Package ‘hues’

December 1, 2019

Type Package
Version 0.2.0
Title Distinct Colour Palettes Based on 'iwanthue'
Description Creating effective colour palettes for figures is challenging. This package generates and plot palettes of optimally distinct colours in perceptually uniform colour space, based on 'iwanthue' <http://tools.medialab.sciences-po.fr/iwanthue/>. This is done through k-means clustering of CIE Lab colour space, according to user-selected constraints on hue, chroma, and lightness.
Date 2019-12-01
Depends R (>= 3.2.0)
Imports colorspace, methods
Suggests ggplot2
URL https://github.com/johnbaums/hues
BugReports https://github.com/johnbaums/hues/issues
License LGPL (>= 3)
LazyData true
RoxygenNote 7.0.1
NeedsCompilation no
Author John Baumgartner [aut, cre] (<https://orcid.org/0000-0002-8898-0300>), Russell Dinnage [aut]
Maintainer John Baumgartner <johnbaums@gmail.com>
Repository CRAN
Date/Publication 2019-12-01 17:30:02 UTC
R topics documented:

hues ................................................................. 2
hues-ggplot2-scales .............................................. 2
iwanthue ............................................................ 4
swatch ............................................................... 6

Index

hues

hues: Generate optimally distinct colour palettes

Description

This package generates and plot palettes of optimally distinct colours in perceptually uniform colour space, based on `iwanthue`. This is done through k-means clustering of CIE Lab colour space, according to user-selected constraints on hue, chroma, and lightness.

References

- iwanthue
- iwanthue GitHub repository

hues-ggplot2-scales  iwanthue scales to use with ggplot2

Description

These functions allow you to use `iwanthue()` generated palettes with ggplot2 plots. You need ggplot2 installed for these to work. Note these only work with discrete scales.

Usage

```r
scale_colour_iwanthue(
  ..., 
  hmin = 0, 
  hmax = 360, 
  cmin = 0, 
  cmax = 180, 
  lmin = 0, 
  lmax = 100, 
  random = FALSE, 
  aesthetics = "colour"
)
```

```r
scale_color_iwanthue(
```
...,
hmin = 0,
hmax = 360,
cmin = 0,
cmax = 180,
lmin = 0,
lmax = 100,
random = FALSE,
aesthetics = "colour"
)

scale_fill_iwanthue(
  ...
)
  hmin = 0,
hmax = 360,
cmin = 0,
cmax = 180,
lmin = 0,
lmax = 100,
random = FALSE,
aesthetics = "fill"
)

Arguments

... Arguments to pass on to ggplot2::discrete_scale().
hmin Numeric, in the range [0, 360]. The lower limit of the hue range to be clustered.
hmax Numeric, in the range [0, 360]. The upper limit of the hue range to be clustered.
cmin Numeric, in the range [0, 180]. The lower limit of the chroma range to be clustered.
cmax Numeric, in the range [0, 180]. The upper limit of the chroma range to be clustered.
lmin Numeric, in the range [0, 100]. The lower limit of the luminance range to be clustered.
lmax Numeric, in the range [0, 100]. The upper limit of the luminance range to be clustered.
random Logical. If TRUE, clustering will be determined by the existing RNG state. If FALSE, the seed will be set to 1 for clustering, and on exit, the function will restore the pre-existing RNG state.
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").

Value

A ScaleDiscrete object that can be added to a ggplot object.
**iwanthue**

Generate a colour palette by k-means clustering of CIE Lab colour space.

```r
if (require('ggplot2')) {
  ggplot(iris, aes(x=Petal.Width, y=Petal.Length)) +
  geom_point(aes(color=Species), size=10) +
  scale_colour_iwanthue()

  ggplot(iris, aes(x=Petal.Width, y=Petal.Length)) +
  geom_point(aes(color=Species), size=10) +
  scale_colour_iwanthue(hmax = 90)
}
```

**Description**

Generate a palette of distinct colours through k-means clustering of CIE Lab colour space.

**Usage**

```r
iwanthue(
  n,
  hmin = 0,
  hmax = 360,
  cmin = 0,
  cmax = 180,
  lmin = 0,
  lmax = 100,
  plot = FALSE,
  random = FALSE
)
```

**Arguments**

- `n` Numeric. The number of colours to generate.
- `hmin` Numeric, in the range [0, 360]. The lower limit of the hue range to be clustered.
- `hmax` Numeric, in the range [0, 360]. The upper limit of the hue range to be clustered.
- `cmin` Numeric, in the range [0, 180]. The lower limit of the chroma range to be clustered.
iwanthue

- \texttt{cmax} Numeric, in the range [0, 180]. The upper limit of the chroma range to be clustered.
- \texttt{lmin} Numeric, in the range [0, 100]. The lower limit of the luminance range to be clustered.
- \texttt{lmax} Numeric, in the range [0, 100]. The upper limit of the luminance range to be clustered.
- \texttt{plot} Logical. Should the colour swatches be plotted (using \texttt{swatch()})?
- \texttt{random} Logical. If \texttt{TRUE}, clustering will be determined by the existing RNG state. If \texttt{FALSE}, the seed will be set to 1 for clustering, and on exit, the function will restore the pre-existing RNG state.

\section*{Details}

Note that \texttt{iwanthue} currently doesn’t support \texttt{hmin} greater than \texttt{hmax} (which should be allowed, since hue is circular).

\section*{Value}

A vector of \texttt{n} colours (as hexadecimal strings), representing centers of clusters determined through k-means clustering of the CIE Lab colour space delimited by \texttt{hmin}, \texttt{hmax}, \texttt{cmin}, \texttt{cmax}, \texttt{lmin} and \texttt{lmax}.

\section*{References}

- Examples follow those presented at \texttt{iwanthue - colors for data scientists}
- \texttt{iwanthue} on GitHub

\section*{See Also}

\texttt{swatch}

\section*{Examples}

\begin{verbatim}
iwanthue(5)
iwanthue(5, plot=TRUE)
iwanthue(5, 0, 240, 0, 24, 0, 100, plot=TRUE) # shades
iwanthue(5, 0, 360, 0, 54, 67, 100, plot=TRUE) # pastel
iwanthue(5, 0, 360, 54, 180, 27, 67, plot=TRUE) # pimp
iwanthue(5, 0, 360, 36, 180, 13, 73, plot=TRUE) # intense
iwanthue(3, 0, 300, 0, 180, 73, 100, plot=TRUE) # fluoro
iwanthue(3, 220, 260, 12, 150, 53, 100, plot=TRUE) # blue ocean
\end{verbatim}
Description

Plot named colour swatches for a vector of colours.

Usage

swatch(x)

Arguments

x  a vector of colours, specified as: colour names (i.e. colour names returned by colors()); numeric indices into palette(), or hexadecimal strings in the form "#RRGGBB", where RR, GG, and BB are pairs of hexadecimal digits representing red, green, and blue components, in the range 00 to FF.

Value

NULL. The colour swatch is plotted to the active plotting device.

See Also

iwanthue

Examples

swatch(colours()[1:10])
swatch(1:4)
swatch(iwanthue(5))
Index

colors().6

ggplot2::discrete_scale(), 3

hues.2
hues-ggplot2-scales.2

iwanthue, 4, 6
iwanthue().2

palette().6

scale_color_iwanthue
  (hues-ggplot2-scales), 2
scale_colour_iwanthue
  (hues-ggplot2-scales), 2
scale_fill_iwanthue
  (hues-ggplot2-scales), 2

swatch, 5, 6
swatch().5