Package ‘Rbearcat’

March 23, 2026

Type Package

Title University of Cincinnati Themes and Utilities for Econometrics and Data Science

Version 0.2.2

Description Provides plotting helpers, table-formatting utilities, and report templates for econometrics, model development, and applied data analysis. Includes University of Cincinnati branded themes for 'ggplot2', 'modelsummary', 'flextable', 'rmarkdown', 'bookdown', and 'quarto'.

License MIT + file LICENSE

URL https://github.com/saannidhya/Rbearcat

BugReports https://github.com/saannidhya/Rbearcat/issues

Encoding UTF-8

RoxygenNote 7.3.3

Imports knitr, kableExtra, magrittr, flextable, purrr, officer, colorspace, ggplot2, scales, bookdown, forcats, dplyr, lubridate, stringr, usethis, readr, rmarkdown, xaringan, git2r, gert, modelsummary, broom, patchwork, lmtest, data.table, tools, rlang

Suggests testthat (>= 2.1.0), tibble, gapminder, ggrepel, gt, reactable, RColorBrewer, fivethirtyeight, gcookbook, haven, here, tidyverse, viridis, fixest, plm, ivreg, forecast, xts, zoo, car, janitor, ssh, askpass, crayon, rstudioapi, ragg, systemfonts

VignetteBuilder knitr

NeedsCompilation no

Author Saannidhya Rawat [aut, cre]

Maintainer Saannidhya Rawat <rawatsa@mail.uc.edu>

Repository CRAN

Date/Publication 2026-03-23 07:20:08 UTC

Contents

bcat_beamer — *UC Beamer presentation*

Description

An R Markdown output format for Beamer presentations (PDF).

Usage

```
bcat_beamer(in_header = NULL, ...)
```

Arguments

`in_header`	relative path to a tex formatting document included in the preamble. If `NULL` (default), function will use the preamble.tex file included with the package.
`...`	other options passed to `rmarkdown::beamer_presentation`

Value

An `rmarkdown::output_format` object for rendering Beamer presentations.

Author(s)

Saannidhya Rawat

bcat_cor_table — *UC-branded correlation matrix*

Description

Produce a formatted correlation matrix with significance stars and UC styling. By default shows only the lower triangle.

Usage

```
bcat_cor_table(
  df,
  method = c("pearson", "spearman", "kendall"),
  full_matrix = FALSE,
  digits = 2,
  stars = TRUE,
  p_threshold = c(0.01, 0.05, 0.1),
  caption = NULL,
  font_size = 12,
  header_bg_color = palette_UC[["UC Red"]],
  header_txt_color = palette_UC[["White"]],
```

```
  striped = TRUE,
  doc_type = NULL,
  ...
)
```

Arguments

df	A data frame of numeric columns.
method	Character. Correlation method: "pearson" (default), "spearman", or "kendall".
full_matrix	Logical. Show full matrix? Default is FALSE (lower triangle only).
digits	Integer. Decimal places. Default is 2.
stars	Logical. Show significance stars? Default is TRUE.
p_threshold	Numeric vector of p-value thresholds for stars. Default: c(0.01, 0.05, 0.1).
caption	Character. Table caption.
font_size	Numeric. Font size. Default is 12.
header_bg_color	Background color for header.
header_txt_color	Text color for header.
striped	Logical. Zebra striping?
doc_type	Character. Force output format. Auto-detected if NULL.
...	Additional arguments passed to table formatting.

Value

A formatted table object.

Author(s)

Saannidhya Rawat

See Also

Other tables: bcat_fmt_style_table(), bcat_reg_table(), bcat_sum_table()

Examples

```
bcat_cor_table(mtcars[, c("mpg", "wt", "hp", "disp")])
bcat_cor_table(mtcars[, c("mpg", "wt", "hp")], method = "spearman")
```

bcat_extract_metadata *Extract metadata*

Description

This utility function is useful for extracting column values that will be used for inline text by saving metadata column values in a nested (named) list.

Usage

```
bcat_extract_metadata(df, id_col)
```

Arguments

df	input data
id_col	ID column used identify observations

Value

A named list. Each top-level element corresponds to a metadata column in `df`, and each nested element is named by `id_col` values and stores the matching column value.

Author(s)

Saannidhya Rawat

See Also

Other utilities: `bcat_import_data()`, `bcat_new_quarto()`, `bcat_setup_rmd()`, `bcat_source_rmd()`

Examples

```
library(Rbearcat)
library(tibble)

lob_df <- tibble(lob = c("drl", "mla", "rac"),
                 balance = bcat_fmt_dollar(c(11, 26, 7)),
                 nco = bcat_fmt_dollar(c(80, 45, 800)))

# extract metadata from each column and save to list
lob_meta <- bcat_extract_metadata(lob_df, lob)

# extract individual values from list
lob_meta$balance$mla
lob_meta$nco$rac
```

bcat_fmt_comma *Label numbers in decimal format*

Description

A wrapper for `scales::label_comma` to format a numeric vector to force decimal display of numbers with comma separators.

Usage

```
bcat_fmt_comma(x, accuracy = NULL, scale = 1, ...)
```

Arguments

`x`	a numeric vector
`accuracy`	A number to round to. Use (e.g.) `0.01` to show 2 decimal places of precision. If `NULL`, the default, uses a heuristic that should ensure breaks have the minimum number of digits needed to show the difference between adjacent values.
`scale`	A scaling factor: `x` will be multiplied by `scale` before formatting.
`...`	other arguments passed to label_comma

Value

A character vector of formatted decimal labels with the same length as `x`.

See Also

Other formatting: `bcat_fmt_date()`, `bcat_fmt_dollar()`, `bcat_fmt_percent()`, `bcat_fmt_pvalue()`, `bcat_fmt_scientific()`

Examples

```
bcat_fmt_comma(c(5000, 10000, 8000))
bcat_fmt_comma(c(5000, 10000, 8000), scale = 10e-4, accuracy = 1, suffix = "K")
```

bcat_fmt_date *Label dates/times*

Description

A wrapper for `base::format` to convert character or date vector to specfied date/time format.

Usage

```
bcat_fmt_date(x, format = "%B %e, %Y", ...)
```

Arguments

`x`	a character or date/datetime vector
`format`	a date/time format string using standard POSIX specification
`...`	other arguments passed to format

Value

A character vector of formatted date or datetime labels with the same length as x.

See Also

Other formatting: `bcat_fmt_comma()`, `bcat_fmt_dollar()`, `bcat_fmt_percent()`, `bcat_fmt_pvalue()`, `bcat_fmt_scientific()`

Examples

```
bcat_fmt_date(Sys.Date())
bcat_fmt_date(c("2019-12-03", "2020-05-22"))
```

`bcat_fmt_dollar` *Label currencies*

Description

A wrapper for `scales::label_dollar` to format a numeric vector to dollars and cents.

Usage

```
bcat_fmt_dollar(
  x,
  accuracy = NULL,
  scale = 1,
  largest_with_cents = 1e+05,
  style_negative = c("hyphen", "parens"),
  ...
)
```

Arguments

`x`	a numeric vector
`accuracy`	Number to round to. If NULL, the default, values will be rounded to the nearest integer
`scale`	A scaling factor: `x` will be multiplied by `scale` before formatting.
`largest_with_cents`	largest numeric value for which cents will be displayed
`style_negative`	Character. How to display negative values: `"hyphen"` (default) or `"parens"`.
`...`	other arguments passed on to label_dollar

Value

A character vector of formatted currency labels with the same length as x.

See Also

Other formatting: bcat_fmt_comma(), bcat_fmt_date(), bcat_fmt_percent(), bcat_fmt_pvalue(), bcat_fmt_scientific()

Examples

```
bcat_fmt_dollar(1:4)
bcat_fmt_dollar(c(-20, -10, 10, 20))
bcat_fmt_dollar(c(-20, -10, 10, 20), style_negative = "parens")
```

---

bcat_fmt_percent *Label percentages*

Description

A wrapper for scales::label_percent to format a numeric vector to percentages.

Usage

```
bcat_fmt_percent(x, accuracy = NULL, scale = 100, ...)
```

Arguments

x	a numeric vector
accuracy	A number to round to. Use (e.g.) 0.01 to show 2 decimal places of precision. If NULL, the default, uses a heuristic that should ensure breaks have the minimum number of digits needed to show the difference between adjacent values.
scale	A scaling factor: x will be multiplied by scale before formatting.
...	other arguments passed to label_percent

Value

A character vector of formatted percentage labels with the same length as x.

See Also

Other formatting: bcat_fmt_comma(), bcat_fmt_date(), bcat_fmt_dollar(), bcat_fmt_pvalue(), bcat_fmt_scientific()

Examples

```
bcat_fmt_percent(c(0.01, 0.05, 0.02))
bcat_fmt_percent(c(0.0151321, 0.090115, 0.022141))
bcat_fmt_percent(c(0.0151321, 0.090115, 0.022141), accuracy = 0.001)
bcat_fmt_percent(c(1.31, 2.44, 3.0), scale = 1)
```

bcat_fmt_pvalue *Label p-values*

Description

A wrapper for `scales::label_pvalue` to format a numeric vector to p-values, using "<" and ">" for p-values close to 0 and 1.

Usage

```
bcat_fmt_pvalue(x, accuracy = 0.001, add_p = FALSE, ...)
```

Arguments

`x`	a numeric vector
`accuracy`	A number to round to. Use (e.g.) `0.01` to show 2 decimal places of precision. If `NULL`, the default, uses a heuristic that should ensure breaks have the minimum number of digits needed to show the difference between adjacent values.
`add_p`	logical. Add "p=" before the value?
`...`	other arguments passed to label_pvalue

Value

A character vector of formatted p-value labels with the same length as `x`.

See Also

Other formatting: `bcat_fmt_comma()`, `bcat_fmt_date()`, `bcat_fmt_dollar()`, `bcat_fmt_percent()`, `bcat_fmt_scientific()`

Examples

```
bcat_fmt_pvalue(c(0.01, 0.05, 0.02))
bcat_fmt_pvalue(c(0.000001, 0.999115, 0.022141))
bcat_fmt_pvalue(c(0.000001, 0.999115, 0.022141), add_p = TRUE)
```

bcat_fmt_scientific *Label numbers with scientific notation*

Description

A wrapper for `scales::label_scientific` to format a numeric vector to scientific notation.

Usage

```
bcat_fmt_scientific(x, digits = 3, scale = 1, ...)
```

Arguments

`x`	a numeric vector
`digits`	number of digits to show before exponent.
`scale`	A scaling factor: `x` will be multiplied by `scale` before formatting.
`...`	other arguments passed to label_scientific

Value

A character vector of scientific-notation labels with the same length as `x`.

See Also

Other formatting: `bcat_fmt_comma()`, `bcat_fmt_date()`, `bcat_fmt_dollar()`, `bcat_fmt_percent()`, `bcat_fmt_pvalue()`

Examples

```
bcat_fmt_scientific(c(1:10)*10e5)
```

bcat_fmt_style_table *UC table formatting*

Description

Format tables for output to final documentation. Automatically detects output format (HTML, PDF, Word, PPTX) and applies UC styling.

Usage

```
bcat_fmt_style_table(
  df,
  caption = NULL,
  footer = NULL,
  header = NULL,
  header_bg_color = palette_UC[["UC Red"]],
  header_txt_color = palette_UC[["White"]],
  align,
  font_size = 12,
  striped = TRUE,
  full_width = FALSE,
  scale_down = FALSE,
  longtable = FALSE,
  col_names = NA,
  threeparttable = FALSE,
 doc_type = c(knitr::opts_knit$get("rmarkdown.pandoc.to"), "html", "pdf", "docx",
    "pptx"),
  ...
)
```

Arguments

`df`	Data frame to format.
`caption`	Character. Table caption.
`footer`	Character. Table footnote.
`header`	Character. Spanning header text above all columns.
`header_bg_color`	Background fill color for header. Default is UC Red.
`header_txt_color`	Text color for header. Default is white.
`align`	Character vector of column alignments: 'l' (left), 'c' (center), 'r' (right).
`font_size`	Numeric. Font size for table. Default is 12.
`striped`	Logical. Enable zebra striping? Default is TRUE.
`full_width`	Logical. Should HTML table be full width? Default is FALSE.
`scale_down`	Logical. Scale LaTeX table to fit page? Default is FALSE.
`longtable`	Logical. Use longtable for multi-page PDF tables? Default is FALSE.
`col_names`	Character vector of column names. Set to NULL to remove header row.
`threeparttable`	Logical. Use LaTeX threeparttable for footnotes? Default is FALSE.
`doc_type`	Manually set document type. Auto-detected if not specified.
`...`	Additional arguments passed to `knitr::kable`.

Value

A formatted table object (kableExtra for HTML/PDF, flextable for Word/PPTX).

Author(s)

Saannidhya Rawat

See Also

Other tables: bcat_cor_table(), bcat_reg_table(), bcat_sum_table()

Examples

```
bcat_fmt_style_table(iris[1:10, ])
bcat_fmt_style_table(iris[1:5, ], caption = "Iris Sample", striped = TRUE)
```

---

bcat_gitbook *Rbearcat Gitbook*

Description

An R Markdown output format for the Gitbook format (HTML).

Usage

```
bcat_gitbook(css = NULL, after_body = NULL, collapse = "subsection", ...)
```

Arguments

css	relative path to a css formatting document. If NULL (default), function will use the style.css file included with the package.
after_body	relative path to HTML file to include after body of each chapter. If NULL (default), function will use the footer.html file included with the package.
collapse	specify level at which to collapse the table of contents.
...	other options passed to bookdown::gitbook

Value

An rmarkdown::output_format object for rendering Gitbook output.

Author(s)

Saannidhya Rawat

`bcat_import_data` *Import data*

Description

Utility to import multiple data sets from a directory into a named list.

Usage

```
bcat_import_data(location, extension, import_function, recursive = TRUE)
```

Arguments

`location`	Character. Directory from which to import data.
`extension`	Character. File extension to match (without ".").
`import_function`	Function used to read each file (e.g., `readr::read_csv`).
`recursive`	Logical. Search subdirectories? Default is `TRUE`.

Value

A named list of imported objects. Names are derived from filenames without their extensions, and each element contains the result returned by `import_function()` for one matching file.

Author(s)

Saannidhya Rawat

See Also

Other utilities: `bcat_extract_metadata()`, `bcat_new_quarto()`, `bcat_setup_rmd()`, `bcat_source_rmd()`

Examples

```
tmp_dir <- tempfile("rbearcat-data-")
dir.create(tmp_dir)
csv_path <- file.path(tmp_dir, "example.csv")
utils::write.csv(data.frame(x = 1:3, y = letters[1:3]), csv_path, row.names = FALSE)

datasets <- bcat_import_data(tmp_dir, "csv", utils::read.csv)
names(datasets)
datasets$example

unlink(tmp_dir, recursive = TRUE)
```

bcat_new_quarto — *Create a UC-branded Quarto document*

Description

Copies a UC Quarto template into the specified directory, ready to edit and render.

Usage

```
bcat_new_quarto(
  type = c("html", "pdf", "revealjs"),
  path = getwd(),
  overwrite = FALSE
)
```

Arguments

`type`	Character. Template type: `"html"`, `"pdf"`, or `"revealjs"`.
`path`	Character. Directory to copy the template into. Default is current working directory.
`overwrite`	Logical. Overwrite existing files? Default is FALSE.

Value

An invisible character scalar giving the path to the created `template.qmd` file.

Author(s)

Saannidhya Rawat

See Also

Other utilities: `bcat_extract_metadata()`, `bcat_import_data()`, `bcat_setup_rmd()`, `bcat_source_rmd()`

Examples

```
out_dir <- file.path(tempdir(), "uc-quarto")
created <- bcat_new_quarto("html", path = out_dir)
basename(created)
unlink(out_dir, recursive = TRUE)
```

`bcat_pdf_book` *Rbearcat PDF Book*

Description

An R Markdown output format for the PDF books.

Usage

```
bcat_pdf_book(in_header = NULL, before_body = NULL, toc_depth = 4, ...)
```

Arguments

`in_header`	relative path to a tex formatting document included in the preamble. If `NULL` (default), function will use the preamble.tex file included with the package.
`before_body`	relative path to a tex document included in before the body of the document. If `NULL` (default), function will use the before_body.tex file included with the package.
`toc_depth`	depth of headers to include in table of contents
`...`	other options passed to `bookdown::pdf_book`

Value

An `rmarkdown::output_format` object for rendering PDF books.

Author(s)

Saannidhya Rawat

`bcat_plt_area` *Area plot utility*

Description

Create an area plot using ggplot2 graphics. This function is a wrapper to create commonly used styles of area plots. Additional layers can be added to this plot as needed. More complicated plots can be creating using individual ggplot2 layers.

Usage

```
bcat_plt_area(
  df,
  x = NULL,
  y = NULL,
  fill = NULL,
  facet = NULL,
  position = c("stack", "fill"),
  x_lab = ggplot2::waiver(),
  y_lab = ggplot2::waiver(),
  title = ggplot2::waiver(),
  subtitle = ggplot2::waiver(),
  caption = ggplot2::waiver(),
  legend_lab = ggplot2::waiver(),
  legend_position = "right",
  legend_hide = FALSE,
  x_scale = NULL,
  y_scale = NULL,
  fill_scale = Rbearcat::scale_fill_UC(),
  facet_scale = c("fixed", "free_y", "free_x", "free"),
  nrow = NULL,
  ncol = NULL,
  x_refline = NULL,
  y_refline = NULL
)
```

Arguments

`df`	The data to be displayed
`x`	variable to map to the x-axis
`y`	Variable to map to the y-axis
`fill`	Variable to map to the fill aesthetic
`facet`	Facetting variable(s). Note: must wrap in `vars`, e.g, `facet = vars(var1, var2)`
`position`	Either stack values in cumulative fashion (`position = "stack"`) or stack such that proportions at each value of `x` sum to 1 (`position = "fill"`).
`x_lab`	Label for x-axis
`y_lab`	Label for y-axis
`title`	Plot title
`subtitle`	Plot subtitle
`caption`	Plot caption
`legend_lab`	Legend title
`legend_position`	legend position. "bottom" or "right"
`legend_hide`	Set to `TRUE` to hide the legend

x_scale
: scale_x_ function to apply to x-axis. If position = "fill", it is recommended to use the expand = c(0,0) option in the x_scale.

y_scale
: scale_y_ function to apply to y-axis. If position = "fill", it is recommended to use the expand = c(0,0) option in the y_scale.

fill_scale
: scale_fill_ function to apply to colors.

facet_scale
: Shoud facet scales be fixed ("fixed", the default), free ("free"), or free in one dimension ("free_x", "free_y")

nrow
: Number of facet rows

ncol
: Number of facet columns

x_refline
: Vector of x-values at which to draw vertical reference lines

y_refline
: Vector of y-values at which to draw horizontal reference lines

Value

A ggplot2 plot object.

Author(s)

Saannidhya Rawat

See Also

Other plots: bcat_plt_bar(), bcat_plt_box(), bcat_plt_coef(), bcat_plt_diag(), bcat_plt_hist(), bcat_plt_line(), bcat_plt_point(), bcat_plt_ts()

Examples

```
set.seed(1234)

d <- data.frame(t=rep(0:23,each=4),var=rep(LETTERS[1:4],4),val=round(runif(4*24,0,50)))

# stacked area plot
bcat_plt_area(df = d, x = t, y = val, fill = var,
              position = "stack",
              fill_scale = ggplot2::scale_fill_viridis_d(),
              legend_lab = NULL)

# filled area plot
bcat_plt_area(df = d, x = t, y = val, fill = var,
              position = "fill",
              fill_scale = ggplot2::scale_fill_viridis_d(),
              legend_lab = NULL)
```

`bcat_plt_bar` *Bar plot utility*

Description

Create a bar plot using ggplot2 graphics. This function is a wrapper to create commonly used styles of bar plots. Additional layers can be added to this plot as needed. More complicated plots can be creating using individual ggplot2 layers.

Usage

```
bcat_plt_bar(
  df,
  x = NULL,
  y = NULL,
  fill = NULL,
  facet = NULL,
  stat = c("freq", "identity", "sum", "mean", "median"),
  position = c("stack", "dodge", "fill"),
  x_lab = ggplot2::waiver(),
  y_lab = ggplot2::waiver(),
  title = ggplot2::waiver(),
  subtitle = ggplot2::waiver(),
  caption = ggplot2::waiver(),
  legend_lab = ggplot2::waiver(),
  legend_position = "bottom",
  legend_hide = FALSE,
  x_scale = NULL,
  y_scale = NULL,
  fill_scale = Rbearcat::scale_fill_UC(),
  facet_scale = c("fixed", "free_y", "free_x", "free"),
  nrow = NULL,
  ncol = NULL,
  x_refline = NULL,
  y_refline = NULL,
  coord_flip = FALSE,
  order = FALSE
)
```

Arguments

`df`	The data to be displayed
`x`	Categorical variable to map to the x-axis
`y`	Variable to map to the y-axis. Only applicable if `stat = "identity"`
`fill`	Variable to map to the fill aesthetic

facet	Facetting variable(s). Note: must wrap in vars(), e.g, facet = vars(var1, var2)
stat	Statistic to map to y-axis. Default ("freq") calculates frequencies, "identity" plots the data as-is. "sum", "mean", and "median" calculate and display the respective summary stats Must specify y if using "identity".
position	For grouped data, either stack bars (position = "stack"), place bars side-by-side (position = "dodge"), or stack such that relative proportions within each group sum to 1 (position = "fill").
x_lab	Label for x-axis
y_lab	Label for y-axis
title	Plot title
subtitle	Plot subtitle
caption	Plot caption
legend_lab	Legend title
legend_position	legend position. "bottom" or "right"
legend_hide	Set to TRUE to hide the legend
x_scale	scale_x_ function to apply to x-axis.
y_scale	scale_y_ function to apply to y-axis.
fill_scale	scale_fill_ function to apply to colors.
facet_scale	Shoud facet scales be fixed ("fixed", the default), free ("free"), or free in one dimension ("free_x", "free_y")?
nrow	Number of facet rows
ncol	Number of facet columns
x_refline	Vector of x-values at which to draw vertical reference lines
y_refline	Vector of y-values at which to draw horizontal reference lines
coord_flip	Set to TRUE to flip x and y-axis. Useful if levels of x have long names.
order	Set to TRUE to arrange bars by frequency (if stat = "freq"), values of y (if stat = "identity"), or by other provided stat. Use if levels of x do not have a natural ordering.

Value

A ggplot2 plot object.

Author(s)

Saannidhya Rawat

See Also

Other plots: bcat_plt_area(), bcat_plt_box(), bcat_plt_coef(), bcat_plt_diag(), bcat_plt_hist(), bcat_plt_line(), bcat_plt_point(), bcat_plt_ts()

Examples

```
library(ggplot2)
library(dplyr)
library(scales)

# basic plot of frequencies
bcat_plt_bar(df = mpg,
             x = toupper(class),
             order = TRUE,
             x_lab = NULL,
             y_lab = NULL,
             title = "Number of Vehicles by Class and Drive Type",
             legend_lab = "Drive Type")

# can plot relative frequencies for each `x` by `fill` using `position = "fill"`
bcat_plt_bar(df = mpg,
             x = toupper(class),
             fill = drv,
             position = "fill",
             x_lab = NULL,
             y_lab = NULL,
             y_scale = scale_y_continuous(labels = percent_format()),
             title = "Percent of Vehicle Class for Each Drive Type",
             legend_lab = "Drive Type")

# use `stat` to compute and plot other statistics of interest
bcat_plt_bar(mpg,
             x = toupper(class),
             y = hwy,
             fill = factor(year),
             stat = "mean",
             order = TRUE,
             position = "dodge",
             y_refline = round(mean(mpg$hwy), 2),
             coord_flip = TRUE,
             x_lab = NULL,
             y_lab = "Highway MPG",
             title = "Average Highway MPG by Vehicle Class",
             legend_lab = NULL)

# use `stat = "identity"` to plot data directly from data set
mpg %>%
  group_by(year, class) %>%
  summarise(hwy = mean(hwy)) %>%
  bcat_plt_bar(x = toupper(class),
               y = hwy,
               fill = factor(year),
               stat = "identity",
               order = TRUE,
               position = "dodge",
               y_refline = round(mean(mpg$hwy), 2),
               coord_flip = TRUE,
```

```
x_lab = NULL,
y_lab = "Highway MPG",
title = "Average Highway MPG by Vehicle Class",
legend_lab = NULL)
```

`bcat_plt_box` *Box plot / violin plot utility*

Description

Create a box plot or violin plot with UC styling. Includes optional jittered point overlay and outlier highlighting.

Usage

```
bcat_plt_box(
  df,
  x,
  y,
  fill = NULL,
  facet = NULL,
  violin = FALSE,
  jitter = TRUE,
  jitter_width = 0.2,
  notch = FALSE,
  order = FALSE,
  coord_flip = FALSE,
  alpha = 0.3,
  x_lab = ggplot2::waiver(),
  y_lab = ggplot2::waiver(),
  title = ggplot2::waiver(),
  subtitle = ggplot2::waiver(),
  caption = ggplot2::waiver(),
  legend_lab = ggplot2::waiver(),
  legend_position = "bottom",
  legend_hide = FALSE,
  x_scale = NULL,
  y_scale = NULL,
  fill_scale = scale_fill_UC(),
  facet_scale = c("fixed", "free_y", "free_x", "free"),
  nrow = NULL,
  ncol = NULL,
  x_refline = NULL,
  y_refline = NULL
)
```

Arguments

df	The data to be displayed.
x	Categorical variable for the x-axis.
y	Numeric variable for the y-axis.
fill	Variable to map to fill aesthetic. If NULL, uses x for coloring.
facet	Facetting variable(s). Wrap in `vars()`.
violin	Logical. Use violin plot instead of box plot? Default is FALSE.
jitter	Logical. Overlay jittered points? Default is TRUE.
jitter_width	Jitter width. Default is 0.2.
notch	Logical. Add notches? Default is FALSE.
order	Logical. Reorder x by median of y? Default is FALSE.
coord_flip	Logical. Flip coordinates? Default is FALSE.
alpha	Point transparency. Default is 0.3.
x_lab	Label for x-axis.
y_lab	Label for y-axis.
title	Plot title.
subtitle	Plot subtitle.
caption	Plot caption.
legend_lab	Legend title.
legend_position	Legend position.
legend_hide	Logical. Hide legend?
x_scale	`scale_x_` function.
y_scale	`scale_y_` function.
fill_scale	`scale_fill_` function.
facet_scale	Facet scales.
nrow	Number of facet rows.
ncol	Number of facet columns.
x_refline	Vertical reference lines.
y_refline	Horizontal reference lines.

Value

A ggplot object.

Author(s)

Saannidhya Rawat

See Also

Other plots: bcat_plt_area(), bcat_plt_bar(), bcat_plt_coef(), bcat_plt_diag(), bcat_plt_hist(), bcat_plt_line(), bcat_plt_point(), bcat_plt_ts()

Examples

```
library(ggplot2)

# Basic box plot
bcat_plt_box(mtcars, x = factor(cyl), y = mpg)

# Violin plot
bcat_plt_box(mtcars, x = factor(cyl), y = mpg, violin = TRUE)

# Ordered and flipped
bcat_plt_box(mtcars, x = factor(cyl), y = mpg, order = TRUE, coord_flip = TRUE)
```

---

bcat_plt_coef *Coefficient plot (forest plot)*

Description

Visualize regression coefficients and confidence intervals for one or more models. Uses broom::tidy() to extract estimates.

Usage

```
bcat_plt_coef(
  models,
  conf_level = 0.95,
  intercept = FALSE,
  coef_rename = NULL,
  highlight = NULL,
  dodge_width = 0.4,
  x_lab = "Estimate",
  y_lab = ggplot2::waiver(),
  title = ggplot2::waiver(),
  subtitle = ggplot2::waiver(),
  caption = ggplot2::waiver(),
  legend_lab = "Model",
  legend_position = "bottom",
  legend_hide = FALSE,
  color_scale = scale_colour_UC()
)
```

Arguments

`models`	A model object or a named list of model objects.
`conf_level`	Numeric. Confidence level. Default is 0.95.
`intercept`	Logical. Include intercept? Default is FALSE.
`coef_rename`	Named character vector to rename coefficients. If NULL (default), auto-cleans names to Title Case.
`highlight`	Character vector of term names to highlight in UC Red.
`dodge_width`	Numeric. Dodge width for multi-model comparison. Default is 0.4.
`x_lab`	Label for x-axis (coefficient values).
`y_lab`	Label for y-axis (term names).
`title`	Plot title.
`subtitle`	Plot subtitle.
`caption`	Plot caption.
`legend_lab`	Legend title.
`legend_position`	Legend position.
`legend_hide`	Logical. Hide legend?
`color_scale`	`scale_color_` function.

Value

A ggplot object.

Author(s)

Saannidhya Rawat

See Also

Other plots: `bcat_plt_area()`, `bcat_plt_bar()`, `bcat_plt_box()`, `bcat_plt_diag()`, `bcat_plt_hist()`, `bcat_plt_line()`, `bcat_plt_point()`, `bcat_plt_ts()`

Examples

```
library(ggplot2)

m <- lm(mpg ~ wt + hp + cyl, data = mtcars)
bcat_plt_coef(m)

m1 <- lm(mpg ~ wt + hp, data = mtcars)
m2 <- lm(mpg ~ wt + hp + cyl + disp, data = mtcars)
bcat_plt_coef(list("Base" = m1, "Full" = m2))
```

bcat_plt_diag *Regression diagnostic dashboard*

Description

Produce a multi-panel diagnostic plot for a fitted model with UC styling. Optionally prints assumption test results (Breusch-Pagan, Shapiro-Wilk, Durbin-Watson) to the console.

Usage

```
bcat_plt_diag(
  model,
  which = c(1, 2, 3, 4),
  labels = TRUE,
  n_labels = 3,
  tests = TRUE,
  nrow = NULL,
  ncol = NULL
)
```

Arguments

`model`	A fitted model object (e.g., `lm`, `glm`).
`which`	Integer vector specifying which panels to include: 1 = Residuals vs Fitted, 2 = Q-Q Plot, 3 = Scale-Location, 4 = Residuals vs Leverage. Default: `c(1, 2, 3, 4)`.
`labels`	Logical. Label influential observations? Default is TRUE.
`n_labels`	Integer. Number of extreme observations to label. Default is 3.
`tests`	Logical. Print assumption test results to console? Default is TRUE.
`nrow`	Number of panel rows.
`ncol`	Number of panel columns.

Value

A patchwork object combining the diagnostic panels.

Author(s)

Saannidhya Rawat

See Also

Other plots: `bcat_plt_area()`, `bcat_plt_bar()`, `bcat_plt_box()`, `bcat_plt_coef()`, `bcat_plt_hist()`, `bcat_plt_line()`, `bcat_plt_point()`, `bcat_plt_ts()`

Examples

```
m <- lm(mpg ~ wt + hp + cyl, data = mtcars)
bcat_plt_diag(m)
bcat_plt_diag(m, which = c(1, 2))
```

`bcat_plt_hist` *Histogram utility*

Description

Create a histogram using ggplot2 graphics with UC styling. Includes optional density curve overlay and mean reference line.

Usage

```
bcat_plt_hist(
  df,
  x,
  bins = NULL,
  fill = palette_UC[["Steger Silver"]],
  facet = NULL,
  density = FALSE,
  mean_line = TRUE,
  rug = FALSE,
  x_lab = ggplot2::waiver(),
  y_lab = ggplot2::waiver(),
  title = ggplot2::waiver(),
  subtitle = ggplot2::waiver(),
  caption = ggplot2::waiver(),
  legend_position = "bottom",
  legend_hide = FALSE,
  x_scale = NULL,
  y_scale = NULL,
  facet_scale = c("fixed", "free_y", "free_x", "free"),
  nrow = NULL,
  ncol = NULL,
  x_refline = NULL,
  y_refline = NULL
)
```

Arguments

`df`	The data to be displayed.
`x`	Variable to map to the x-axis.
`bins`	Number of bins. If NULL (default), uses ggplot2's automatic selection.
`fill`	Fill color for bars. Default is UC Steger Silver.

facet	Facetting variable(s). Wrap in vars().
density	Logical. Overlay density curve? Default is FALSE.
mean_line	Logical. Draw dashed vertical line at mean? Default is TRUE.
rug	Logical. Add rug plot at bottom? Default is FALSE.
x_lab	Label for x-axis.
y_lab	Label for y-axis.
title	Plot title.
subtitle	Plot subtitle.
caption	Plot caption.
legend_position	Legend position: "bottom" or "right".
legend_hide	Logical. Hide legend?
x_scale	scale_x_ function.
y_scale	scale_y_ function.
facet_scale	Facet scales: "fixed", "free", "free_x", "free_y".
nrow	Number of facet rows.
ncol	Number of facet columns.
x_refline	Vector of x-values for vertical reference lines.
y_refline	Vector of y-values for horizontal reference lines.

Value

A ggplot object.

Author(s)

Saannidhya Rawat

See Also

Other plots: bcat_plt_area(), bcat_plt_bar(), bcat_plt_box(), bcat_plt_coef(), bcat_plt_diag(), bcat_plt_line(), bcat_plt_point(), bcat_plt_ts()

Examples

```
library(ggplot2)

# Basic histogram
bcat_plt_hist(mtcars, x = mpg)

# With density curve and mean line
bcat_plt_hist(mtcars, x = mpg, density = TRUE, mean_line = TRUE)

# Faceted by cylinders
bcat_plt_hist(mtcars, x = mpg, facet = vars(cyl), facet_scale = "free_x")
```

bcat_plt_line *Line plot utility*

Description

Create a line plot using ggplot2 graphics. This function is a wrapper to create commonly used styles of line plots. Additional layers can be added to this plot as needed. More complicated plots can be creating using individual ggplot2 layers.

Usage

```
bcat_plt_line(
  df,
  x,
  y,
  color = NULL,
  linetype = NULL,
  facet = NULL,
  x_lab = ggplot2::waiver(),
  y_lab = ggplot2::waiver(),
  title = ggplot2::waiver(),
  subtitle = ggplot2::waiver(),
  caption = ggplot2::waiver(),
  legend_lab = ggplot2::waiver(),
  legend_position = "bottom",
  legend_hide = FALSE,
  x_scale = NULL,
  y_scale = NULL,
  color_scale = Rbearcat::scale_colour_UC(),
  facet_scale = c("fixed", "free_y", "free_x", "free"),
  nrow = NULL,
  ncol = NULL,
  layer_points = FALSE,
  x_refline = NULL,
  y_refline = NULL,
  x_highlight_min = NULL,
  x_highlight_max = NULL,
  y_highlight_min = NULL,
  y_highlight_max = NULL,
  y_ribbon_min = NULL,
  y_ribbon_max = NULL,
  y_error_min = NULL,
  y_error_max = NULL,
  y_error_width = 1
)
```

Arguments

df	The data to be displayed
x	Variable to map to the x-axis
y	Variable to map to the y-axis
color	Variable to map to the color aesthetic
linetype	Variable to map to the linetype aesthetic
facet	Facetting variable(s). Note: must wrap in `vars()`, e.g, `facet = vars(var1, var2)`
x_lab	Label for x-axis
y_lab	Label for y-axis
title	Plot title
subtitle	Plot subtitle
caption	Plot caption
legend_lab	Legend title
legend_position	legend position. "bottom" or "right"
legend_hide	Set to `TRUE` to hide the legend
x_scale	`scale_x_` function to apply to x-axis.
y_scale	`scale_y_` function to apply to y-axis.
color_scale	`scale_color_` function to apply to colors.
facet_scale	Shoud facet scales be fixed ("fixed", the default), free ("free"), or free in one dimension ("free_x", "free_y")?
nrow	Number of facet rows
ncol	Number of facet columns
layer_points	Set to `TRUE` to plot points
x_refline	Vector of x-values at which to draw vertical reference lines
y_refline	Vector of y-values at which to draw horizontal reference lines
x_highlight_min	Vector of min x-values at which to start draw rectangle annotation
x_highlight_max	Vector of max x-values at which to end rectangle annotation
y_highlight_min	Vector of min y-values at which to start draw rectangle annotation
y_highlight_max	Vector of max y-values at which to end rectangle annotation
y_ribbon_min	Variable to use as minimum values for ribbon around y at each value of x
y_ribbon_max	Variable to use as maximum values for ribbon around y at each value of x
y_error_min	Variable to use as minimum values for error bars around y at each value of x
y_error_max	Variable to use as maximum values for error bars around y at each value of x
y_error_width	Width of error bars

Value

A ggplot2 plot object.

Author(s)

Saannidhya Rawat

See Also

Other plots: bcat_plt_area(), bcat_plt_bar(), bcat_plt_box(), bcat_plt_coef(), bcat_plt_diag(), bcat_plt_hist(), bcat_plt_point(), bcat_plt_ts()

Examples

```
library(ggplot2)
library(scales)

# basic time series plot with a reference line
bcat_plt_line(df = economics,
              x = date,
              y = unemploy,
              y_scale = scale_y_continuous(labels = comma_format()),
              y_refline = 10000)

# line plot with facets and highlight periods
bcat_plt_line(df = economics_long,
             x = date,
             y = value,
             color = variable,
             facet = vars(variable),
             x_lab = "Decade",
             y_lab = "Value",
             legend_lab = NULL,
             facet_scale = "free_y",
             x_highlight_min = as.Date(c("2000-01-01", "2008-01-01")),
             x_highlight_max = as.Date(c("2002-01-01", "2010-01-01")),
             ncol = 1)
```

---

bcat_plt_point *Scatter plot utility*

Description

Create a scatter plot using ggplot2 graphics. This function is a wrapper to create commonly used styles of scatter plots. Additional layers can be added to this plot as needed. More complicated scatter plots can be creating using individual ggplot2 layers.

Usage

```
bcat_plt_point(
  df,
  x,
  y,
  color = NULL,
  size = NULL,
  facet = NULL,
  jitter = FALSE,
  jitter_width = NULL,
  smooth = FALSE,
  method = "lm",
  se = TRUE,
  x_lab = ggplot2::waiver(),
  y_lab = ggplot2::waiver(),
  title = ggplot2::waiver(),
  subtitle = ggplot2::waiver(),
  caption = ggplot2::waiver(),
  legend_lab = ggplot2::waiver(),
  legend_position = "bottom",
  legend_hide = FALSE,
  x_scale = NULL,
  y_scale = NULL,
  color_scale = Rbearcat::scale_colour_UC(),
  fill_scale = Rbearcat::scale_fill_UC(),
  facet_scale = c("fixed", "free_y", "free_x", "free"),
  alpha = 0.6,
  nrow = NULL,
  ncol = NULL,
  x_refline = NULL,
  y_refline = NULL,
  identity_line = FALSE,
  x_highlight_min = NULL,
  x_highlight_max = NULL,
  y_highlight_min = NULL,
  y_highlight_max = NULL,
  y_error_min = NULL,
  y_error_max = NULL,
  y_error_width = 1
)
```

Arguments

df	The data to be displayed
x	Variable to map to the x-axis
y	Variable to map to the y-axis
color	Variable to map to the color aesthetic

size	Variable to map to the size aesthetic
facet	Facetting variable(s). Note: must wrap in vars(), e.g, facet = vars(var1, var2)
jitter	Set to TRUE to enable jittering.
jitter_width	Set the jitter width. Leave as NULL to use the default ggplot2 settings
smooth	Add fit line to plot
method	Method to use for fit line. "lm" is default
se	Set to FALSE to remove standard error band around fit line
x_lab	Label for x-axis
y_lab	Label for y-axis
title	Plot title
subtitle	Plot subtitle
caption	Plot caption
legend_lab	Legend title
legend_position	legend position. "bottom" or "right"
legend_hide	Set to TRUE to hide the legend
x_scale	scale_x_ function to apply to x-axis.
y_scale	scale_y_ function to apply to y-axis.
color_scale	scale_color_ function to apply to colors.
fill_scale	scale_fill_ function to apply to colors. only applicable if smooth = TRUE`
facet_scale	Shoud facet scales be fixed ("fixed", the default), free ("free"), or free in one dimension ("free_x", "free_y")?
alpha	level of point transparency. lower alpha leads to more transparency.
nrow	Number of facet rows
ncol	Number of facet columns
x_refline	Vector of x-values at which to draw vertical reference lines
y_refline	Vector of y-values at which to draw horizontal reference lines
identity_line	Set to TRUE to draw 45 degree identity line
x_highlight_min	Vector of min x-values at which to start draw rectangle annotation
x_highlight_max	Vector of max x-values at which to end rectangle annotation
y_highlight_min	Vector of min y-values at which to start draw rectangle annotation
y_highlight_max	Vector of max y-values at which to end rectangle annotation
y_error_min	Variable to use as minimum values for error bars around y at each value of x
y_error_max	Variable to use as maximum values for error bars around y at each value of x
y_error_width	Width of error bars

Value

A ggplot2 plot object.

Author(s)

Saannidhya Rawat

See Also

Other plots: bcat_plt_area(), bcat_plt_bar(), bcat_plt_box(), bcat_plt_coef(), bcat_plt_diag(), bcat_plt_hist(), bcat_plt_line(), bcat_plt_ts()

Examples

```
library(ggplot2)

# basic scatter plot
bcat_plt_point(df = iris,
               x = Sepal.Length,
               y = Sepal.Width,
               x_lab = "Length",
               y_lab = "Width",
               title = "Sepal Width vs Length")

# scatter plot with LOESS fit line
bcat_plt_point(df = iris,
               x = Sepal.Length,
               y = Sepal.Width,
               x_lab = "Length",
               y_lab = "Width",
               smooth = TRUE,
               method = "loess",
               title = "Sepal Width vs Length",
               subtitle = "Loess Fit")

 # scatter plot with faceting and LM fit lines
 bcat_plt_point(df = iris,
                x = Sepal.Length,
                y = Sepal.Width,
                color = Species,
                facet = vars(Species),
                smooth = TRUE,
                x_lab = "Length",
                y_lab = "Width",
                legend_lab = NULL,
                title = "Sepal Width vs Length by Species",
                subtitle = "Linear Fit",
                nrow = 1)
```

bcat_plt_ts *Time series plot utility*

Description

Create a time series visualization with UC styling. Supports decomposition into trend/seasonal/remainder, ACF/PACF display, and recession shading.

Usage

```
bcat_plt_ts(
  df,
  x = NULL,
  y = NULL,
  color = NULL,
  decompose = FALSE,
  acf = FALSE,
  recession_bars = NULL,
  x_lab = ggplot2::waiver(),
  y_lab = ggplot2::waiver(),
  title = ggplot2::waiver(),
  subtitle = ggplot2::waiver(),
  caption = ggplot2::waiver(),
  legend_lab = ggplot2::waiver(),
  legend_position = "bottom",
  legend_hide = FALSE,
  x_scale = NULL,
  y_scale = NULL,
  color_scale = scale_colour_UC(),
  layer_points = FALSE,
  x_highlight_min = NULL,
  x_highlight_max = NULL,
  y_refline = NULL
)
```

Arguments

df	A data frame with date and value columns.
x	Date column.
y	Value column.
color	Variable for color aesthetic.
decompose	Logical. Show 4-panel decomposition? Default is FALSE.
acf	Logical. Show ACF/PACF side-by-side? Default is FALSE.
recession_bars	Data frame with `start` and `end` date columns. Default is NULL.
x_lab	Label for x-axis.

y_lab	Label for y-axis.
title	Plot title.
subtitle	Plot subtitle.
caption	Plot caption.
legend_lab	Legend title.
legend_position	Legend position.
legend_hide	Logical. Hide legend?
x_scale	scale_x_ function.
y_scale	scale_y_ function.
color_scale	scale_color_ function.
layer_points	Logical. Show points on line? Default is FALSE.
x_highlight_min	Date(s) at which to start highlight region(s).
x_highlight_max	Date(s) at which to end highlight region(s).
y_refline	Horizontal reference line(s).

Value

A ggplot or patchwork object.

Author(s)

Saannidhya Rawat

See Also

Other plots: bcat_plt_area(), bcat_plt_bar(), bcat_plt_box(), bcat_plt_coef(), bcat_plt_diag(), bcat_plt_hist(), bcat_plt_line(), bcat_plt_point()

Examples

```
library(ggplot2)
library(scales)

bcat_plt_ts(economics, x = date, y = unemploy,
            y_scale = scale_y_continuous(labels = comma_format()))
```

`bcat_reg_table` *UC-branded regression table*

Description

Produce publication-quality regression tables with UC styling. Wraps `modelsummary::modelsummary()` with UC defaults for formatting, colors, and statistical conventions.

Usage

```
bcat_reg_table(
  models,
  stars = c(`*` = 0.1, `**` = 0.05, `***` = 0.01),
  se_type = "default",
  coef_rename = NULL,
  gof_map = c("nobs", "r.squared", "adj.r.squared", "statistic"),
  caption = NULL,
  footer = NULL,
  font_size = 12,
  header_bg_color = palette_UC[["UC Red"]],
  header_txt_color = palette_UC[["White"]],
  striped = TRUE,
  doc_type = NULL,
  ...
)
```

Arguments

`models`	A model object, or a list of model objects. Supports `lm`, `glm`, `fixest::feols`, `plm`, `ivreg`, and any model supported by `broom::tidy()`.
`stars`	Named numeric vector for significance stars. Set to `FALSE` to suppress stars. Default: `c("*" = 0.1, "**" = 0.05, "***" = 0.01)`.
`se_type`	Character. Standard error type passed to `modelsummary`'s `vcov` argument. Options: `"default"`, `"HC1"`, `"HC3"`, or a function/named list. Default is `"default"`.
`coef_rename`	Named character vector to rename coefficients. If `NULL` (default), coefficients are auto-cleaned to Title Case.
`gof_map`	Character vector of goodness-of-fit statistics to include. Default: `c("nobs", "r.squared", "adj.r.squared", "statistic")`.
`caption`	Character. Table caption.
`footer`	Character. Table footnote.
`font_size`	Numeric. Font size. Default is 12.
`header_bg_color`	Background color for header. Default is UC Red.
`header_txt_color`	Text color for header. Default is white.

striped	Logical. Zebra striping? Default is TRUE.
doc_type	Character. Force output format. Auto-detected if NULL.
...	Additional arguments passed to modelsummary::modelsummary().

Value

A formatted table object.

Author(s)

Saannidhya Rawat

See Also

Other tables: bcat_cor_table(), bcat_fmt_style_table(), bcat_sum_table()

Examples

```
# Single model
m1 <- lm(mpg ~ wt + hp, data = mtcars)
bcat_reg_table(m1)

# Compare models side-by-side
m2 <- lm(mpg ~ wt + hp + cyl, data = mtcars)
bcat_reg_table(list("Base" = m1, "Extended" = m2))
```

---

bcat_setup_rmd *Defaults for Rmd Setup Chunk*

Description

Sets default code chunk options and configures options for kable tables. Call at the start of Rmd file in "setup" chunk.

Usage

```
bcat_setup_rmd()
```

Value

No return value, called for side effects on knitr chunk defaults and table-related options.

Author(s)

Saannidhya Rawat

See Also

Other utilities: bcat_extract_metadata(), bcat_import_data(), bcat_new_quarto(), bcat_source_rmd()

Examples

```
# Call in your Rmd setup chunk:
bcat_setup_rmd()
```

`bcat_source_rmd` *Source Rmd Files*

Description

Utility function to run all code chunks in an Rmd file.

Usage

```
bcat_source_rmd(rmd)
```

Arguments

`rmd`	character. Rmd file from which to run all code chunks.

Value

An invisible list, as returned by `base::source()`, containing the value of the last evaluated expression and whether it was visible.

Author(s)

Saannidhya Rawat

Source

https://stackoverflow.com/questions/10966109/how-to-source-r-markdown-file-like-sourcemyfile-r

See Also

Other utilities: `bcat_extract_metadata()`, `bcat_import_data()`, `bcat_new_quarto()`, `bcat_setup_rmd()`

bcat_sum_table *UC-branded summary statistics table*

Description

Produce a descriptive statistics table with UC styling. Displays mean, SD, min, median, max, N, and percent missing for numeric columns.

Usage

```
bcat_sum_table(
  df,
  by = NULL,
  stats = c("mean", "sd", "min", "median", "max", "n", "pct_missing"),
  digits = 2,
  caption = NULL,
  footer = NULL,
  font_size = 12,
  header_bg_color = palette_UC[["UC Red"]],
  header_txt_color = palette_UC[["White"]],
  striped = TRUE,
  doc_type = NULL,
  ...
)
```

Arguments

`df`	A data frame, tibble, or data.table.
`by`	Character. Column name for grouped summaries. Default is NULL.
`stats`	Character vector of statistics to compute. Default includes all.
`digits`	Integer. Number of decimal places. Default is 2.
`caption`	Character. Table caption.
`footer`	Character. Table footnote.
`font_size`	Numeric. Font size. Default is 12.
`header_bg_color`	Background color for header. Default is UC Red.
`header_txt_color`	Text color for header. Default is white.
`striped`	Logical. Zebra striping? Default is TRUE.
`doc_type`	Character. Force output format. Auto-detected if NULL.
`...`	Additional arguments passed to table formatting.

Value

A formatted table object.

Author(s)

Saannidhya Rawat

See Also

Other tables: `bcat_cor_table()`, `bcat_fmt_style_table()`, `bcat_reg_table()`

Examples

```
bcat_sum_table(mtcars[, c("mpg", "wt", "hp")])
bcat_sum_table(mtcars[, c("mpg", "wt", "cyl")], by = "cyl")
```

---

bcat_xaringan   *UC Xaringan presentation*

Description

An R Markdown output format for remark.js slides (HTML).

Usage

```
bcat_xaringan(css = NULL, after_body = NULL, ...)
```

Arguments

`css`	A vector of CSS file paths. If `NULL` (default), function will use the css files included with the package.
`after_body`	relative path to HTML document to include after body of each slide. If `NULL` (default), function will use the insert-logo.html file included with the package.
`...`	arguments passed to `xaringan::moon_reader`

Value

An `rmarkdown::output_format` object for rendering Xaringan presentations.

Author(s)

Saannidhya Rawat

palette_OkabeIto *Color palette proposed by Okabe and Ito & UC*

Description

Two color palettes taken from the article "Color Universal Design" by Okabe and Ito. The variant `palette_OkabeIto` contains a gray color, while `palette_OkabeIto_black` contains black instead. `palette_OkabeIto_light` contains a lightened version of the `palette_OkabeIto` colors. Note: To extract hex colors, convert palette using `as.vector`.

Usage

```
palette_OkabeIto

palette_OkabeIto_black

palette_OkabeIto_light
```

Format

An object of class `character` of length 8.

An object of class `character` of length 8.

An object of class `character` of length 8.

Value

A named character vector of hexadecimal color values. The exported objects provide the standard, black-substituted, and lightened Okabe-Ito variants, respectively.

See Also

Other palettes: `palette_UC`, `scale_OkabeIto()`, `scale_UC()`

palette_UC *UC Color Palette*

Description

Official University of Cincinnati brand colors.

Usage

```
palette_UC
```

Format

An object of class character of length 17.

Details

The palette combines the current primary colors from UC's visual identity system with the expanded palette recommended for data visualization. Canonical brand names are included alongside a small set of legacy aliases used by earlier versions of this package. Note: To extract hex colors, convert palette using as.vector.

Value

A named character vector of hexadecimal University of Cincinnati brand colors.

See Also

Other palettes: palette_OkabeIto, scale_OkabeIto(), scale_UC()

---

scale_OkabeIto *Okabe-Ito color scale*

Description

This is a color-blind friendly, qualitative scale with eight different colors from the colorblindr package (https://github.com/clauswilke/colorblindr).

Arguments

use_black	If TRUE, scale includes black, otherwise includes gray.
order	Numeric vector listing the order in which the colors should be used. Default is 1:8.
darken	Relative amount by which the scale should be darkened (for positive values) or lightened (for negative values).
alpha	Alpha transparency level of the color. Default is no transparency.
...	common discrete scale parameters: name, breaks, labels, na.value, limits, guide, and aesthetics. see ggplot2::discrete_scale for more details.

Value

A discrete ggplot2 scale object. The helper aliases return the same scale with aesthetics preset for colour or fill mappings.

See Also

Other palettes: palette_OkabeIto, palette_UC, scale_UC()

Examples

```
library(ggplot2)
ggplot(iris, aes(Sepal.Length, Sepal.Width, color = Species)) +
  geom_point() + scale_color_OkabeIto()
ggplot(iris, aes(Sepal.Length, fill = Species)) +
  geom_density(alpha = 0.7) + scale_fill_OkabeIto(order = c(1, 3, 5))
```

`scale_UC` *UC color scale*

Description

Apply the official University of Cincinnati expanded palette to ggplot graphics. The default ordering prioritizes the primary UC colors, followed by expanded brand accents intended for data visualization.

Arguments

`order`	Numeric vector listing the order in which the colors should be used.
`darken`	Relative amount by which the scale should be darkened (for positive values) or lightened (for negative values).
`alpha`	Alpha transparency level of the color. Default is no transparency.
`...`	common discrete scale parameters: `name`, `breaks`, `labels`, `na.value`, `limits`, `guide`, and `aesthetics`. see `ggplot2::discrete_scale` for more details.

Value

A discrete `ggplot2` scale object. The helper aliases return the same scale with `aesthetics` preset for colour or fill mappings.

See Also

Other palettes: `palette_OkabeIto`, `palette_UC`, `scale_OkabeIto()`

Examples

```
library(ggplot2)
ggplot(iris, aes(Sepal.Length, Sepal.Width, color = Species)) +
  geom_point() + scale_color_UC()
ggplot(iris, aes(Sepal.Length, fill = Species)) +
  geom_density(alpha = 0.7) + scale_fill_UC()
```

scale_x_dt *Date Scale*

Description

Wrapper for ggplot2::scale_x_date that generates labels for data variable on x-axis based on input data.

Usage

```
scale_x_dt(
  df,
  x,
  break_by = "year",
  round_unit = "year",
  labels = scales::date_format("'%y"),
  ...
)
```

Arguments

df	data set used for plotting
x	date variable mapped to x-axis
break_by	a character string specifying a time unit to use for axis breaks
round_unit	a character string specifying a time unit or a multiple of a unit to be rounded to for the axis breaks. Valid base units are second, minute, hour, day, week, month, bimonth, quarter, season, halfyear and year.
labels	format to use for displaying labels
...	additional arguments passed to ggplot2::scale_x_date

Value

A ggplot2 date scale object created by ggplot2::scale_x_date().

Author(s)

Saannidhya Rawat

Examples

```
library(ggplot2)

# use default ggplot2 scale
bcat_plt_line(df = economics,
              x = date,
              y = unemploy,
```

```
               y_ref = 10000)

# use scale_x_dt to break by every 5 years
bcat_plt_line(df = economics,
              x = date,
              y = unemploy,
              y_ref = 10000,
              x_scale = scale_x_dt(economics, date, round_unit = "5 years"))
```

`set_UC_geoms` *UC default geoms*

Description

Apply UC default colors to ggplot2 geoms. Note: only applies during session.

Usage

```
set_UC_geoms()
```

Value

No return value, called for side effects on `ggplot2` geom defaults for the current R session.

Author(s)

Saannidhya Rawat

`theme_UC` *Various UC themes for use with ggplot charts*

Description

These functions allow you to add University of Cincinnati themes to ggplot graphics using the official UC color system and digital typography fallbacks.

Usage

```
theme_UC(
  border = TRUE,
  legend_position = c("bottom", "right"),
  legend_hide = FALSE
)

theme_UC_hgrid(border = FALSE, ...)

theme_UC_vgrid(border = FALSE, ...)

theme_UC_nogrid(border = FALSE, ...)
```

Arguments

`border`	If `FALSE`, removes outer border from plot.
`legend_position`	legend position. "bottom" or "right"
`legend_hide`	If `TRUE`, hides legend in final plot.
`...`	Additional arguments passed into `theme_UC`

Details

The main theme is `theme_UC`, while `theme_UC_hgrid` and `theme_UC_vgrid` are used for plots that benefit from a single grid direction. `theme_UC_nogrid` removes all gridlines.

Value

A complete `ggplot2` theme object. The exported variants return horizontal-grid, vertical-grid, or no-grid versions of the base theme.

Author(s)

Saannidhya Rawat

Examples

```
library(ggplot2)

p <- ggplot(
  data = iris,
  mapping = aes(x = Petal.Width, y = Petal.Length, color = Species)
) +
  geom_jitter(size = 1.5) +
  labs(
    x = "Petal Width",
    y = "Petal Length",
    title = "Iris Example",
    subtitle = "Width vs Length by Species",
    caption = "Note: This is a caption"
  ) +
  facet_wrap(~Species)

p + theme_UC()
p + theme_UC_hgrid()
p + theme_UC_vgrid()
```

UC_html_document   *Rbearcat HTML document*

Description

An R Markdown output format for HTML documents.

Usage

```
UC_html_document(
  number_sections = FALSE,
  toc = TRUE,
  toc_depth = 2,
  css = NULL,
  ...
)
```

Arguments

`number_sections`	logical. Set to `FALSE` to leave sections unnumbered.
`toc`	logical. set to `FALSE` to excluded the table of contents side-bar.
`toc_depth`	number of layers in TOC.
`css`	relative path to a css formatting document. If `NULL` (default), function will use the style.css file included with the package.
`...`	other options passed to `bookdown::html_document2`. See `html_document2` and `html_document` for all available options.

Value

An `rmarkdown::output_format` object for rendering HTML documents.

Author(s)

Saannidhya Rawat

UC_pdf_document *Rbearcat PDF document*

Description

An R Markdown output format for PDF documents.

Usage

```
UC_pdf_document(
  number_sections = FALSE,
  toc = FALSE,
  toc_depth = 2,
  highlight_bw = FALSE,
  in_header = NULL,
  ...
)
```

Arguments

number_sections	logical. Set to TRUE to number sections.
toc	logical. set to TRUE to include a table of contents.
toc_depth	number of layers in TOC.
highlight_bw	set to TRUE to convert colors in syntax highlighted code blocks to grayscale.
in_header	relative path to a tex formatting document included in the preamble. If NULL (default), function will use the preamble.tex file included with the package.
...	other options passed to bookdown::pdf_document2. See pdf_document2 and pdf_document for all available options.

Value

An rmarkdown::output_format object for rendering PDF documents.

Author(s)

Saannidhya Rawat

UC_word_document *Rbearcat word document*

Description

An R Markdown output format for Word documents (docx).

Usage

```
UC_word_document(toc = FALSE, toc_depth = 2, reference_docx = NULL, ...)
```

Arguments

toc	logical. set to TRUE to include a table of contents.
toc_depth	number of layers in TOC.
reference_docx	relative path to a docx reference document. If NULL (default), function will use the style.docx file included with the package.
...	other options passed to bookdown::word_document2. See word_document2 and word_document for all available options.

Value

An rmarkdown::output_format object for rendering Word documents.

Author(s)

Saannidhya Rawat

util_git_check_uncommitted *Check if git repo has uncommitted changes*

Description

Check if git repo has uncommitted changes

Usage

```
util_git_check_uncommitted(path = ".", dir_ignore = "docs/")
```

Arguments

path	the path to the directory to check.
dir_ignore	name of subdirectory to skip over for uncommitted files. should generally be the output directory for rmarkdown rendered files (e.g., "docs/" for bookdown projects.)

Value

Invisible NULL if path is not a Git repository or has no relevant uncommitted changes. Throws an error otherwise.

Author(s)

Saannidhya Rawat

---

util_git_commit_id *Get git commit ID*

Description

Get git commit ID

Usage

```
util_git_commit_id(path = ".")
```

Arguments

path	the path to the directory

Value

A length-one character vector containing the current Git SHA, or NULL if path is not a Git repository.

Author(s)

Saannidhya Rawat

Index