

Package: ROvis.utils (via r-universe)

October 21, 2026

Title Utilities for Rijksoverheid-Styled Data Visualizations

Version 0.1.0

Description Provides shared utilities for the ROvis suite, including Rijksoverheid color palettes, palette validation, and font-availability checks, so that themes across the ROvis packages stay consistent with official Rijksoverheid design guidelines.

License EUPL (>= 1.2)

Depends R (>= 4.5.0)

Imports cli, dplyr, ggplot2, magrittr (>= 2.0.3), rlang (>= 1.1.0), plotly, scales, stringr, systemfonts, tibble

Suggests covr (>= 3.6.4), pkgdown (>= 2.0.9), quarto (>= 1.4), remotes, knitr, lubridate, tidyr, testthat (>= 3.0.0)

VignetteBuilder quarto

Config/testthat/edition 3

Encoding UTF-8

Config/roxygen2/markdown TRUE

Config/roxygen2/version 8.1.0

Config/pak/sysreqs cmake libfontconfig1-dev libfreetype6-dev make libicu-dev libuv1-dev libssl-dev

Repository <https://rivm-syso.r-universe.dev>

Date/Publication 2026-10-21 07:22:35 UTC

RemoteUrl <https://github.com/rivm-syso/ROvis.utils>

RemoteRef HEAD

RemoteSha 6d7e97dfabbce9cbca06b0d21514a446169b7237

Contents

ro_check_if_font_available	2
ro_color	3
ro_color_categorical	4

ro_color_palette	5
ro_color_seq	6
ro_color_seq_named	7
ro_get_color_palette	8
ro_group_badge	9
ro_show_colors	10
Index	12

ro_check_if_font_available
<i>Check whether a font family is available on the system</i>

Description

[DT][echarts4r][ggplot2][gt][plotly]

Used by ro_gg_theme and ro_e_theme to resolve the font family before building a chart. When the RijksoverheidSansWebText font is requested but not installed, the function tries "Verdana", then "Arial", and only uses one that is actually installed. If neither is available either (for example on Linux/CI images that ship none of them), the function informs the user and falls back to the first font it can find on the system. When any other font is requested but not installed, the function aborts with an actionable error message.

Usage

```
ro_check_if_font_available(  
  target_font_family,  
  arg = caller_arg(target_font_family),  
  call = caller_env()  
)
```

Arguments

target_font_family	Character. Font family name to check. No default: this is a validation function, so a sensible default (e.g. "RijksoverheidSansWebText") belongs in the calling plotting functions (e.g. ro_gg_theme(), ro_ply_theme()), not here.
arg, call	Passed on to <code>rlang::abort()/cli::cli_abort()</code> so that messages cite the argument name used by the calling function (e.g. base_family in ro_gg_theme()) instead of this function's own target_font_family parameter. Only relevant for programmatic callers that want to override the automatically detected argument name.

Value

The resolved font family name: target_font_family itself if installed; otherwise, for the RO font, the first available font among "Verdana" and "Arial", or (if neither is installed) the first font found on the system. Aborts for any other unavailable font.

See Also

Other ggplot2: [get_color_palette\(\)](#), [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_color_validate\(\)](#), [ro_get_color_palette\(\)](#), [ro_gg_create_palette\(\)](#), [ro_show_colors\(\)](#), [ro_show_colors_get_df\(\)](#)

Other ggplotly: [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_get_color_palette\(\)](#)

Other plotly: [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_get_color_palette\(\)](#)

Other echarts4r: [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#)

ro_color

Return hex code of RIVM color

Description

[DT][echarts4r][ggplot2][gt][plotly]

Returns hex code(s) of color_name. All 18 RIVM colors and their lighter shades can be accessed with this function. The names of RIVM colors are listed at the [rijkshuisstijl](#) website, or, can be viewed by running another function, [ro_show_colors](#).

To access a vector of hex color codes of the categorical palette, use the ROvis function [ro_color_categorical](#).

Usage

```
ro_color(color_name = "hemelblauw")
```

Arguments

- | | |
|------------|--|
| color_name | <p>Character or vector of characters. Default="hemelblauw". Can be one of:</p> <ul style="list-style-type: none"> • The name of an RIVM color, e.g. "robijnrood". All colors come at different shades: 'tint15', 'tint30', 'tint45', 'tint60', or 'tint75'. A color shade can be accessed by combining the color name with the suffix "_tintXX". The seven grey tints are available as 'grijs_1' (lightest) to 'grijs_7' (darkest). • A character vector of RIVM color names. Example: c("hemelblauw", "donkergeel") |
|------------|--|

Value

returns the (array of) hex color code(s)

See Also

Other ggplot2: [get_color_palette\(\)](#), [ro_check_if_font_available\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_color_validate\(\)](#), [ro_get_color_palette\(\)](#), [ro_gg_create_palette\(\)](#), [ro_show_colors\(\)](#), [ro_show_colors_get_df\(\)](#)

Other ggplotly: [ro_check_if_font_available\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_get_color_palette\(\)](#)

Other plotly: [ro_check_if_font_available\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_get_color_palette\(\)](#)

Other echarts4r: [ro_check_if_font_available\(\)](#), [ro_color_categorical\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#)

Examples

```
ro_color()
ro_color("robinrood")
ro_color("donkergeel_tint75")
ro_color("grijs_3")
ro_color(c("lintblauw", "robinrood", "mintgroen_tint30"))
```

`ro_color_categorical` *Show hex codes of RIVM categorical color palette*

Description

[DT][echarts4r][ggplot2][gt][plotly]

Returns a vector of hex color codes of the categorical RIVM color palette. This is especially useful when you don't need to assign a specific color to a specific category. If you do, for example for male/female, we recommend to assign the colors explicitly using the [ro_color](#) function.

Usage

```
ro_color_categorical()
```

Value

Returns a character vector of hex color codes of the categorical palette.

See Also

Other ggplotly: [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_get_color_palette\(\)](#)

Other plotly: [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_get_color_palette\(\)](#)

Other echarts4r: [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#)

Other ggplot2: [get_color_palette\(\)](#), [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_color_validate\(\)](#), [ro_get_color_palette\(\)](#), [ro_gg_create_palette\(\)](#), [ro_show_colors\(\)](#), [ro_show_colors_get_df\(\)](#)

Examples

```
ro_color_categorical()
```

ro_color_palette	<i>Return named list with hexcodes RIVM palette</i>
------------------	---

Description

[echarts4r][ggplot2][plotly]

Usage

```
ro_color_palette(palette_name)
```

Arguments

palette_name one of:

- "full": all 18 RIVM colors (full color, 100% tint)
- "categorical": best palette for visualizing categories (no variation in tint, good contrast differences)
- "gender_con": gender conventional
- "gender_unc": gender unconventional
- "greys": all 7 grey tints

Value

named vector with hex color codes

Dependencies

- [ro_color](#)

See Also

Other ggplot2: [get_color_palette\(\)](#), [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_color_validate\(\)](#), [ro_get_color_palette\(\)](#), [ro_gg_create_palette\(\)](#), [ro_show_colors\(\)](#), [ro_show_colors_get_df\(\)](#)

Other ggplotly: [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_get_color_palette\(\)](#)

Other plotly: [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_get_color_palette\(\)](#)

Examples

```
ro_color_palette("categorical")
```

ro_color_seq

Return sequence of hex color codes between two diverging colors

Description

[DT][echarts4r][ggplot2][gt][plotly]

Useful to visualize ordered categorical data, for example regions on a map. The function takes two hex color codes and returns a color sequence of the desired length where these colors gradually fade into one another. This could be a transition from a light to a dark shade, or one color to a different color, see examples. Uses `pal_seq_gradient()` from the `scales` package under the hood.

To access the hex color code of a specific color, use the [ro_color](#) function.

Usage

```
ro_color_seq(low, high, n)
```

Arguments

low	Character. The hexadecimal code (#RRGGBB) of the color for 'low' values. This color will be the first hex code of the sequence.
high	Character. The hexadecimal code (#RRGGBB) of the color for 'high' values. This color will be the last hex code of the sequence.
n	Integer. Desired amount of colors in the sequence (including the specified colors).

Value

sequence of hex color codes sorted from the 'low' to the 'high' color.

See Also

Other ggplotly: [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq_named\(\)](#), [ro_get_color_palette\(\)](#)

Other plotly: [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq_named\(\)](#), [ro_get_color_palette\(\)](#)

Other echarts4r: [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_seq_named\(\)](#)

Other ggplot2: [get_color_palette\(\)](#), [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq_named\(\)](#), [ro_color_validate\(\)](#), [ro_get_color_palette\(\)](#), [ro_gg_create_palette\(\)](#), [ro_show_colors\(\)](#), [ro_show_colors_get_df\(\)](#)

Examples

```
ro_color_seq("#F7D9E7", "#CA005D", 5)
ro_color_seq(ro_color("robijnrood_tint15"), ro_color("robijnrood"), 7)
ro_color_seq(ro_color("hemelblauw_tint30"), ro_color("hemelblauw"), 4)
ro_color_seq(ro_color("hemelblauw"), ro_color("robijnrood"), 8)
```

ro_color_seq_named	<i>Set RIVM colors to specific levels to use in scale_color_manual for maps or other visualizations.</i>
--------------------	--

Description

[DT][echarts4r][ggplot2][gt][plotly]

A wrapper function that maps RIVM colors to specific levels/classes to use in `scale_color_manual` for maps or other visualizations. Note that this function is dependent on the functions `ro_color()` and `ro_color_seq()`.

Usage

```
ro_color_seq_named(
  cat_names,
  low_col = "robijnrood",
  high_col = "robijnrood_tint15",
  NA_cat_name = "NA",
  NA_cat_col = "grijs_5"
)
```

Arguments

cat_names	Character. Vector that holds categories.
low_col	Character. The hexadecimal code (#RRGGBB) of the color for 'low' values. This color will be the first hex code of the sequence. Default: "robijnrood".
high_col	Character. The hexadecimal code (#RRGGBB) of the color for 'high' values. This color will be the last hex code of the sequence. Default: "robijnrood_tint15".
NA_cat_name	Character, specified 'no-record' category to search for. Default: "Geen meldingen".
NA_cat_col	Character: sets the hexadecimal code (#RRGGBB) for the 'no_record' category. Default: "grijs_5".

See Also

Other ggplotly: `ro_check_if_font_available()`, `ro_color()`, `ro_color_categorical()`, `ro_color_palette()`, `ro_color_seq()`, `ro_get_color_palette()`

Other plotly: `ro_check_if_font_available()`, `ro_color()`, `ro_color_categorical()`, `ro_color_palette()`, `ro_color_seq()`, `ro_get_color_palette()`

Other echarts4r: `ro_check_if_font_available()`, `ro_color()`, `ro_color_categorical()`, `ro_color_seq()`

Other ggplot2: `get_color_palette()`, `ro_check_if_font_available()`, `ro_color()`, `ro_color_categorical()`, `ro_color_palette()`, `ro_color_seq()`, `ro_color_validate()`, `ro_get_color_palette()`, `ro_gg_create_palette()`, `ro_show_colors()`, `ro_show_colors_get_df()`

Examples

```
ro_color_seq_named(
  cat_names = c("<=99", "100 t/m 399", "400+", "Geen meldingen"),
  NA_cat_name = "Geen meldingen"
)

ro_color_seq_named(
  cat_names = c("<10", "10-15", "16+", "NA"),
  low_col = "lichtblauw",
  high_col = "hemelblauw",
  NA_cat_col = "rood"
)
```

`ro_get_color_palette` *Return named list with hexcodes RIVM palette*

Description

[echarts4r][ggplot2][plotly]

Usage

```
ro_get_color_palette(palette_name)
```

Arguments

`palette_name` one of:

- "full": all 18 RIVM colors (full color, 100% tint)
- "categorical": best palette for visualizing categories (no variation in tint, good contrast differences)
- "gender_con": gender conventional
- "gender_unc": gender unconventional
- "greys": all 7 grey tints

Value

named vector with hex color codes

Dependencies

- [ro_color](#)

See Also

Other ggplot2: [get_color_palette\(\)](#), [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_color_validate\(\)](#), [ro_gg_create_palette\(\)](#), [ro_show_colors\(\)](#), [ro_show_colors_get_df\(\)](#)

Other ggplotly: [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#)

Other plotly: [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#)

Examples

```
R0vis.utils::ro_get_color_palette("categorical")
```

ro_group_badge

Build backend group badge(s) for use in roxygen documentation

Description

Generates Rd markup for badge(s) indicating which plotting/output backend(s) (ggplot2, echarts4r, plotly, gt, or DT) a function belongs to. Mirrors the mechanism used by [badge](#): the HTML help page shows an SVG badge from `man/figures/`, while non-HTML help (e.g. the console) falls back to bracketed text. It is a documentation-time helper (not exported), used via ``r ro_group_badge("ggplot2")`` inside roxygen comments.

Usage

```
ro_group_badge(group)
```

Arguments

group	Character vector with one or more of "ggplot2", "echarts4r", "plotly", "gt", "DTdevto". When more than one is supplied, the badges are placed next to each other, sorted alphabetically (regardless of the order passed in).
-------	--

Value

A string containing Rd markup. Ends with `\cr` so any text following it in the same roxygen paragraph starts on a new line.

ro_show_colors	<i>Show RIVM color palette</i>
----------------	--------------------------------

Description

[DT][echarts4r][ggplot2][gt][plotly]

Plots the full RIVM color palette, a specified RIVM color palette, or a custom color palette. Hoover over the palette to view the hex color codes.

Usage

```
ro_show_colors(palette_name = "full", custom_colors = NULL)
```

Arguments

palette_name	Character. Default="full". Can be one of: • The name of an RIVM palette. Available palettes are: – full: all 18 RIVM colors (full color, 100% tint) – categorical: best palette for visualizing categories (no variation in tint, good contrast differences) – gender_con: gender conventional – gender_unc: gender unconventional – greys: all 7 grey tints • 'custom'. use this if you want to specify your own custom_colors.
custom_colors	Character or vector of characters. Should be hexadecimal code(s) (#RRGGBB). It is advised to use the other ROvis color functions. To access the hex color code of a specific color, use the ro_color function. To access a vector of hex color codes of the categorical palette, use the ROvis function ro_color_categorical .

Value

Color palette in the Viewer pane.

Dependencies

- [ro_color_palette](#)

See Also

Other ggplot2: [get_color_palette\(\)](#), [ro_check_if_font_available\(\)](#), [ro_color\(\)](#), [ro_color_categorical\(\)](#), [ro_color_palette\(\)](#), [ro_color_seq\(\)](#), [ro_color_seq_named\(\)](#), [ro_color_validate\(\)](#), [ro_get_color_palette\(\)](#), [ro_gg_create_palette\(\)](#), [ro_show_colors_get_df\(\)](#)

Examples

```
## Not run:
ro_show_colors()
ro_show_colors("categorical")
ro_show_colors("custom", c(ro_color("donkergeel"), ro_color("paars_tint90"), ro_color("mosgroen")))
ro_show_colors("custom", "#76d2b6")
ro_show_colors("custom", c(
  donkergeel = ro_color("donkergeel"), paars_tint90 = ro_color("paars_tint90"),
  mosgroen = ro_color("mosgroen")
))
ro_show_colors("custom", ro_color_seq(ro_color("hemelblauw_tint15"), ro_color("hemelblauw"), 5))
ro_show_colors("custom", ro_color_seq(ro_color("robijnrood_tint15"), ro_color("robijnrood"), 12))

## End(Not run)
```

Index

- * **DT**
 - ro_check_if_font_available, [2](#)
- * **echarts4r**
 - ro_check_if_font_available, [2](#)
 - ro_color, [3](#)
 - ro_color_categorical, [4](#)
 - ro_color_seq, [6](#)
 - ro_color_seq_named, [7](#)
- * **ggplot2**
 - ro_check_if_font_available, [2](#)
 - ro_color, [3](#)
 - ro_color_categorical, [4](#)
 - ro_color_palette, [5](#)
 - ro_color_seq, [6](#)
 - ro_color_seq_named, [7](#)
 - ro_get_color_palette, [8](#)
 - ro_show_colors, [10](#)
- * **ggplotly**
 - ro_check_if_font_available, [2](#)
 - ro_color, [3](#)
 - ro_color_categorical, [4](#)
 - ro_color_palette, [5](#)
 - ro_color_seq, [6](#)
 - ro_color_seq_named, [7](#)
 - ro_get_color_palette, [8](#)
- * **plotly**
 - ro_check_if_font_available, [2](#)
 - ro_color, [3](#)
 - ro_color_categorical, [4](#)
 - ro_color_palette, [5](#)
 - ro_color_seq, [6](#)
 - ro_color_seq_named, [7](#)
 - ro_get_color_palette, [8](#)

badge, [9](#)

cli::cli_abort(), [2](#)

get_color_palette(), [3-6, 8-10](#)

rlang::abort(), [2](#)

ro_check_if_font_available, [2](#)

ro_check_if_font_available(), [4-6, 8-10](#)

ro_color, [3, 4-6, 9, 10](#)

ro_color(), [3-6, 8-10](#)

ro_color_categorical, [3, 4, 10](#)

ro_color_categorical(), [3-6, 8-10](#)

ro_color_palette, [5, 10](#)

ro_color_palette(), [3-6, 8-10](#)

ro_color_seq, [6](#)

ro_color_seq(), [3-5, 8-10](#)

ro_color_seq_named, [7](#)

ro_color_seq_named(), [3-6, 9, 10](#)

ro_color_validate(), [3-6, 8-10](#)

ro_get_color_palette, [8](#)

ro_get_color_palette(), [3-6, 8, 10](#)

ro_gg_create_palette(), [3-6, 8-10](#)

ro_group_badge, [9](#)

ro_show_colors, [3, 10](#)

ro_show_colors(), [3-6, 8, 9](#)

ro_show_colors_get_df(), [3-6, 8-10](#)