# Package 'qualpalr'

July 22, 2025

**Title** Automatic Generation of Qualitative Color Palettes

**Version** 0.4.4

**Description** Automatic generation of maximally distinct qualitative color palettes, optionally tailored to color deficiency. A list of colors or a subspace of a color space is used as input and then projected to the DIN99d color space, where colors that are maximally distinct are chosen algorithmically.

**Depends** R (>= 3.3.0)

**Imports** randtoolbox (>= 1.17), graphics, stats, grDevices, utils, assertthat, Rcpp

**Suggests** testthat, knitr, maps, rmarkdown, rgl, spelling, covr

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**URL** https://jolars.github.io/qualpalr/

**BugReports** https://github.com/jolars/qualpalr/issues

**VignetteBuilder** knitr

**LinkingTo** Rcpp (>= 0.12.9), RcppArmadillo (>= 0.7.600.1.0)

**Language** en-US

**NeedsCompilation** yes

**Author** Johan Larsson [aut, cre] (ORCID: <https://orcid.org/0000-0002-4029-5945>)

**Maintainer** Johan Larsson <johanlarsson@outlook.com>

**Repository** CRAN

**Date/Publication** 2023-09-13 07:30:02 UTC

## Contents

---

`autopal` *Optimize color palette for color vision deficiency*

---

## Description

This function adapts color palettes to color vision deficiency (CVD) by optimizing the CVD severity to try reach a target color difference (DIN99d $\delta E$) of the user's choosing. Basically, it will choose a color palette that is as close as possible to the target $\delta E$ by tweaking the CVD severity option in `qualpal`.

## Usage

```
autopal(
  n,
  colorspace = "pretty",
  cvd = c("protan", "deutan", "tritan"),
  target = 20
)
```

## Arguments

| | |
|---|---|
| `n` | Number of colors to generate. |
| `colorspace` | Either 1) a list of three named numeric vectors: `h` (hue), `s` (saturation), and `l` (lightness), all of length 2 specifying a min and max value for the range. The values has to be in the range -360 to 360 for `h`, and 0 to 1 for `s` and `l` 2), or 2) a *character vector* specifying one of the predefined color spaces (see below). |
| `cvd` | Color vision deficiency adaptation to adapt the color palette to. |
| `target` | Target color difference. |

## Details

The rationale for this function is that when there are few colors in a color palette, there is no cost involved in adapting colors to CVD – the colors will still remain distinct. As more an more colors are added to the palette, however, adapting the color palette to CVD will eventually lead to colors that are too similar. This function gradually loosens the adaptation to CVDs by lowering the severity of CVD to simulate to before picking colors (the `cvd_severity` argument in `qualpal`)

## Value

A list of class `qualpal` with the following components.

| | |
|---|---|
| `HSL` | A matrix of the colors in the HSL color space. |
| `DIN99d` | A matrix of the colors in the DIN99d color space (after power transformations). |
| `RGB` | A matrix of the colors in the sRGB color space. |
| `hex` | A character vector of the colors in hex notation. |
| `de_DIN99d` | A distance matrix of color differences according to delta E DIN99d. |
| `min_de_DIN99d` | The smallest pairwise DIN99d color difference. |

## Examples

```
pal <- autopal(3, cvd = "protan", target = 15)
plot(pal)
```

---

# `pairs.qualpal` *Scatterplot matrix of qualitative color palette*

---

## Description

Plots the colors in an object of class `"qualpal"` as a scatterplot matrix on either the DIN99d (the default) or HSL color space.

## Usage

```
## S3 method for class 'qualpal'
pairs(x, colorspace = c("DIN99d", "HSL", "RGB"), ...)
```

## Arguments

| | |
|---|---|
| `x` | A list object of class `"qualpal"` generated from `qualpal`. |
| `colorspace` | The color space in which to plot the colors ("DIN99d", "HSL", or "RGB"). |
| `...` | Arguments to pass on to `pairs`. |

## See Also

`qualpal`, `plot.qualpal`, `pairs`

## Examples

```
col_pal <- qualpal(3)
pairs(col_pal)
pairs(col_pal, colorspace = "HSL")
```

## plot.qualpal *Multidimensional scaling map of qualitative color palette*

### Description

Uses the colors in a qualpal object to compute and plot a multidimensional scaling (MDS) map using cmdscale on the Delta E DIN99d distance matrix.

### Usage

```
## S3 method for class 'qualpal'
plot(x, ...)
```

### Arguments

| | |
|---|---|
| x | An object of class "qualpal" generated from qualpal. |
| ... | Arguments to pass on to plot. |

### See Also

qualpal, pairs.qualpal, plot

### Examples

```
col_pal <- qualpal(3)
plot(col_pal)
```

## print.qualpal *Print qualpal palette*

### Description

Print the result from a call to qualpal.

### Usage

```
## S3 method for class 'qualpal'
print(x, colorspace = c("HSL", "DIN99d", "RGB"), digits = 2, ...)
```

### Arguments

| | |
|---|---|
| x | An object of class "qualpal". |
| colorspace | Color space to print colors in. |
| digits | Number of significant digits for the output. (See print.default.) Setting it to NULL uses getOption("digits"). |
| ... | Arguments to pass to print.default. |

## Value

Prints the colors as a matrix in the specified color space as well as a distance matrix of the color differences. Invisibly returns x.

## Examples

```
f <- qualpal(3)
print(f, colorspace = "DIN99d", digits = 3)
```

---

`qualpal` *Generate qualitative color palettes*

---

## Description

Given a color space or collection of colors, `qualpal()` projects these colors to the DIN99d color space, where it generates a color palette from the most visually distinct colors, optionally taking color vision deficiency into account.

## Usage

```
qualpal(
  n,
  colorspace = "pretty",
  cvd = c("protan", "deutan", "tritan"),
  cvd_severity = 0,
  n_threads
)
```

## Arguments

`n` The number of colors to generate.

`colorspace` A color space to generate colors from. Can be any of the following:

- A `list` with the following *named* vectors, each of length two, giving a range for each item.
  - `h` Hue, in the range [-360, 360]
  - `s` Saturation, in the range [0, 1]
  - `l` Lightness, in the range [0, 1]
- A `character` vector of length one specifying one of these predefined color spaces:
  - `pretty` Tries to provide aesthetically pleasing, but still distinct color palettes. Hue ranges from 0 to 360, saturation from 0.1 to 0.5, and lightness from 0.5 to 0.85. This palette is not suitable for high `n`
  - `pretty_dark` Like `pretty` but darker. Hue ranges from 0 to 360, saturation from 0.1 to 0.5, and lightness from 0.2 to 0.4.
  - `rainbow` Uses all hues, chromas, and most of the lightness range. Provides distinct but not aesthetically pleasing colors.

`pastels` Pastel colors from the complete range of hues (0-360), with saturation between 0.2 and 0.4, and lightness between 0.8 and 0.9.

- A `matrix` of colors from the sRGB color space, each row representing a unique color.
- A `data.frame` that can be converted to a matrix via data.matrix

| | |
|---|---|
| `cvd` | Color vision deficiency adaptation. Use `cvd_severity` to set the severity of color vision deficiency to adapt to. Permissible values are `"protan"`, `"deutan"`, and `"tritan"`. |
| `cvd_severity` | Severity of color vision deficiency to adapt to. Can take any value from 0, for normal vision (the default), and 1, for dichromatic vision. |
| `n_threads` | Previously the number of threads to use, but this argument is now deprecated. |

## Details

The function takes a color subspace in the HSL color space, where lightness and saturation take values from 0 to 1. Hue take values from -360 to 360, although negative values are brought to lie in the range {0, 360}; this behavior exists to enable color subspaces that span all hues being that the hue space is circular.

The HSL color subspace that the user provides is projected into the DIN99d color space, which is approximately perceptually uniform, i.e. color difference is proportional to the euclidean distance between two colors. A distance matrix is computed and, as an additional step, is transformed using power transformations discovered by Huang 2015 in order to fine tune differences.

`qualpal` then searches the distance matrix for the most distinct colors; it does this iteratively by first selecting a random set of colors and then iterates over each color, putting colors back into the total set and replaces it with a new color until it has gone through the whole range without changing any of the colors.

Optionally, `qualpal` can adapt palettes to cater to color vision deficiency (cvd). This is accomplished by taking the colors provided by the user and transforming them to colors that someone with cvd would see, that is, simulating cvd. qualpal then chooses colors from these new colors.

`qualpal` currently only supports the sRGB color space with the D65 white point reference.

## Value

A list of class `qualpal` with the following components.

| | |
|---|---|
| `HSL` | A matrix of the colors in the HSL color space. |
| `DIN99d` | A matrix of the colors in the DIN99d color space (after power transformations). |
| `RGB` | A matrix of the colors in the sRGB color space. |
| `hex` | A character vector of the colors in hex notation. |
| `de_DIN99d` | A distance matrix of color differences according to delta E DIN99d. |
| `min_de_DIN99d` | The smallest pairwise DIN99d color difference. |

## See Also

`plot.qualpal`, `pairs.qualpal`

## Examples

```
# Generate 3 distinct colors from the default color space
qualpal(3)

# Provide a custom color space
qualpal(n = 3, list(h = c(35, 360), s = c(0.5, 0.7), l = c(0, 0.45)))

qualpal(3, "pretty")

# Adapt palette to deuteranopia
qualpal(5, colorspace = "pretty_dark", cvd = "deutan", cvd_severity = 1)

# Adapt palette to protanomaly with severity 0.4
qualpal(8, colorspace = "pretty_dark", cvd = "protan", cvd_severity = 0.4)

## Not run:
# The range of hue cannot exceed 360
qualpal(3, list(h = c(-20, 360), s = c(0.5, 0.7), l = c(0, 0.45)))

## End(Not run)
```

# Index