Package ‘dichromat’

May 2, 2022

Version 2.0-0.1
Date 2013-01-23
Title Color Schemes for Dichromats
Description Collapse red-green or green-blue distinctions to simulate the effects of different types of color-blindness.
Depends R (>= 2.10), stats
License GPL-2
LazyLoad Yes
Author Thomas Lumley [aut, cre],
Ken Knoblauch [ctb],
Scott Waichler [ctb],
Achim Zeileis [ctb]
Maintainer Thomas Lumley <tlumley@u.washington.edu>
Repository CRAN
Date/Publication 2022-05-02 13:01:50 UTC
NeedsCompilation no

R topics documented:
colorschemes ................................................................. 2
dalton .............................................................. 3
dichromat ............................................................. 4

Index 7
### Description

17 color schemes suitable for people with deficient or anomalous red-green vision.

### Usage

colorschemes

### Format

- BrowntoBlue.10
- BrowntoBlue.12
- BluetoDarkOrange.12
- BluetoDarkOrange.18
- DarkRedtoBlue.12
- DarkRedtoBlue.18
- BluetoGreen.14
- BluetoGray.8
- BluetoOrangeRed.14
- BluetoOrange.10
- BluetoOrange.12
- BluetoOrange.8
- LightBluetoDarkBlue.10
- LightBluetoDarkBlue.7
- Categorical.12
- GreentoMagenta.16
- SteppedSequential.5

### Author(s)

Scott Waichler <scott.waichler@pnl.gov>

### Source

[https://geography.uoregon.edu/datagraphics/color_scales.htm](https://geography.uoregon.edu/datagraphics/color_scales.htm)

### References

Examples

```r
pal <- function(col, ...) 
  image(seq_along(col), 1, matrix(seq_along(col), ncol = 1), 
  col = col, axes = FALSE, ...)

opar <- par(mar = c(1, 2, 1, 1))
layout(matrix(1:6, ncol = 1))
pal(colorschemes$BrowntoBlue.10, main = "Brown to Blue (10)"
pal(colorRampPalette(colorschemes$BrowntoBlue.10, space = "Lab")(100), 
  main = "Brown to Blue Ramp")
pal(dichromat(colorschemes$BrowntoBlue.10), 
  main = "Brown to Blue (10) -- deuteranopia")
pal(colorschemes$Categorical.12, main = "Categorical (12)"
pal(dichromat(colorschemes$Categorical.12), 
  main = "Categorical (12) -- deuteranopia")
pal(dichromat(colorschemes$Categorical.12, "protan"), 
  main = "Categorical (12) -- protanopia")
par(opar)
```

dalton

Effects of Daltonism (Red-Green Color Blindness)

Description

A 256-color palette as it would appear with normal vision, and with two types of red-green color blindness: protanopia and deuteronopia. Furthermore, a color palette for a type of green-blue color blindness (tritanopia) is available.

Usage

```r
data("dalton")
```

Format

dalton A 256 x 3 x 4 array. The columns index the red, green and blue color values, the layers index the vision type.
dalton.colors: A list of colors, with names normal, protan, deutan, tritan.

Details

The dalton array lists a grid of 256 colors in RGB coordinates for "normal" vision as well as corresponding RGB coordinates in which certain red-green ("deutan" and "protan") or green-blue contrasts ("tritan") are collapsed.

The formulas for mapping the RGB coordinates to the collapsed coordinates have been developed in a series of papers by Brettel, Mollon, and Viénot. The protan/deutan mapping is provided in Table 2 of Viénot et al. (1999) while the tritan case is discussed in Brettel at al. (1997).

The dalton.colors list contains the sRGB colors (as R color strings) as computed with rgb from dalton.

For an automatic mapping of a given color vector to its dichromatic counterpart, see dichromat.
The data were kindly provided by Françoise Viénot.
The deutan/protan data (rounded to integers) is also available in Table 2 of Viénot et al. (1999).

References


See Also

dichromat, rgb

Examples

data("dalton", package = "dichromat")
par(mfrow = c(4, 1))
image(matrix(1:256, 128), col = dalton.colors$normal)
image(matrix(1:256, 128), col = dalton.colors$deutan)
image(matrix(1:256, 128), col = dalton.colors$protan)
image(matrix(1:256, 128), col = dalton.colors$tritan)

---

dichromat

Remove Red-Green or Green-Blue Contrasts from Colors

Description

Collapses red-green or green-blue color distinctions to approximate the effect of the three forms of dichromacy: protanopia and deuteranopia (red-green color blindness), and tritanopia (green-blue color blindness). deuteranopia.

Usage

dichromat(colours, type = c("deutan", "protan", "tritan"))

Arguments

- **colours**: A vector of R colors, either color names or color hex strings.
- **type**: Type of color-blindness to simulate,
Details

Someone with the specified form of color blindness will find that the transformation has little effect on the appearance of colors. Colors that are indistinguishable after transformation were likely indistinguishable to them before transformation. About 10% of men (and almost no women) have some degree of red-green color blindness. Tritanopia is much less common but occurs in both males and females.

The mapping from the original color vector to the dichromatic counterpart is based on a sequence of papers by Brettel, Mollon, and Viénot. For more details, see the references and also the underlying data set `dalton`.

Value

A vector of R colors.

Author(s)

Thomas Lumley

References


See Also

dalton, rgb

Examples

```r
## from example(pie)
par(mfrow = c(2, 2))
pie.sales <- c(0.12, 0.3, 0.26, 0.16, 0.04, 0.12)
names(pie.sales) <- c("Blueberry", "Cherry", "Apple", "Boston Cream", "Other", "Vanilla Cream")
pie(pie.sales, # default colors
   col = c("white", "lightblue", "mistyrose", "lightcyan", "lavender", "cornsilk"))
pie(pie.sales,
```
dichromat

col = c("purple", "violetred1", "green3", "cornsilk", "cyan", "white")
pie(pie.sales, col = dichromat(c("white", "lightblue", "mistyrose", "lightcyan", "lavender", "cornsilk")))
pie(pie.sales, col = dichromat(c("purple", "violetred1", "green3", "cornsilk", "cyan", "white")))

## standard color schemes
pie(rep(1,10), col = heat.colors(10))
pie(rep(1,10), col = dichromat(heat.colors(10)))
pie(rep(1,8), col = palette())
pie(rep(1,8), col = dichromat(palette()))
pie(rep(1,15), col = topo.colors(15))
pie(rep(1,15), col = dichromat(topo.colors(15)))
pie(rep(1,15), col = terrain.colors(15))
pie(rep(1,15), col = dichromat(terrain.colors(15)))
pie(rep(1,15), col = cm.colors(15))
pie(rep(1,15), col = dichromat(cm.colors(15)))

## color ramp schemes
bluescale <- colorRampPalette(c("FFFCFC", "C7E9B4", "7FCDBB", 
"40B6C4", "2C7FB8", "253494"))
redgreen <- colorRampPalette(c("red", "green3"))
pie(rep(1,15), col = bluescale(15))
pie(rep(1,15), col = dichromat(bluescale(15)))
par(mfrow = c(2, 4))
x <- matrix(rnorm(10 * 10), 10)
image(1:10, 1:10, x, col = bluescale(10), main = "blue-yellow scale")
image(1:10, 1:10, x, col = dichromat(bluescale(10), "deutan"), main = "deutan")
image(1:10, 1:10, x, col = dichromat(bluescale(10), "protan"), main = "protan")
image(1:10, 1:10, x, col = dichromat(bluescale(10), "tritan"), main = "tritan")
image(1:10, 1:10, x, col = redgreen(10), main = "red-green scale")
image(1:10, 1:10, x, col = dichromat(redgreen(10), "deutan"), main = "deutan")
image(1:10, 1:10, x, col = dichromat(redgreen(10), "protan"), main = "protan")
image(1:10, 1:10, x, col = dichromat(redgreen(10), "tritan"), main = "tritan")
Index

* color
  dichromat. 4
* datasets
  colorschemes. 2
dalton. 3
colorschemes. 2
dalton. 3, 5
dichromat. 3, 4, 4
rgb. 3–5