

Package ‘schemr’

October 14, 2022

Type Package

Title Convert Images to Usable Color Schemes

Version 0.1.0

Description A fast and adaptable tool to convert photos and images into usable colour schemes for data visualisation. Contains functionality to extract colour palettes from images, as well for the conversion of images between colour spaces.

License GPL-3

Encoding UTF-8

LazyData true

Imports dplyr, stringr, ggplot2, magrittr, purrr, apcluster, OpenImageR, methods

RoxygenNote 7.1.1

Suggests testthat

NeedsCompilation no

Author Stuart Morrison [aut, cre]

Maintainer Stuart Morrison <stuart@spheric.al>

Repository CRAN

Date/Publication 2021-03-10 18:40:02 UTC

R topics documented:

hex_to_lab	2
hex_to_rgb	3
hex_to_xyz	3
image_to_palette	4
lab_to_hex	5
lab_to_rgb	6
lab_to_xyz	7
palette,schemr-method	7
plot,schemr,ANY-method	8
rgb_to_hex	8

rgb_to_lab	9
rgb_to_xyz	10
schemr-class	10
xyz_to_hex	11
xyz_to_lab	12
xyz_to_rgb	12

Index	14
--------------	-----------

hex_to_lab	<i>Convert hex RGB values to Lab space.</i>
------------	---

Description

Convert hex RGB values to Lab space.

Usage

```
hex_to_lab(hex, transformation = "sRGB", linear_func = NULL)
```

Arguments

hex	A character vector containing hex representations of RGB colours.
transformation	An option in c("sRGB", "Adobe") for a built-in transformation or, alternatively, a custom 3x3 transformation matrix.
linear_func	A function to convert RGB colour space into non-linear RGB space. Used only if a custom transformation matrix is provided. Transformation skips if no function is provided under a user-defined transformation matrix. See: https://en.wikipedia.org/wiki/SRGB .

Value

A tibble of L, a and b colour space values.

Examples

```
red <- sample(x = 1:255, size = 10, replace = TRUE)
green <- sample(x = 1:255, size = 10, replace = TRUE)
blue <- sample(x = 1:255, size = 10, replace = TRUE)
hex_to_lab(rgb_to_hex(data.frame(r = red, g = green, b = blue)))
```

hex_to_rgb	<i>Convert hexadecimal colours to RGB colour channels.</i>
------------	--

Description

Convert hexadecimal colours to RGB colour channels.

Usage

```
hex_to_rgb(hex)
```

Arguments

hex	A character vector containing hex representations of RGB colours.
-----	---

Value

A tibble of red, green and blue colour channels.

Examples

```
hex_to_rgb(c("#5f9e3a"))
```

hex_to_xyz	<i>Convert hex RGB values to XYZ space.</i>
------------	---

Description

Convert hex RGB values to XYZ space.

Usage

```
hex_to_xyz(hex, transformation = "sRGB", linear_func = NULL)
```

Arguments

hex	A character vector containing hex representations of RGB colours.
transformation	An option in c("sRGB", "Adobe") for a built-in transformation or, alternatively, a custom 3x3 transformation matrix.
linear_func	A function to convert RGB colour space into non-linear RGB space. Used only if a custom transformation matrix is provided. Transformation skips if no function is provided under a user-defined transformation matrix. See: https://en.wikipedia.org/wiki/SRGB .

Value

A tibble of X, Y and Z colour space values.

Examples

```
red <- sample(x = 1:255, size = 10, replace = TRUE)
green <- sample(x = 1:255, size = 10, replace = TRUE)
blue <- sample(x = 1:255, size = 10, replace = TRUE)
hex_to_xyz(rgb_to_hex(data.frame(r = red, g = green, b = blue)))
```

image_to_palette *Develop a usable colour palette form an image.*

Description

Develop a usable colour palette form an image.

Usage

```
image_to_palette(
  image_path,
  resize_factor = NULL,
  colour_space = "sRGB",
  rgb_to_linear_func = NULL,
  rgb_to_nonlinear_func = NULL,
  method = "slic",
  superpixel = 200,
  compactness = 20,
  verbose = TRUE,
  s = negDistMat(r = 2),
  summary_method = mean,
  ...
)
```

Arguments

image_path	A character path to the image to cluster. Reads images of type .png, .jpeg, .jpg, .tiff.
resize_factor	A numeric scalar that reduces (or increases) the size of the image before any processing.
colour_space	The colour space of the original image. The clustering is undertaken in the Lab space. This is an an option in c("sRGB", "Adobe") for a built-in transformation or, alternatively, a custom 3x3 transformation matrix.
rgb_to_linear_func	The clustering is undertaken in the Lab space. This is a function to convert RGB colour space into linear RGB space. Used only if a custom transformation matrix is provided. Transformation skips if no function is provided under a user-defined transformation matrix. See: https://en.wikipedia.org/wiki/SRGB .

rgb_to_nonlinear_func	The clustering is undertaken in the Lab space. This is a function to convert linear RGB colour space into non-linear RGB space. Used only if a custom transformation matrix is provided. Transformation skips if no function is provided under a user-defined transformation matrix. See: https://en.wikipedia.org/wiki/SRGB .
method	From <code>OpenImageR::superpixels</code> . A character string specifying the method to use. Either "slic" or "slico".
superpixel	From <code>OpenImageR::superpixels</code> . A numeric value specifying the number of superpixels to use.
compactness	From <code>OpenImageR::superpixels</code> . A numeric value specifying the compactness parameter. The compactness parameter is needed only if method is "slic". The "slico" method adaptively chooses the compactness parameter for each superpixel differently.
verbose	From <code>OpenImageR::superpixels</code> . A boolean. If TRUE then information will be printed in the R session.
s	From <code>apcluster::apcluster</code> . An $l \times l$ similarity matrix or a similarity function either specified as the name of a package-provided similarity function as character string or a user provided function object. <code>s</code> may also be a sparse matrix according to the <code>Matrix</code> package. Internally, <code>apcluster</code> uses the <code>dgTMatrix</code> class; all other sparse matrices are cast to this class (if possible, otherwise the function quits with an error). If <code>s</code> is any other object of class <code>Matrix</code> , <code>s</code> is cast to a regular matrix internally (if possible, otherwise the function quits with an error).
summary_method	Function to summarise colours in clustered superpixels. Defaults to <code>mean</code> .
...	Other arguments to be passed to the <code>apcluster</code> algorithm. For the methods with signatures <code>character,ANY</code> and <code>function,ANY</code> , all other arguments are passed to the selected similarity function as they are; for the methods with signatures <code>Matrix,missing</code> and <code>sparseMatrix,missing</code> , further arguments are passed on to the <code>apcluster</code> methods with signatures <code>Matrix,missing</code> and <code>dgTMatrix,missing</code> , respectively.

Value

A `schemr` object containing colour scheme colours and image properties and clusters.

lab_to_hex

Convert from Lab space into hex RGB colour values.

Description

Convert from Lab space into hex RGB colour values.

Usage

```
lab_to_hex(lab, transformation = "sRGB", linear_func = NULL)
```

Arguments

lab	A dataframe or matrix with L, a and b colour channels located in the columns 1 to 3, respectively.
transformation	An option in <code>c("sRGB", "Adobe")</code> for a built-in transformation or, alternatively, a custom 3x3 transformation matrix.
linear_func	A function to convert RGB colour space into non-linear RGB space. Used only if a custom transformation matrix is provided. Transformation skips if no function is provided under a user-defined transformation matrix. See: https://en.wikipedia.org/wiki/SRGB .

Value

A character vector with hex representations of RGB colour channels.

Examples

```
red <- sample(x = 1:255, size = 10, replace = TRUE)
green <- sample(x = 1:255, size = 10, replace = TRUE)
blue <- sample(x = 1:255, size = 10, replace = TRUE)
lab_to_hex(rgb_to_lab(data.frame(r = red, g = green, b = blue)))
```

lab_to_rgb

Convert from Lab space into RGB colour channels.

Description

Convert from Lab space into RGB colour channels.

Usage

```
lab_to_rgb(lab, transformation = "sRGB", linear_func = NULL)
```

Arguments

lab	A dataframe or matrix with L, a and b colour channels located in the columns 1 to 3, respectively.
transformation	An option in <code>c("sRGB", "Adobe")</code> for a built-in transformation or, alternatively, a custom 3x3 transformation matrix.
linear_func	A function to convert RGB colour space into non-linear RGB space. Used only if a custom transformation matrix is provided. Transformation skips if no function is provided under a user-defined transformation matrix. See: https://en.wikipedia.org/wiki/SRGB .

Value

A tibble of red, green and blue colour channels.

Examples

```
red <- sample(x = 1:255, size = 10, replace = TRUE)
green <- sample(x = 1:255, size = 10, replace = TRUE)
blue <- sample(x = 1:255, size = 10, replace = TRUE)
lab_to_rgb(rgb_to_lab(data.frame(r = red, g = green, b = blue)))
```

`lab_to_xyz`*Convert from Lab space to XYZ colour channels.*

Description

Convert from Lab space to XYZ colour channels.

Usage

```
lab_to_xyz(lab)
```

Arguments

`lab` A dataframe or matrix with L, a and b colour channels located in the columns 1 to 3, respectively.

Value

A tibble of X, Y and Z colour channels.

Examples

```
l <- sample(x = 40:60, size = 10, replace = TRUE)
a <- sample(x = -128:128, size = 10, replace = TRUE)
b <- sample(x = -128:128, size = 10, replace = TRUE)
lab_to_xyz(data.frame(l = l, a = a, b = b))
```

`palette,schemr-method` *Plot the colour palette*

Description

Plot the colour palette

Usage

```
## S4 method for signature 'schemr'
palette(value)
```

Arguments

value A schemr class object

Value

No return value, calls a barplot of the colour palette.

plot, schemr, ANY-method

Plot the clustered image data

Description

Plot the clustered image data

Usage

```
## S4 method for signature 'schemr,ANY'
plot(x, y = NULL, ...)
```

Arguments

x A schemr class object
y Not used, NULL
... Other arguments to pass onto 'plot'

Value

No return value, calls a raster plot of the clustered image data.

rgb_to_hex

Convert RGB colour channels to hex colour codes.

Description

Convert RGB colour channels to hex colour codes.

Usage

```
rgb_to_hex(rgb)
```

Arguments

rgb A dataframe or matrix with red, green and blue colour channels located in the columns 1 to 3, respectively. Colour channel values should be between 0 and 255, inclusive.

Value

A character vector with hex representations of RGB colour channels.

Examples

```
red <- sample(x = 1:255, size = 10, replace = TRUE)
green <- sample(x = 1:255, size = 10, replace = TRUE)
blue <- sample(x = 1:255, size = 10, replace = TRUE)
rgb_to_hex(data.frame(r = red, g = green, b = blue))
```

rgb_to_lab

Convert from RGB colour channels to Lab space.

Description

Convert from RGB colour channels to Lab space.

Usage

```
rgb_to_lab(rgb, transformation = "sRGB", linear_func = NULL)
```

Arguments

- | | |
|----------------|---|
| rgb | A dataframe or matrix with red, green and blue colour channels located in the columns 1 to 3, respectively. Colour channel values should be between 0 and 255, inclusive. |
| transformation | An option in c("sRGB", "Adobe") for a built-in transformation or, alternatively, a custom 3x3 transformation matrix. |
| linear_func | A function to convert RGB colour space into linear RGB space. Used only if a custom transformation matrix is provided. Transformation skips if no function is provided under a user-defined transformation matrix. See: https://en.wikipedia.org/wiki/SRGB . |

Value

A tibble of L, a and b colour space values.

Examples

```
red <- sample(x = 1:255, size = 10, replace = TRUE)
green <- sample(x = 1:255, size = 10, replace = TRUE)
blue <- sample(x = 1:255, size = 10, replace = TRUE)
rgb_to_lab(data.frame(r = red, g = green, b = blue), transformation = "Adobe")
```

rgb_to_xyz	<i>Convert from RGB colour channels to XYZ space.</i>
------------	---

Description

Convert from RGB colour channels to XYZ space.

Usage

```
rgb_to_xyz(rgb, transformation = "sRGB", linear_func = NULL)
```

Arguments

rgb	A dataframe or matrix with red, green and blue colour channels located in the columns 1 to 3, respectively. Colour channel values should be between 0 and 255, inclusive.
transformation	An option in c("sRGB", "Adobe") for a built-in transformation or, alternatively, a custom 3x3 transformation matrix.
linear_func	A function to convert RGB colour space into linear RGB space. Used only if a custom transformation matrix is provided. Transformation skips if no function is provided under a user-defined transformation matrix. See: https://en.wikipedia.org/wiki/SRGB .

Value

A tibble of X, Y and Z colour channels.

Examples

```
red <- sample(x = 1:255, size = 10, replace = TRUE)
green <- sample(x = 1:255, size = 10, replace = TRUE)
blue <- sample(x = 1:255, size = 10, replace = TRUE)
rgb_to_xyz(data.frame(r = red, g = green, b = blue), transformation = "Adobe")
```

schemr-class	<i>Create the schemr class, which holds the palette and image data</i>
--------------	--

Description

Create the schemr class, which holds the palette and image data

Fields

image	An array of dimension (Image width) by (Image height) by (3 colour channels) that contains the data of the original image
clustered_image	An array of dimension (Image width) by (Image height) by (3 colour channels) that contains the data of the image with clustered colour blocks
palette	A character vector that contains the colours of the resulting colour palette

Methods

`print(x)` Print the colour palette.

xyz_to_hex

Convert from XYZ space into hex RGB colour values.

Description

Convert from XYZ space into hex RGB colour values.

Usage

```
xyz_to_hex(xyz, transformation = "sRGB", linear_func = NULL)
```

Arguments

<code>xyz</code>	A dataframe or matrix with X, Y and Z colour channels located in the columns 1 to 3, respectively.
<code>transformation</code>	An option in <code>c("sRGB", "Adobe")</code> for a built-in transformation or, alternatively, a custom 3x3 transformation matrix.
<code>linear_func</code>	A function to convert RGB colour space into non-linear RGB space. Used only if a custom transformation matrix is provided. Transformation skips if no function is provided under a user-defined transformation matrix. See: https://en.wikipedia.org/wiki/SRGB .

Value

A character vector with hex representations of RGB colour channels.

Examples

```
x <- sample(x = 40:60, size = 10, replace = TRUE)
y <- sample(x = 40:60, size = 10, replace = TRUE)
z <- sample(x = 40:60, size = 10, replace = TRUE)
xyz_to_hex(data.frame(x = x, y = y, z = z))
```

xyz_to_lab *Convert from XYZ colour channels to Lab space.*

Description

Convert from XYZ colour channels to Lab space.

Usage

```
xyz_to_lab(xyz)
```

Arguments

xyz A dataframe or matrix with X, Y and Z colour channels located in the columns 1 to 3, respectively.

Value

A tibble of L, a and b colour space values.

Examples

```
x <- sample(x = 40:60, size = 10, replace = TRUE)
y <- sample(x = 40:60, size = 10, replace = TRUE)
z <- sample(x = 40:60, size = 10, replace = TRUE)
xyz_to_lab(data.frame(x = x, y = y, z = z))
```

xyz_to_rgb *Convert from RGB colour channels to XYZ space.*

Description

Convert from RGB colour channels to XYZ space.

Usage

```
xyz_to_rgb(xyz, transformation = "sRGB", linear_func = NULL)
```

Arguments

xyz A dataframe or matrix with X, Y and Z colour channels located in the columns 1 to 3, respectively.

transformation An option in c("sRGB", "Adobe") for a built-in transformation or, alternatively, a custom 3x3 transformation matrix.

linear_func A function to convert linear RGB colour space into RGB space. Used only if a custom transformation matrix is provided. Transformation skips if no function is provided under a user-defined transformation matrix. See: <https://en.wikipedia.org/wiki/SRGB>.

Value

A tibble of red, green and blue colour channels.

Examples

```
x <- sample(x = 40:60, size = 10, replace = TRUE)
y <- sample(x = 40:60, size = 10, replace = TRUE)
z <- sample(x = 40:60, size = 10, replace = TRUE)
xyz_to_rgb(data.frame(x = x, y = y, z = z))
```

Index

[hex_to_lab](#), [2](#)

[hex_to_rgb](#), [3](#)

[hex_to_xyz](#), [3](#)

[image_to_palette](#), [4](#)

[lab_to_hex](#), [5](#)

[lab_to_rgb](#), [6](#)

[lab_to_xyz](#), [7](#)

[palette](#), [schemr-method](#), [7](#)

[plot](#), [schemr](#), [ANY-method](#), [8](#)

[rgb_to_hex](#), [8](#)

[rgb_to_lab](#), [9](#)

[rgb_to_xyz](#), [10](#)

[schemr-class](#), [10](#)

[xyz_to_hex](#), [11](#)

[xyz_to_lab](#), [12](#)

[xyz_to_rgb](#), [12](#)