Package 'tricolore'

May 15, 2024

Type Package

Title A Flexible Color Scale for Ternary Compositions

Version 1.2.4

Description Compositional data consisting of three-parts can be color mapped with a ternary color scale. Such a scale is provided by the tricolore packages with options for discrete and continuous colors, mean-centering and scaling. See Jonas Schöley (2021) ``The centered ternary balance scheme. A technique to visualize surfaces of unbalanced three-part compositions" <doi:10.4054/DemRes.2021.44.19>, Jonas Schöley, Frans Willekens (2017) ``Visualizing compositional data on the Lexis surface" <doi:10.4054/DemRes.2017.36.21>, and Ilya Kashnitsky, Jonas Schöley (2018) ``Regional population structures at a glance" <doi:10.1016/S0140-6736(18)31194-2>.

License GPL-3

URL https://github.com/jschoeley/tricolore

Encoding UTF-8

LazyData true

Depends R (>= 4.0)

Imports grDevices, ggplot2 (>= 3.4.0), ggtern (>= 3.4.0), rlang (>= 1.1.0), shiny, assertthat

RoxygenNote 7.3.1

Suggests testthat, knitr, rmarkdown, sf, leaflet, httpuv, dplyr

VignetteBuilder knitr

NeedsCompilation no

Author Jonas Schöley [aut, cre] (<https://orcid.org/0000-0002-3340-8518>), Ilya Kashnitsky [aut] (<https://orcid.org/0000-0003-1835-8687>)

Maintainer Jonas Schöley <jschoeley@gmail.com>

Repository CRAN

Date/Publication 2024-05-15 15:00:02 UTC

R topics documented:

`DemoTricolore` *Interactive Tricolore Demonstration*

Description

An interactive demonstration of the tricolore color scale inspired by the colorbrewer2.org application. Helps in picking the right color scale for your data.

Usage

```
DemoTricolore()
```

Value

Opens a shiny app session.

`euro_basemap` *Flat Map of European Continent*

Description

A ggplot object rendering a flat background map of the European continent.

Usage

```
euro_basemap
```

Format

An object of class gg (inherits from `ggplot`) of length 9.

Source

Derived from Eurostats European Geodata. (c) EuroGeographics for the administrative boundaries.
https://ec.europa.eu/eurostat/web/gisco/geodata/reference-data/administrative-units-statistical-uni

`euro_example` *NUTS-2 Level Geodata and Compositional Data for Europe*

Description

A simple-features dataframe containing the NUTS-2 level polygons of European regions along with regional compositional data on education and labor-force.

Usage

```
euro_example
```

Format

A data frame with 312 rows and 9 variables:

id NUTS-2 code.

name Name of NUTS-2 region.

ed_0to2 Share of population with highest attained education "lower secondary or less".

ed_3to4 Share of population with highest attained education "upper secondary".

ed_5to8 Share of population with highest attained education "tertiary".

lf_pri Share of labor-force in primary sector.

lf_sec Share of labor-force in secondary sector.

lf_ter Share of labor-force in tertiary sector.

geometry Polygon outlines for regions in sf package format.

Details

Variables starting with "ed" refer to the relative share of population ages 25 to 64 by educational attainment in the European NUTS-2 regions 2016.

Variables starting with "lf" refer to the relative share of workers by labor-force sector in the European NUTS-2 regions 2016. The original NACE (rev. 2) codes have been recoded into the three sectors "primary" (A), "secondary" (B-E & F) and "tertiary" (all other NACE codes).

Source

Derived from Eurostats European Geodata. (c) EuroGeographics for the administrative boundaries.
https://ec.europa.eu/eurostat/web/gisco/geodata/reference-data/administrative-units-statistical-uni

Education data derived from Eurostats table "edat_lfse_04".

Labor-force data derived from Eurostats table "lfst_r_lfe2en2".

`Tricolore` *Ternary Balance Color Scale*

Description

Color-code three-part compositions with a ternary balance color scale and return a color key.

Usage

```
Tricolore(
  df,
  p1,
  p2,
  p3,
  center = rep(1/3, 3),
  breaks = ifelse(identical(center, rep(1/3, 3)), 4, Inf),
  hue = 0.2,
  chroma = 0.7,
  lightness = 0.8,
  contrast = 0.4,
  spread = 1,
  legend = TRUE,
  show_data = TRUE,
  show_center = ifelse(identical(center, rep(1/3, 3)), FALSE, TRUE),
  label_as = ifelse(identical(center, rep(1/3, 3)), "pct", "pct_diff"),
  crop = FALSE,
  input_validation = TRUE
)
```

Arguments

`df`	Data frame of compositional data.
`p1`	Column name for variable in df giving first proportion of ternary composition (string).
`p2`	Column name for variable in df giving second proportion of ternary composition (string).
`p3`	Column name for variable in df giving third proportion of ternary composition (string).
`center`	Ternary coordinates of the color scale center. (default = 1/3,1/3,1/3). NA puts center over the compositional mean of the data.
`breaks`	Number of per-axis breaks in the discrete color scale. An integer >1. Values above 99 imply no discretization.
`hue`	Primary hue of the first ternary element (0 to 1).
`chroma`	Maximum possible chroma of mixed colors (0 to 1).
`lightness`	Lightness of mixed colors (0 to 1).

contrast	Lightness contrast of the color scale (0 to 1).
spread	The spread of the color scale. Choose values > 1 to focus the color scale on the center.
legend	Should a legend be returned along with the colors? (default=TRUE)
show_data	Should the data be shown on the legend? (default=TRUE)
show_center	Should the center be shown on the legend? (default=FALSE if center is at c(1/3, 1/3, 1/3), otherwise TRUE)
label_as	"pct" for percent-share labels or "pct_diff" for percent-point-difference from center labels. (default='pct' if center is at c(1/3, 1/3, 1/3), otherwise 'pct_diff')
crop	Should the legend be cropped to the data? (default=FALSE)
input_validation	Should the function arguments be validated? (default=TRUE)

Value

- legend=FALSE: A vector of rgbs hex-codes representing the ternary balance scheme colors.
- legend=TRUE: A list with elements "rgb" and "key".

Examples

```
P <- as.data.frame(prop.table(matrix(runif(3^6), ncol = 3), 1))
Tricolore(P, 'V1', 'V2', 'V3')
```

TricoloreSextant *Ternary Sextant Color Scale*

Description

Color-code three-part compositions with a ternary sextant color scale and return a color key.

Usage

```
TricoloreSextant(
  df,
  p1,
  p2,
  p3,
  center = rep(1/3, 3),
  values = c("#FFFF00", "#B3DCC3", "#01A0C6", "#B8B3D8", "#F11D8C", "#FFB3B3"),
  legend = TRUE,
  show_data = TRUE,
  show_center = TRUE,
  label_as = ifelse(identical(center, rep(1/3, 3)), "pct", "pct_diff"),
  crop = FALSE,
  input_validation = TRUE
)
```

Arguments

df	Data frame of compositional data.
p1	Column name for variable in df giving first proportion of ternary composition (string).
p2	Column name for variable in df giving second proportion of ternary composition (string).
p3	Column name for variable in df giving third proportion of ternary composition (string).
center	Ternary coordinates of the color scale center. (default = 1/3,1/3,1/3). NA puts center over the compositional mean of the data.
values	6 element character vector of rgb-codes.
legend	Should a legend be returned along with the colors? (default=TRUE)
show_data	Should the data be shown on the legend? (default=TRUE)
show_center	Should the center be shown on the legend? (default=FALSE if center is at c(1/3, 1/3, 1/3), otherwise TRUE)
label_as	"pct" for percent-share labels or "pct_diff" for percent-point-difference from center labels. (default='pct' if center is at c(1/3, 1/3, 1/3), otherwise 'pct_diff')
crop	Should the legend be cropped to the data? (default=FALSE)
input_validation	Should the function arguments be validated? (default=TRUE)

Value

- legend=FALSE: A vector of rgbs hex-codes representing the ternary balance scheme colors.
- legend=TRUE: A list with elements "rgb" and "key".

Examples

```
P <- as.data.frame(prop.table(matrix(runif(3^6), ncol = 3), 1))
TricoloreSextant(P, 'V1', 'V2', 'V3')
```

Index