Package: palettecore (via r-universe)

November 3, 2026

Title Derive, Optimise and Audit a Scientific Colour Palette from One Seed Colour

Version 0.4.1

Description Generates sequential, diverging and categorical colour palettes from a single seed colour in OKLCH (the cylindrical lightness-chroma-hue representation of the Oklab perceptual colour space), with spacing measured by the CIEDE2000 colour-difference formula of the International Commission on Illumination. Audits every palette under simulated colour-vision deficiency, greyscale conversion, the standard Red Green Blue (sRGB) gamut and Web Content Accessibility Guidelines (WCAG) contrast. Colour-vision deficiency is simulated at severity 1.0 with the model of Machado, Oliveira and Fernandes (2009) <doi:10.1109/TVCG.2009.113>; the design rationale follows Crameri, Shephard and Heron (2020) <doi:10.1038/s41467-020-19160-7>. Mirrors the 'Python' reference implementation maintained in the same repository and is validated against shared parity fixtures. Thresholds are configurable design rules, not established accessibility cut-offs.

License Apache License (>= 2)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Depends R (>= 4.1)

Imports stats, utils

Suggests ggplot2, jsonlite

URL https://github.com/heidihelena/palettecore

BugReports https://github.com/heidihelena/palettecore/issues

Repository https://heidihelena.r-universe.dev

Date/Publication 2026-08-05 11:08:29 UTC

RemoteUrl https://github.com/heidihelena/palettecore

RemoteRef HEAD

RemoteSha e88898ef8950221c7e93aa5a78922365deb0c484

RemoteSubdir r/palettecore

Contents

ciede2000 . . . 2
contrast_ratio . . . 3
generate_palette . . . 3
hex_to_oklch . . . 4
hex_to_srgb . . . 5
max_chroma . . . 5
oklch_to_hex . . . 6
out_of_gamut_excursion . . . 6
scale_colour_accessible . . . 7
scale_fill_accessible . . . 7
simulate_cvd . . . 8
srgb_to_hex . . . 8

Index **9**

`ciede2000` *CIEDE2000 colour difference between two sRGB colours*

Description

Computed via CIELAB under D65 (Sharma et al. 2005 formulation).

Usage

```
ciede2000(rgb1, rgb2)
```

Arguments

`rgb1`, `rgb2`	Numeric vectors of length 3, sRGB channels in [0, 1].

Value

CIEDE2000 distance (0 for identical colours, about 100 for black vs white).

contrast_ratio *WCAG 2.x contrast ratio between two sRGB colours*

Description

WCAG 2.x contrast ratio between two sRGB colours

Usage

```
contrast_ratio(rgb1, rgb2)
```

Arguments

rgb1, rgb2	Numeric vectors of length 3, sRGB channels in [0, 1].

Value

Contrast ratio between 1 and 21.

generate_palette *Generate an audited palette from one seed colour*

Description

Generate an audited palette from one seed colour

Usage

```
generate_palette(
  seed,
  n = 8L,
  kind = "sequential",
  background = "#FFFFFF",
  use = "data_fill",
  thresholds = NULL,
  anchor = "path",
  vividness = 0,
  rotations = 1
)
```

Arguments

seed	Seed colour as HEX, e.g. "#8B6FC9".
n	Number of colours (2-24).
kind	"sequential", "diverging" or "categorical".
background	Intended background as HEX.
use	"data_fill", "text", "line" or "UI".
thresholds	Named list overriding the default dE floors.
anchor	"path" or "exact" (categorical always contains the seed).
vividness	0.0 (default, seed-faithful) to 1.0 (chroma pushed toward the gamut edge). Lifts chroma only, never lightness, on every kind; the default reproduces earlier versions exactly.
rotations	Helix only: how many full hue turns the cubehelix makes over its lightness range (fractional or negative allowed). Rejected if non-default for other kinds.

Value

List with hexes, diagnostics and warnings.

Examples

```
r <- generate_palette("#8B6FC9", n = 4, kind = "sequential")
r$hexes
r$warnings
```

---

hex_to_oklch *Convert a HEX colour to OKLCH*

Description

Convert a HEX colour to OKLCH

Usage

```
hex_to_oklch(hex_str)
```

Arguments

hex_str	Colour as HEX string.

Value

Numeric vector c(L, C, H): OKLab lightness, chroma, hue in degrees.

hex_to_srgb *Convert a HEX colour to sRGB*

Description

Convert a HEX colour to sRGB

Usage

```
hex_to_srgb(hex_str)
```

Arguments

hex_str	Colour as HEX string, e.g. "#8B6FC9" (3- or 6-digit).

Value

Numeric vector of length 3, sRGB channels in [0, 1].

max_chroma *Largest OKLCH chroma at (L, H) inside the sRGB gamut*

Description

Binary search; matches the Python reference implementation.

Usage

```
max_chroma(L, H, c_hi = 0.4, tol = 1e-04)
```

Arguments

L	OKLab lightness in [0, 1].
H	OKLCH hue in degrees.
c_hi	Upper search bound for chroma.
tol	Search tolerance.

Value

Maximum displayable chroma (0 if (L, H) itself is out of gamut).

oklch_to_hex *Convert an OKLCH colour to HEX*

Description

Convert an OKLCH colour to HEX

Usage

```
oklch_to_hex(lch)
```

Arguments

lch	Numeric vector c(L, C, H).

Value

HEX string (sRGB, channel values clipped to [0, 1]).

out_of_gamut_excursion *Pre-clamp gamut excursion of a CVD simulation*

Description

Largest linear-RGB excursion outside [0, 1] before clamping. 0 means the simulated colour was displayable as-is; larger values mean clamping altered the colour and measured separations involving it should be read with that in mind.

Usage

```
out_of_gamut_excursion(rgb, condition)
```

Arguments

rgb	Numeric vector of length 3, sRGB channels in [0, 1].
condition	One of "protanopia", "deuteranopia", "tritanopia".

Value

Non-negative excursion magnitude in linear RGB.

scale_colour_accessible *Discrete colour scale from an audited palettecore palette*

Description

Generates the palette with generate_palette() and surfaces its audit warnings via warning(), so an inaccessible combination never passes silently into a figure.

Usage

```
scale_colour_accessible(seed, kind = "categorical", ...)

scale_color_accessible(seed, kind = "categorical", ...)
```

Arguments

seed	Seed colour as HEX.
kind	Palette kind; "categorical" is the usual choice for discrete scales.
...	Passed to generate_palette() (background, use, thresholds, ...).

Value

A ggplot2 discrete scale object (class `ScaleDiscrete`, a `ggproto` object) for the `colour` aesthetic, to be added to a ggplot with +. When the plot is built, the scale generates an audited palette of hex colours for the number of levels in the data and emits any audit warnings (colour-vision deficiency, greyscale, gamut or contrast failures) via `warning()`.

scale_fill_accessible *Discrete fill scale from an audited palettecore palette*

Description

Generates the palette with generate_palette() and surfaces its audit warnings via warning(), so an inaccessible combination never passes silently into a figure.

Usage

```
scale_fill_accessible(seed, kind = "categorical", ...)
```

Arguments

seed	Seed colour as HEX.
kind	Palette kind; "categorical" is the usual choice for discrete scales.
...	Passed to generate_palette() (background, use, thresholds, ...).

Value

A ggplot2 discrete scale object (class `ScaleDiscrete`, a `ggproto` object) for the `fill` aesthetic, to be added to a ggplot with `+`. When the plot is built, the scale generates an audited palette of hex colours for the number of levels in the data and emits any audit warnings (colour-vision deficiency, greyscale, gamut or contrast failures) via `warning()`.

`simulate_cvd` *Simulate colour-vision deficiency (complete dichromacy)*

Description

Machado et al. (2009) matrices at severity 1.0 on the paper's [0, 1] scale, applied in linear RGB and clamped to the displayable range.

Usage

```
simulate_cvd(rgb, condition)
```

Arguments

`rgb`	Numeric vector of length 3, sRGB channels in [0, 1].
`condition`	One of "protanopia", "deuteranopia", "tritanopia".

Value

Simulated sRGB colour, clamped to [0, 1].

`srgb_to_hex` *Convert an sRGB colour to HEX*

Description

Convert an sRGB colour to HEX

Usage

```
srgb_to_hex(rgb)
```

Arguments

`rgb`	Numeric vector of length 3, sRGB channels in [0, 1].

Value

HEX string, e.g. "#8B6FC9".

Index

ciede2000, 2
contrast_ratio, 3

generate_palette, 3

hex_to_oklch, 4
hex_to_srgb, 5

max_chroma, 5

oklch_to_hex, 6
out_of_gamut_excursion, 6

scale_color_accessible
        *(*scale_colour_accessible*)*, 7
scale_colour_accessible, 7
scale_fill_accessible, 7
simulate_cvd, 8
srgb_to_hex, 8