

Package ‘biopalette’

September 2, 2026

Title Image-Inspired Color Palettes for Biomedical Visualization

Version 0.2.2

Description Provides a curated collection of image-inspired color palettes for biomedical visualization. The palettes are organized as qualitative, sequential, or diverging scales and include documented source context and intended use. The package provides functions to retrieve, inspect, preview, and apply these palettes in base R and 'ggplot2' graphics, together with utilities for working with palette definitions.

License MIT + file LICENSE

URL <https://github.com/evanbio/biopalette>,
<https://evanbio.github.io/biopalette/>

BugReports <https://github.com/evanbio/biopalette/issues>

Depends R (>= 4.1)

Imports cli, ggplot2, jsonlite, scales

Encoding UTF-8

Language en-US

RoxygenNote 7.3.2

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr, rmarkdown

Config/testthat/edition 3

NeedsCompilation no

Author Yibin Zhou [aut, cre] (ORCID: <<https://orcid.org/0009-0009-4600-8175>>)

Maintainer Yibin Zhou <evanzhou.bio@gmail.com>

Repository CRAN

Date/Publication 2026-09-02 12:10:02 UTC

Contents

biopalette	2
create_palette	3
get_palette	4
hex2rgb	5
list_palettes	6
palette_gallery	6
palette_info	7
preview_palette	8
remove_palette	9
rgb2hex	10
scale_color_biopalette	10
scale_color_biopalette_gradient	12
Index	15

biopalette	<i>biopalette: Image-Inspired Color Palettes for Biomedical Visualization</i>
------------	---

Description

The **biopalette** package provides image-inspired color palettes for biomedical visualization. Each palette is sourced from a real image — a film still, a scientific figure, or an artwork — and translated into a reproducible color system.

Details

- Main features:
- Retrieve palettes by name, type, and size
 - Preview and browse palettes interactively
 - Apply palettes as discrete or continuous ggplot2 scales
 - Create and manage custom palettes
 - Convert between HEX and RGB color formats

Author(s)

Maintainer: Yibin Zhou <evanzhou.bio@gmail.com> ([ORCID](#))

See Also

- Useful links:
- <https://github.com/evanbio/biopalette>
 - <https://evanbio.github.io/biopalette/>
 - Report bugs at <https://github.com/evanbio/biopalette/issues>

create_palette	<i>Create and Save a Custom Color Palette</i>
----------------	---

Description

Save a named color palette as a JSON file in a collection directory. The palette is usable immediately: point any reading function at the same palettes_dir. The JSON is written to a same-directory temporary file, validated, and then committed; a failed overwrite leaves the previous palette intact.

Usage

```
create_palette(
  name,
  type = c("sequential", "diverging", "qualitative"),
  colors,
  palettes_dir,
  overwrite = FALSE
)
```

Arguments

name	Character. Palette name (e.g., "blues").
type	Character. One of "sequential", "diverging", or "qualitative".
colors	Character vector of HEX color values (e.g., "#E64B35" or "#E64B35B2").
palettes_dir	Character. Directory to write the palette into. Required: there is deliberately no default, so a palette can never be written into the collection that ships with the package.
overwrite	Logical. If TRUE, overwrite existing palette file. Default: FALSE.

Value

Invisibly returns a list with path and info.

Examples

```
temp_dir <- file.path(tempdir(), "palettes")
create_palette("blues", "sequential", c("#deebf7", "#9ecae1", "#3182bd"),
  palettes_dir = temp_dir)
create_palette("qual_vivid", "qualitative", c("#E64B35", "#4DBBD5", "#00A087"),
  palettes_dir = temp_dir)

# Overwrite an existing palette explicitly
create_palette("blues", "sequential", c("#c6dbef", "#6baed6", "#2171b5"),
  palettes_dir = temp_dir, overwrite = TRUE)

unlink(temp_dir, recursive = TRUE)
```

get_palette

Get a Color Palette

Description

Retrieve a named palette by name and type, returning a vector of HEX colors. Automatically checks for type mismatch and provides smart suggestions.

Usage

```
get_palette(name, type = NULL, n = NULL, reverse = FALSE, palettes_dir = NULL)
```

Arguments

name	Character. Name of the palette (e.g. "qual_vivid").
type	Character. One of "sequential", "diverging", "qualitative". If NULL, type is auto-detected.
n	Integer. Number of colors to return. If NULL, returns all colors. See <i>What n means</i> above. Default is NULL.
reverse	Logical. Reverse the palette before n is applied, so the two arguments stay independent: reverse hands back a different palette and n then selects from it. Default: FALSE.
palettes_dir	Character. Directory holding a palette collection (sequential/, diverging/, qualitative/ subdirectories of JSON files). If NULL, the palettes bundled with the package are used.

Value

Character vector of HEX color codes.

What n means

n is resolved according to what the palette's type says the colors *are*, because "give me 3 colors" means two different things:

qualitative The colors are unordered categories, so n takes the first n of them. Asking for more than the palette holds is an error — there is no way to invent a category that the palette does not contain.

sequential, diverging The colors are stops along a ramp, so n returns n steps spanning the *whole* ramp, interpolating as needed. Interpolation takes place in Lab colour space, matching the package's ggplot2 gradient scales. Any n works, above or below the number of stops. Taking the first n stops instead would silently hand back one end of the ramp — the light half of a sequential scale, or one arm of a diverging one.

n equal to the number of stops returns the palette untouched. When reverse = TRUE the palette is flipped first, so n selects from the reversed palette.

Examples

```

get_palette("gene_red", type = "qualitative")

# Qualitative: the first n categories
get_palette("walter_white2", type = "qualitative", n = 2)

# Sequential: n steps across the whole ramp, not the first n stops
get_palette("mitonuclear_blue")
get_palette("mitonuclear_blue", n = 3)

# Ramps can also be stretched beyond the stops they were drawn from
get_palette("walter_white", type = "diverging", n = 9)

# Flip a ramp end to end
get_palette("mitonuclear_blue", reverse = TRUE)

```

hex2rgb

*Convert HEX Colors to RGB***Description**

Convert a character vector of HEX color codes to a data.frame with columns hex, r, g, b, and alpha.

Usage

```
hex2rgb(hex)
```

Arguments

hex	Character vector of HEX color codes (e.g. "#FF8000" or "#FF8000B2"). Both 6-digit and 8-digit (with alpha) codes are accepted. The # prefix is required. No NA values allowed.
-----	--

Value

A data.frame with hex (character) and integer r, g, b, and alpha columns. Channels lie in [0, 255]. alpha is NA_integer_ for a 6-digit input and the explicit alpha channel for an 8-digit input.

Examples

```

hex2rgb("#FF8000")
hex2rgb(c("#FF8000", "#00FF00"))

```

list_palettes	<i>List Available Color Palettes</i>
---------------	--------------------------------------

Description

Return a data.frame of all available palette metadata, optionally filtered by type.

Usage

```
list_palettes(type = NULL, sort = TRUE, palettes_dir = NULL)
```

Arguments

type	Palette type(s) to filter: "sequential", "diverging", "qualitative". Default NULL returns all.
sort	Whether to sort by type, n_color, name. Default: TRUE.
palettes_dir	Character. Directory holding a palette collection (sequential/, diverging/, qualitative/ subdirectories of JSON files). If NULL, the palettes bundled with the package are used.

Value

A data.frame with columns: name, type, n_color, colors.

Examples

```
list_palettes()
list_palettes(type = "qualitative")
list_palettes(type = c("sequential", "diverging"))
```

palette_gallery	<i>Visualize All Palettes in a Gallery View</i>
-----------------	---

Description

Display palettes in a paged gallery format, returning a named list of ggplot objects.

Usage

```
palette_gallery(
  type = NULL,
  max_palettes = 30,
  max_row = 12,
  verbose = TRUE,
  palettes_dir = NULL
)
```

Arguments

type	Palette types to include: "sequential", "diverging", "qualitative". Default NULL returns all.
max_palettes	Number of palettes per page. Default: 30.
max_row	Max colors per row. Default: 12.
verbose	Whether to print progress info. Default: TRUE.
palettes_dir	Character. Directory holding a palette collection (sequential/, diverging/, qualitative/ subdirectories of JSON files). If NULL, the palettes bundled with the package are used.

Value

A named list of ggplot objects (one per page).

Examples

```
palette_gallery()
palette_gallery(type = "qualitative")
palette_gallery(type = c("sequential", "diverging"), max_palettes = 10)
```

palette_info

Get Metadata for One Color Palette

Description

Return the runtime metadata for a single named palette. This is the one-palette counterpart to [list_palettes\(\)](#) and uses the same lookup rules as [get_palette\(\)](#).

Usage

```
palette_info(name, type = NULL, palettes_dir = NULL)
```

Arguments

name	Character. Palette name.
type	Character. One of "sequential", "diverging", or "qualitative". If NULL, the type is detected automatically.
palettes_dir	Character. Directory holding a palette collection. If NULL, the palettes bundled with the package are used.

Value

A one-row data.frame with columns name, type, n_color, and colors. The colors column is a list-column containing the complete HEX color vector.

Examples

```
palette_info("walter_white")
palette_info("babel", type = "qualitative")
```

preview_palette	<i>Preview a Color Palette</i>
-----------------	--------------------------------

Description

Visualize a palette using various plot styles.

Usage

```
preview_palette(
  name,
  type = NULL,
  n = NULL,
  reverse = FALSE,
  plot_type = c("bar", "pie", "point", "rect", "circle"),
  title = NULL,
  palettes_dir = NULL
)
```

Arguments

name	Character. Name of the palette.
type	Character. One of "sequential", "diverging", "qualitative". If NULL, auto-detected.
n	Integer. Number of colors to use. If NULL, uses all. Default: NULL.
reverse	Logical. Reverse the palette before n is applied. Passed straight through to get_palette() . Default: FALSE.
plot_type	Character. One of "bar", "pie", "point", "rect", "circle". Default: "bar".
title	Character. Plot title. If NULL, defaults to palette name.
palettes_dir	Character. Directory holding a palette collection (sequential/, diverging/, qualitative/ subdirectories of JSON files). If NULL, the palettes bundled with the package are used.

Details

"bar", "pie", "point" and "circle" are drawn with base graphics; "rect" is drawn with ggplot2. Either way the plot goes straight to the active device and nothing is returned — use [palette_gallery\(\)](#) when you want plot objects you can modify, arrange or save.

Value

NULL, invisibly. Called for its plotting side effect — the return shape is the same for every `plot_type`.

See Also

`palette_gallery()`, which returns `ggplot` objects instead of drawing.

Examples

```
preview_palette("gene_red", plot_type = "bar")
preview_palette("walter_white", plot_type = "pie")
preview_palette("walter_white2", n = 2, plot_type = "circle")
```

remove_palette	<i>Remove a Saved Palette JSON</i>
----------------	------------------------------------

Description

Remove a palette JSON file by name, searching across types if needed.

Usage

```
remove_palette(name, type = NULL, palettes_dir)
```

Arguments

name	Character. Palette name (without '.json' suffix).
type	Character. One of "sequential", "diverging", "qualitative". If NULL, searches all types.
palettes_dir	Character. Directory holding the palette collection. Required: there is deliberately no default, so the collection that ships with the package can never be removed from.

Value

Invisibly TRUE if removed successfully, FALSE otherwise.

Examples

```
temp_dir <- tempfile("biopalette-palettes-")
create_palette(
  "example_palette",
  "qualitative",
  c("#E64B35", "#4DBBD5", "#00A087"),
  palettes_dir = temp_dir
)
```

```
remove_palette("example_palette", palettes_dir = temp_dir)
unlink(temp_dir, recursive = TRUE)
```

rgb2hex

Convert RGB Values to HEX Color Codes

Description

Convert RGB or RGBA values to HEX color codes. Accepts either a numeric vector or a data.frame, symmetrically with [hex2rgb\(\)](#).

Usage

```
rgb2hex(rgb)
```

Arguments

rgb	A numeric vector of length 3 (r, g, b) or 4 (r, g, b, alpha), or a data.frame with r, g, b, and optional alpha columns. Matrices and arrays are not accepted. Values must lie in [0, 255] and are rounded to the nearest integer, with exact halves rounded up. In a data.frame, alpha = NA emits 6-digit HEX; a finite alpha value emits 8-digit HEX. Data frames must contain at least one row; unrelated extra columns are ignored.
-----	--

Value

A character vector of uppercase 6- or 8-digit HEX color codes.

Examples

```
rgb2hex(c(255, 128, 0))
rgb2hex(c(255, 128, 0, 178))
rgb2hex(hex2rgb(c("#FF8000", "#00FF0080")))
```

scale_color_biopalette

Discrete ggplot2 Scales from a biopalette Palette

Description

Use a named biopalette palette for a discrete colour or fill mapping. The scale asks [get_palette\(\)](#) for exactly as many colours as the trained data has levels. Qualitative palettes therefore use their first n colours, while sequential and diverging palettes sample n colours across their complete ramps.

Usage

```
scale_color_biopalette(  
  palette,  
  ...,  
  type = NULL,  
  reverse = FALSE,  
  palettes_dir = NULL  
)  
  
scale_colour_biopalette(  
  palette,  
  ...,  
  type = NULL,  
  reverse = FALSE,  
  palettes_dir = NULL  
)  
  
scale_fill_biopalette(  
  palette,  
  ...,  
  type = NULL,  
  reverse = FALSE,  
  palettes_dir = NULL  
)
```

Arguments

palette	Character. Name of the palette.
...	Passed to <code>ggplot2::discrete_scale()</code> , for example name, breaks, labels, limits, na.value, drop, and guide.
type	Character. One of "sequential", "diverging", or "qualitative". If NULL, the type is detected automatically.
reverse	Logical. Reverse the palette. Default: FALSE.
palettes_dir	Character. Directory holding a palette collection. If NULL, the palettes bundled with the package are used.

Details

All three palette types are supported. Qualitative palettes suit unordered categories; sequential and diverging palettes can be useful for ordered categories.

Value

A ggplot2 discrete scale.

Examples

```
library(ggplot2)

ggplot(iris, aes(Sepal.Length, Sepal.Width, colour = Species)) +
  geom_point() +
  scale_color_biopalette("three_body")

ggplot(iris, aes(Species, Sepal.Length, fill = Species)) +
  geom_boxplot() +
  scale_fill_biopalette("three_body", guide = "none")
```

scale_color_biopalette_gradient

Continuous ggplot2 Gradients from a biopalette Palette

Description

Build a continuous colour or fill gradient from a sequential or diverging biopalette palette. The complete palette supplies the gradient stops to [ggplot2::scale_colour_gradientn\(\)](#) or [ggplot2::scale_fill_gradientn\(\)](#). Gradients are interpolated in Lab colour space, the same space used when [get_palette\(\)](#) resamples sequential and diverging palettes.

Usage

```
scale_color_biopalette_gradient(
  palette,
  ...,
  type = NULL,
  reverse = FALSE,
  values = NULL,
  midpoint = NULL,
  transform = "identity",
  palettes_dir = NULL
)

scale_colour_biopalette_gradient(
  palette,
  ...,
  type = NULL,
  reverse = FALSE,
  values = NULL,
  midpoint = NULL,
  transform = "identity",
  palettes_dir = NULL
)
```

```

scale_fill_biopalette_gradient(
  palette,
  ...,
  type = NULL,
  reverse = FALSE,
  values = NULL,
  midpoint = NULL,
  transform = "identity",
  palettes_dir = NULL
)

```

Arguments

palette	Character. Name of the palette.
...	Passed to <code>ggplot2::scale_colour_gradientn()</code> or <code>ggplot2::scale_fill_gradientn()</code> , for example name, breaks, labels, limits, transform, na.value, and guide.
type	Character. "sequential" or "diverging". If NULL, the type is detected automatically.
reverse	Logical. Reverse the palette. When values is supplied, its spacing is reversed with the colours. Default: FALSE.
values	Optional numeric positions for the palette colours, as accepted by <code>ggplot2::scale_colour_gradientn()</code> . Must have one value per palette colour, be non-decreasing, and lie in $[0, 1]$.
midpoint	Optional finite numeric value to place at the visual centre of a diverging palette. Values are rescaled with equal units on either side of this point, so the shorter side of an asymmetric data range does not reach the palette's extreme colour. midpoint cannot be combined with values or a custom rescaler. The default, NULL, applies ordinary range-based scaling.
transform	A transformation specification accepted by <code>ggplot2::continuous_scale()</code> . The same transformation is applied to midpoint before the gradient is rescaled, so the centre remains correct under transformations such as "log10". Default: "identity".
palettes_dir	Character. Directory holding a palette collection. If NULL, the palettes bundled with the package are used.

Details

Qualitative palettes are rejected because interpolating unordered category colours does not produce a meaningful continuous scale. Use `scale_color_biopalette()` or `scale_fill_biopalette()` for those palettes.

Value

A ggplot2 continuous scale.

Examples

```
library(ggplot2)

ggplot(mtcars, aes(wt, mpg, colour = hp)) +
  geom_point(size = 3) +
  scale_color_biopalette_gradient("mitonuclear_blue")

ggplot(mtcars, aes(factor(cyl), factor(gear), fill = mpg - mean(mpg))) +
  geom_tile() +
  scale_fill_biopalette_gradient("walter_white", midpoint = 0)
```

Index

- * **biomedical**
 - biopalette, [2](#)
- * **color**
 - biopalette, [2](#)
- * **palette**
 - biopalette, [2](#)
- * **visualization**
 - biopalette, [2](#)

biopalette, [2](#)
biopalette-package (biopalette), [2](#)

create_palette, [3](#)

get_palette, [4](#)
get_palette(), [7](#), [8](#), [10](#), [12](#)
ggplot2::continuous_scale(), [13](#)
ggplot2::discrete_scale(), [11](#)
ggplot2::scale_colour_gradientn(), [12](#),
[13](#)
ggplot2::scale_fill_gradientn(), [12](#), [13](#)

hex2rgb, [5](#)
hex2rgb(), [10](#)

list_palettes, [6](#)
list_palettes(), [7](#)

palette_gallery, [6](#)
palette_gallery(), [8](#), [9](#)
palette_info, [7](#)
preview_palette, [8](#)

remove_palette, [9](#)
rgb2hex, [10](#)

scale_color_biopalette, [10](#)
scale_color_biopalette(), [13](#)
scale_color_biopalette_gradient, [12](#)
scale_colour_biopalette
(scale_color_biopalette), [10](#)
scale_colour_biopalette_gradient
(scale_color_biopalette_gradient),
[12](#)
scale_fill_biopalette
(scale_color_biopalette), [10](#)
scale_fill_biopalette(), [13](#)
scale_fill_biopalette_gradient
(scale_color_biopalette_gradient),
[12](#)