Package: washi (via r-universe)

October 22, 2026

Title Washington Soil Health Initiative Branding

Version 0.2.1.9000

Description Create plots and tables in a consistent style with WaSHI (Washington Soil Health Initiative) branding. Use 'washi' to easily style your 'ggplot2' plots and 'flextable' tables.

License MIT + file LICENSE

URL https://github.com/WA-Department-of-Agriculture/washi, https://wa-department-of-agriculture.github.io/washi/

BugReports https://github.com/WA-Department-of-Agriculture/washi/issues/

Depends R (>= 4.1.0)

Imports cli, flextable, ggplot2, graphics, grDevices, officer, scales, systemfonts

Suggests covr, forcats, ragg, testthat (>= 3.0.0), vdiffr

Config/testthat/edition 3

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Config/pak/sysreqs libcairo2-dev cmake libfontconfig1-dev libfreetype6-dev libfribidi-dev make libharfbuzz-dev libicu-dev libjpeg-dev libpng-dev libtiff-dev libuv1-dev libwebp-dev libxml2-dev libssl-dev

Repository https://wa-department-of-agriculture.r-universe.dev

Date/Publication 2025-09-16 19:22:57 UTC

RemoteUrl https://github.com/WA-Department-of-Agriculture/washi

RemoteRef HEAD

RemoteSha d2d4f884e8ad015e376cc14b430b48284062b15e

Contents

example_data_long . . . 2
example_data_wide . . . 3
washi_flextable . . . 3
washi_install_fonts . . . 4
washi_pal . . . 5
washi_pal_setup . . . 5
washi_pal_view . . . 6
washi_scale . . . 7
washi_theme . . . 8

Index **11**

`example_data_long` *Example data in long (tidy) format*

Description

A subset WaSHI State of the Soils Assessment dataset that has been anonymized. This dataset is tidied, so each measurement is in its own row.

Usage

```
example_data_long
```

Format

`example_data_long` **A data frame with 1,800 rows and 14 columns::**

year Year of sample

sample_id, producer_id, field_id Anonymized IDs

farm_name, producer_name, field_name Anonymized names

longitude, latitude Truncated coordinates

texture Measured soil texture

measurement Measurement name with units

value Measurement result ...

Source

https://agr.wa.gov/departments/land-and-water/natural-resources/soil-health/state-of-the-soils

example_data_wide *Example data in wide format*

Description

A subset WaSHI State of the Soils Assessment dataset that has been anonymized. This dataset presents each sample in its own row, with columns for each measurement.

Usage

```
example_data_wide
```

Format

`example_data_wide` **A data frame with 30 rows and 72:**

columns:

year Year of sample

sample_id, producer_id, field_id Anonymized IDs

farm_name, producer_name, field_name Anonymized names

longitude, latitude Truncated coordinates

texture Measured soil texture

other columns Column name includes measurement and units; value is the measurement results

...

Source

https://agr.wa.gov/departments/land-and-water/natural-resources/soil-health/state-of-the-soils

washi_flextable *WaSHI flextable style*

Description

Creates a flextable in WaSHI's style.

Usage

```
washi_flextable(
  data,
  cols_bold = NULL,
  header_font = "Lato",
  body_font = "Poppins",
  header_font_color = "white",
  header_bg_color = washi_pal[["standard"]][["green"]],
  border_color = washi_pal[["standard"]][["tan"]]
)
```

Arguments

`data`	Dataframe for the table.
`cols_bold`	Numeric indices of columns to bold. Defaults to `NULL`.
`header_font`	Font family for header text. Defaults to "Lato".
`body_font`	Font family for table body text. Defaults to "Poppins".
`header_font_color`	Hexcode color for header font. Defaults to white.
`header_bg_color`	Hexcode color for header background. Defaults to WaSHI green.
`border_color`	Hexcode color for horizontal borders. Defaults to WaSHI tan.

Value

A flextable formatted in WaSHI's style.

Examples

```
subset(
  example_data_wide,
  select = c(
    "sample_id",
    "county",
    "crop",
    "total_n_percent",
    "total_c_percent"
  )
) |>
  washi_flextable(cols_bold = 1)
```

`washi_install_fonts` *Install Lato and Poppins fonts on your system*

Description

Opens the font files to install on your computer.

Usage

```
washi_install_fonts()
```

Value

Opens a directory containing font files.

washi_pal *WaSHI color palettes*

Description

Color palettes are stored in a named list.

Usage

```
washi_pal
```

Format

An object of class `list` of length 11.

Value

List of available `washi` color palettes

See Also

Other color palette functions: `washi_pal_setup()`, `washi_pal_view()`

Examples

```
# List names of available palettes
names(washi_pal)

# Get hex codes from a palette using dollar `$name` or
# double bracket
# `[["name"]]` operators for extracting list elements
washi_pal$standard

washi_pal[["green_gradient"]]

# Extract a color from the standard WaSHI palette
washi_pal[["standard"]][["green"]]
```

washi_pal_setup *Setup a color palette*

Description

Choose desired number of colors and whether the colors are reversed.

Usage

```
washi_pal_setup(palette = "standard", n, reverse = FALSE)
```

Arguments

palette	Character name of palette in washi_pal. See names(washi_pal) for a list of available palettes.
n	Number of colors in palette.
reverse	Boolean indicating whether the palette should be reversed. Default is FALSE.

Value

A vector of color hex codes.

See Also

Other color palette functions: washi_pal, washi_pal_view()

Examples

```
washi_pal_setup("color_blind")

washi_pal_setup("green_gradient", 12)
```

---

washi_pal_view *View a WaSHI palette*

Description

Show the colors within a palette in a plot.

Usage

```
washi_pal_view(palette = "color_blind", n, reverse = FALSE)
```

Arguments

palette	Character name of palette in washi_pal. See names(washi_pal) for a list of available palettes.
n	Number of colors in palette.
reverse	Boolean indicating whether the palette should be reversed. Default is FALSE.

Value

A plot with each color displayed.

See Also

Other color palette functions: washi_pal, washi_pal_setup()

Examples

```
washi_pal_view("standard")

washi_pal_view("color_blind")

washi_pal_view("blue_gradient", 4, reverse = TRUE)
```

`washi_scale` *Scales for plotting with WaSHI palettes*

Description

Provides compatibility with ggplot2.

Usage

```
washi_scale(
  palette = "color_blind",
  aesthetics = c("color", "fill"),
  alpha = 1,
  reverse = FALSE,
  discrete = TRUE,
  ...
)
```

Arguments

palette	Character name of palette in washi_pal. See `names(washi_pal)` for a list of available palettes.
aesthetics	Character string or vector of character strings listing the name(s) of the aesthetic(s) that this scale works with. Defaults to c("color", "fill"), which applies the palette to both the color and fill aesthetics at the same time.
alpha	Numeric transparency level of the color from 0 to 1. Default is 1 (not transparent).
reverse	Boolean indicating whether the palette should be reversed. Default is FALSE.
discrete	Boolean indicating whether color aesthetic is discrete or not. Default is TRUE.
...	Additional arguments passed to discrete_scale() or scale_color_gradientn(), used respectively when discrete is TRUE or FALSE

Value

A `ScaleContinuous` or `ScaleDiscrete` object that can be added to a `ggplot` object.

See Also

Other ggplot2 functions: `washi_theme()`

Examples

```
library(ggplot2)

# Discrete scale
example_data_wide |>
  subset(crop %in% c("Apple", "Cherry", "Potato")) |>
  ggplot(aes(x = ph, y = mn_mg_kg, color = crop)) +
  geom_point(size = 2.5) +
  theme_minimal() +
  washi_scale()

# Continuous scale
example_data_wide |>
  ggplot(aes(x = total_c_percent, y = poxc_mg_kg, color = poxc_mg_kg)) +
  geom_point(size = 2.5) +
  theme_minimal() +
  washi_scale("green_gradient", reverse = TRUE, discrete = FALSE)
```

`washi_theme` *Create standard WaSHI plots*

Description

All changed defaults from this function can be overridden by another call to `ggplot2::theme()` with the desired changes.

Usage

```
washi_theme(
  header_font = "Lato Black",
  header_color = "#151414",
  body_font = "Poppins",
  body_color = "#151414",
  text_scale = 1,
  legend_position = "top",
  facet_space = 2,
  color_gridline = washi_pal[["standard"]][["tan"]],
  gridline_y = TRUE,
  gridline_x = TRUE,
  ...
)
```

Arguments

`header_font`	Font family for title and subtitle. Defaults to "Lato Black".
`header_color`	Font color for title and subtitle. Defaults to almost black.
`body_font`	Font family for all other text Defaults to "Poppins".

body_color	Font color for all other text Defaults to almost black.
text_scale	Scalar that will grow/shrink all text defined within.
legend_position	Position of legend ("none", "left", "right", "bottom", "top", or two-element numeric vector). Defaults to "top".
facet_space	Controls how far apart facets are from each other.
color_gridline	Gridline color. Defaults to WaSHI tan.
gridline_y	Boolean indicating whether major gridlines are displayed for the y axis. Default is TRUE.
gridline_x	Boolean indicating whether major gridlines are displayed for the x axis. Default is TRUE.
...	Pass any parameters from theme that are not already defined within.

Value

ggplot2 object

Source

Adapted from `glitr::si_style()`.

See Also

Other ggplot2 functions: `washi_scale()`

Examples

```
library(ggplot2)
# NOTE: These examples do not use Poppins or Lato in order to pass
# automated checks on computers without these fonts installed.

# Single geom_point plot
example_data_wide |>
  subset(crop %in% c("Apple", "Cherry", "Potato")) |>
  ggplot(aes(x = ph, y = mn_mg_kg, color = crop)) +
  labs(
    title = "Scatter plot of pH and Mn (mg/kg)",
    subtitle = "Example with geom_point().",
    caption = "This is a caption."
  ) +
  geom_point(size = 2.5) +
  washi_theme(
    header_font = "sans",
    body_font = "sans"
  ) +
  washi_scale()

# Bar plot
if (requireNamespace("forcats")) {
```

```
  example_data_wide |>
    ggplot(aes(x = forcats::fct_rev(forcats::fct_infreq(crop)))) +
    geom_bar(fill = washi_pal[["standard"]][["blue"]]) +
    geom_text(
      aes(
        y = after_stat(count),
        label = after_stat(count)
      ),
      stat = "count",
      hjust = 2.5,
      color = "white"
    ) +
    # Flip coordinates to accommodate long crop names
    coord_flip() +
    labs(
      title = "Number of samples in each crop",
      subtitle = "Example plot with geom_bar() without gridlines.",
      y = NULL,
      x = NULL
    ) +
    # Turn gridlines off
    washi_theme(
      gridline_y = FALSE,
      gridline_x = FALSE,
      header_font = "sans",
      body_font = "sans"
    ) +
    # Remove x-axis
    theme(axis.text.x = element_blank())
}

# Facetted geom_density plots
example_data_long |>
  subset(measurement %in% c("total_c_percent", "poxc_mg_kg") &
    !texture == "Loamy Sand") |>
  ggplot(aes(x = value, fill = texture, color = texture)) +
  labs(
    title = "Distribution of POXC (mg/kg) and Total C (%)",
    subtitle = "Example with geom_density() and facet_wrap()."
  ) +
  geom_density(alpha = 0.4) +
  facet_wrap(. ~ measurement, scales = "free") +
  washi_theme(
    legend_position = "bottom",
    header_font = "sans",
    body_font = "sans"
  ) +
  washi_scale() +
  xlab(NULL) +
  guides(col = guide_legend(nrow = 2, byrow = TRUE))
```

Index

∗ **color palette functions**
    washi_pal, 5
    washi_pal_setup, 5
    washi_pal_view, 6
∗ **datasets**
    example_data_long, 2
    example_data_wide, 3
    washi_pal, 5
∗ **font functions**
    washi_install_fonts, 4
∗ **ggplot2 functions**
    washi_scale, 7
    washi_theme, 8
∗ **table functions**
    washi_flextable, 3

example_data_long, 2
example_data_wide, 3

ggplot2::theme(), *8*

washi_flextable, 3
washi_install_fonts, 4
washi_pal, 5, *6*
washi_pal_setup, 5, *5, 6*
washi_pal_view, *5*, 6, *6*
washi_scale, 7, *9*
washi_theme, *7*, 8