

Package ‘Canton’

September 22, 2026

Title Consistent Fonts and Figure Export for Scientific Publication
Workflows

Version 0.0.8

Description Provides a unified interface for exporting figures created with base graphics, 'ggplot2', 'grid', 'pheatmap', and 'ComplexHeatmap' to PDF, PNG, JPEG, and TIFF files. Selects an appropriate rendering strategy based on the plot object and supports exporting a figure to multiple formats in one call. Also provides font configuration and diagnostics, reusable figure presets, publication-oriented themes, and colour palettes. The 'ggplot2' graphics framework is described by Wickham (2016, ISBN:978-3-319-24277-4).

License GPL (>= 3)

URL <https://github.com/ScienceAdvances/Canton>

BugReports <https://github.com/ScienceAdvances/Canton/issues>

Encoding UTF-8

Depends R (>= 3.5)

Imports ggplot2 (>= 3.4.0), systemfonts

Suggests ComplexHeatmap, pheatmap, testthat (>= 3.0.0)

Config/testthat/edition 3

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Tim Holy [aut, cre]

Maintainer Tim Holy <email@timholy.com>

Repository CRAN

Date/Publication 2026-09-22 06:30:02 UTC

Contents

canton_scales	2
figure_preset	3
fontcheck	4

fontlist	4
hue	5
imagesave	5
mkdir	7
pwd	7
resetfont	8
setfont	8
theme_canton	9
using	10
Index	11

canton_scales	<i>Canton colour scales for ggplot2</i>
---------------	---

Description

Apply a palette returned by [hue\(\)](#) to discrete or continuous ggplot2 colour and fill scales.

Usage

```
scale_colour_canton_d(palette = "NPG", ...)  
scale_color_canton_d(palette = "NPG", ...)  
scale_fill_canton_d(palette = "NPG", ...)  
scale_colour_canton_c(palette = "NPG", ...)  
scale_color_canton_c(palette = "NPG", ...)  
scale_fill_canton_c(palette = "NPG", ...)
```

Arguments

- palette Canton palette name.
- ... Additional arguments passed to the corresponding ggplot2 scale.

Value

A ggplot2 scale object.

figure_preset	Create a figure export preset
---------------	-------------------------------

Description

Create reusable dimensions and output settings for `imagesave()`. Presets are starting points rather than journal-specific guarantees; always compare them with the current author instructions for the target journal.

Usage

```
figure_preset(
  name = c("publication", "single_column", "double_column", "high_resolution",
    "presentation"),
  format = NULL,
  width = NULL,
  height = NULL,
  units = NULL,
  dpi = NULL,
  bg = NULL,
  quality = NULL,
  compression = NULL,
  pointsize = NULL
)
```

Arguments

name	Built-in preset name: "publication", "single_column", "double_column", "high_resolution", or "presentation".
format, width, height, units, dpi, bg, quality, compression, pointsize	Optional values that override the built-in preset.

Value

An object of class `canton_figure_preset`.

Examples

```
figure_preset("single_column")
figure_preset("publication", format = c("pdf", "png"), dpi = 600)
```

fontcheck	<i>Check whether a plotting font is available</i>
-----------	---

Description

Check whether a plotting font is available

Usage

```
fontcheck(family = "Arial", error = FALSE, quiet = FALSE)
```

Arguments

family	Font family to check.
error	Whether to throw an informative error when the font is missing.
quiet	Whether to suppress the status message.

Value

A logical scalar.

Examples

```
fontcheck("sans", quiet = TRUE)
```

fontlist	<i>List available font families</i>
----------	-------------------------------------

Description

List font families detected by the operating system and by the systemfonts registry.

Usage

```
fontlist(pattern = NULL)
```

Arguments

pattern	Optional regular expression used to filter family names.
---------	--

Value

A sorted character vector of font family names.

Examples

```
head(fontlist(), 10)  
fontlist("Arial")
```

hue	<i>Return a Canton colour palette</i>
-----	---------------------------------------

Description

Return a named colour palette. Call `hue()` without a name to list the available palettes.

Usage

```
hue(name = NULL)
```

Arguments

name	Palette name. Matching is case-insensitive. If <code>NULL</code> , list the available palette names.
------	--

Value

A character vector of colours, or invisibly a character vector of available palette names when `name` is `NULL`.

Examples

```
hue("NPG")
hue("dark2")
```

imagesave	<i>Save plots in one or more image formats</i>
-----------	--

Description

`imagesave()` saves `ggplot2` plots, `pheatmap` and `ComplexHeatmap` objects, grid graphical objects, recorded base R plots, lattice plots, and plotting functions. When `plot` is `NULL`, the current plot is captured with `grDevices::recordPlot()`.

Usage

```
imagesave(
  plot = NULL,
  name = "plot",
  outdir,
  format = "pdf",
  preset = NULL,
  width = 9,
  height = 9,
  units = "in",
```

```

    dpi = 300,
    bg = "white",
    quality = 95,
    compression = "lzw",
    pointsize = 12,
    family = base::getOption("Canton.font_family", NULL),
    overwrite = TRUE
  )

```

Arguments

plot	A plot object or a zero-argument function that draws a plot. Supported objects include <code>ggplot</code> , <code>pheatmap</code> , <code>ComplexHeatmap</code> <code>Heatmap</code> and <code>HeatmapList</code> objects, <code>grob/gtable</code> , <code>recordedplot</code> , and <code>trellis</code> . If <code>NULL</code> , capture the current plot.
name	Output filename stem. A supported extension in name is used when format is omitted.
outdir	Required output directory. No files are written unless an explicit, non-empty directory is supplied.
format	One or more of "pdf", "png", "jpg", "jpeg", "tif", or "tiff".
preset	Optional built-in preset name or an object returned by <code>figure_preset()</code> . Explicitly supplied arguments override preset values.
width, height	Plot dimensions.
units	Units for width and height: "in", "cm", or "mm".
dpi	Resolution for raster formats.
bg	Background colour.
quality	JPEG quality from 0 to 100.
compression	TIFF compression method.
pointsize	Default text point size for graphics devices.
family	Optional font family passed to graphics devices.
overwrite	Whether existing files may be overwritten.

Details

`ggplot2` and grid-based objects are saved with `ggplot2::ggsave()`. Recorded plots and plotting functions are redrawn on a format-specific graphics device.

Value

Invisibly, a named character vector containing the output paths.

Examples

```
p <- ggplot2::ggplot(mtcars, ggplot2::aes(mpg, wt)) +  
  ggplot2::geom_point()  
imagesave(p, "scatter", outdir = tempdir(), format = c("png", "pdf"))  
  
imagesave(function() plot(mtcars$mpg, mtcars$wt),  
          name = "base-plot", outdir = tempdir(), format = "png")
```

mkdir	Create directories
-------	--------------------

Description

Create one or more directories recursively. Existing directories are left unchanged, making the function safe to call repeatedly.

Usage

```
mkdir(directory)
```

Arguments

directory One or more directory paths.

Value

Invisibly, normalized directory paths.

Examples

```
mkdir(file.path(tempdir(), "canton-example"))
```

pwd	Return the current working directory
-----	--------------------------------------

Description

Return the absolute path of the current working directory.

Usage

```
pwd()
```

Value

A character scalar containing the current working directory.

Examples

```
pwd()
```

resetfont*Compatibility helper for scoped font settings*

Description

Font settings now restore automatically when the code block in `setfont()` exits. This compatibility helper does not modify user settings.

Usage

```
resetfont(quiet = FALSE)
```

Arguments

`quiet` Suppress the status message.

Value

Invisibly, FALSE; no persistent settings need restoring.

Examples

```
resetfont(quiet = TRUE)
```

setfont*Configure a font locally*

Description

Without code, validate and return a font family without changing settings. With code, apply the font during evaluation and restore the previous options, ggplot2 theme, and existing device's font on exit, including errors. No graphics device is opened and no persistent plotting hook is installed. Print ggplot objects inside the block to render them with the temporary theme.

Usage

```
setfont(family = "Arial", fallback = NULL, quiet = FALSE, code)
```

Arguments

family	Font family. Defaults to "Arial".
fallback	Optional available fallback, used with a warning.
quiet	Suppress the status message.
code	Optional expression evaluated with temporary font settings. For base graphics, open an explicit device before calling this function.

Value

Without code, invisibly the validated family; otherwise the result of evaluating code.

Examples

```
family <- setfont("sans", quiet = TRUE)
theme_canton(base_family = family)
setfont("sans", quiet = TRUE, code = {
  theme_canton()
})
```

theme_canton	<i>A publication-oriented ggplot2 theme</i>
--------------	---

Description

A compact classic theme with explicit font, line, tick, legend, and grid defaults suitable as a starting point for scientific figures.

Usage

```
theme_canton(
  base_size = 10,
  base_family = base::getOption("Canton.font_family", "sans"),
  grid = c("none", "major", "both"),
  legend_position = "right"
)
```

Arguments

base_size	Base font size in points.
base_family	Base font family. Defaults to the temporary family inside a <code>setfont()</code> code block, or "sans" otherwise.
grid	Which panel grid lines to show: "none", "major", or "both".
legend_position	ggplot2 legend position.

Value

A ggplot2 theme object.

Examples

```
ggplot2::ggplot(mtcars, ggplot2::aes(mpg, wt)) +  
  ggplot2::geom_point() +  
  theme_canton()
```

using

Load multiple packages quietly

Description

Attach one or more installed packages while suppressing startup messages. Package names may be supplied as bare names, strings, or character vectors.

Usage

```
using(...)
```

Arguments

... Package names supplied as bare names, character strings, or character vectors.

Value

Invisibly, a named logical vector indicating which packages were loaded successfully.

Examples

```
using(ggplot2)  
using(c("ggplot2", "grid"))
```

Index

canton_scales, [2](#)

figure_preset, [3](#)
figure_preset(), [6](#)
fontcheck, [4](#)
fontlist, [4](#)

ggplot2::ggsave(), [6](#)
grDevices::recordPlot(), [5](#)

hue, [5](#)
hue(), [2](#)

imagesave, [5](#)
imagesave(), [3](#)

mkdir, [7](#)

pwd, [7](#)

resetfont, [8](#)

scale_color_canton_c(canton_scales), [2](#)
scale_color_canton_d(canton_scales), [2](#)
scale_colour_canton_c(canton_scales), [2](#)
scale_colour_canton_d(canton_scales), [2](#)
scale_fill_canton_c(canton_scales), [2](#)
scale_fill_canton_d(canton_scales), [2](#)
setfont, [8](#)
setfont(), [8](#), [9](#)

theme_canton, [9](#)

using, [10](#)