Package ‘schemr’

June 25, 2024

Type Package
Title Convert Images to Usable Color Schemes
Version 0.3.0
BugReports https://github.com/stuart-morrison/schemr/issues
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
Imports dplyr, stringr, magrittr, purrr, apcluster, OpenImageR, methods
RoxygenNote 7.3.1
Suggests testthat
NeedsCompilation no
Author Stuart Morrison [aut, cre]
Maintainer Stuart Morrison <smorrison@ucdavis.edu>
Repository CRAN
Date/Publication 2024-06-25 02:30:02 UTC

Contents

hex_to_lab .......................................................... 2
hex_to_rgb .......................................................... 3
hex_to_xyz .......................................................... 3
hsl_to_hsv .......................................................... 4
hsl_to_lab .......................................................... 5
hsl_to_rgb .......................................................... 5
hsl_to_xyz .......................................................... 6
hsv_to_hsl .......................................................... 6
hsv_to_lab .......................................................... 7
hsv_to_rgb .......................................................... 8
hex_to_lab

Convert hex RGB values to Lab space.

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.
**hex_to_rgb**

**Examples**

```r
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**  
*Convert hexadecimal colours to RGB colour channels.*

**Description**

Convert hexadecimal colours to RGB colour channels.

**Usage**

```r
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**

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

---

**hex_to_xyz**  
*Convert hex RGB values to XYZ space.*

**Description**

Convert hex RGB values to XYZ space.

**Usage**

```r
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.
hsl_to_hsv

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)))

hsl_to_hsv

Convert HSL to HSV

Description

Convert HSL to HSV

Usage

hsl_to_hsv(hsl)

Arguments

hsl A dataframe or matrix with H, S and L colour channels located in the columns 1 to 3, respectively. H in degrees in [0, 360], S and L in [0, 1]

Value

A tibble of H, S and V colour channels. Hue is constant between colour spaces, while saturation differs.

Examples

H <- sample(x = 0:360, size = 10, replace = TRUE)
S <- runif(n = 10)
L <- runif(n = 10)
hsl_to_hsv(data.frame(h = H, s = S, l = L))
**hsl_to_lab**

*Convert HSL to Lab*

**Description**

Convert HSL to Lab

**Usage**

\[
\text{hsl}_1 \rightarrow \text{lab} \left( hsl, \text{transformation} = "sRGB", \text{linear_func} = \text{NULL} \right)
\]

**Arguments**

- **hsl**: A dataframe or matrix with H, S and L colour channels located in the columns 1 to 3, respectively. H in degrees in \([0, 360]\), S and L in \([0, 1]\).
- **transformation**: An option in \(\text{c("sRGB", "Adobe")}\) for a built-in transformation or, alternatively, a custom \(3 \times 3\) 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](https://en.wikipedia.org/wiki/SRGB).

**Value**

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

**hsl_to_rgb**

*Convert HSL space into RGB space*

**Description**

Convert HSL space into RGB space

**Usage**

\[
\text{hsl}_1 \rightarrow \text{rgb} \left( hsl \right)
\]

**Arguments**

- **hsl**: A dataframe or matrix with H, S and L colour channels located in the columns 1 to 3, respectively. H in degrees in \([0, 360]\), S and L in \([0, 1]\).

**Value**

A tibble of red, green and blue colour channels.
Examples

```r
H <- sample(x = 0:360, size = 10, replace = TRUE)
S <- runif(n = 10)
L <- runif(n = 10)
hsl_to_rgb(data.frame(h = H, s = S, l = L))
```

---

**hsl_to_xyz**

Convert HSL to XYZ

**Description**

Convert HSL to XYZ

**Usage**

```r
hsl_to_xyz(hsl, transformation = "sRGB", linear_func = NULL)
```

**Arguments**

- **hsl**: A dataframe or matrix with H, S and L colour channels located in the columns 1 to 3, respectively. H in degrees in [0, 360], S and L in [0, 1]
- **transformation**: An option in \(\text{c("sRGB", "Adobe")}\) 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.

---

**hsv_to_hsl**

Convert HSV to HSL

**Description**

Convert HSV to HSL

**Usage**

```r
hsv_to_hsl(hsv)
```

**Arguments**

- **hsv**: A dataframe or matrix with H, S and V colour channels located in the columns 1 to 3, respectively. H in degrees in [0, 360], S and V in [0, 1]
Value

A tibble of H, S and L colour channels. Hue is constant between colour spaces, while saturation differs.

Examples

```r
H <- sample(x = 0:360, size = 10, replace = TRUE)
S <- runif(n = 10)
V <- runif(n = 10)
hsv_to_hsl(data.frame(h = H, s = S, v = V))
```

---

**hsv_to_lab**  
(Convert HSV to Lab)

Description

Convert HSV to Lab

Usage

```r
hsv_to_lab(hsv, transformation = "sRGB", linear_func = NULL)
```

Arguments

- `hsv`: A dataframe or matrix with H, S and V colour channels located in the columns 1 to 3, respectively. H in degrees in [0, 360], S and L in [0, 1]
- `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.
## hsv_to_rgb

### Convert HSV to RGB

**Description**

Convert HSV to RGB

**Usage**

```r
hsv_to_rgb(hsv)
```

**Arguments**

- `hsv` A dataframe or matrix with H, S and V colour channels located in the columns 1 to 3, respectively. H in degrees in [0, 360], S and L in [0, 1]

**Value**

A tibble of red, green and blue colour channels.

## hsv_to_xyz

### Convert HSV to XYZ

**Description**

Convert HSV to XYZ

**Usage**

```r
hsv_to_xyz(hsv, transformation = "sRGB", linear_func = NULL)
```

**Arguments**

- `hsv` A dataframe or matrix with H, S and V colour channels located in the columns 1 to 3, respectively. H in degrees in [0, 360], S and L in [0, 1]
- `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.
image_to_pallette  Develop a usable colour palette form an image.

Description

Develop a usable colour palette form an image.

Usage

```r
image_to_pallette(
  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 OpenImageR::superpixels. A character string specifying the method to use. Either "slic" or "slico".
**superpixel**  
From OpenImageR::superpixels. A numeric value specifying the number of superpixels to use.

**compactness**  
From OpenImageR::superpixels. 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 OpenImageR::superpixels. A boolean. If TRUE then information will be printed in the R session.

**s**  
From apcluster::apcluster. 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. $s$ may also be a sparse matrix according to the Matrix package. Internally, apcluster uses the dgTMatrix class; all other sparse matrices are cast to this class (if possible, otherwise the function quits with an error). If $s$ is any other object of class Matrix, $s$ 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 mean.

**...**  
Other arguments to be passed to the apcluster algorithm. For the methods with signatures character, ANY and function, ANY, all other arguments are passed to the selected similarity function as they are; for the methods with signatures Matrix, missing and sparseMatrix, missing, further arguments are passed on to the apcluster methods with signatures Matrix, missing and dgTMatrix, missing, 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**

```r
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 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.
lab_to_hsl

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_hsl

Convert Lab to HSL

Description

Convert Lab to HSL

Usage

lab_to_hsl(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 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 H, S and L colour channels.
---

**lab_to_hsv**  
*Convert Lab to HSV*

**Description**

Convert Lab to HSV

**Usage**

lab_to_hsv(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 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 H, S and V colour channels.

---

**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 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.
lab_to_xyz

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_hsl**  
*Convert RGB space into HSL space*

**Description**  
Convert RGB space into HSL space

**Usage**  
`rgb_to_hsl(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 tibble of H, S and L 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_hsl(data.frame(r = red, g = green, b = blue))

rgb_to_hsv

Convert RGB to HSV

Description

Convert RGB to HSV

Usage

rgb_to_hsv(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 tibble of H, S and V colour channels.

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)
rgb_to_xyz

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")

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

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
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 RGB colour space into non-linear RGB space. Used only
if a custom transformation matrix is provided. Transformation skips if no func-
tion is provided under a user-defined transformation matrix. See: https://en.wikipedia.org/wiki/SRGB.
xyz_to_hsl

Value

A character vector with hex representations of RGB colour channels.

Examples

```r
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_hsl (Convert XYZ to HSL)

Description

Convert XYZ to HSL

Usage

`xyz_to_hsl(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 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 H, S and L colour channels.
xyz_to_hsv

Convert XYZ to HSV

Description

Convert XYZ to HSV

Usage

xyz_to_hsv(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 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 H, S and V colour channels.

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.
**xyz_to_rgb**

Convert from RGB colour channels to XYZ space.

**Usage**

```r
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**

```r
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
hsl_to_hsv, 4
hsl_to_lab, 5
hsl_to_rgb, 5
hsl_to_xyz, 6
hsv_to_hsl, 6
hsv_to_lab, 7
hsv_to_rgb, 8
hsv_to_xyz, 8

image_to_pallette, 9

lab_to_hex, 10
lab_to_hsl, 11
lab_to_hsv, 12
lab_to_rgb, 12
lab_to_xyz, 13

palette, schemr-method, 14
plot, schemr, ANY-method, 14

rgb_to_hex, 15
rgb_to_hsl, 15
rgb_to_hsv, 16
rgb_to_lab, 16
rgb_to_xyz, 17

schemr-class, 18

xyz_to_hex, 18
xyz_to_hsl, 19
xyz_to_hsv, 20
xyz_to_lab, 20
xyz_to_rgb, 21