci-r/gghsci

RemoteRef HEAD

RemoteSha 5a40cb7a685443b0f474e8f761fe279330833f2c

Contents

`coloropt_na_value` *extract the coloropt final neutral gray value for different palettes*

Description

extract the coloropt final neutral gray value for different palettes

Usage

```
coloropt_na_value(n, option = "normal")
```

Arguments

`n`	the number of colors required from the palette (max 13 for normal, 7 for the other palettes)
`option`	coloropt palette from normal, bright, dark, fancy and tarnish

---

`coloropt_pal` *coloropt palette*

Description

coloropt palette

Usage

```
coloropt_pal(option = "normal")
```

Arguments

`option`	coloropt palette from normal, bright, dark, fancy and tarnish

`convert_palette_to_grayscale` *convert a palette to grayscale*

Description

convert a palette to grayscale

Usage

```
convert_palette_to_grayscale(palette)
```

Arguments

`palette`	palette to convert

`save_plots` *Save plots to PNG + SVG files*

Description

Save plots to PNG + SVG files

Usage

```
save_plots(dir, basename, plot, width = 6, height = 5, units = "in", dpi = 300)
```

Arguments

`dir`	directory to save the plots to
`basename`	the base name for the plot files
`plot`	the plot to save
`width`	plot width
`height`	plot height
`units`	the units in which width & height are measured
`dpi`	plot resolution (dots per inch)

`scale_coloropt` *coloropt discrete color/fill scale*

Description

coloropt discrete color/fill scale

Usage

```
scale_coloropt(..., option = "normal", aesthetics = c("colour", "fill"))
```

Arguments

`...` Arguments passed on to `discrete_scale`

`palette` A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take (e.g., `scales::hue_pal()`).

`breaks` One of:

- `NULL` for no breaks
- `waiver()` for the default breaks (the scale limits)
- A character vector of breaks
- A function that takes the limits as input and returns breaks as output. Also accepts rlang lambda function notation.

`limits` One of:

- `NULL` to use the default scale values
- A character vector that defines possible values of the scale and their order
- A function that accepts the existing (automatic) values and returns new ones. Also accepts rlang lambda function notation.

`drop` Should unused factor levels be omitted from the scale? The default, `TRUE`, uses the levels that appear in the data; `FALSE` uses all the levels in the factor.

`na.translate` Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.

`scale_name` The name of the scale that should be used for error messages associated with this scale.

`name` The name of the scale. Used as the axis or legend title. If `waiver()`, the default, the name of the scale is taken from the first mapping used for that aesthetic. If `NULL`, the legend title will be omitted.

`labels` One of:

- `NULL` for no labels
- `waiver()` for the default labels computed by the transformation object
- A character vector giving labels (must be same length as `breaks`)

- An expression vector (must be the same length as breaks). See ?plotmath for details.
- A function that takes the breaks as input and returns labels as output. Also accepts rlang lambda function notation.

`guide` A function used to create a guide or its name. See `guides()` for more information.

`expand` For position scales, a vector of range expansion constants used to add some padding around the data to ensure that they are placed some distance away from the axes. Use the convenience function `expansion()` to generate the values for the `expand` argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.

`position` For position scales, The position of the axis. `left` or `right` for y axes, `top` or `bottom` for x axes.

`super` The super class to use for the constructed scale

`option` coloropt palette from normal, bright, dark, fancy and tarnish

`aesthetics` Character string or vector of character strings listing the name(s) of the aesthetic(s) that this scale works with. This can be useful, for example, to apply colour settings to the `colour` and `fill` aesthetics at the same time, via `aesthetics = c("colour", "fill")`.

`scale_coloropt_grayscale` *grayscale version of the coloropt palette*

Description

grayscale version of the coloropt palette

Usage

```
scale_coloropt_grayscale(
  ...,
  option = "normal",
  aesthetics = c("colour", "fill")
)
```

Arguments

`...` Arguments passed on to `discrete_scale`

`palette` A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take (e.g., `scales::hue_pal()`).

`breaks` One of:

- `NULL` for no breaks

- `waiver()` for the default breaks (the scale limits)
- A character vector of breaks
- A function that takes the limits as input and returns breaks as output. Also accepts rlang lambda function notation.

`limits` One of:

- `NULL` to use the default scale values
- A character vector that defines possible values of the scale and their order
- A function that accepts the existing (automatic) values and returns new ones. Also accepts rlang lambda function notation.

`drop` Should unused factor levels be omitted from the scale? The default, `TRUE`, uses the levels that appear in the data; `FALSE` uses all the levels in the factor.

`na.translate` Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.

`scale_name` The name of the scale that should be used for error messages associated with this scale.

`name` The name of the scale. Used as the axis or legend title. If `waiver()`, the default, the name of the scale is taken from the first mapping used for that aesthetic. If `NULL`, the legend title will be omitted.

`labels` One of:

- `NULL` for no labels
- `waiver()` for the default labels computed by the transformation object
- A character vector giving labels (must be same length as `breaks`)
- An expression vector (must be the same length as breaks). See ?plotmath for details.
- A function that takes the breaks as input and returns labels as output. Also accepts rlang lambda function notation.

`guide` A function used to create a guide or its name. See `guides()` for more information.

`expand` For position scales, a vector of range expansion constants used to add some padding around the data to ensure that they are placed some distance away from the axes. Use the convenience function `expansion()` to generate the values for the `expand` argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.

`position` For position scales, The position of the axis. `left` or `right` for y axes, `top` or `bottom` for x axes.

`super` The super class to use for the constructed scale

`option` coloropt palette from normal, bright, dark, fancy and tarnish

`aesthetics` Character string or vector of character strings listing the name(s) of the aesthetic(s) that this scale works with. This can be useful, for example, to apply colour settings to the `colour` and `fill` aesthetics at the same time, via `aesthetics = c("colour", "fill")`.

`scale_viridis_c_grayscale` *grayscale version of the continuous viridis palette*

Description

grayscale version of the continuous viridis palette

Usage

```
scale_viridis_c_grayscale(
  ...,
  alpha = 1,
  begin = 0,
  end = 1,
  direction = 1,
  option = "D",
  values = NULL,
  space = "Lab",
  na.value = "grey50",
  guide = "colourbar",
  aesthetics = c("colour", "fill")
)
```

Arguments

`...`	Other arguments passed on to `discrete_scale()`, `continuous_scale()`, or `binned_scale()` to control name, limits, breaks, labels and so forth.
`alpha`	The alpha transparency, a number in [0,1], see argument alpha in `hsv`.
`begin, end`	The (corrected) hue in `[0,1]` at which the color map begins and ends.
`direction`	Sets the order of colors in the scale. If 1, the default, colors are ordered from darkest to lightest. If -1, the order of colors is reversed.
`option`	A character string indicating the color map option to use. Eight options are available: • "magma" (or "A") • "inferno" (or "B") • "plasma" (or "C") • "viridis" (or "D") • "cividis" (or "E") • "rocket" (or "F") • "mako" (or "G") • "turbo" (or "H")

values	if colours should not be evenly positioned along the gradient this vector gives the position (between 0 and 1) for each colour in the colours vector. See rescale() for a convenience function to map an arbitrary range to between 0 and 1.
space	colour space in which to calculate gradient. Must be "Lab" - other values are deprecated.
na.value	Missing values will be replaced with this value.
guide	A function used to create a guide or its name. See guides() for more information.
aesthetics	Character string or vector of character strings listing the name(s) of the aesthetic(s) that this scale works with. This can be useful, for example, to apply colour settings to the colour and fill aesthetics at the same time, via aesthetics = c("colour", "fill").

See Also

ggplot2::scale_colour_viridis_c

---

theme_hsci *HSCI ggplot theme*

Description

HSCI ggplot theme

Usage

```
theme_hsci(base_size = 12, base_family = "sans")
```

Arguments

base_size	base font size, given in pts.
base_family	base font family

See Also

ggplot

theme_hsci_continuous *HSCI ggplot theme with viridis continuous palette*

Description

HSCI ggplot theme with viridis continuous palette

Usage

```
theme_hsci_continuous(
  base_size = 12,
  base_family = "sans",
  palette = "viridis"
)
```

Arguments

base_size	base font size, given in pts.
base_family	base font family
palette	viridis palette to use. Four options are available: "magma" (or "A"), "inferno" (or "B"), "plasma" (or "C"), "viridis" (or "D", the default option) and "cividis" (or "E").

See Also

theme_hsci

theme_hsci_continuous_grayscale *HSCI ggplot theme with grayscale viridis continuous palette*

Description

HSCI ggplot theme with grayscale viridis continuous palette

Usage

```
theme_hsci_continuous_grayscale(
  base_size = 12,
  base_family = "sans",
  palette = "viridis"
)
```

Arguments

base_size	base font size, given in pts.
base_family	base font family
palette	viridis palette to use. Four options are available: "magma" (or "A"), "inferno" (or "B"), "plasma" (or "C"), "viridis" (or "D", the default option) and "cividis" (or "E").

See Also

theme_hsci

---

theme_hsci_discrete *HSCI ggplot theme with a discrete coloropt palette.*

Description

HSCI ggplot theme with a discrete coloropt palette.

Usage

```
theme_hsci_discrete(base_size = 12, base_family = "sans", palette = "normal")
```

Arguments

base_size	base font size, given in pts.
base_family	base font family
palette	coloropt palette from normal, bright, dark, fancy and tarnish

See Also

theme_hsci

---

theme_hsci_discrete_grayscale *HSCI ggplot theme with a grayscale coloropt discrete palette.*

Description

HSCI ggplot theme with a grayscale coloropt discrete palette.

Usage

```
theme_hsci_discrete_grayscale(
  base_size = 12,
  base_family = "sans",
  palette = "normal"
)
```

Arguments

base_size	base font size, given in pts.
base_family	base font family
palette	coloropt palette from normal, bright, dark, fancy and tarnish

See Also

theme_hsci

Index