Package ‘ggblanket’

February 23, 2024

Title Simplify 'ggplot2' Visualisation

Version 6.0.0

Description Simplify 'ggplot2' visualisation with 'ggblanket' wrapper functions.

License MIT + file LICENSE

URL https://davidhodge931.github.io/ggblanket/,
     https://github.com/davidhodge931/ggblanket

BugReports https://github.com/davidhodge931/ggblanket/issues

Imports dplyr (>= 1.0.4), farver,forcats,ggplot2 (>= 3.5.0), grid,
       hms (>= 0.5.0), lubridate (>= 1.7.8), magrittr,purr,rlang
       (>= 1.1.0), scales (>= 1.3.0), snakecase,stringr (>= 1.3.0),
       tidy (>= 1.0.0), tidyselect (>= 1.2.0), viridisLite (>= 0.4.0)

Suggests hexbin, isoband, knitr, palmerpenguins, patchwork, quantreg,
       rmarkdown, sf, testthat (>= 3.0.0), tibble, vdiffr

VignetteBuilder knitr

Config/Needs/website concaveman, corrr, deldir, distributional,
       farver, geomtextpath, ggbeeswarm, ggblend, ggdensity, ggdist,
       ggeasy, ggforce, ggh4x, gghighlight, ggiraph, ggnewscale,
       grepel, gggridges, glue, mdthemes, plotly, RColorBrewer,
       showtext, sysfonts

Config/testthat/edition 3

Encoding UTF-8

Language en-GB

RoxygenNote 7.3.1

NeedsCompilation no

Author David Hodge [aut, cre, cph] (<https://orcid.org/0000-0002-3868-7501>)

Maintainer David Hodge <davidhodge931@gmail.com>

Repository CRAN

Date/Publication 2024-02-23 21:40:02 UTC
### R topics documented:

- `aes_contrast` .......................................................... 3
- `blue` ..................................................................... 4
- `darkness` ................................................................ 5
- `dark_mode_b` .............................................................. 5
- `dark_mode_n` .............................................................. 6
- `dark_mode_r` .............................................................. 7
- `dark_mode_t` .............................................................. 8
- `gg_area` ................................................................... 8
- `gg_bar` .................................................................... 14
- `gg_bin_2d` ................................................................ 19
- `gg_blanket` ............................................................... 25
- `gg_boxplot` ............................................................... 30
- `gg_col` .................................................................... 36
- `gg_contour` ............................................................... 41
- `gg_contour_filled` ..................................................... 46
- `gg_crossbar` .............................................................. 52
- `gg_density` ............................................................... 57
- `gg_density_2d` .......................................................... 63
- `gg_density_2d_filled` ................................................. 68
- `gg_errorbar` .............................................................. 74
- `gg_freqpoly` ............................................................. 79
- `gg_function` ............................................................. 85
- `gg_hex` ................................................................... 90
- `gg_histogram` .......................................................... 95
- `gg_jitter` .................................................................. 101
- `gg_label` ................................................................ 107
- `gg_line` .................................................................. 112
- `gg_linerange` ........................................................... 118
- `gg_path` ................................................................ 123
- `gg_point` ................................................................ 129
- `gg_pointrange` ........................................................ 134
- `gg_polygon` ............................................................. 140
- `gg_qq` .................................................................... 145
- `gg_quantile` ............................................................. 151
- `gg_raster` ................................................................. 156
- `gg_rect` .................................................................. 161
- `gg_ribbon` ............................................................... 167
- `gg_rug` ................................................................... 173
- `gg_segment` .............................................................. 178
- `gg_sf` .................................................................... 183
- `gg_smooth` .............................................................. 189
- `gg_step` .................................................................. 194
- `gg_text` .................................................................. 200
- `gg_tile` ................................................................... 205
- `gg_violin` ................................................................. 211
- `greyness` ................................................................ 216
## aes_contrast

A colour aesthetic that automatically contrasts with fill.

### Description

A colour aesthetic that automatically contrasts with fill. Can be spliced into `ggplot2::aes` with `rlang::!!!.`.

### Usage

```r
aes_contrast(col_pal = c("black", "white"))
```

### Arguments

- **col_pal**
  
  A vector of a dark colour and then a light colour. Defaults to `c("black", "white")`. Use `lightness`, `greyness` or `darkness` with the applicable `*_mode_*` theme.

### Value

An aesthetic

### Examples

```r
library(palmerpenguins)
library(dplyr)
library(ggplot2)
library(stringr)

penguins |> 
```
blue

A blue colour

Description
A blue colour derived from viridisLite::mako(9)[5]

Usage
blue

Format
An object of class character of length 1.
**darkness**

**Value**

A character vector.

**Examples**

```r
ggplot() + geom_point(aes(x = x, y = y))
```

**Description**

A vector of colours used in the dark_mode_* themes.

**Usage**

darkness

**Format**

An object of class character of length 3.

**Value**

A character vector.

**Examples**

```r
ggplot() + geom_point(aes(x = x, y = y))
```

**dark_mode_b**  
*Dark mode theme with bottom legend*

**Description**

Dark mode theme for a ggplot visualisation with bottom legend. It uses the colours from darkness.

**Usage**

dark_mode_b(base_size = 11, base_family = "", x_title = TRUE, y_title = TRUE)

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>base_size</td>
<td>The base size of the text. Defaults to 11.</td>
</tr>
<tr>
<td>base_family</td>
<td>The base family of the text. Defaults to &quot;&quot;.</td>
</tr>
<tr>
<td>x_title</td>
<td>TRUE or FALSE whether to have a x axis title. Defaults to TRUE.</td>
</tr>
<tr>
<td>y_title</td>
<td>TRUE or FALSE whether to have a y axis title. Defaults to TRUE.</td>
</tr>
</tbody>
</table>
Value
A ggplot theme.

Examples
library(palmerpenguins)
library(ggplot2)

penguins |> 
gg_point(
  x = flipper_length_mm,
  y = body_mass_g,
  col = species,
  mode = dark_mode_b()
)

dark_mode_n

Dark mode theme with no legend

Description
Dark mode theme for a ggplot visualisation with no legend. It uses the colours from darkness.

Usage
dark_mode_n(base_size = 11, base_family = "", x_title = TRUE, y_title = TRUE)

Arguments
base_size The base size of the text. Defaults to 11.
base_family The base family of the text. Defaults to "".
x_title TRUE or FALSE whether to have a x axis title. Defaults to TRUE.
y_title TRUE or FALSE whether to have a y axis title. Defaults to TRUE.

Value
A ggplot theme.

Examples
library(palmerpenguins)
library(ggplot2)

penguins |> 
gg_jitter(
  x = species,
  y = body_mass_g,
dark_mode_r

    col = species,
    mode = dark_mode_r()

Dark mode theme with right legend

Description

Dark mode theme for a ggplot visualisation with legend at right. It uses the colours from darkness.

Usage

dark_mode_r(base_size = 11, base_family = "", x_title = TRUE, y_title = TRUE)

Arguments

base_size      The base size of the text. Defaults to 11.
base_family    The base family of the text. Defaults to "."

x_title        TRUE or FALSE whether to have a x axis title. Defaults to TRUE.
y_title        TRUE or FALSE whether to have a y axis title. Defaults to TRUE.

Value

A ggplot theme.

Examples

library(palmerpenguins)
library(ggplot2)

penguins |> 
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = species,
    mode = dark_mode_r()
  )
**dark_mode_t**  
*Dark mode theme with top legend*

**Description**

Dark mode theme for a ggplot visualisation with top legend. It uses the colours from *darkness*.

**Usage**

```
dark_mode_t(base_size = 11, base_family = "", x_title = TRUE, y_title = TRUE)
```

**Arguments**

- `base_size`: The base size of the text. Defaults to 11.
- `base_family`: The base family of the text. Defaults to "".
- `x_title`: TRUE or FALSE whether to have a x axis title. Defaults to TRUE.
- `y_title`: TRUE or FALSE whether to have a y axis title. Defaults to TRUE.

**Value**

A ggplot theme.

**Examples**

```r
library(palmerpenguins)
library(ggplot2)

penguins |>  
  gg_point(  
    x = flipper_length_mm,  
    y = body_mass_g,  
    col = species,  
    mode = dark_mode_t()  
  )
```

---

**gg_area**  
*Area ggplot*

**Description**

Create an area ggplot with a wrapper around `ggplot2::ggplot() + geom_area()`. 
Usage

gg_area(
  data = NULL,
  ..., 
  stat = "align",
  position = "stack",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
  x_title = NULL,
  x_transform = NULL,
  y_breaks = NULL,
  y_expand = NULL,
  y_expand_limits = NULL,
  y_labels = NULL,
  y_limits = NULL,
  y_oob = scales::oob_keep,
  y_position = "left",
  y_title = NULL,
  y_transform = NULL,
  col_breaks = NULL,
  col_expand_limits = NULL,
  col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

data
A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat
A statistical transformation to use on the data. A snakecase character string of a
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position
A position adjustment. A snakecase character string of a ggproto Position sub-
class object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).
coord  A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode  A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x   Unquoted x aesthetic variable.
xmin  Unquoted xmin aesthetic variable.
xmax  Unquoted xmax aesthetic variable.
xend  Unquoted xend aesthetic variable.
y   Unquoted y aesthetic variable.
ymin  Unquoted ymin aesthetic variable.
ymax  Unquoted ymax aesthetic variable.
yend  Unquoted yend aesthetic variable.
z   Unquoted z aesthetic variable.
col  Unquoted col aesthetic variable.
alpha  Unquoted alpha aesthetic variable.
facet  Unquoted facet aesthetic variable.
facet2  Unquoted facet2 aesthetic variable.
group  Unquoted group aesthetic variable.
subgroup  Unquoted subgroup aesthetic variable.
label  Unquoted label aesthetic variable.
text  Unquoted text aesthetic variable.
sample  Unquoted sample aesthetic variable.
mapping  Set of additional aesthetic mappings within ggplot2::aes() for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.
x_breaks, y_breaks  A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.
x_expanded, y_expanded  Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).
x_expanded_limits, y_expanded_limits  For a continuous variable, any values that the limits should encompass (e.g. 0).
x_labels, y_labels  A function that takes the breaks as inputs (e.g. \(\text{stringr::str_to_sentence(x)}\) or scales::label_comma()), or a vector of labels.
x_limits, y_limits  A vector of length 2 to determine the limits of the axis.
x_oob, y_oob  For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using
y_position = "top" with a *mode_* theme, add caption = "" or caption = "\n".

x_title, y_title
Axis title string. Use "" for no title.

x_transform, y_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10())
or character string of this minus the transform_prefix (e.g. "log10").

col_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector
of breaks.

colexpand_limits
For a continuous variable, any values that the limits should encompass (e.g. \0).

col_labels
A function that takes the breaks as inputs (e.g. \(\text{\texttt{x}}\) stringr::str_to_sentence(x)
or scales::label_comma()), or a vector of labels.

col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a scales::oob_* function of how to handle
values outside of limits. Defaults to scales::oob_keep.

col_pal
Colours to use. A character vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
For a continuous variable, a scales::rescale() function.

col_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. De-
defaults to FALSE, which colours in a gradient.

col_title
Axis title string. Use "" for no title.

col_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10())
or character string of this minus the transform_prefix (e.g. "log10").

alpha_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector
of breaks.

alpha_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. \0).

alpha_labels
A function that takes the breaks as inputs (e.g. \(\text{\texttt{x}}\) stringr::str_to_sentence(x)
or scales::label_comma()), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow
The number of columns and rows for the legend guide.

alpha_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
A vector of length 2 to determine the limits of the axis.

alpha_oob
For a continuous scale variable, a scales::oob_* function of how to handle
values outside of limits. Defaults to scales::oob_keep.
**alpha_pal**  
Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

**alpha_pal_na**  
Alpha value to use for the NA value. A integer between 0 and 1.

**alpha_title**  
Axis title string. Use "" for no title.

**alpha_transform**  
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_prefix (e.g. "log10").

**facet_axes**  
Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

**facet_axis_labels**  
Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

**facet_labels**  
A function that takes the breaks as inputs (e.g. \((x)\) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

**facet_labels_position**  
When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

**facet_labels_switch**  
When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

**facet_layout**  
Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

**facet_ncol**  
The number of columns of facets. Only applies to a facet layout of "wrap".

**facet_nrow**  
The number of rows of facets. Only applies to a facet layout of "wrap".

**facet_scales**  
Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

**facet_space**  
When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

**title**  
Title string.

**subtitle**  
Subtitle string.

**caption**  
Caption title string.

**titles_to_case**  
A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

**Value**  
A ggplot object.
Examples

```r
library(ggplot2)
library(dplyr)
economics |>
  gg_area(
    x = date,
    y = unemploy,
    y_title = "Unemployment",
  )
```

---

**gg_bar**  
*Bar ggplot*

**Description**

Create a bar ggplot with a wrapper around `ggplot2::ggplot()` + `geom_bar()`.

**Usage**

```r
gg_bar(
  data = NULL,
  ..., 
  stat = "count",
  position = "stack",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
```
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand Limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand Limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

data A data frame or tibble.
... Other arguments passed to within a params list in layer().
stat A statistical transformation to use on the data. A snakecase character string of a
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position A position adjustment. A snakecase character string of a ggproto Position sub-
class object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).
coord A coordinate system. A coord_*() function that outputs a constructed ggproto
Coord subclass object (e.g. ggplot2::coord_cartesian()).
mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()).
        This argument adds the theme with side-effects, as the gg_* function will re-
        moves selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme
        on to the output of gg_*.
x Unquoted x aesthetic variable.
xmin Unquoted xmin aesthetic variable.
xmax Unquoted xmax aesthetic variable.
xend Unquoted xend aesthetic variable.
y Unquoted y aesthetic variable.
ymin Unquoted ymin aesthetic variable.
ymax Unquoted ymax aesthetic variable.
yend Unquoted yend aesthetic variable.
z Unquoted z aesthetic variable.
col Unquoted col aesthetic variable.
alpha Unquoted alpha aesthetic variable.
facet Unquoted facet aesthetic variable.
facet2 Unquoted facet2 aesthetic variable.
group Unquoted group aesthetic variable.
subgroup Unquoted subgroup aesthetic variable.
label  Unquoted label aesthetic variable.
text   Unquoted text aesthetic variable.
sample Unquoted sample aesthetic variable.
mapping Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

\textit{x_breaks, y_breaks}
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

\textit{x_expand, y_expand}
Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. c(0, 0)).

\textit{x_expand_limits, y_expand_limits}
For a continuous variable, any values that the limits should encompass (e.g. 0).

\textit{x_labels, y_labels}
A function that takes the breaks as inputs (e.g. \texttt{\textbackslash (x) stringr::str_to_sentence(x)} or scales::label_comma()), or a vector of labels.

\textit{x_limits, y_limits}
A vector of length 2 to determine the limits of the axis.

\textit{x_oob, y_oob}
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

\textit{x_position, y_position}
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".

\textit{x_title, y_title}
Axis title string. Use "" for no title.

\textit{x_transform, y_transform}
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

\textit{col_breaks}
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

\textit{col_expand_limits}
For a continuous variable, any values that the limits should encompass (e.g. 0).

\textit{col_labels}
A function that takes the breaks as inputs (e.g. \texttt{\textbackslash (x) stringr::str_to_sentence(x)} or scales::label_comma()), or a vector of labels.

\textit{col_legend_ncol, col_legend_nrow}
The number of columns and rows for the legend guide.

\textit{col_legend_rev}
Reverse the elements of the legend guide. Defaults to FALSE.

\textit{col_limits}
A vector of length 2 to determine the limits of the axis.

\textit{col_oob}
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

\textit{col_pal}
Colours to use. A character vector of hex codes (or names).

\textit{col_pal_na}
Colour to use for NA values. A character vector of a hex code (or name).
For a continuous variable, a `scales::rescale()` function.

For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_prefix (e.g. "log10").

A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

For a continuous variable, any values that the limits should encompass (e.g. 0).

A function that takes the breaks as inputs (e.g. `(x) stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

The number of columns and rows for the legend guide.

Reverse the elements of the legend guide. Defaults to FALSE.

A vector of length 2 to determine the limits of the axis.

For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

Alpha value to use for the NA value. A integer between 0 and 1.

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_prefix (e.g. "log10").

Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

A function that takes the breaks as inputs (e.g. `(x) stringr::str_to_sentence(x)`), or a named vector of labels (e.g. c("value1" = "label1", ...)).

When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

The number of columns of facets. Only applies to a facet layout of "wrap".

The number of rows of facets. Only applies to a facet layout of "wrap".
**facet_scales**  Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

**facet_space**  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

**title**  Title string.

**subtitle**  Subtitle string.

**caption**  Caption title string.

**titles_to_case**  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

**Value**

A ggplot object.

**Examples**

```r
library(ggplot2)
library(dplyr)
library(palmerpenguins)

penguins |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  gg_bar(
    y = species,
    col = sex,
    position = position_dodge(preserve = "single"),
    width = 0.75,
  )
```

---

**gg_bin_2d**  Bin_2d ggplot

**Description**

Create a bin2d ggplot with a wrapper around ggplot2::ggplot() + geom_bin_2d().

**Usage**

```r
gg_bin_2d(
  data = NULL,
  ...,
  stat = "bin2d",
  position = "identity",
)```
coord = ggplot2::coord_cartesian(clip = "off"),
mode = NULL,
x = NULL,
xmin = NULL,
xmax = NULL,
xend = NULL,
y = NULL,
ymin = NULL,
ymax = NULL,
yend = NULL,
z = NULL,
col = NULL,
alpha = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
Arguments

data A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A _mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()).

This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.
x Unquoted x aesthetic variable.
xmin Unquoted xmin aesthetic variable.
xmax Unquoted xmax aesthetic variable.
xend Unquoted xend aesthetic variable.
y Unquoted y aesthetic variable.
ymin Unquoted ymin aesthetic variable.
ymax Unquoted ymax aesthetic variable.
yend Unquoted yend aesthetic variable.
z Unquoted z aesthetic variable.
col Unquoted col aesthetic variable.
alpha Unquoted alpha aesthetic variable.
facet Unquoted facet aesthetic variable.
facet2 Unquoted facet2 aesthetic variable.
group Unquoted group aesthetic variable.
subgroup Unquoted subgroup aesthetic variable.
label Unquoted label aesthetic variable.
text Unquoted text aesthetic variable.
sample Unquoted sample aesthetic variable.
mapping Set of additional aesthetic mappings within ggplot2::aes() for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.
x_breaks, y_breaks
   A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.
x_expand, y_expand
   Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).
x_expand_limits, y_expand_limits
   For a continuous variable, any values that the limits should encompass (e.g. 0).
x_labels, y_labels
   A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.
x_limits, y_limits
   A vector of length 2 to determine the limits of the axis.
x_oob, y_oob
   For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
x_position, y_position
   The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".
x_title, y_title
   Axis title string. Use "" for no title.
x_transform, y_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_prefix (e.g. "log10").

col_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

col_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

col_pal
Colours to use. A vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
For a continuous variable, a `scales::rescale()` function.

col_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title
Axis title string. Use "" for no title.

col_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_prefix (e.g. "log10").

alpha_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

alpha_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow
The number of columns and rows for the legend guide.

alpha_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
A vector of length 2 to determine the limits of the axis.

alpha_oob
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

alpha_pal
Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na
Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title
Axis title string. Use "" for no title.

alpha_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_prefix (e.g. "log10").
facet_axes  Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels  Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels  A function that takes the breaks as inputs (e.g. \(\text{x} \) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position  When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch  When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout  Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol  The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow  The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales  Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title  Title string.

subtitle  Subtitle string.

caption  Caption title string.

titles_to_case  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value

A ggplot object.

Examples

library(ggplot2)
library(dplyr)

diamonds |>
  gg_bin_2d(
    x = carat,
    y = price,
  )
Description

Create a blanket ggplot with a wrapper around `ggplot2::ggplot()` + `layer()` with `geom_blank()` defaults. This function underlies all other `gg_*` functions. It contains a `geom` argument for maximum flexibility.

Usage

```r
gg_blanket(
  data = NULL,
  ..., 
  geom = "blank",
  stat = "identity",
  position = "identity",
  coord = NULL,
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
  x_title = NULL,
  x_transform = NULL,
  )
```
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

data          A data frame or tibble.

...           Other arguments passed to within a params list in layer().

geom          A geometric object to display the data. A snakecase character string of a ggproto 
              Geom subclass object minus the Geom prefix (e.g. "point").

stat          A statistical transformation to use on the data. A snakecase character string of a ggproto 
              Stat subclass object minus the Stat prefix (e.g. "identity").

position      A position adjustment. A snakecase character string of a ggproto Position subclass object 
              minus the Position prefix (e.g. "identity"), or a position_*() function that outputs 
              a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord         A coordinate system. A coord_*() function that outputs a constructed ggproto 
              Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode          A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). 
              This argument adds the theme with side-effects, as the gg_* function will re-
              moves selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme 
              on to the output of gg_*.

x             Unquoted x aesthetic variable.

xmin          Unquoted xmin aesthetic variable.

xmax          Unquoted xmax aesthetic variable.

xend          Unquoted xend aesthetic variable.

y             Unquoted y aesthetic variable.

ymin          Unquoted ymin aesthetic variable.

ymax          Unquoted ymax aesthetic variable.

yend          Unquoted yend aesthetic variable.

z             Unquoted z aesthetic variable.

col           Unquoted col1 aesthetic variable.

alpha         Unquoted alpha aesthetic variable.

facet         Unquoted facet aesthetic variable.

facet2        Unquoted facet2 aesthetic variable.

group         Unquoted group aesthetic variable.

subgroup      Unquoted subgroup aesthetic variable.

label         Unquoted label aesthetic variable.

text          Unquoted text aesthetic variable.

sample        Unquoted sample aesthetic variable.

mapping       Set of additional aesthetic mappings within ggplot2::aes() for non-supported 
              aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evalua-

x_breaks, y_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

x_expand, y_expand
Padding to the limits with the `ggplot2:::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

x_expand_limits, y_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels
A function that takes the breaks as inputs (e.g. `\(x\)` `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

x_limits, y_limits
A vector of length 2 to determine the limits of the axis.

x_oob, y_oob
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y_position = "top"` with a *_mode_* theme, add `caption = ""` or `caption = "\n"`.

x_title, y_title
Axis title string. Use "" for no title.

x_transform, y_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()` or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

col_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
A function that takes the breaks as inputs (e.g. `\(x\)` `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

col_pal
Colours to use. A character vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
For a continuous variable, a `scales::rescale()` function.

col_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title
Axis title string. Use "" for no title.

col_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()` or character string of this minus the transform_ prefix (e.g. "log10").
facet_axes  Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels  Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels  A function that takes the breaks as inputs (e.g. \( (x) \) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position  When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch  When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout  Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol  The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow  The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales  Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

alpha_breaks  A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

alpha_expand_limits  For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels  A function that takes the breaks as inputs (e.g. \( (x) \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow  The number of columns and rows for the legend guide.

alpha_legend_rev  Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits  A vector of length 2 to determine the limits of the axis.

alpha_oob  For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na  Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title  Axis title string. Use "" for no title.

alpha_transform  For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").
gg_boxplot

<table>
<thead>
<tr>
<th>title</th>
<th>Title string.</th>
</tr>
</thead>
<tbody>
<tr>
<td>subtitle</td>
<td>Subtitle string.</td>
</tr>
<tr>
<td>caption</td>
<td>Caption title string.</td>
</tr>
<tr>
<td>titles_to_case</td>
<td>A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.</td>
</tr>
</tbody>
</table>

**Value**

A ggplot object.

**Examples**

```r
library(ggplot2)
library(dplyr)
library(palmerpenguins)

penguins |> tidyr::drop_na(sex) |> mutate(across(sex, \(x) stringr::str_to_sentence(x))) |> gg_blanket(
  geom = "violin",
  stat = "ydensity",
  position = "dodge",
  x = sex,
  y = body_mass_g,
  col = sex,
  facet = species,
  mode = grey_mode_b(),
)
```

---

**Description**

Create a boxplot ggplot with a wrapper around ggplot2::ggplot() + geom_boxplot().

**Usage**

```r
gg_boxplot(
  data = NULL,
  ...,
  stat = "boxplot",
  position = "dodge2",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
)```
xmin = NULL,
 xmax = NULL,
 xend = NULL,
 y = NULL,
 ymin = NULL,
 ymax = NULL,
 yend = NULL,
 z = NULL,
 col = NULL,
 alpha = NULL,
 facet = NULL,
 facet2 = NULL,
 group = NULL,
 subgroup = NULL,
 label = NULL,
 text = NULL,
 sample = NULL,
 mapping = NULL,
 x_breaks = NULL,
 x_expand = NULL,
 x_expand_limits = NULL,
 x_labels = NULL,
 x_limits = NULL,
 x_oob = scales::oob_keep,
 x_position = "bottom",
 x_title = NULL,
 x_transform = NULL,
 y_breaks = NULL,
 y_expand = NULL,
 y_expand_limits = NULL,
 y_labels = NULL,
 y_limits = NULL,
 y_oob = scales::oob_keep,
 y_position = "left",
 y_title = NULL,
 y_transform = NULL,
 col_breaks = NULL,
 col_expand = NULL,
 col_expand_limits = NULL,
 col_label = NULL,
 col_legend_ncol = NULL,
 col_legend_nrow = NULL,
 col_legend_rev = FALSE,
 col_limits = NULL,
 col_oob = scales::oob_keep,
 col_pal = NULL,
 col_pal_na = "darkgrey",
 col_rescale = scales::rescale(),
 col_steps = FALSE,
Arguments

data A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.
Unquoted x_max aesthetic variable.

Unquoted x_end aesthetic variable.

Unquoted y aesthetic variable.

Unquoted y_min aesthetic variable.

Unquoted y_max aesthetic variable.

Unquoted y_end aesthetic variable.

Unquoted z aesthetic variable.

Unquoted col aesthetic variable.

Unquoted alpha aesthetic variable.

Unquoted facet aesthetic variable.

Unquoted facet2 aesthetic variable.

Unquoted group aesthetic variable.

Unquoted subgroup aesthetic variable.

Unquoted label aesthetic variable.

Unquoted text aesthetic variable.

Unquoted sample aesthetic variable.

Set of additional aesthetic mappings within ggplot2::aes() for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).

For a continuous variable, any values that the limits should encompass (e.g. 0).

A function that takes the breaks as inputs (e.g. \((x)\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

A vector of length 2 to determine the limits of the axis.

For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a _mode_* theme, add caption = "" or caption = "\n".

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_* prefix (e.g. "log10").
**col_breaks**
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

**col_expand_limits**
For a continuous variable, any values that the limits should encompass (e.g. 0).

**col_labels**
A function that takes the breaks as inputs (e.g. \( (x) \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

**col_legend_ncol, col_legend_nrow**
The number of columns and rows for the legend guide.

**col_legend_rev**
Reverse the elements of the legend guide. Defaults to FALSE.

**col_limits**
A vector of length 2 to determine the limits of the axis.

**col_oob**
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

**col_pal**
Colours to use. A character vector of hex codes (or names).

**col_pal_na**
Colour to use for NA values. A character vector of a hex code (or name).

**col_rescale**
For a continuous variable, a scales::rescale() function.

**col_steps**
For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

**col_title**
Axis title string. Use "" for no title.

**col_transform**
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_prefix (e.g. "log10").

**alpha_breaks**
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

**alpha_expand_limits**
For a continuous variable, any values that the limits should encompass (e.g. 0).

**alpha_labels**
A function that takes the breaks as inputs (e.g. \( (x) \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

**alpha_legend_ncol, alpha_legend_nrow**
The number of columns and rows for the legend guide.

**alpha_legend_rev**
Reverse the elements of the legend guide. Defaults to FALSE.

**alpha_limits**
A vector of length 2 to determine the limits of the axis.

**alpha_oob**
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

**alpha_pal**
Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

**alpha_pal_na**
Alpha value to use for the NA value. A integer between 0 and 1.

**alpha_title**
Axis title string. Use "" for no title.

**alpha_transform**
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_prefix (e.g. "log10").

**facet_axes**
Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".
facet_axis_labels
Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels
A function that takes the breaks as inputs (e.g. \(\texttt{x}\) stringr::str_to_sentence(x)),
or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position
When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch
When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout
Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol
The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow
The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales
Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space
When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title
Title string.

subtitle
Subtitle string.

caption
Caption title string.

titles_to_case
A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value
A ggplot object.

Examples

library(ggplot2)
library(dplyr)
library(palmerpenguins)

penguins |>
  tidyr::drop_na(sex) |>
  mutate(across(sex, \(\texttt{x}\) stringr::str_to_sentence(x))) |>
  gg_boxplot(
    x = flipper_length_mm,
    y = sex,
    col = species,
    mode = light_mode_b(),
  )
**gg_col**

*Col ggplot*

**Description**

Create a col ggplot with a wrapper around `ggplot2::ggplot()` + `geom_col()`.

**Usage**

```r
gg_col(
  data = NULL,
  ..., 
  stat = "identity",
  position = "stack",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
  x_title = NULL,
  x_transform = NULL,
  y_breaks = NULL,
  y_expand = NULL,
  y_expand_limits = NULL,
```
```r
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
```
Arguments

- **data**: A data frame or tibble.
- **...**: Other arguments passed to within a `params` list in `layer()`.
- **stat**: A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
- **position**: A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. `ggplot2::position_identity()`).
- **coord**: A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. `ggplot2::coord_cartesian()`).
- **mode**: A *_mode_* theme (e.g. `grey_mode_b()`, `grey_mode_r()`, or `dark_mode_r()`). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.
- **x**: Unquoted x aesthetic variable.
- **xmin**: Unquoted xmin aesthetic variable.
- **xmax**: Unquoted xmax aesthetic variable.
- **xend**: Unquoted xend aesthetic variable.
- **y**: Unquoted y aesthetic variable.
- **ymin**: Unquoted ymin aesthetic variable.
- **ymax**: Unquoted ymax aesthetic variable.
- **yend**: Unquoted yend aesthetic variable.
- **z**: Unquoted z aesthetic variable.
- **col**: Unquoted col aesthetic variable.
- **alpha**: Unquoted alpha aesthetic variable.
- **facet**: Unquoted facet aesthetic variable.
- **facet2**: Unquoted facet2 aesthetic variable.
- **group**: Unquoted group aesthetic variable.
- **subgroup**: Unquoted subgroup aesthetic variable.
- **label**: Unquoted label aesthetic variable.
- **text**: Unquoted text aesthetic variable.
- **sample**: Unquoted sample aesthetic variable.
- **mapping**: Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.
- **x_breaks, y_breaks**: A scales::breaks_* function (e.g. `scales::breaks_pretty()`), or a vector of breaks.
- **x_expand, y_expand**: Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. c(0, 0)).
x\_expand\_limits, y\_expand\_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

x\_labels, y\_labels
A function that takes the breaks as inputs (e.g. \(\texttt{x})\) stringr::str\_to\_sentence(x) or scales::label\_comma()), or a vector of labels.

x\_limits, y\_limits
A vector of length 2 to determine the limits of the axis.

x\_oob, y\_oob
For a continuous scale variable, a scales::oob\_\* function of how to handle values outside of limits. Defaults to scales::oob\_keep.

x\_position, y\_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y\_position = "top" with a \*\_mode\_\* theme, add caption = "" or caption = "\n".

x\_title, y\_title
Axis title string. Use "" for no title.

x\_transform, y\_transform
For a numeric scale, a transformation object (e.g. scales::transform\_log10()) or character string of this minus the transform\_ prefix (e.g. "log10").

col\_breaks
A scales::breaks\_\* function (e.g. scales::breaks\_pretty()), or a vector of breaks.

col\_expand\_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col\_labels
A function that takes the breaks as inputs (e.g. \(\texttt{x})\) stringr::str\_to\_sentence(x) or scales::label\_comma()), or a vector of labels.

col\_legend\_ncol, col\_legend\_nrow
The number of columns and rows for the legend guide.

col\_legend\_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col\_limits
A vector of length 2 to determine the limits of the axis.

col\_oob
For a continuous scale variable, a scales::oob\_\* function of how to handle values outside of limits. Defaults to scales::oob\_keep.

col\_pal
Colours to use. A character vector of hex codes (or names).

col\_pal\_na
Colour to use for NA values. A character vector of a hex code (or name).

col\_rescale
For a continuous variable, a scales::rescale() function.

col\_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col\_title
Axis title string. Use "" for no title.

col\_transform
For a numeric scale, a transformation object (e.g. scales::transform\_log10()) or character string of this minus the transform\_ prefix (e.g. "log10").

alpha\_breaks
A scales::breaks\_\* function (e.g. scales::breaks\_pretty()), or a vector of breaks.

alpha\_expand\_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha\_labels
A function that takes the breaks as inputs (e.g. \(\texttt{x})\) stringr::str\_to\_sentence(x) or scales::label\_comma()), or a vector of labels.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>alpha_legend_ncol, alpha_legend_nrow</td>
<td>The number of columns and rows for the legend guide.</td>
</tr>
<tr>
<td>alpha_legend_rev</td>
<td>Reverse the elements of the legend. Defaults to FALSE.</td>
</tr>
<tr>
<td>alpha_limits</td>
<td>A vector of length 2 to determine the limits of the axis.</td>
</tr>
<tr>
<td>alpha_oob</td>
<td>For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.</td>
</tr>
<tr>
<td>alpha_pal</td>
<td>Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.</td>
</tr>
<tr>
<td>alpha_pal_na</td>
<td>Alpha value to use for the NA value. A integer between 0 and 1.</td>
</tr>
<tr>
<td>alpha_title</td>
<td>Axis title string. Use &quot;&quot; for no title.</td>
</tr>
<tr>
<td>alpha_transform</td>
<td>For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_prefix (e.g. &quot;log10&quot;).</td>
</tr>
<tr>
<td>facet_axes</td>
<td>Whether to add interior axes and ticks with &quot;margins&quot;, &quot;all&quot;, &quot;all_x&quot;, or &quot;all_y&quot;.</td>
</tr>
<tr>
<td>facet_axis_labels</td>
<td>Whether to add interior axis labels with &quot;margins&quot;, &quot;all&quot;, &quot;all_x&quot;, or &quot;all_y&quot;.</td>
</tr>
<tr>
<td>facet_labels</td>
<td>A function that takes the breaks as inputs (e.g. (x) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c(&quot;value1&quot; = &quot;label1&quot;, ...)).</td>
</tr>
<tr>
<td>facet_labels_position</td>
<td>When the facet layout is &quot;wrap&quot;, the position of the facet labels. Either &quot;top&quot;, &quot;right&quot;, &quot;bottom&quot; or &quot;left&quot;.</td>
</tr>
<tr>
<td>facet_labels_switch</td>
<td>When the facet layout is &quot;grid&quot;, whether to switch the facet labels to the opposite side of the plot. Either &quot;x&quot;, &quot;y&quot; or &quot;both&quot;.</td>
</tr>
<tr>
<td>facet_layout</td>
<td>Whether the layout is to be &quot;wrap&quot; or &quot;grid&quot;. If NULL and a single facet (or facet2) argument is provided, then defaults to &quot;wrap&quot;. If NULL and both facet and facet2 arguments are provided, defaults to &quot;grid&quot;.</td>
</tr>
<tr>
<td>facet_ncol</td>
<td>The number of columns of facets. Only applies to a facet layout of &quot;wrap&quot;.</td>
</tr>
<tr>
<td>facet_nrow</td>
<td>The number of rows of facets. Only applies to a facet layout of &quot;wrap&quot;.</td>
</tr>
<tr>
<td>facet_scales</td>
<td>Whether facet scales should be &quot;fixed&quot; across facets, &quot;free&quot; in both directions, or free in just one direction (i.e. &quot;free_x&quot; or &quot;free_y&quot;). Defaults to &quot;fixed&quot;.</td>
</tr>
<tr>
<td>facet_space</td>
<td>When the facet layout is &quot;grid&quot; and facet scales are not &quot;fixed&quot;, whether facet space should be &quot;fixed&quot; across facets, &quot;free&quot; to be proportional in both directions, or free to be proportional in just one direction (i.e. &quot;free_x&quot; or &quot;free_y&quot;). Defaults to &quot;fixed&quot;.</td>
</tr>
<tr>
<td>title</td>
<td>Title string.</td>
</tr>
<tr>
<td>subtitle</td>
<td>Subtitle string.</td>
</tr>
<tr>
<td>caption</td>
<td>Caption title string.</td>
</tr>
<tr>
<td>titles_to_case</td>
<td>A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.</td>
</tr>
</tbody>
</table>
Value

A ggplot object.

Examples

```r
library(ggplot2)
library(dplyr)
library(palmerpenguins)

penguins |>  # penguins
  tidyr::drop_na(sex) |>  # drop missing values for sex
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>  # convert sex to sentence
  group_by(sex, species) |>  # group by sex and species
  summarise(across(flipper_length_mm, \(x) mean(x, na.rm = TRUE))) |>  # calculate mean flipper length
  gg_col(  # add contour plot
    x = flipper_length_mm,  # x variable is flipper length
    y = species,  # y variable is species
    col = sex,  # color by sex
    position = position_dodge(preserve = "single"),  # dodge positions
    width = 0.75  # set width
  )
```

Description

Create a contour ggplot with a wrapper around `ggplot2::ggplot()` + `geom_contour()`.

Usage

```r
gg_contour(
  data = NULL,
  ...,  # additional arguments
  stat = "contour",  # use contour statistic
  position = "identity",  # use identity position
  coord = ggplot2::coord_cartesian(clip = "off"),  # use cartesian coordinates
  mode = NULL,  # mode
  x = NULL,  # x variable
  xmin = NULL,  # minimum x value
  xmax = NULL,  # maximum x value
  xend = NULL,  # x end value
  y = NULL,  # y variable
  ymin = NULL,  # minimum y value
  ymax = NULL,  # maximum y value
  yend = NULL,  # y end value
```
z = NULL,
col = NULL,
alpha = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
Arguments

data A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_.*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>yend</code></td>
<td>Unquoted <code>yend</code> aesthetic variable.</td>
</tr>
<tr>
<td><code>z</code></td>
<td>Unquoted <code>z</code> aesthetic variable.</td>
</tr>
<tr>
<td><code>col</code></td>
<td>Unquoted <code>col</code> aesthetic variable.</td>
</tr>
<tr>
<td><code>alpha</code></td>
<td>Unquoted <code>alpha</code> aesthetic variable.</td>
</tr>
<tr>
<td><code>facet</code></td>
<td>Unquoted <code>facet</code> aesthetic variable.</td>
</tr>
<tr>
<td><code>facet2</code></td>
<td>Unquoted <code>facet2</code> aesthetic variable.</td>
</tr>
<tr>
<td><code>group</code></td>
<td>Unquoted <code>group</code> aesthetic variable.</td>
</tr>
<tr>
<td><code>subgroup</code></td>
<td>Unquoted <code>subgroup</code> aesthetic variable.</td>
</tr>
<tr>
<td><code>label</code></td>
<td>Unquoted <code>label</code> aesthetic variable.</td>
</tr>
<tr>
<td><code>text</code></td>
<td>Unquoted <code>text</code> aesthetic variable.</td>
</tr>
<tr>
<td><code>sample</code></td>
<td>Unquoted <code>sample</code> aesthetic variable.</td>
</tr>
<tr>
<td><code>mapping</code></td>
<td>Set of additional aesthetic mappings within <code>ggplot2::aes()</code> for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.</td>
</tr>
<tr>
<td><code>x_breaks, y_breaks</code></td>
<td>A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_pretty()</code>), or a vector of breaks.</td>
</tr>
<tr>
<td><code>x_expand, y_expand</code></td>
<td>Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code>).</td>
</tr>
<tr>
<td><code>xExpand_limits, yExpand_limits</code></td>
<td>For a continuous variable, any values that the limits should encompass (e.g. 0).</td>
</tr>
<tr>
<td><code>x_labels, y_labels</code></td>
<td>A function that takes the breaks as inputs (e.g. (x) <code>stringr::str_to_sentence(x)</code> or <code>scales::label_comma()</code>), or a vector of labels.</td>
</tr>
<tr>
<td><code>x_limits, y_limits</code></td>
<td>A vector of length 2 to determine the limits of the axis.</td>
</tr>
<tr>
<td><code>x_oob, y_oob</code></td>
<td>For a continuous scale variable, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code>.</td>
</tr>
<tr>
<td><code>x_position, y_position</code></td>
<td>The position of the axis (i.e. &quot;left&quot;, &quot;right&quot;, &quot;bottom&quot; or &quot;top&quot;). If using <code>y_position = &quot;top&quot;</code> with a <code>_mode_</code> theme, add <code>caption = &quot;&quot;</code> or <code>caption = &quot;\n&quot;</code>.</td>
</tr>
<tr>
<td><code>x_title, y_title</code></td>
<td>Axis title string. Use &quot;&quot; for no title.</td>
</tr>
<tr>
<td><code>x_transform, y_transform</code></td>
<td>For a numeric scale, a transformation object (e.g. <code>scales::transform_log10()</code>) or character string of this minus the <code>transform_</code> prefix (e.g. &quot;log10&quot;).</td>
</tr>
<tr>
<td><code>col_breaks</code></td>
<td>A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_pretty()</code>), or a vector of breaks.</td>
</tr>
<tr>
<td><code>col_expand_limits</code></td>
<td>For a continuous variable, any values that the limits should encompass (e.g. 0).</td>
</tr>
<tr>
<td><code>col_labels</code></td>
<td>A function that takes the breaks as inputs (e.g. (x) <code>stringr::str_to_sentence(x)</code> or <code>scales::label_comma()</code>), or a vector of labels.</td>
</tr>
</tbody>
</table>
col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

col_pal
Colours to use. A character vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_steps
For a continuous variable, a `scales::rescale()` function.

col_title
Axis title string. Use "" for no title.

col_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

alpha_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
A function that takes the breaks as inputs (e.g. \(x\)` stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow
The number of columns and rows for the legend guide.

alpha_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
A vector of length 2 to determine the limits of the axis.

alpha_oob
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

alpha_pal
Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na
Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title
Axis title string. Use "" for no title.

alpha_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes
Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels
Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels
A function that takes the breaks as inputs (e.g. \(x\)` stringr::str_to_sentence(x)`), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position
When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".
When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

The number of columns of facets. Only applies to a facet layout of "wrap".

The number of rows of facets. Only applies to a facet layout of "wrap".

Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

Title string.

Subtitle string.

Caption title string.

A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

A `ggplot` object.

library(ggplot2)
library(dplyr)

ggplot2::faithful |>
  gg_contour(
    x = waiting,
    y = eruptions,
    z = density,
  )
Usage

```r
gg_contour_filled(
  data = NULL,
  ..., 
  stat = "contour_filled",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
  x_title = NULL,
  x_transform = NULL,
  y_breaks = NULL,
  y_expand = NULL,
  y_expand_limits = NULL,
  y_labels = NULL,
  y_limits = NULL,
  y_oob = scales::oob_keep,
  y_position = "left",
  y_title = NULL,
  y_transform = NULL,
  col_breaks = NULL,
  col_expand_limits = NULL,
  col_labels = NULL,
)```
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

data              A data frame or tibble.

...               Other arguments passed to within a params list in layer().

stat              A statistical transformation to use on the data. A snakecase character string of a
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position          A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).
coord A coordinate system. A `coord_*()` function that outputs a constructed ggproto Coord subclass object (e.g. `ggplot2::coord_cartesian()`).

mode A *mode_* theme (e.g. `grey_mode_b()`, `grey_mode_r()`, or `dark_mode_r()`). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.

yend Unquoted yend aesthetic variable.

z Unquoted z aesthetic variable.

col Unquoted col aesthetic variable.

alpha Unquoted alpha aesthetic variable.

facet Unquoted facet aesthetic variable.

facet2 Unquoted facet2 aesthetic variable.

group Unquoted group aesthetic variable.

subgroup Unquoted subgroup aesthetic variable.

label Unquoted label aesthetic variable.

text Unquoted text aesthetic variable.

sample Unquoted sample aesthetic variable.

mapping Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks A scales::breaks_* function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

x_expand, y_expand Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. c(0, 0)).

x_expand_limits, y_expand_limits For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels A function that takes the breaks as inputs (e.g. \(\backslash(x)\) stringr::str_to_sentence(x) or `scales::label_comma()`), or a vector of labels.

x_limits, y_limits A vector of length 2 to determine the limits of the axis.

x_oob, y_oob For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to `scales::oob_keep.`
x_position, y_position
   The position of the axis (i.e. "left", "right", "bottom" or "top"). If using
   y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".

x_title, y_title
   Axis title string. Use "" for no title.

x_transform, y_transform
   For a numeric scale, a transformation object (e.g. scales::transform_log10())
   or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks
   A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector
   of breaks.

col_expand_limits
   For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
   A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)
   or scales::label_comma()), or a vector of labels.

col_legend_ncol, col_legend_nrow
   The number of columns and rows for the legend guide.

col_legend_rev
   Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
   A vector of length 2 to determine the limits of the axis.

col_oob
   For a continuous scale variable, a scales::oob_* function of how to handle
   values outside of limits. Defaults to scales::oob_keep.

col_pal
   Colours to use. A character vector of hex codes (or names).

col_pal_na
   Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
   For a continuous variable, a scales::rescale() function.

col_steps
   For a continuous variable, TRUE or FALSE of whether to colour in steps. De-
  faults to FALSE, which colours in a gradient.

col_title
   Axis title string. Use "" for no title.

col_transform
   For a numeric scale, a transformation object (e.g. scales::transform_log10())
   or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks
   A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector
   of breaks.

alpha_expand_limits
   For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
   A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)
   or scales::label_comma()), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow
   The number of columns and rows for the legend guide.

alpha_legend_rev
   Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
   A vector of length 2 to determine the limits of the axis.

alpha_oob
   For a continuous scale variable, a scales::oob_* function of how to handle
   values outside of limits. Defaults to scales::oob_keep.
alpha_pal  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.
alpha_pal_na Alpha value to use for the NA value. A integer between 0 and 1.
alpha_title Axis title string. Use "" for no title.
alpha_transform For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_prefix (e.g. "log10").
facet_axes Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".
facet_axis_labels Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_labels A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)`), or a named vector of labels (e.g. c("value1" = "label1", ...)).
facet_labels_position When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".
facet_labels_switch When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".
facet_layout Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
facet_ncol The number of columns of facets. Only applies to a facet layout of "wrap".
facet_nrow The number of rows of facets. Only applies to a facet layout of "wrap".
facet_scales Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title Title string.
subtitle Subtitle string.
caption Caption title string.
titles_to_case A function to format unspecified titles_to_case. Defaults to `snakecase::to_sentence_case`.

Value
A ggplot object.
Examples

```r
library(ggplot2)
library(dplyr)

faithfuld |>
  gg_contour_filled(
    x = waiting,
    y = eruptions,
    z = density,
    bins = 8,
  )
```

**Description**

Create a crossbar ggplot with a wrapper around `ggplot2::ggplot()` + `geom_crossbar()`.

**Usage**

```r
gg_crossbar(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
)```
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",}
Faceting Arguments

- **facet_labels**: A character vector of facet labels. This can be NULL, a character vector, or a function that returns a character vector.
- **facet_labels_position**: The position of facet labels can be set to "top", "right", "bottom", or "left".
- **facet_labels_switch**: A boolean indicating whether to switch facet labels.
- **facet_layout**: A character vector of layout names or "fill".
- **facet_ncol**: The maximum number of facets per row.
- **facet_nrow**: The maximum number of facets per column.
- **facet_scales**: A name of scales to use in facets.
- **facet_space**: An element of "fixed" or "free", or a function that returns a ggproto Scale subclass object.
- **title**: A character vector of titles.
- **subtitle**: A character vector of subtitles.
- **caption**: A character vector of captions.
- **titles_to_case**: A function that returns a character string with titles in sentence case.

**Arguments**

- **data**: A data frame or tibble.
- **...**: Other arguments passed to within a `params` list in `layer()`.
- **stat**: A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
- **position**: A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_() function that outputs a ggproto Position subclass object (e.g. `ggplot2::position_identity()`).
- **coord**: A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. `ggplot2::coord_cartesian()`).
- **mode**: A *_mode_* theme (e.g. `grey_mode_b()`, `grey_mode_r()`, or `dark_mode_r()`). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.
- **x**: Unquoted x aesthetic variable.
- **xmin**: Unquoted xmin aesthetic variable.
- **xmax**: Unquoted xmax aesthetic variable.
- **xend**: Unquoted xend aesthetic variable.
- **y**: Unquoted y aesthetic variable.
- **ymin**: Unquoted ymin aesthetic variable.
- **ymax**: Unquoted ymax aesthetic variable.
- **yend**: Unquoted yend aesthetic variable.
- **z**: Unquoted z aesthetic variable.
- **col**: Unquoted col aesthetic variable.
- **alpha**: Unquoted alpha aesthetic variable.
- **facet**: Unquoted facet aesthetic variable.
- **facet2**: Unspecified facet2 aesthetic variable.
- **group**: Unquoted group aesthetic variable.
subgroup  Unquoted subgroup aesthetic variable.
label    Unquoted label aesthetic variable.
text     Unquoted text aesthetic variable.
sample   Unquoted sample aesthetic variable.
mapping  Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

x_expand, y_expand
Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

x_expand_limits, y_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels
A function that takes the breaks as inputs (e.g. `\(x\)` `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

x_limits, y_limits
A vector of length 2 to determine the limits of the axis.

x_oob, y_oob  For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y_position = "top"` with a *_mode_* theme, add `caption = ""` or `caption = \"n\"`.

x_title, y_title
Axis title string. Use ““ for no title.

x_transform, y_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

col_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
A function that takes the breaks as inputs (e.g. `\(x\)` `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev
Reverse the elements of the legend guide. Defaults to `FALSE`.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

col_pal
Colours to use. A character vector of hex codes (or names).
col_pal_na  Colour to use for NA values. A character vector of a hex code (or name).
col_rescale For a continuous variable, a scales::rescale() function.
col_steps  For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.
col_title  Axis title string. Use "" for no title.
col_transform For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").
alpha_breaks A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.
alpha_expand_limits For a continuous variable, any values that the limits should encompass (e.g. 0).
alpha_labels A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.
alpha_legend_ncol, alpha_legend_nrow  The number of columns and rows for the legend guide.
alpha_legend_rev  Reverse the elements of the legend guide. Defaults to FALSE.
alpha_limits A vector of length 2 to determine the limits of the axis.
alpha_oob  For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
alpha_pal  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.
alpha_pal_na  Alpha value to use for the NA value. A integer between 0 and 1.
alpha_title  Axis title string. Use "" for no title.
alpha_transform For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").
facet_axes  Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".
facet_axis_labels Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_labels A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).
facet_labels_position When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".
facet_labels_switch When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".
facet_layout  Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
facet_ncol  The number of columns of facets. Only applies to a facet layout of "wrap".
gg_density

facet_nrow  The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales  Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title  Title string.

subtitle  Subtitle string.

caption  Caption title string.

titles_to_case  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value

A ggplot object.

Examples

library(ggplot2)
library(dplyr)
data.frame(
  trt = factor(c(1, 1, 2, 2)),
  resp = c(1, 5, 3, 4),
  group = factor(c(1, 2, 1, 2)),
  upper = c(1.1, 5.3, 3.3, 4.2),
  lower = c(0.8, 4.6, 2.4, 3.6)) |> gg_crossbar(
  x = trt,
  y = resp,
  ymin = lower,
  ymax = upper,
  col = group,
  width = 0.5,
  x_title = "Treatment",
  y_title = "Response",
)

---

gg_density  Density ggplot

Description

Create a density ggplot with a wrapper around ggplot2::ggplot() + geom_density().
Usage

```r
gg_density(
  data = NULL,
  ..., 
  stat = "density",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
  x_title = NULL,
  x_transform = NULL,
  y_breaks = NULL,
  y_expand = NULL,
  y_expand_limits = NULL,
  y_labels = NULL,
  y_limits = NULL,
  y_oob = scales::oob_keep,
  y_position = "left",
  y_title = NULL,
  y_transform = NULL,
  col_breaks = NULL,
  col_expand_limits = NULL,
  col_labels = NULL,
)```
Arguments

**data**

A data frame or tibble.

**...**

Other arguments passed to within a `params` list in `layer()`.

**stat**

A statistical transformation to use on the data. A snakecase character string of a `ggproto` Stat subclass object minus the Stat prefix (e.g. "identity").

**position**

A position adjustment. A snakecase character string of a `ggproto` Position subclass object minus the Position prefix (e.g. "identity"), or a `position_*()` function that outputs a `ggproto` Position subclass object (e.g. `ggplot2::position_identity()`).
coord A coordinate system. A `coord_*()` function that outputs a constructed `ggproto` Coord subclass object (e.g. `ggplot2::coord_cartesian()`).

mode A *_mode_* theme (e.g. `grey_mode_b()`, `grey_mode_r()`, or `dark_mode_r()`). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x Unquoted x aesthetic variable.
xmin Unquoted xmin aesthetic variable.
xmax Unquoted xmax aesthetic variable.
xend Unquoted xend aesthetic variable.
y Unquoted y aesthetic variable.
ymin Unquoted ymin aesthetic variable.
ymax Unquoted ymax aesthetic variable.
yend Unquoted yend aesthetic variable.
z Unquoted z aesthetic variable.
col Unquoted col aesthetic variable.
alpha Unquoted alpha aesthetic variable.
facet Unquoted facet aesthetic variable.
facet2 Unquoted facet2 aesthetic variable.
group Unquoted group aesthetic variable.
subgroup Unquoted subgroup aesthetic variable.
label Unquoted label aesthetic variable.
text Unquoted text aesthetic variable.
sample Unquoted sample aesthetic variable.
mapping Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks A scales::breaks_* function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

x_expand, y_expand Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

x_expand_limits, y_expand_limits For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels A function that takes the breaks as inputs (e.g. `\(x\)` `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

x_limits, y_limits A vector of length 2 to determine the limits of the axis.

x_oob, y_oob For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to `scales::oob_keep`. 
x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using
y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".

x_title, y_title
Axis title string. Use "" for no title.

x_transform, y_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks
A scales::breaks_* function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

col_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
A function that takes the breaks as inputs (e.g. \( \langle x \rangle \) stringr::str_to_sentence(x) or `scales::label_comma()`), or a vector of labels.

col_legends_ncol, col_legends_nrow
The number of columns and rows for the legend guide.

col_legends_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

col_pal
Colours to use. A character vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
For a continuous variable, a scales::rescale() function.

col_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title
Axis title string. Use "" for no title.

col_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks
A scales::breaks_* function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

alpha_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
A function that takes the breaks as inputs (e.g. \( \langle x \rangle \) stringr::str_to_sentence(x) or `scales::label_comma()`), or a vector of labels.

alpha_legends_ncol, alpha_legends_nrow
The number of columns and rows for the legend guide.

alpha_legends_rev
Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
A vector of length 2 to determine the limits of the axis.

alpha_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
alpha_pal  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na  Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title  Axis title string. Use "" for no title.

alpha_transform  For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes  Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels  Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels  A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position  When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch  When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout  Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol  The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow  The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales  Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title  Title string.

subtitle  Subtitle string.

caption  Caption title string.

titles_to_case  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value  A ggplot object.
Examples

```r
library(ggplot2)
library(dplyr)
library(palmerpenguins)

penguins |>
    mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
    tidyr::drop_na(sex) |>
    gg_density(
        x = flipper_length_mm,
        col = species,
        mode = light_mode_t(),
    )
```

Description

Create a density_2d ggplot with a wrapper around `ggplot2::ggplot()` + `geom_density_2d()`.

Usage

```r
gg_density_2d(
    data = NULL,
    ...,
    stat = "density_2d",
    position = "identity",
    coord = ggplot2::coord_cartesian(clip = "off"),
    mode = NULL,
    x = NULL,
    xmin = NULL,
    xmax = NULL,
    xend = NULL,
    y = NULL,
    ymin = NULL,
    ymax = NULL,
    yend = NULL,
    z = NULL,
    col = NULL,
    alpha = NULL,
    facet = NULL,
    facet2 = NULL,
    group = NULL,
    subgroup = NULL,
    label = NULL,
)```
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,


`gg_density_2d`

```r
gg_density_2d
```

Arguments

- **data**: A data frame or tibble.
- **...**: Other arguments passed to within a `params` list in `layer()`.
- **stat**: A statistical transformation to use on the data. A snakecase character string of a `ggproto` Stat subclass object minus the Stat prefix (e.g. "identity").
- **position**: A position adjustment. A snakecase character string of a `ggproto` Position subclass object minus the Position prefix (e.g. "identity"), or a `position_*(...)` function that outputs a `ggproto` Position subclass object (e.g. `ggplot2::position_identity()`).
- **coord**: A coordinate system. A `coord_*(...)` function that outputs a constructed `ggproto` Coord subclass object (e.g. `ggplot2::coord_cartesian()`).
- **mode**: A *_mode_* theme (e.g. `grey_mode_b()`, `grey_mode_r()`, or `dark_mode_r()`). This argument adds the theme with side-effects, as the `gg_*` function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of `gg_*`.
- **x**: Unquoted x aesthetic variable.
- **xmin**: Unquoted xmin aesthetic variable.
- **xmax**: Unquoted xmax aesthetic variable.
- **xend**: Unquoted xend aesthetic variable.
- **y**: Unquoted y aesthetic variable.
- **ymin**: Unquoted ymin aesthetic variable.
- **ymax**: Unquoted ymax aesthetic variable.
- **yend**: Unquoted yend aesthetic variable.
- **z**: Unquoted z aesthetic variable.
- **col**: Unquoted col aesthetic variable.
- **alpha**: Unquoted alpha aesthetic variable.
- **facet**: Unquoted facet aesthetic variable.
- **facet2**: Unquoted facet2 aesthetic variable.

---

**facet_axis_labels**: "margins",
**facet_labels**: NULL,
**facet_labels_position**: "top",
**facet_labels_switch**: NULL,
**facet_layout**: NULL,
**facet_ncol**: NULL,
**facet_nrow**: NULL,
**facet_scales**: "fixed",
**facet_space**: "fixed",
**title**: NULL,
**subtitle**: NULL,
**caption**: NULL,
**titles_to_case**: `snakecase::to_sentence_case`
group  Unquoted group aesthetic variable.
subgroup Unquoted subgroup aesthetic variable.
label  Unquoted label aesthetic variable.
text  Unquoted text aesthetic variable.
sample Unquoted sample aesthetic variable.
mapping Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

**x_breaks, y_breaks**
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

**x_expand, y_expand**
Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

**x_expand_limits, y_expand_limits**
For a continuous variable, any values that the limits should encompass (e.g. 0).

**x_labels, y_labels**
A function that takes the breaks as inputs (e.g. `\(x\) stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

**x_limits, y_limits**
A vector of length 2 to determine the limits of the axis.

**x_oob, y_oob**
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

**x_position, y_position**
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y_position = "top"` with a *_mode_* theme, add `caption = ""` or `caption = "\n"`.

**x_title, y_title**
Axis title string. Use "" for no title.

**x_transform, y_transform**
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the `transform_` prefix (e.g. "log10").

**col_breaks**
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

**col_expand_limits**
For a continuous variable, any values that the limits should encompass (e.g. 0).

**col_labels**
A function that takes the breaks as inputs (e.g. `\(x\) stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

**col_legend_ncol, col_legend_nrow**
The number of columns and rows for the legend guide.

**col_legend_rev**
Reverse the elements of the legend guide. Defaults to FALSE.

**col_limits**
A vector of length 2 to determine the limits of the axis.

**col_oob**
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`. 
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>col_pal</td>
<td>Colours to use. A character vector of hex codes (or names).</td>
</tr>
<tr>
<td>col_pal_na</td>
<td>Colour to use for NA values. A character vector of a hex code (or name).</td>
</tr>
<tr>
<td>col_rescale</td>
<td>For a continuous variable, a <code>scales::rescale()</code> function.</td>
</tr>
<tr>
<td>col_steps</td>
<td>For a continuous variable, TRUE or FALSE of whether to colour in steps.</td>
</tr>
<tr>
<td>col_title</td>
<td>Axis title string. Use &quot;&quot; for no title.</td>
</tr>
<tr>
<td>col_transform</td>
<td>For a numeric scale, a transformation object (e.g. <code>scales::transform_log10()</code>).</td>
</tr>
<tr>
<td>alpha_breaks</td>
<td>A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_pretty()</code>), or a vector of breaks.</td>
</tr>
<tr>
<td>alpha_expand_limits</td>
<td>For a continuous variable, any values that the limits should encompass (e.g. 0).</td>
</tr>
<tr>
<td>alpha_labels</td>
<td>A function that takes the breaks as inputs (e.g. (x) <code>stringr::str_to_sentence(x)</code> or <code>scales::label_comma()</code>), or a vector of labels.</td>
</tr>
<tr>
<td>alpha_legend_ncol, alpha_legend_nrow</td>
<td>The number of columns and rows for the legend guide.</td>
</tr>
<tr>
<td>alpha_legend_rev</td>
<td>Reverse the elements of the legend guide. Defaults to FALSE.</td>
</tr>
<tr>
<td>alpha_limits</td>
<td>A vector of length 2 to determine the limits of the axis.</td>
</tr>
<tr>
<td>alpha_oob</td>
<td>For a continuous scale variable, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code>.</td>
</tr>
<tr>
<td>alpha_pal</td>
<td>Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.</td>
</tr>
<tr>
<td>alpha_pal_na</td>
<td>Alpha value to use for the NA value. A integer between 0 and 1.</td>
</tr>
<tr>
<td>alpha_title</td>
<td>Axis title string. Use &quot;&quot; for no title.</td>
</tr>
<tr>
<td>alpha_transform</td>
<td>For a numeric scale, a transformation object (e.g. <code>scales::transform_log10()</code> or character string of this minus the transform_ prefix (e.g. &quot;log10&quot;).</td>
</tr>
<tr>
<td>facet_axes</td>
<td>Whether to add interior axes and ticks with &quot;margins&quot;, &quot;all&quot;, &quot;all_x&quot;, or &quot;all_y&quot;.</td>
</tr>
<tr>
<td>facet_axis_labels</td>
<td>Whether to add interior axis labels with &quot;margins&quot;, &quot;all&quot;, &quot;all_x&quot;, or &quot;all_y&quot;.</td>
</tr>
<tr>
<td>facet_labels</td>
<td>A function that takes the breaks as inputs (e.g. (x) <code>stringr::str_to_sentence(x)</code>), or a named vector of labels (e.g. c(&quot;value1&quot; = &quot;label1&quot;, ...)).</td>
</tr>
<tr>
<td>facet_labels_position</td>
<td>When the facet layout is &quot;wrap&quot;, the position of the facet labels. Either &quot;top&quot;, &quot;right&quot;, &quot;bottom&quot; or &quot;left&quot;.</td>
</tr>
<tr>
<td>facet_labels_switch</td>
<td>When the facet layout is &quot;grid&quot;, whether to switch the facet labels to the opposite side of the plot. Either &quot;x&quot;, &quot;y&quot; or &quot;both&quot;.</td>
</tr>
<tr>
<td>facet_layout</td>
<td>Whether the layout is to be &quot;wrap&quot; or &quot;grid&quot;. If NULL and a single facet (or facet2) argument is provided, then defaults to &quot;wrap&quot;. If NULL and both facet and facet2 arguments are provided, defaults to &quot;grid&quot;.</td>
</tr>
</tbody>
</table>
facet_ncol  The number of columns of facets. Only applies to a facet layout of "wrap".
facet_nrow  The number of rows of facets. Only applies to a facet layout of "wrap".
facet_scales  Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title  Title string.
subtitle  Subtitle string.
caption  Caption title string.
titles_to_case  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value
A ggplot object.

Examples

library(ggplot2)
library(dplyr)

faithful |>
gg_density_2d(
  x = waiting,
  y = eruptions,
  bins = 8,
)

---

**gg_density_2d_filled**  *Density_2d_filled ggplot*

**Description**

Create a density_2d_filled ggplot with a wrapper around `ggplot2::ggplot() + geom_density_2d_filled()`.

**Usage**

```r
gg_density_2d_filled(
  data = NULL,
  ..., 
  stat = "density_2d_filled",
  position = "identity",
)```
coord = ggplot2::coord_cartesian(clip = "off"),
mode = NULL,
x = NULL,
xmin = NULL,
xmax = NULL,
xend = NULL,
y = NULL,
ymin = NULL,
ymax = NULL,
yend = NULL,
z = NULL,
col = NULL,
alpha = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
xexpand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
yexpand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
Arguments

data A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.
x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.

yend Unquoted yend aesthetic variable.

z Unquoted z aesthetic variable.

col Unquoted col aesthetic variable.

alpha Unquoted alpha aesthetic variable.

facet Unquoted facet aesthetic variable.

facet2 Unquoted facet2 aesthetic variable.

group Unquoted group aesthetic variable.

subgroup Unquoted subgroup aesthetic variable.

label Unquoted label aesthetic variable.

text Unquoted text aesthetic variable.

sample Unquoted sample aesthetic variable.

mapping Set of additional aesthetic mappings within ggplot2::aes() for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

x_expand, y_expand Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).

x_expand_limits, y_expand_limits For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

x_limits, y_limits A vector of length 2 to determine the limits of the axis.

x_oob, y_oob For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

x_position, y_position The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".

x_title, y_title Axis title string. Use "" for no title.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>x_transform</code>, <code>y_transform</code></td>
<td>For a numeric scale, a transformation object (e.g. <code>scales::transform_log10()</code>) or character string of this minus the <code>transform_</code> prefix (e.g. &quot;log10&quot;).</td>
</tr>
<tr>
<td><code>col_breaks</code></td>
<td>A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_pretty()</code>), or a vector of breaks.</td>
</tr>
<tr>
<td><code>col_expand_limits</code></td>
<td>For a continuous variable, any values that the limits should encompass (e.g. 0).</td>
</tr>
<tr>
<td><code>col_labels</code></td>
<td>A function that takes the breaks as inputs (e.g. <code>\(x\)</code> <code>stringr::str_to_sentence(x)</code> or <code>scales::label_comma()</code>) or a vector of labels.</td>
</tr>
<tr>
<td><code>col_legend.ncol</code>, <code>col_legend.nrow</code></td>
<td>The number of columns and rows for the legend guide.</td>
</tr>
<tr>
<td><code>col_legend_rev</code></td>
<td>Reverse the elements of the legend guide. Defaults to FALSE.</td>
</tr>
<tr>
<td><code>col_limits</code></td>
<td>A vector of length 2 to determine the limits of the axis.</td>
</tr>
<tr>
<td><code>col_oob</code></td>
<td>For a continuous scale variable, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code>.</td>
</tr>
<tr>
<td><code>col_pal</code></td>
<td>Colours to use. A character vector of hex codes (or names).</td>
</tr>
<tr>
<td><code>col_pal_na</code></td>
<td>Colour to use for NA values. A character vector of a hex code (or name).</td>
</tr>
<tr>
<td><code>col_rescale</code></td>
<td>For a continuous variable, a <code>scales::rescale()</code> function.</td>
</tr>
<tr>
<td><code>col_steps</code></td>
<td>For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.</td>
</tr>
<tr>
<td><code>col_title</code></td>
<td>Axis title string. Use &quot;&quot; for no title.</td>
</tr>
<tr>
<td><code>col_transform</code></td>
<td>For a numeric scale, a transformation object (e.g. <code>scales::transform_log10()</code>) or character string of this minus the <code>transform_</code> prefix (e.g. &quot;log10&quot;).</td>
</tr>
<tr>
<td><code>alpha_breaks</code></td>
<td>A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_pretty()</code>), or a vector of breaks.</td>
</tr>
<tr>
<td><code>alpha_expand_limits</code></td>
<td>For a continuous variable, any values that the limits should encompass (e.g. 0).</td>
</tr>
<tr>
<td><code>alpha_labels</code></td>
<td>A function that takes the breaks as inputs (e.g. <code>\(x\)</code> <code>stringr::str_to_sentence(x)</code> or <code>scales::label_comma()</code>) or a vector of labels.</td>
</tr>
<tr>
<td><code>alpha_legend.ncol</code>, <code>alpha_legend.nrow</code></td>
<td>The number of columns and rows for the legend guide.</td>
</tr>
<tr>
<td><code>alpha_legend_rev</code></td>
<td>Reverse the elements of the legend guide. Defaults to FALSE.</td>
</tr>
<tr>
<td><code>alpha_limits</code></td>
<td>A vector of length 2 to determine the limits of the axis.</td>
</tr>
<tr>
<td><code>alpha_oob</code></td>
<td>For a continuous scale variable, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code>.</td>
</tr>
<tr>
<td><code>alpha_pal</code></td>
<td>Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.</td>
</tr>
<tr>
<td><code>alpha_pal_na</code></td>
<td>Alpha value to use for the NA value. A integer between 0 and 1.</td>
</tr>
<tr>
<td><code>alpha_title</code></td>
<td>Axis title string. Use &quot;&quot; for no title.</td>
</tr>
<tr>
<td><code>alpha_transform</code></td>
<td>For a numeric scale, a transformation object (e.g. <code>scales::transform_log10()</code>) or character string of this minus the <code>transform_</code> prefix (e.g. &quot;log10&quot;).</td>
</tr>
</tbody>
</table>
gg_density_2d_filled

facet_axes

Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels

Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels

A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position

When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch

When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout

Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol

The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow

The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales

Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space

When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title

Title string.

subtitle

Subtitle string.

caption

Caption title string.

titles_to_case

A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value

A ggplot object.

Examples

library(ggplot2)
library(dplyr)

faithful |>
  gg_density_2d_filled(
    x = waiting,
    y = eruptions,
    bins = 8,
  )
Description

Create an errorbar ggplot with a wrapper around `ggplot2::ggplot()` + `geom_errorbar()`.

Usage

```r
gg_errorbar(
  data = NULL,
  ..., 
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
  x_title = NULL,
  x_transform = NULL,
  y_breaks = NULL,
  y_expand = NULL,
  y_expand = NULL,
  y_expand_limits = NULL,
)```
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case)
## gg_errorbar

### Arguments

- **data**
  A data frame or tibble.

- **...**
  Other arguments passed to within a `params` list in `layer()`.

- **stat**
  A statistical transformation to use on the data. A snakecase character string of a `ggproto` `Stat` subclass object minus the `Stat` prefix (e.g. "identity").

- **position**
  A position adjustment. A snakecase character string of a `ggproto` `Position` subclass object minus the `Position` prefix (e.g. "identity"), or a position_*() function that outputs a `ggproto` Position subclass object (e.g. `ggplot2::position_identity()`).

- **coord**
  A coordinate system. A coord_*() function that outputs a constructed `ggproto` Coord subclass object (e.g. `ggplot2::coord_cartesian()`).

- **mode**
  A *mode_* theme (e.g. `grey_mode_b()`, `grey_mode_r()`, or `dark_mode_r()`). This argument adds the theme with side-effects, as the `gg_*` function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of `gg_*`.

- **x**
  Unquoted x aesthetic variable.

- **xmin**
  Unquoted xmin aesthetic variable.

- **xmax**
  Unquoted xmax aesthetic variable.

- **xend**
  Unquoted xend aesthetic variable.

- **y**
  Unquoted y aesthetic variable.

- **ymin**
  Unquoted ymin aesthetic variable.

- **ymax**
  Unquoted ymax aesthetic variable.

- **yend**
  Unquoted yend aesthetic variable.

- **z**
  Unquoted z aesthetic variable.

- **col**
  Unquoted col aesthetic variable.

- **alpha**
  Unquoted alpha aesthetic variable.

- **facet**
  Unquoted facet aesthetic variable.

- **facet2**
  Unquoted facet2 aesthetic variable.

- **group**
  Unquoted group aesthetic variable.

- **subgroup**
  Unquoted subgroup aesthetic variable.

- **label**
  Unquoted label aesthetic variable.

- **text**
  Unquoted text aesthetic variable.

- **sample**
  Unquoted sample aesthetic variable.

- **mapping**
  Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

- **x_breaks**, **y_breaks**
  A scales::breaks_* function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

- **x_expand**, **y_expand**
  Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. c(0, 0)).
x_expand_limits, y_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels
A function that takes the breaks as inputs (e.g. \( \text{stringr::str_to_sentence}(x) \) or \text{scales::label_comma()}), or a vector of labels.

x_limits, y_limits
A vector of length 2 to determine the limits of the axis.

x_oob, y_oob
For a continuous scale variable, a \text{scales::oob_*} function of how to handle values outside of limits. Defaults to \text{scales::oob_keep}.

x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using \( y\_position = "\text{top}" \) with a *mode_* theme, add \text{caption = ""} or \text{caption = "\n"}.

x_title, y_title
Axis title string. Use "" for no title.

x_transform, y_transform
For a numeric scale, a transformation object (e.g. \text{scales::transform_log10()}) or character string of this minus the \text{transform_*} prefix (e.g. "log10").

col_breaks
A \text{scales::breaks_*} function (e.g. \text{scales::breaks_pretty()}), or a vector of breaks.

col_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
A function that takes the breaks as inputs (e.g. \( \text{stringr::str_to_sentence}(x) \) or \text{scales::label_comma()}), or a vector of labels.

col_legends, col_legends_nrow
The number of columns and rows for the legend guide.

col_legends_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a \text{scales::oob_*} function of how to handle values outside of limits. Defaults to \text{scales::oob_keep}.

col_pal
Colours to use. A character vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
For a continuous variable, a \text{scales::rescale()} function.

col_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title
Axis title string. Use "" for no title.

col_transform
For a numeric scale, a transformation object (e.g. \text{scales::transform_log10()}) or character string of this minus the \text{transform_*} prefix (e.g. "log10").

alpha_breaks
A \text{scales::breaks_*} function (e.g. \text{scales::breaks_pretty()}), or a vector of breaks.

alpha_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
A function that takes the breaks as inputs (e.g. \( \text{stringr::str_to_sentence}(x) \) or \text{scales::label_comma()}), or a vector of labels.
alpha_legend_ncol, alpha_legend_nrow
   The number of columns and rows for the legend guide.
alpha_legend_rev
   Reverse the elements of the legend guide. Defaults to FALSE.
alpha_limits
   A vector of length 2 to determine the limits of the axis.
alpha_oob
   For a continuous scale variable, a scales::oob_* function of how to handle
   values outside of limits. Defaults to scales::oob_keep.
alpha_pal
   Alpha values to use. For a continuous variable, a vector of length 2 between 0
   and 1. For a discrete variable, a vector of integers between 0 and 1.
alpha_pal_na
   Alpha value to use for the NA value. A integer between 0 and 1.
alpha_title
   Axis title string. Use "" for no title.
alpha_transform
   For a numeric scale, a transformation object (e.g. scales::transform_log10())
   or character string of this minus the transform_ prefix (e.g. "log10").
facet_axes
   Whether to add interior axes and ticks with "margins", "all", "all_x", or
   "all_y".
facet_axis_labels
   Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_labels
   A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)),
   or a named vector of labels (e.g. c("value1" = "label1", ...)).
facet_labels_position
   When the facet layout is "wrap", the position of the facet labels. Either "top",
   "right", "bottom" or "left".
facet_labels_switch
   When the facet layout is "grid", whether to switch the facet labels to the oppo-
   site side of the plot. Either "x", "y" or "both".
facet_layout
   Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or
   facet2) argument is provided, then defaults to "wrap". If NULL and both facet
   and facet2 arguments are provided, defaults to "grid".
facet_ncol
   The number of columns of facets. Only applies to a facet layout of "wrap".
facet_nrow
   The number of rows of facets. Only applies to a facet layout of "wrap".
facet_scales
   Whether facet scales should be "fixed" across facets, "free" in both directions,
   or free in just one direction (i.e. "free_x" or "free_y"). Defaults to
   "fixed".
facet_space
   When the facet layout is "grid" and facet scales are not "fixed", whether
   facet space should be "fixed" across facets, "free" to be proportional in both
   directions, or free to be proportional in just one direction (i.e. "free_x" or
   "free_y"). Defaults to "fixed".
title
   Title string.
subtitle
   Subtitle string.
caption
   Caption title string.
titles_to_case
   A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.
Value

A ggplot object.

Examples

```r
library(ggplot2)
library(dplyr)

data.frame(
  trt = factor(c(1, 1, 2, 2)),
  resp = c(1, 5, 3, 4),
  group = factor(c(1, 2, 1, 2)),
  upper = c(1.1, 5.3, 3.3, 4.2),
  lower = c(0.8, 4.6, 2.4, 3.6)
) |> |
gg_errorbar(
  x = trt,
  ymin = lower,
  ymax = upper,
  col = group,
  width = 0.1,
  x_title = "Treatment",
  y_title = "Response",
)
```

Description

Create a freqpoly ggplot with a wrapper around `ggplot2::ggplot() + geom_freqpoly()`.

Usage

```r
gg_freqpoly(
  data = NULL,
  ...
  stat = "bin",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ```
ymin = NULL,
ymax = NULL,
yend = NULL,
z = NULL,
col = NULL,
alpha = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

data A data frame or tibble.
...
Other arguments passed to within a params list in layer().
stat A statistical transformation to use on the data. A snakecase character string of a

ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position A position adjustment. A snakecase character string of a ggproto Position sub-
class object minus the Position prefix (e.g. "identity"), or a position_*() 
function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).
coord A coordinate system. A coord_*() function that outputs a constructed ggproto
Coord subclass object (e.g. ggplot2::coord_cartesian()).
mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()).
This argument adds the theme with side-effects, as the gg_* function will re-
moves selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme 
on to the output of gg_*.
x Unquoted x aesthetic variable.
xmin Unquoted xmin aesthetic variable.
xmax Unquoted xmax aesthetic variable.
xend Unquoted xend aesthetic variable.
y Unquoted y aesthetic variable.
ymin  Unquoted ymin aesthetic variable.
ymax  Unquoted ymax aesthetic variable.
yend  Unquoted yend aesthetic variable.
z     Unquoted z aesthetic variable.
col   Unquoted col aesthetic variable.
alpha Unquoted alpha aesthetic variable.
facet Unquoted facet aesthetic variable.
facet2 Unquoted facet2 aesthetic variable.
group Unquoted group aesthetic variable.
subgroup Unquoted subgroup aesthetic variable.
label Unquoted label aesthetic variable.
text  Unquoted text aesthetic variable.
sample Unquoted sample aesthetic variable.
mapping Set of additional aesthetic mappings within ggplot2::aes() for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

x_expand, y_expand
Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).

x_expand_limits, y_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels
A function that takes the breaks as inputs (e.g. \(x) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

x_limits, y_limits
A vector of length 2 to determine the limits of the axis.

x_oob, y_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a _mode_* theme, add caption = "" or caption = "\n".

x_title, y_title
Axis title string. Use "" for no title.

x_transform, y_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.
col_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

col_pal
Colours to use. A character vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
For a continuous variable, a scales::rescale() function.

col_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title
Axis title string. Use "" for no title.

col_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

alpha_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow
The number of columns and rows for the legend guide.

alpha_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
A vector of length 2 to determine the limits of the axis.

alpha_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal
Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na
Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title
Axis title string. Use "" for no title.

alpha_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes
Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels
Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_labels  A function that takes the breaks as inputs (e.g. \( x \) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position  When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch  When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout  Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol  The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow  The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales  Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title  Title string.

subtitle  Subtitle string.

caption  Caption title string.

titles_to_case  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value
A ggplot object.

Examples

```r
library(ggplot2)
library(dplyr)
library(palmerpenguins)

penguins |>
  mutate(across(sex, \( x \) stringr::str_to_sentence(x))) |>
  gg_freqpoly(
    x = flipper_length_mm,
    col = sex,
    col_title = "",
    mode = light_mode_t(),
  )
```
**gg_function**

Function ggplot

Description

Create a function `ggplot` with a wrapper around `ggplot2::ggplot()` + `geom_function()`.

Usage

```r
gg_function(
  data = NULL,
  ..., 
  stat = "function",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  xexpand = NULL,
  xexpand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
  x_title = NULL,
  x_transform = NULL,
  y_breaks = NULL,
  y_expand = NULL,
  y_expand_limits = NULL,
)```

y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expanded_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expanded_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case)
Arguments

data  A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.

yend Unquoted yend aesthetic variable.

z Unquoted z aesthetic variable.

col Unquoted col aesthetic variable.

alpha Unquoted alpha aesthetic variable.

facet Unquoted facet aesthetic variable.

facet2 Unquoted facet2 aesthetic variable.

group Unquoted group aesthetic variable.

subgroup Unquoted subgroup aesthetic variable.

label Unquoted label aesthetic variable.

text Unquoted text aesthetic variable.

sample Unquoted sample aesthetic variable.

mapping Set of additional aesthetic mappings within ggplot2::aes() for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

x_expand, y_expand Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).
x_expand_limits, y_expand_limits

For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels

A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

x_limits, y_limits

A vector of length 2 to determine the limits of the axis.

x_oob, y_oob

For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

x_position, y_position

The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a _theme, add caption = "" or caption = "\n".

x_title, y_title

Axis title string. Use "" for no title.

x_transform, y_transform

For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks

A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

col_expand_limits

For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels

A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

col_legends, col_legend_nrow, col_legends_rev

The number of columns and rows for the legend guide.

col_legends_rev

Reverse the elements of the legend guide. Defaults to FALSE.

col_limits

A vector of length 2 to determine the limits of the axis.

col_oob

For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

col_pal

Colours to use. A character vector of hex codes (or names).

col_pal_na

Colour to use for NA values. A character vector of a hex code (or name).

col_rescale

For a continuous variable, a scales::rescale() function.

col_steps

For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title

Axis title string. Use "" for no title.

col_transform

For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

col_limits

A scales::transform_* function (e.g. scales::transform_pretty()), or a vector of breaks.

col_limits

For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels

A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.
alpha_legend_ncol, alpha_legend_nrow

The number of columns and rows for the legend guide.

alpha_legend_rev

Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits

A vector of length 2 to determine the limits of the axis.

alpha_oob

For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal

Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na

Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title

Axis title string. Use "" for no title.

alpha_transform

For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes

Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels

Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels

A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position

When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch

When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout

Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol

The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow

The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales

Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space

When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title

Title string.

subtitle

Subtitle string.

caption

Caption title string.

titles_to_case

A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.
Value

A ggplot object.

Examples

library(ggplot2)
library(dplyr)

gg_function(
  fun = \( \text{dnorm}(x, \text{mean} = 0, \text{sd} = 5) \),
  x_limits = qnorm(p = c(0.005, 0.995), mean = 0, sd = 5),
  y_expand_limits = 0,
)

Description

Create a hex ggplot with a wrapper around `ggplot2::ggplot()` + `geom_hex()`.

Usage

```r
gg_hex(
  data = NULL,
  ..., 
  stat = "binhex",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
)```
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
Arguments

data A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.

yend Unquoted yend aesthetic variable.

z Unquoted z aesthetic variable.

col Unquoted col aesthetic variable.

alpha Unquoted alpha aesthetic variable.

facet Unquoted facet aesthetic variable.
gg_hex

facet2 Unquoted facet2 aesthetic variable.
group Unquoted group aesthetic variable.
subgroup Unquoted subgroup aesthetic variable.
label Unquoted label aesthetic variable.
text Unquoted text aesthetic variable.
sample Unquoted sample aesthetic variable.
mapping Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

\texttt{x\_breaks, y\_breaks}
A `scales::breaks_` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

\texttt{x\_expand, y\_expand}
Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

\texttt{x\_expand\_limits, y\_expand\_limits}
For a continuous variable, any values that the limits should encompass (e.g. 0).

\texttt{x\_labels, y\_labels}
A function that takes the breaks as inputs (e.g. \texttt{\(x\) stringr::str_to_sentence(x)} or `scales::label_comma()`), or a vector of labels.

\texttt{x\_limits, y\_limits}
A vector of length 2 to determine the limits of the axis.

\texttt{x\_oob, y\_oob}
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

\texttt{x\_position, y\_position}
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y\_position = "top"` with a *_mode_* theme, add `caption = ""` or `caption = "\n"`.

\texttt{x\_title, y\_title}
Axis title string. Use "" for no title.

\texttt{x\_transform, y\_transform}
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_ prefix (e.g. "log10").

\texttt{col\_breaks}
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

\texttt{col\_expand\_limits}
For a continuous variable, any values that the limits should encompass (e.g. 0).

\texttt{col\_labels}
A function that takes the breaks as inputs (e.g. \texttt{\(x\) stringr::str_to_sentence(x)} or `scales::label_comma()`), or a vector of labels.

\texttt{col\_legend\_ncol, col\_legend\_nrow}
The number of columns and rows for the legend guide.

\texttt{col\_legend\_rev}
Reverse the elements of the legend guide. Defaults to FALSE.

\texttt{col\_limits}
A vector of length 2 to determine the limits of the axis.
col_oob  For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

col_pal  Colours to use. A character vector of hex codes (or names).

col_pal_na  Colour to use for NA values. A character vector of a hex code (or name).

col_rescale  For a continuous variable, a `scales::rescale()` function.

col_steps  For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title  Axis title string. Use "" for no title.

col_transform  For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the `transform_` prefix (e.g. "log10").

alpha_breaks  A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

alpha_expand_limits  For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels  A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow  The number of columns and rows for the legend guide.

alpha_legend_rev  Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits  A vector of length 2 to determine the limits of the axis.

alpha_oob  For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

alpha_pal  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na  Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title  Axis title string. Use "" for no title.

alpha_transform  For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the `transform_` prefix (e.g. "log10").

facet_axes  Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels  Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels  A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)`), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position  When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch  When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".
facet_layout  Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol  The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow  The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales  Whether facet scales should be “fixed” across facets, “free” in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to “fixed”.

facet_space  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title  Title string.

subtitle  Subtitle string.

caption  Caption title string.

titles_to_case  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value

A ggplot object.

Examples

```r
library(ggplot2)
library(dplyr)

diamonds |>
  gg_hex(
    x = carat,
    y = price,
    coord = coord_cartesian(clip = "on"),
    y_limits = c(0, 20000),
  )
```

Description

Create a histogram ggplot with a wrapper around `ggplot2::ggplot() + geom_histogram()`.
Usage

gg_histogram(
  data = NULL,
  ..., 
  stat = "bin",
  position = "stack",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
  x_title = NULL,
  x_transform = NULL,
  y_breaks = NULL,
  y_expand = NULL,
  y_expand_limits = NULL,
  y_labels = NULL,
  y_limits = NULL,
  y_oob = scales::oob_keep,
  y_position = "left",
  y_title = NULL,
  y_transform = NULL,
  col_breaks = NULL,
  col_expand_limits = NULL,
  col_labels = NULL,
Arguments

- **data**
  A data frame or tibble.

- **...**
  Other arguments passed to within a `params` list in `layer()`.

- **stat**
  A statistical transformation to use on the data. A snakecase character string of a `ggproto` Stat subclass object minus the Stat prefix (e.g. "identity").

- **position**
  A position adjustment. A snakecase character string of a `ggproto` Position subclass object minus the Position prefix (e.g. "identity"), or a `position_*()` function that outputs a `ggproto` Position subclass object (e.g. `ggplot2::position_identity()`).
coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.

yend Unquoted yend aesthetic variable.

z Unquoted z aesthetic variable.

col Unquoted col aesthetic variable.

alpha Unquoted alpha aesthetic variable.

facet Unquoted facet aesthetic variable.

facet2 Unquoted facet2 aesthetic variable.

group Unquoted group aesthetic variable.

subgroup Unquoted subgroup aesthetic variable.

label Unquoted label aesthetic variable.

text Unquoted text aesthetic variable.

sample Unquoted sample aesthetic variable.

mapping Set of additional aesthetic mappings within ggplot2::aes() for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

x_expand, y_expand Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).

x_expand_limits, y_expand_limits For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels A function that takes the breaks as inputs (e.g. \(x) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

x_limits, y_limits A vector of length 2 to determine the limits of the axis.

x_oob, y_oob For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

For a continuous variable, any values that the limits should encompass (e.g. 0).

A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

The number of columns and rows for the legend guide.

Reverse the elements of the legend guide. Defaults to FALSE.

A vector of length 2 to determine the limits of the axis.

For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

Colours to use. A character vector of hex codes (or names).

Colour to use for NA values. A character vector of a hex code (or name).

For a continuous variable, a scales::rescale() function.

For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

For a continuous variable, any values that the limits should encompass (e.g. 0).

A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

The number of columns and rows for the legend guide.

Reverse the elements of the legend guide. Defaults to FALSE.

A vector of length 2 to determine the limits of the axis.

For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
gg_histogram

alpha_pal  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title Axis title string. Use "" for no title.

alpha_transform For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_prefix (e.g. "log10").

facet_axes Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels A function that takes the breaks as inputs (e.g. `\(x\) stringr::str_to_sentence(x)`), or a named vector of labels (e.g. `c("value1" = "label1", ...)`).

facet_labels_position When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title Title string.

subtitle Subtitle string.

caption Caption title string.

titles_to_case A function to format unspecified titles_to_case. Defaults to `snakecase::to_sentence_case`.

Value

A ggplot object.
Examples

```r
library(ggplot2)
library(dplyr)
library(palmerpenguins)

penguins |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  gg_histogram(
    x = flipper_length_mm,
    col = sex,
    facet = species,
    bins = 50,
    mode = light_mode_b(),
  )
```

Description

Create a jitter ggplot with a wrapper around `ggplot2::ggplot()` + `geom_jitter()`.

Usage

```r
gg_jitter(
  data = NULL,
  ...,
  stat = "identity",
  position = "jitter",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
)```
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_limit = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
gg_jitter

facets = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

data
A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat
A statistical transformation to use on the data. A snakecase character string of a
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position
A position adjustment. A snakecase character string of a ggproto Position sub-
class object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord
A coordinate system. A coord_*() function that outputs a constructed ggproto
Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode
A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()).
This argument adds the theme with side-effects, as the gg_* function will re-
moves selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme
on to the output of gg_*.

x
Unquoted x aesthetic variable.

xmin
Unquoted xmin aesthetic variable.

xmax
Unquoted xmax aesthetic variable.

xend
Unquoted xend aesthetic variable.

y
Unquoted y aesthetic variable.

ymin
Unquoted ymin aesthetic variable.

ymax
Unquoted ymax aesthetic variable.

yend
Unquoted yend aesthetic variable.

z
Unquoted z aesthetic variable.

col
Unquoted col aesthetic variable.

alpha
Unquoted alpha aesthetic variable.

facet
Unquoted facet aesthetic variable.
facet2  Unquoted facet2 aesthetic variable.
group   Unquoted group aesthetic variable.
subgroup Unquoted subgroup aesthetic variable.
label   Unquoted label aesthetic variable.
text    Unquoted text aesthetic variable.
sample  Unquoted sample aesthetic variable.
mapping Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.
x_breaks, y_breaks
       A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.
x_expand, y_expand
       Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).
x_expand_limits, y_expand_limits
       For a continuous variable, any values that the limits should encompass (e.g. 0).
x_labels, y_labels
       A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.
x_limits, y_limits
       A vector of length 2 to determine the limits of the axis.
x_oob, y_oob
       For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.
x_position, y_position
       The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y_position = "top"` with a *_mode_* theme, add caption = "" or caption = "\n".
x_title, y_title
       Axis title string. Use "" for no title.
x_transform, y_transform
       For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the `transform_` prefix (e.g. "log10").
col_breaks  A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.
col_expand_limits
       For a continuous variable, any values that the limits should encompass (e.g. 0).
col_labels  A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.
col_legend_ncol, col_legend_nrow
       The number of columns and rows for the legend guide.
col_legend_rev  Reverse the elements of the legend guide. Defaults to `FALSE`.
col_limits  A vector of length 2 to determine the limits of the axis.
col_oob For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

col_pal Colours to use. A character vector of hex codes (or names).

col_pal_na Colour to use for NA values. A character vector of a hex code (or name).

col_rescale For a continuous variable, a scales::rescale() function.

col_steps For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title Axis title string. Use "" for no title.

col_transform For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

alpha_expnd_limits For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow The number of columns and rows for the legend guide.

alpha_legend_rev Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits A vector of length 2 to determine the limits of the axis.

alpha_oob For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title Axis title string. Use "" for no title.

alpha_transform For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>facet_layout</td>
<td>Whether the layout is to be &quot;wrap&quot; or &quot;grid&quot;. If NULL and a single facet (or facet2) argument is provided, then defaults to &quot;wrap&quot;. If NULL and both facet and facet2 arguments are provided, defaults to &quot;grid&quot;.</td>
</tr>
<tr>
<td>facet_ncol</td>
<td>The number of columns of facets. Only applies to a facet layout of &quot;wrap&quot;.</td>
</tr>
<tr>
<td>facet_nrow</td>
<td>The number of rows of facets. Only applies to a facet layout of &quot;wrap&quot;.</td>
</tr>
<tr>
<td>facet_scales</td>
<td>Whether facet scales should be &quot;fixed&quot; across facets, &quot;free&quot; in both directions, or free in just one direction (i.e. &quot;free_x&quot; or &quot;free_y&quot;). Defaults to &quot;fixed&quot;.</td>
</tr>
<tr>
<td>facet_space</td>
<td>When the facet layout is &quot;grid&quot; and facet scales are not &quot;fixed&quot;, whether facet space should be &quot;fixed&quot; across facets, &quot;free&quot; to be proportional in both directions, or free to be proportional in just one direction (i.e. &quot;free_x&quot; or &quot;free_y&quot;). Defaults to &quot;fixed&quot;.</td>
</tr>
<tr>
<td>title</td>
<td>Title string.</td>
</tr>
<tr>
<td>subtitle</td>
<td>Subtitle string.</td>
</tr>
<tr>
<td>caption</td>
<td>Caption title string.</td>
</tr>
<tr>
<td>titles_to_case</td>
<td>A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.</td>
</tr>
</tbody>
</table>

**Value**

A ggplot object.

**Examples**

```r
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set.seed(123)

penguins |> gg_jitter(
  x = species,
  y = body_mass_g,
  col = flipper_length_mm,
  position = position_jitter(height = 0),
  y_expand_limits = 0,
  col_steps = TRUE,
)
```
Description

Create a label ggplot with a wrapper around \texttt{ggplot2::ggplot()} + \texttt{geom_label()}.

Usage

\begin{verbatim}
gg_label(
  data = NULL,
  ..., 
  stat = "identity", 
  position = "identity", 
  coord = ggplot2::coord_cartesian(clip = "off"), 
  mode = NULL, 
  x = NULL, 
  xmin = NULL, 
  xmax = NULL, 
  xend = NULL, 
  y = NULL, 
  ymin = NULL, 
  ymax = NULL, 
  yend = NULL, 
  z = NULL, 
  col = NULL, 
  alpha = NULL, 
  facet = NULL, 
  facet2 = NULL, 
  group = NULL, 
  subgroup = NULL, 
  label = NULL, 
  text = NULL, 
  sample = NULL, 
  mapping = NULL, 
  x_breaks = NULL, 
  x_expand = NULL, 
  x_expand_limits = NULL, 
  x_labels = NULL, 
  x_limits = NULL, 
  x_oob = scales::oob_keep, 
  x_position = "bottom", 
  x_title = NULL, 
  x_transform = NULL, 
  y_breaks = NULL, 
  y_expand = NULL, 
  y_expand = NULL, 
  y_expand_limits = NULL, 
\end{verbatim}
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
Arguments

data A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_.*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.

yend Unquoted yend aesthetic variable.

z Unquoted z aesthetic variable.

col Unquoted col aesthetic variable.

alpha Unquoted alpha aesthetic variable.

facet Unquoted facet aesthetic variable.

date2 Unquoted date2 aesthetic variable.

subgroup Unquoted subgroup aesthetic variable.

text Unquoted text aesthetic variable.

sample Unquoted sample aesthetic variable.

mapping Set of additional aesthetic mappings within ggplot2::aes() for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

x_expand, y_expand Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).
x_expand_limits, y_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels
A function that takes the breaks as inputs (e.g. \( \backslash (x) \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

x_limits, y_limits
A vector of length 2 to determine the limits of the axis.

x_oob, y_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob.keep.

x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".

x_title, y_title
Axis title string. Use "" for no title.

x_transform, y_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

col_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
A function that takes the breaks as inputs (e.g. \( \backslash (x) \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob.keep.

col_pal
Colours to use. A character vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
For a continuous variable, a scales::rescale() function.

col_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title
Axis title string. Use "" for no title.

col_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

alpha_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
A function that takes the breaks as inputs (e.g. \( \backslash (x) \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.
alpha_legend_ncol, alpha_legend_nrow
The number of columns and rows for the legend guide.

alpha_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
A vector of length 2 to determine the limits of the axis.

alpha_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal
Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na
Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title
Axis title string. Use "" for no title.

alpha_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes
Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels
Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels
A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position
When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch
When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout
Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol
The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow
The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales
Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space
When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title
Title string.

subtitle
Subtitle string.

caption
Caption title string.

titles_to_case
A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.
gg_line

Value

A ggplot object.

Examples

```r
library(ggplot2)
library(dplyr)
bind_rows(
  mtcars |> slice_min(order_by = mpg),
  mtcars |> slice_max(order_by = mpg)) |> 
tibble::rownames_to_column("model") |> 
gg_label(
  x = model,
  y = mpg,
  col = mpg,
  label = model,
  size = 3.53,
  y_expand_limits = 0,
  y_title = "Miles per gallon",
  col_pal = c(orange, "white", teal),
)
```

Description

Create a line ggplot with a wrapper around `ggplot2::ggplot()` + `geom_line()`.

Usage

```r
gg_line(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
```
yend = NULL,
z = NULL,
col = NULL,
alpha = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
ox_breaks = NULL,
ox_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha.legend_nrow = NULL,
alpha.legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

data
A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat
A statistical transformation to use on the data. A snakecase character string of a

ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position
A position adjustment. A snakecase character string of a ggproto Position sub-
class object minus the Position prefix (e.g. "identity"), or a position_*( )
function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity( )).

coord
A coordinate system. A coord_*( ) function that outputs a constructed ggproto
Coord subclass object (e.g. ggplot2::coord_cartesian( )).

mode
A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()).
This argument adds the theme with side-effects, as the gg_* function will re-
moves selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme
on to the output of gg_*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.
Unquoted ymax aesthetic variable.

Unquoted yend aesthetic variable.

Unquoted z aesthetic variable.

Unquoted col aesthetic variable.

Unquoted alpha aesthetic variable.

Unquoted facet aesthetic variable.

Unquoted facet2 aesthetic variable.

Unquoted group aesthetic variable.

Unquoted subgroup aesthetic variable.

Unquoted label aesthetic variable.

Unquoted text aesthetic variable.

Unquoted sample aesthetic variable.

Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

For a continuous variable, any values that the limits should encompass (e.g. 0).

A function that takes the breaks as inputs (e.g. \(\text{x} \) stringr::str_to_sentence(x) or `scales::label_comma()`), or a vector of labels.

A vector of length 2 to determine the limits of the axis.

For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y_position = "top"` with a *_mode_* theme, add caption = "" or caption = "\n".

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_ prefix (e.g. "log10").

A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

For a continuous variable, any values that the limits should encompass (e.g. 0).
col_labels  A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma), or a vector of labels.
col_legend_ncol, col_legend_nrow  The number of columns and rows for the legend guide.
col_legend_rev  Reverse the elements of the legend guide. Defaults to FALSE.
col_limits  A vector of length 2 to determine the limits of the axis.
col_oob  For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
col_pal  Colours to use. A character vector of hex codes (or names).
col_pal_na  Colour to use for NA values. A character vector of a hex code (or name).
col_rescale  For a continuous variable, a scales::rescale() function.
col_steps  For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.
col_title  Axis title string. Use "" for no title.
col_transform  For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").
alpha_breaks  A scales::breaks_* function (e.g. scales::breaks_pretty), or a vector of breaks.
alpha_expand_limits  For a continuous variable, any values that the limits should encompass (e.g. 0).
alpha_labels  A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma), or a vector of labels.
alpha_legend_ncol, alpha_legend_nrow  The number of columns and rows for the legend guide.
alpha_legend_rev  Reverse the elements of the legend guide. Defaults to FALSE.
alpha_limits  A vector of length 2 to determine the limits of the axis.
alpha_oob  For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
alpha_pal  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.
alpha_pal_na  Alpha value to use for the NA value. A integer between 0 and 1.
alpha_title  Axis title string. Use "" for no title.
alpha_transform  For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").
facet_axes  Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".
facet_axis_labels  Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_labels  A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).
When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

The number of columns of facets. Only applies to a facet layout of "wrap".

The number of rows of facets. Only applies to a facet layout of "wrap".

Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

Title string.

Subtitle string.

Caption title string.

A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

A ggplot object.

library(ggplot2)
library(dplyr)
economics |>
gg_line(
  x = date,
  y = unemploy,
  y_expand_limits = 0,
  y_title = "Unemployment",
)
Description

Create a linerange ggplot with a wrapper around `ggplot2::ggplot() + geom_linerange().`

Usage

```r
gg_linerange(
  data = NULL,
  ..., 
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
  x_title = NULL,
  x_transform = NULL,
  y_breaks = NULL,
  y_expand = NULL,
  y_expand_limits = NULL,
)```
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case}
Arguments

data  A data frame or tibble.

...  Other arguments passed to within a `params` list in `layer()`.

stat  A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position  A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a `position_*()` function that outputs a ggproto Position subclass object (e.g. `ggplot2::position_identity()`).

coord  A coordinate system. A `coord_*()` function that outputs a constructed ggproto Coord subclass object (e.g. `ggplot2::coord_cartesian()`).

mode  A *_mode_* theme (e.g. `grey_mode_b()`, `grey_mode_r()`, or `dark_mode_r()`). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x  Unquoted x aesthetic variable.

xmin  Unquoted xmin aesthetic variable.

xmax  Unquoted xmax aesthetic variable.

xend  Unquoted xend aesthetic variable.

y  Unquoted y aesthetic variable.

ymin  Unquoted ymin aesthetic variable.

ymax  Unquoted ymax aesthetic variable.

yend  Unquoted yend aesthetic variable.

z  Unquoted z aesthetic variable.

col  Unquoted col aesthetic variable.

alpha  Unquoted alpha aesthetic variable.

facet  Unquoted facet aesthetic variable.

facet2  Unquoted facet2 aesthetic variable.

group  Unquoted group aesthetic variable.

subgroup  Unquoted subgroup aesthetic variable.

label  Unquoted label aesthetic variable.

text  Unquoted text aesthetic variable.

sample  Unquoted sample aesthetic variable.

mapping  Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks  A scales::breaks_* function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

x_expand, y_expand  Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. c(0, 0)).
For a continuous variable, any values that the limits should encompass (e.g. 0).

A function that takes the breaks as inputs (e.g. \( \text{\texttt{\textbackslash{}(x) stringr::str_to_sentence(x) or scales::label_comma()}} \)), or a vector of labels.

A vector of length 2 to determine the limits of the axis.

For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

For a continuous variable, any values that the limits should encompass (e.g. 0).

A function that takes the breaks as inputs (e.g. \( \text{\texttt{\textbackslash{}(x) stringr::str_to_sentence(x) or scales::label_comma()}} \)), or a vector of labels.

The number of columns and rows for the legend guide.

Reverse the elements of the legend guide. Defaults to FALSE.

A vector of length 2 to determine the limits of the axis.

For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

Colours to use. A character vector of hex codes (or names).

Colour to use for NA values. A character vector of a hex code (or name).

For a continuous variable, a scales::rescale() function.

For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

For a continuous variable, any values that the limits should encompass (e.g. 0).

A function that takes the breaks as inputs (e.g. \( \text{\texttt{\textbackslash{}(x) stringr::str_to_sentence(x) or scales::label_comma()}} \)), or a vector of labels.
alpha_legend_ncol, alpha_legend_nrow
The number of columns and rows for the legend guide.
alpha_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.
alpha_limits
A vector of length 2 to determine the limits of the axis.
alpha_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
alpha_pal
Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.
alpha_pal_na
Alpha value to use for the NA value. A integer between 0 and 1.
alpha_title
Axis title string. Use "" for no title.
alpha_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").
facet_axes
Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".
facet_axis_labels
Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_labels
A function that takes the breaks as inputs (e.g. \(x) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).
facet_labels_position
When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".
facet_labels_switch
When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".
facet_layout
Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
facet_ncol
The number of columns of facets. Only applies to a facet layout of "wrap".
facet_nrow
The number of rows of facets. Only applies to a facet layout of "wrap".
facet_scales
Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space
When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title
Title string.
subtitle
Subtitle string.
caption
Caption title string.
titles_to_case
A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.
gg_path

Value

A ggplot object.

Examples

```r
library(ggplot2)
library(dplyr)

data.frame(
  trt = factor(c(1, 1, 2, 2)),
  resp = c(1, 5, 3, 4),
  group = factor(c(1, 2, 1, 2)),
  upper = c(1.1, 5.3, 3.3, 4.2),
  lower = c(0.8, 4.6, 2.4, 3.6)) |>
  gg_linerange(
    x = trt,
    ymin = lower,
    ymax = upper,
    col = group,
    position = position_dodge(width = 0.2),
    x_title = "Treatment",
    y_title = "Response",
  )
```

---

**gg_path**  
Path ggplot

Description

Create a path ggplot with a wrapper around `ggplot2::ggplot()` + `geom_path()`.

Usage

```r
gg_path(
  data = NULL,
  ...
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ```
ymax = NULL,
yend = NULL,
z = NULL,
col = NULL,
alpha = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
Arguments

data
A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.
Unquoted `ymin` aesthetic variable.

Unquoted `yend` aesthetic variable.

Unquoted `z` aesthetic variable.

Unquoted `col` aesthetic variable.

Unquoted `alpha` aesthetic variable.

Unquoted `facet` aesthetic variable.

Unquoted `facet2` aesthetic variable.

Unquoted `group` aesthetic variable.

Unquoted `subgroup` aesthetic variable.

Unquoted `label` aesthetic variable.

Unquoted `text` aesthetic variable.

Unquoted `sample` aesthetic variable.

Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. `shape`, `linetype`, `linewidth`, or `size`) or for delayed evaluation.

A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

For a continuous variable, any values that the limits should encompass (e.g. 0).

A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

A vector of length 2 to determine the limits of the axis.

For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y_position = "top"` with a `_mode_` theme, add `caption = ""` or `caption = "n"`.

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the `transform_` prefix (e.g. "log10").

A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

For a continuous variable, any values that the limits should encompass (e.g. 0).
col_labels A function that takes the breaks as inputs (e.g. \( x \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev Reverse the elements of the legend guide. Defaults to FALSE.

col_limits A vector of length 2 to determine the limits of the axis.

col_oob For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

col_pal Colours to use. A character vector of hex codes (or names).

col_pal_na Colour to use for NA values. A character vector of a hex code (or name).

col_rescale For a continuous variable, a scales::rescale() function.

col_steps For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title Axis title string. Use "" for no title.

col_transform For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

alphaexpand limits For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels A function that takes the breaks as inputs (e.g. \( x \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow
The number of columns and rows for the legend guide.

alpha_legend_rev Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits A vector of length 2 to determine the limits of the axis.

alpha_oob For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title Axis title string. Use "" for no title.

alpha_transform For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels A function that takes the breaks as inputs (e.g. \( x \) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).
facet_labels_position
When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch
When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout
Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol
The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow
The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales
Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space
When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title
Title string.

subtitle
Subtitle string.

caption
Caption title string.

titles_to_case
A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value
A ggplot object.

Examples

library(ggplot2)
library(dplyr)

economics |> mutate(unemploy_rate = unemploy / pop) |> gg_path(
  x = unemploy_rate,
  y = psavert,
  x_title = "Unemployment rate",
  y_expand_limits = 0,
  y_title = "Personal savings rate",
)
Description

Create a point ggplot with a wrapper around `ggplot2::ggplot()` + `geom_point()`.

Usage

```r
gg_point(
  data = NULL,
  ..., stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
  x_title = NULL,
  x_transform = NULL,
  y_breaks = NULL,
  y_expand = NULL,
  y_expand_limits = NULL,
  y_label = NULL,
  y_limits = NULL,
  y_oob = scales::oob_keep,
  y_position = "bottom",
  y_title = NULL,
  y_transform = NULL,
  z_breaks = NULL,
  z_expand = NULL,
  z_expand_limits = NULL,
  z_labels = NULL,
  z_limits = NULL,
  z_oob = scales::oob_keep,
  z_position = "bottom",
  z_title = NULL,
  z_transform = NULL,
  layout = NULL,
)
```
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
Arguments

data A data frame or tibble.
... Other arguments passed to within a params list in layer().
stat A statistical transformation to use on the data. A snakecase character string of a
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position A position adjustment. A snakecase character string of a ggproto Position sub-
class object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).
coord A coordinate system. A coord_*() function that outputs a constructed ggproto
Coord subclass object (e.g. ggplot2::coord_cartesian()).
mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()).
This argument adds the theme with side-effects, as the gg_* function will re-
moves selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme
on to the output of gg_*.
x Unquoted x aesthetic variable.
xmin Unquoted xmin aesthetic variable.
xmax Unquoted xmax aesthetic variable.
xend Unquoted xend aesthetic variable.
y Unquoted y aesthetic variable.
ymin Unquoted ymin aesthetic variable.
ymax Unquoted ymax aesthetic variable.
yend Unquoted yend aesthetic variable.
z Unquoted z aesthetic variable.
col Unquoted col aesthetic variable.
alpha Unquoted alpha aesthetic variable.
facet Unquoted facet aesthetic variable.
facet2 Unquoted facet2 aesthetic variable.
group Unquoted group aesthetic variable.
subgroup Unquoted subgroup aesthetic variable.
label Unquoted label aesthetic variable.
text Unquoted text aesthetic variable.
sample Unquoted sample aesthetic variable.
mapping Set of additional aesthetic mappings within ggplot2::aes() for non-supported
aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evalua-
tion.
x_breaks, y_breaks A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector
of breaks.
x_expand, y_expand Padding to the limits with the ggplot2::expansion() function, or a vector of
length 2 (e.g. c(0, 0)).
**x\_expand\_limits, y\_expand\_limits**
For a continuous variable, any values that the limits should encompass (e.g. 0).

**x\_labels, y\_labels**
A function that takes the breaks as inputs (e.g. \(\text{\texttt{\textbackslash(x}}\text{\texttt{ stringr::str\_to\_sentence(x)}}\text{\texttt{)}}\)) or \texttt{scales::label\_comma()}, or a vector of labels.

**x\_limits, y\_limits**
A vector of length 2 to determine the limits of the axis.

**x\_oob, y\_oob**
For a continuous scale variable, a \texttt{scales::oob\_*} function of how to handle values outside of limits. Defaults to \texttt{scales::oob\_keep}.

**x\_position, y\_position**
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using \(y\_position = "\text{top}"\) with a \_*\_mode\_* theme, add \texttt{caption = \"\"} or \texttt{caption = \"\text{\textbackslash n}\"}.

**x\_title, y\_title**
Axis title string. Use \"\" for no title.

**x\_transform, y\_transform**
For a numeric scale, a transformation object (e.g. \texttt{scales::transform\_log10()}), or character string of this minus the \texttt{transform\_} prefix (e.g. "log10").

**col\_breaks**
A \texttt{scales::breaks\_*} function (e.g. \texttt{scales::breaks\_pretty()}), or a vector of breaks.

**col\_expand\_limits**
For a continuous variable, any values that the limits should encompass (e.g. 0).

**col\_labels**
A function that takes the breaks as inputs (e.g. \(\text{\texttt{\textbackslash(x}}\text{\texttt{ stringr::str\_to\_sentence(x)}}\text{\texttt{)}}\)) or \texttt{scales::label\_comma()}, or a vector of labels.

**col\_legend\_ncol, col\_legend\_nrow**
The number of columns and rows for the legend guide.

**col\_legend\_rev**
Reverse the elements of the legend guide. Defaults to \texttt{FALSE}.

**col\_limits**
A vector of length 2 to determine the limits of the axis.

**col\_oob**
For a continuous scale variable, a \texttt{scales::oob\_*} function of how to handle values outside of limits. Defaults to \texttt{scales::oob\_keep}.

**col\_pal**
Colours to use. A character vector of hex codes (or names).

**col\_pal\_na**
Colour to use for \texttt{NA} values. A character vector of a hex code (or name).

**col\_rescale**
For a continuous variable, a \texttt{scales::rescale\]() function.

**col\_steps**
For a continuous variable, \texttt{TRUE} or \texttt{FALSE} of whether to colour in steps. Defaults to \texttt{FALSE}, which colours in a gradient.

**col\_title**
Axis title string. Use \"\" for no title.

**col\_transform**
For a numeric scale, a transformation object (e.g. \texttt{scales::transform\_log10()}), or character string of this minus the \texttt{transform\_} prefix (e.g. "log10").

**alpha\_breaks**
A \texttt{scales::breaks\_*} function (e.g. \texttt{scales::breaks\_pretty()}), or a vector of breaks.

**alpha\_expand\_limits**
For a continuous variable, any values that the limits should encompass (e.g. 0).

**alpha\_labels**
A function that takes the breaks as inputs (e.g. \(\text{\texttt{\textbackslash(x}}\text{\texttt{ stringr::str\_to\_sentence(x)}}\text{\texttt{)}}\)) or \texttt{scales::label\_comma()}, or a vector of labels.
alpha_legend_ncol, alpha_legend_nrow

The number of columns and rows for the legend guide.

alpha_legend_rev

Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits

A vector of length 2 to determine the limits of the axis.

alpha_oob

For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal

Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na

Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title

Axis title string. Use "" for no title.

alpha_transform

For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes

Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels

Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels

A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position

When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch

When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout

Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol

The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow

The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales

Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space

When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title

Title string.

subtitle

Subtitle string.

caption

Caption title string.

titles_to_case

A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.
Value

A ggplot object.

Examples

```r
library(ggplot2)
library(dplyr)
library(palmerpenguins)

penguins |>
  mutate(across(sex, \(x\) stringr::str_to_sentence(x))) |>
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = sex,
    facet = species,
    mode = light_mode_b(),
  )
```

Description

Create a pointrange ggplot with a wrapper around `ggplot2::ggplot() + geom_pointrange()`.

Usage

```r
gg_pointrange(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
)```
facets = NULL,
facets2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
col_breaks = NULL,
Arguments

data A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.

yend Unquoted yend aesthetic variable.

z Unquoted z aesthetic variable.


col Unquoted col aesthetic variable.
alpha Unquoted alpha aesthetic variable.
facet Unquoted facet aesthetic variable.
facet2 Unquoted facet2 aesthetic variable.
group Unquoted group aesthetic variable.
subgroup Unquoted subgroup aesthetic variable.
label Unquoted label aesthetic variable.
text Unquoted text aesthetic variable.
sample Unquoted sample aesthetic variable.
mapping Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

x_expand, y_expand
Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

x_expand_limits, y_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels
A function that takes the breaks as inputs (e.g. `\(x) stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

x_limits, y_limits
A vector of length 2 to determine the limits of the axis.

x_oob, y_oob
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y_position = "top"` with a `_mode_*` theme, add `caption = ""` or `caption = "\n"`.

x_title, y_title
Axis title string. Use "" for no title.

x_transform, y_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or a character string of this minus the `transform_*` prefix (e.g. "log10").

col_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

col_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
A function that takes the breaks as inputs (e.g. `\(x) stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.
gg_pointrange

col_legend_rev  Reverse the elements of the legend guide. Defaults to FALSE.
col_limits  A vector of length 2 to determine the limits of the axis.
col_oob  For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
col_pal  Colours to use. A character vector of hex codes (or names).
col_pal_na  Colour to use for NA values. A character vector of a hex code (or name).
col_rescale  For a continuous variable, a scales::rescale() function.
col_steps  For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.
col_title  Axis title string. Use "" for no title.
col_transform  For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_. prefix (e.g. "log10").
alpha_breaks  A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.
alpha_expand_limits  For a continuous variable, any values that the limits should encompass (e.g. 0).
alpha_labels  A function that takes the breaks as inputs (e.g. \(x) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.
alpha_legend_ncol, alpha_legend_nrow  The number of columns and rows for the legend guide.
alpha_legend_rev  Reverse the elements of the legend guide. Defaults to FALSE.
alpha_limits  A vector of length 2 to determine the limits of the axis.
alpha_oob  For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
alpha_pal  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.
alpha_pal_na  Alpha value to use for the NA value. A integer between 0 and 1.
alpha_title  Axis title string. Use "" for no title.
alpha_transform  For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_. prefix (e.g. "log10").
facet_axes  Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".
facet_axis_labels  Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_labels  A function that takes the breaks as inputs (e.g. \(x) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).
facet_labels_position  When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".
When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

The number of columns of facets. Only applies to a facet layout of "wrap".

Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

Title string.

Subtitle string.

Caption title string.

A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

A ggplot object.

library(ggplot2)
library(dplyr)

data.frame(
  trt = factor(c(1, 1, 2, 2)),
  resp = c(1, 5, 3, 4),
  group = factor(c(1, 2, 1, 2)),
  upper = c(1.1, 5.3, 3.3, 4.2),
  lower = c(0.8, 4.6, 2.4, 3.6)) |
> gg_pointrange(
  x = trt,
  y = resp,
  col = group,
  y_min = lower,
  y_max = upper,
  position = position_dodge(width = 0.2),
  size = 0.2,
  x_title = "Treatment",
  y_title = "Response",
)
Description

Create a polygon ggplot with a wrapper around `ggplot2::ggplot()` + `geom_polygon()`.

Usage

```r
gg_polygon(
  data = NULL,
  ..., 
  stat = "identity", 
  position = "identity", 
  coord = ggplot2::coord_cartesian(clip = "off"), 
  mode = NULL, 
  x = NULL, 
  xmin = NULL, 
  xmax = NULL, 
  xend = NULL, 
  y = NULL, 
  ymin = NULL, 
  ymax = NULL, 
  yend = NULL, 
  z = NULL, 
  col = NULL, 
  alpha = NULL, 
  facet = NULL, 
  facet2 = NULL, 
  group = NULL, 
  subgroup = NULL, 
  label = NULL, 
  text = NULL, 
  sample = NULL, 
  mapping = NULL, 
  x_breaks = NULL, 
  x_expand = NULL, 
  x_expand_limits = NULL, 
  x_labels = NULL, 
  x_limits = NULL, 
  x_oob = scales::oob_keep, 
  x_position = "bottom", 
  x_title = NULL, 
  x_transform = NULL, 
  y_breaks = NULL, 
  y_expand = NULL, 
  y_expand = NULL, 
  y_expand_limits = NULL,
)```
```
```r
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)
Arguments

- **data**
  A data frame or tibble.

- **...**
  Other arguments passed to within a `params` list in `layer()`.

- **stat**
  A statistical transformation to use on the data. A snakecase character string of a `ggproto Stat` subclass object minus the Stat prefix (e.g. "identity").

- **position**
  A position adjustment. A snakecase character string of a `ggproto Position` subclass object minus the Position prefix (e.g. "identity"), or a `position_*()` function that outputs a `ggproto Position` subclass object (e.g. `ggplot2::position_identity()`).

- **coord**
  A coordinate system. A `coord_*()` function that outputs a constructed `ggproto Coord` subclass object (e.g. `ggplot2::coord_cartesian()`).

- **mode**
  A `*_mode_*` theme (e.g. `grey_mode_b()`, `grey_mode_r()`, or `dark_mode_r()`). This argument adds the theme with side-effects, as the `gg_*` function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of `gg_*`.

- **x**
  Unquoted x aesthetic variable.

- **xmin**
  Unquoted xmin aesthetic variable.

- **xmax**
  Unquoted xmax aesthetic variable.

- **xend**
  Unquoted xend aesthetic variable.

- **y**
  Unquoted y aesthetic variable.

- **ymin**
  Unquoted ymin aesthetic variable.

- **ymax**
  Unquoted ymax aesthetic variable.

- **yend**
  Unquoted yend aesthetic variable.

- **z**
  Unquoted z aesthetic variable.

- **col**
  Unquoted col aesthetic variable.

- **alpha**
  Unquoted alpha aesthetic variable.

- **facet**
  Unquoted facet aesthetic variable.

- **facet2**
  Unquoted facet2 aesthetic variable.

- **group**
  Unquoted group aesthetic variable.

- **subgroup**
  Unquoted subgroup aesthetic variable.

- **label**
  Unquoted label aesthetic variable.

- **text**
  Unquoted text aesthetic variable.

- **sample**
  Unquoted sample aesthetic variable.

- **mapping**
  Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

- **x_breaks, y_breaks**
  A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

- **x_expand, y_expand**
  Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).
x.expand_limits, y.expand_limits
   For a continuous variable, any values that the limits should encompass (e.g. 0).

x.labels, y.labels
   A function that takes the breaks as inputs (e.g. \( \text{\textbackslash}(x) \) stringr::str_to_sentence(x)
   or scales::label_comma()), or a vector of labels.

x.limits, y.limits
   A vector of length 2 to determine the limits of the axis.

x.oob, y.oob
   For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

x.position, y.position
   The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y.position = "top" with a _mode_* theme, add caption = "" or caption = "\n".

x.title, y.title
   Axis title string. Use "" for no title.

x.transform, y.transform
   For a numeric scale, a transformation object (e.g. scales::transform_log10())
   or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks
   A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

col.expand_limits
   For a continuous variable, any values that the limits should encompass (e.g. 0).

col.labels
   A function that takes the breaks as inputs (e.g. \( \text{\textbackslash}(x) \) stringr::str_to_sentence(x)
   or scales::label_comma()), or a vector of labels.

col.legend_ncol, col.legend_nrow
   The number of columns and rows for the legend guide.

col.legend_rev
   Reverse the elements of the legend guide. Defaults to FALSE.

col.limits
   A vector of length 2 to determine the limits of the axis.

col.oob
   For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

col_pal
   Colours to use. A character vector of hex codes (or names).

col_pal_na
   Colour to use for NA values. A character vector of a hex code (or name).

col.rescale
   For a continuous variable, a scales::rescale() function.

col.steps
   For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col.title
   Axis title string. Use "" for no title.

col.transform
   For a numeric scale, a transformation object (e.g. scales::transform_log10())
   or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks
   A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

alpha.expand_limits
   For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha.labels
   A function that takes the breaks as inputs (e.g. \( \text{\textbackslash}(x) \) stringr::str_to_sentence(x)
   or scales::label_comma()), or a vector of labels.
alpha_legend_ncol, alpha_legend_nrow
The number of columns and rows for the legend guide.

alpha_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
A vector of length 2 to determine the limits of the axis.

alpha_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal
Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na
Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title
Axis title string. Use "" for no title.

alpha_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes
Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels
Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels
A function that takes the breaks as inputs (e.g. \(x) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position
When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch
When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x"", "y" or "both".

facet_layout
Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol
The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow
The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales
Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space
When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title
Title string.

subtitle
Subtitle string.

caption
Caption title string.

titles_to_case
A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.
Value

A ggplot object.

Examples

```r
library(ggplot2)
library(dplyr)

ids <- factor(c("1.1", "2.1", "1.2", "2.2", "1.3", "2.3"))

values <- data.frame(
  id = ids,
  value = c(3, 3.1, 3.1, 3.2, 3.15, 3.5)
)

positions <- data.frame(
  id = rep(ids, each = 4),
  x = c(2, 1, 1.1, 2.2, 1, 0, 0.3, 1.1, 2.2, 1.1, 1.2, 2.5, 1.1, 0.3,
    0.5, 1.2, 2.5, 1.2, 1.3, 2.7, 1.2, 0.5, 0.6, 1.3),
  y = c(-0.5, 0, 1, 0.5, 0, 0.5, 1.5, 1, 0.5, 1, 2.1, 1.7, 1, 1.5,
    2.2, 2.1, 1.7, 2.1, 3.2, 2.8, 2.1, 2.2, 3.3, 3.2)
)

datapoly <- merge(values, positions, by = c("id"))

datapoly |> |
  gg_polygon(
    x = x,
    y = y,
    col = value,
    group = id,
  )
```

---

**gg_qq**


### Description

Create a qq ggplot with a wrapper around `ggplot2::ggplot()` + `geom_qq()`.

#### Usage

```r
gg_qq(
  data = NULL,
  ..., 
  stat = "qq",
  position = "identity",
```
coord = ggplot2::coord_cartesian(clip = "off"),
mode = NULL,
x = NULL,
xmin = NULL,
xmax = NULL,
xend = NULL,
y = NULL,
ymin = NULL,
ymax = NULL,
yend = NULL,
z = NULL,
col = NULL,
alpha = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
Arguments

data          A data frame or tibble.

...         Other arguments passed to within a params list in layer().

stat          A statistical transformation to use on the data. A snakecase character string of a
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position      A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord          A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode              A _mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()).

This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.
x  Unquoted x aesthetic variable.
xmin  Unquoted xmin aesthetic variable.
xmax  Unquoted xmax aesthetic variable.
xend  Unquoted xend aesthetic variable.
y  Unquoted y aesthetic variable.
ymin  Unquoted ymin aesthetic variable.
ymax  Unquoted ymax aesthetic variable.
yend  Unquoted yend aesthetic variable.
z  Unquoted z aesthetic variable.
col  Unquoted col aesthetic variable.
alpha  Unquoted alpha aesthetic variable.
facet  Unquoted facet aesthetic variable.
facet2  Unquoted facet2 aesthetic variable.
group  Unquoted group aesthetic variable.
subgroup  Unquoted subgroup aesthetic variable.
label  Unquoted label aesthetic variable.
text  Unquoted text aesthetic variable.
sample  Unquoted sample aesthetic variable.
mapping  Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks  A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

x_expand, y_expand  Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

x_expand_limits, y_expand_limits  For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels  A function that takes the breaks as inputs (e.g. `\(x\)` `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

x_limits, y_limits  A vector of length 2 to determine the limits of the axis.

x_oob, y_oob  For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

x_position, y_position  The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y_position = "top"` with a *mode_* theme, add `caption = ""` or `caption = "\n"`.

x_title, y_title  Axis title string. Use "" for no title.
x_transform, y_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`)
or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks  A scales::breaks_* function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

col_expand_limits  For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels  A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)
or `scales::label_comma()`), or a vector of labels.

col_legend_ncol, col_legend_nrow  The number of columns and rows for the legend guide.

col_legend_rev  Reverse the elements of the legend guide. Defaults to FALSE.

col_limits  A vector of length 2 to determine the limits of the axis.

col_oob  For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

col_pal  Colours to use. A character vector of hex codes (or names).

col_pal_na  Colour to use for NA values. A character vector of a hex code (or name).

col_rescale  For a continuous variable, a scales::rescale() function.

col_steps  For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title  Axis title string. Use "" for no title.

col_transform  For a numeric scale, a transformation object (e.g. `scales::transform_log10()`)
or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks  A scales::breaks_* function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

alpha_expand_limits  For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels  A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)
or `scales::label_comma()`), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow  The number of columns and rows for the legend guide.

alpha_legend_rev  Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits  A vector of length 2 to determine the limits of the axis.

alpha_oob  For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na  Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title  Axis title string. Use "" for no title.

alpha_transform  For a numeric scale, a transformation object (e.g. `scales::transform_log10()`)
or character string of this minus the transform_ prefix (e.g. "log10").
facet_axes  Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels A function that takes the breaks as inputs (e.g. \( \text{\texttt{stringr::str_to_sentence(x)}} \)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title Title string.

subtitle Subtitle string.

caption Caption title string.

titles_to_case A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value

A ggplot object.

Examples

```r
library(ggplot2)
library(dplyr)
library(palmerpenguins)

penguins |>
  gg_qq(
    sample = body_mass_g,
    facet = species,
    coord = coord_cartesian(clip = "on"),
  ) +
```
gg_quantile

```r
geom_qq_line(
  colour = blue,
)
```

---

**gg_quantile**  
*Quantile ggplot*

**Description**

Create an quantile ggplot with a wrapper around `ggplot2::ggplot() + geom_quantile()`.

**Usage**

```r
gg_quantile(
  data = NULL,
  ...,
  stat = "quantile",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
```
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

data           A data frame or tibble.
...
stat           A statistical transformation to use on the data. A snakecase character string of a
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position       A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).
coord          A coordinate system. A coord_*() function that outputs a constructed ggproto
                Coord subclass object (e.g. ggplot2::coord_cartesian()).
mode           A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()).
                This argument adds the theme with side-effects, as the gg_* function will re-
                moves selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme
                on to the output of gg_*.

x              Unquoted x aesthetic variable.
xmin           Unquoted xmin aesthetic variable.
xmax           Unquoted xmax aesthetic variable.
xend           Unquoted xend aesthetic variable.
y              Unquoted y aesthetic variable.
ymin           Unquoted ymin aesthetic variable.
ymax           Unquoted ymax aesthetic variable.
yend           Unquoted yend aesthetic variable.
z              Unquoted z aesthetic variable.
col            Unquoted col aesthetic variable.
alpha          Unquoted alpha aesthetic variable.
facet          Unquoted facet aesthetic variable.
facet2         Unquoted facet2 aesthetic variable.
group          Unquoted group aesthetic variable.
subgroup       Unquoted subgroup aesthetic variable.
label          Unquoted label aesthetic variable.
text           Unquoted text aesthetic variable.
sample         Unquoted sample aesthetic variable.

mapping        Set of additional aesthetic mappings within ggplot2::aes() for non-supported
                aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evalua-
                tion.
x_breaks, y_breaks
   A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

x_expand, y_expand
   Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).

x_expand_limits, y_expand_limits
   For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels
   A function that takes the breaks as inputs (e.g. \( x \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

x_limits, y_limits
   A vector of length 2 to determine the limits of the axis.

x_oob, y_oob
   For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

x_position, y_position
   The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".

x_title, y_title
   Axis title string. Use "" for no title.

x_transform, y_transform
   For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "\log10").

col_breaks
   A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

col_expand_limits
   For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
   A function that takes the breaks as inputs (e.g. \( x \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

col_legend_ncol, col_legend_nrow
   The number of columns and rows for the legend guide.

col_legend_rev
   Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
   A vector of length 2 to determine the limits of the axis.

col_oob
   For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

col_pal
   Colours to use. A character vector of hex codes (or names).

col_pal_na
   Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
   For a continuous variable, a scales::rescale() function.

col_steps
   For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title
   Axis title string. Use "" for no title.

col_transform
   For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "\log10").
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>alpha_breaks</td>
<td>A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_pretty()</code>), or a vector of breaks.</td>
</tr>
<tr>
<td>alpha_expand_limits</td>
<td>For a continuous variable, any values that the limits should encompass (e.g. 0).</td>
</tr>
<tr>
<td>alpha_labels</td>
<td>A function that takes the breaks as inputs (e.g. (x) <code>stringr::str_to_sentence(x)</code> or <code>scales::label_comma()</code>), or a vector of labels.</td>
</tr>
<tr>
<td>alpha_legend_ncol, alpha_legend_nrow</td>
<td>The number of columns and rows for the legend guide.</td>
</tr>
<tr>
<td>alpha_legend_rev</td>
<td>Reverse the elements of the legend guide. Defaults to FALSE.</td>
</tr>
<tr>
<td>alpha_limits</td>
<td>A vector of length 2 to determine the limits of the axis.</td>
</tr>
<tr>
<td>alpha_oob</td>
<td>For a continuous scale variable, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code>.</td>
</tr>
<tr>
<td>alpha_pal</td>
<td>Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.</td>
</tr>
<tr>
<td>alpha_pal_na</td>
<td>Alpha value to use for the NA value. A integer between 0 and 1.</td>
</tr>
<tr>
<td>alpha_title</td>
<td>Axis title string. Use &quot;&quot; for no title.</td>
</tr>
<tr>
<td>alpha_transform</td>
<td>For a numeric scale, a transformation object (e.g. <code>scales::transform_log10()</code>) or character string of this minus the transform_ prefix (e.g. &quot;log10&quot;).</td>
</tr>
<tr>
<td>facet_axes</td>
<td>Whether to add interior axes and ticks with &quot;margins&quot;, &quot;all&quot;, &quot;all_x&quot;, or &quot;all_y&quot;.</td>
</tr>
<tr>
<td>facet_axis_labels</td>
<td>Whether to add interior axis labels with &quot;margins&quot;, &quot;all&quot;, &quot;all_x&quot;, or &quot;all_y&quot;.</td>
</tr>
<tr>
<td>facet_labels</td>
<td>A function that takes the breaks as inputs (e.g. (x) <code>stringr::str_to_sentence(x)</code>), or a named vector of labels (e.g. c(&quot;value1&quot; = &quot;label1&quot;, ...)).</td>
</tr>
<tr>
<td>facet_labels_position</td>
<td>When the facet layout is &quot;wrap&quot;, the position of the facet labels. Either &quot;top&quot;, &quot;right&quot;, &quot;bottom&quot; or &quot;left&quot;.</td>
</tr>
<tr>
<td>facet_labels_switch</td>
<td>When the facet layout is &quot;grid&quot;, whether to switch the facet labels to the opposite side of the plot. Either &quot;x&quot;, &quot;y&quot; or &quot;both&quot;.</td>
</tr>
<tr>
<td>facet_layout</td>
<td>Whether the layout is to be &quot;wrap&quot; or &quot;grid&quot;. If NULL and a single facet (or facet2) argument is provided, then defaults to &quot;wrap&quot;. If NULL and both facet and facet2 arguments are provided, defaults to &quot;grid&quot;.</td>
</tr>
<tr>
<td>facet_ncol</td>
<td>The number of columns of facets. Only applies to a facet layout of &quot;wrap&quot;.</td>
</tr>
<tr>
<td>facet_nrow</td>
<td>The number of rows of facets. Only applies to a facet layout of &quot;wrap&quot;.</td>
</tr>
<tr>
<td>facet_scales</td>
<td>Whether facet scales should be &quot;fixed&quot; across facets, &quot;free&quot; in both directions, or free in just one direction (i.e. &quot;free_x&quot; or &quot;free_y&quot;). Defaults to &quot;fixed&quot;.</td>
</tr>
<tr>
<td>facet_space</td>
<td>When the facet layout is &quot;grid&quot; and facet scales are not &quot;fixed&quot;, whether facet space should be &quot;fixed&quot; across facets, &quot;free&quot; to be proportional in both directions, or free to be proportional in just one direction (i.e. &quot;free_x&quot; or &quot;free_y&quot;). Defaults to &quot;fixed&quot;.</td>
</tr>
</tbody>
</table>
gg_raster

Description

Create a raster ggplot with a wrapper around `ggplot2::ggplot()` + `geom_raster()`.

Usage

```r
gg_raster(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
```

Value

A ggplot object.

Examples

```r
if (requireNamespace("quantreg", quietly = TRUE)) {
  library(ggplot2)
  library(palmerpenguins)

  penguins |>
    gg_quantile(
      x = flipper_length_mm,
      y = body_mass_g,
    )
}
```
z = NULL,
col = NULL,
alpha = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

data A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.
### gg_raster

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>yend</td>
<td>Unquoted yend aesthetic variable.</td>
</tr>
<tr>
<td>z</td>
<td>Unquoted z aesthetic variable.</td>
</tr>
<tr>
<td>col</td>
<td>Unquoted col aesthetic variable.</td>
</tr>
<tr>
<td>alpha</td>
<td>Unquoted alpha aesthetic variable.</td>
</tr>
<tr>
<td>facet</td>
<td>Unquoted facet aesthetic variable.</td>
</tr>
<tr>
<td>facet2</td>
<td>Unquoted facet2 aesthetic variable.</td>
</tr>
<tr>
<td>group</td>
<td>Unquoted group aesthetic variable.</td>
</tr>
<tr>
<td>subgroup</td>
<td>Unquoted subgroup aesthetic variable.</td>
</tr>
<tr>
<td>label</td>
<td>Unquoted label aesthetic variable.</td>
</tr>
<tr>
<td>text</td>
<td>Unquoted text aesthetic variable.</td>
</tr>
<tr>
<td>sample</td>
<td>Unquoted sample aesthetic variable.</td>
</tr>
<tr>
<td>mapping</td>
<td>Set of additional aesthetic mappings within <code>ggplot2::aes()</code> for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.</td>
</tr>
<tr>
<td>x_breaks, y_breaks</td>
<td>A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_pretty()</code>), or a vector of breaks.</td>
</tr>
<tr>
<td>x_expand, y_expand</td>
<td>Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code>).</td>
</tr>
<tr>
<td>x_expand_limits, y_expand_limits</td>
<td>For a continuous variable, any values that the limits should encompass (e.g. 0).</td>
</tr>
</tbody>
</table>
| x_labels, y_labels | A function that takes the breaks as inputs (e.g. `
(x) stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels. |
| x_limits, y_limits | A vector of length 2 to determine the limits of the axis. |
| x_oob, y_oob | For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`. |
| x_position, y_position | The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y_position = "top"` with a *_mode_* theme, add `caption = ""` or `caption = "n"`. |
| x_title, y_title | Axis title string. Use "" for no title. |
| x_transform, y_transform | For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_ prefix (e.g. "log10"). |
| col_breaks | A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks. |
| col_expand_limits | For a continuous variable, any values that the limits should encompass (e.g. 0). |
| col_labels | A function that takes the breaks as inputs (e.g. `
(x) stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels. |
col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a scales::oob_* function of how to handle
values outside of limits. Defaults to scales::oob_keep.

col_pal
Colours to use. A character vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
For a continuous variable, a scales::rescale() function.

col_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. De-
defaults to FALSE, which colours in a gradient.

col_title
Axis title string. Use "" for no title.

col_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10())
or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector
of breaks.

alpha_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)
or scales::label_comma()), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow
The number of columns and rows for the legend guide.

alpha_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
A vector of length 2 to determine the limits of the axis.

alpha_oob
For a continuous scale variable, a scales::oob_* function of how to handle
values outside of limits. Defaults to scales::oob_keep.

alpha_pal
Alpha values to use. For a continuous variable, a vector of length 2 between 0
and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na
Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title
Axis title string. Use "" for no title.

alpha_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10())
or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes
Whether to add interior axes and ticks with "margins", "all", "all_x", or
"all_y".

facet_axis_labels
Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels
A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)),
or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position
When the facet layout is "wrap", the position of the facet labels. Either "top",
"right", "bottom" or "left".
When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

The number of columns of facets. Only applies to a facet layout of "wrap".

The number of rows of facets. Only applies to a facet layout of "wrap".

Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

Title string.

Subtitle string.

Caption title string.

A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

A ggplot object.

library(ggplot2)
library(dplyr)
faithfuld |>
gg_raster(
  x = waiting,
  y = eruptions,
  col = density,
)

Create a rect ggplot with a wrapper around ggplot2::ggplot() + geom_rect().
Usage

gg_rect(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
  x_title = NULL,
  x_transform = NULL,
  y_breaks = NULL,
  y_expand = NULL,
  y_expand_limits = NULL,
  y_labels = NULL,
  y_limits = NULL,
  y_oob = scales::oob_keep,
  y_position = "left",
  y_title = NULL,
  y_transform = NULL,
  col_breaks = NULL,
  col_expand_limits = NULL,
  col_labels = NULL,
Arguments

- **data**
  
  A data frame or tibble.

- **...**
  
  Other arguments passed to within a `params` list in `layer()`.

- **stat**
  
  A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

- **position**
  
  A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. `ggplot2::position_identity()`).
coord
A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode
A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x
Unquoted x aesthetic variable.

xmin
Unquoted xmin aesthetic variable.

xmax
Unquoted xmax aesthetic variable.

xend
Unquoted xend aesthetic variable.

y
Unquoted y aesthetic variable.

ymin
Unquoted ymin aesthetic variable.

ymax
Unquoted ymax aesthetic variable.

yend
Unquoted yend aesthetic variable.

z
Unquoted z aesthetic variable.

col
Unquoted col aesthetic variable.

alpha
Unquoted alpha aesthetic variable.

facet
Unquoted facet aesthetic variable.

facet2
Unquoted facet2 aesthetic variable.

group
Unquoted group aesthetic variable.

subgroup
Unquoted subgroup aesthetic variable.

label
Unquoted label aesthetic variable.

text
Unquoted text aesthetic variable.

sample
Unquoted sample aesthetic variable.

mapping
Set of additional aesthetic mappings within ggplot2::aes() for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

x_expand, y_expand
Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).

x_expand_limits, y_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels
A function that takes the breaks as inputs (e.g. \( \backslash (x) \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

x_limits, y_limits
A vector of length 2 to determine the limits of the axis.

x_oob, y_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
x_position, y_position
   The position of the axis (i.e. "left", "right", "bottom" or "top"). If using
   y_position = "top" with a _mode_* theme, add caption = "" or caption = "\n".

x_title, y_title
   Axis title string. Use "" for no title.

x_transform, y_transform
   For a numeric scale, a transformation object (e.g. scales::transform_log10())
   or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks
   A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector
   of breaks.

col_expand_limits
   For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
   A function that takes the breaks as inputs (e.g. \( \backslash (x) \) stringr::str_to_sentence(x)
   or scales::label_comma()), or a vector of labels.

col_legend_ncol, col_legend_nrow
   The number of columns and rows for the legend guide.

col_legend_rev
   Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
   A vector of length 2 to determine the limits of the axis.

col_oob
   For a continuous scale variable, a scales::oob_ function of how to handle
   values outside of limits. Defaults to scales::oob_keep.

col_pal
   Colours to use. A character vector of hex codes (or names).

col_pal_na
   Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
   For a continuous variable, a scales::rescale() function.

col_steps
   For a continuous variable, TRUE or FALSE of whether to colour in steps. De-
  faults to FALSE, which colours in a gradient.

col_title
   Axis title string. Use "" for no title.

col_transform
   For a numeric scale, a transformation object (e.g. scales::transform_log10())
   or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks
   A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector
   of breaks.

alpha_expand_limits
   For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
   A function that takes the breaks as inputs (e.g. \( \backslash (x) \) stringr::str_to_sentence(x)
   or scales::label_comma()), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow
   The number of columns and rows for the legend guide.

alpha_legend_rev
   Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
   A vector of length 2 to determine the limits of the axis.

alpha_oob
   For a continuous scale variable, a scales::oob_* function of how to handle
   values outside of limits. Defaults to scales::oob_keep.
alpha_pal  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na  Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title  Axis title string. Use "" for no title.

alpha_transform  For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes  Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels  Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels  A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position  When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch  When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout  Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol  The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow  The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales  Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title  Title string.

subtitle  Subtitle string.

caption  Caption title string.

titles_to_case  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value

A ggplot object.
Examples

```r
library(ggplot2)
library(dplyr)

data.frame(
  x = rep(c(2, 5, 7, 9, 12), 2),
  y = rep(c(1, 2), each = 5),
  z = factor(c(rep(1:3, each = 3), 4)),
  w = rep(diff(c(0, 4, 6, 8, 10, 14)), 2)) |
> mutate(
  xmin = x - w / 2,
  xmax = x + w / 2,
  ymin = y,
  ymax = y + 1
) |
> gg_rect(
  xmin = xmin,
  xmax = xmax,
  ymin = ymin,
  ymax = ymax,
  col = z,
)
```

---

**gg_ribbon**

*Ribbon ggplot*

Description

Create a ribbon ggplot with a wrapper around `ggplot2::ggplot()` + `geom_ribbon()`

Usage

```r
gg_ribbon(
  data = NULL,
  ..., 
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
```
yend = NULL,
z = NULL,
col = NULL,
alpha = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend(ncol = NULL,


Arguments

**data**
A data frame or tibble.

**...**
Other arguments passed to within a params list in `layer()`.

**stat**
A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

**position**
A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

**coord**
A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

**mode**
A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

**x**
Unquoted $x$ aesthetic variable.

**xmin**
Unquoted $x_{min}$ aesthetic variable.

**xmax**
Unquoted $x_{max}$ aesthetic variable.

**xend**
Unquoted $x_{end}$ aesthetic variable.

**y**
Unquoted $y$ aesthetic variable.

**ymin**
Unquoted $y_{min}$ aesthetic variable.
ymax  Unquoted ymax aesthetic variable.
yend  Unquoted yend aesthetic variable.
z  Unquoted z aesthetic variable.
col  Unquoted col aesthetic variable.
alpha  Unquoted alpha aesthetic variable.
facet  Unquoted facet aesthetic variable.
facet2  Unquoted facet2 aesthetic variable.
group  Unquoted group aesthetic variable.
subgroup  Unquoted subgroup aesthetic variable.
label  Unquoted label aesthetic variable.
text  Unquoted text aesthetic variable.
sample  Unquoted sample aesthetic variable.
mapping  Set of additional aesthetic mappings within \texttt{ggplot2::aes()} for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

\texttt{x\_breaks, y\_breaks}
A \texttt{scales::breaks\_\*} function (e.g. \texttt{scales::breaks_pretty()}), or a vector of breaks.

\texttt{x\_expand, y\_expand}
Padding to the limits with the \texttt{ggplot2::expansion()} function, or a vector of length 2 (e.g. \texttt{c(0, 0)}).

\texttt{x\_expand\_limits, y\_expand\_limits}
For a continuous variable, any values that the limits should encompass (e.g. \texttt{0}).

\texttt{x\_labels, y\_labels}
A function that takes the breaks as inputs (e.g. \texttt{\(x\) stringr::str\_to\_sentence(x)} or \texttt{scales::label\_comma()}), or a vector of labels.

\texttt{x\_limits, y\_limits}
A vector of length 2 to determine the limits of the axis.

\texttt{x\_oob, y\_oob}
For a continuous scale variable, a \texttt{scales::oob\_\*} function of how to handle values outside of limits. Defaults to \texttt{scales::oob\_keep}.

\texttt{x\_position, y\_position}
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using \texttt{y\_position = "top"} with a \_mode\_\* theme, add \texttt{caption = ""} or \texttt{caption = "\n"}.

\texttt{x\_title, y\_title}
Axis title string. Use "" for no title.

\texttt{x\_transform, y\_transform}
For a numeric scale, a transformation object (e.g. \texttt{scales::transform\_log10()}) or character string of this minus the transform\_ prefix (e.g. "log10").

\texttt{col\_breaks}
A \texttt{scales::breaks\_\*} function (e.g. \texttt{scales::breaks_pretty()}), or a vector of breaks.

\texttt{col\_expand\_limits}
For a continuous variable, any values that the limits should encompass (e.g. \texttt{0}).
col_labels  A function that takes the breaks as inputs (e.g. \( x \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

col_legend_ncol, col_legend_nrow  
The number of columns and rows for the legend guide.

col_legend_rev  Reverse the elements of the legend guide. Defaults to FALSE.

col_limits  A vector of length 2 to determine the limits of the axis.

col_oob  For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

col_pal  Colours to use. A character vector of hex codes (or names).

col_pal_na  Colour to use for NA values. A character vector of a hex code (or name).

col_rescale  For a continuous variable, a scales::rescale() function.

col_steps  For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title  Axis title string. Use "" for no title.

col_transform  For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_prefix (e.g. "log10").

alpha_breaks  A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

alpha_limits  For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels  A function that takes the breaks as inputs (e.g. \( x \) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow  
The number of columns and rows for the legend guide.

alpha_legend_rev  Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits  A vector of length 2 to determine the limits of the axis.

alpha_oob  For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na  Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title  Axis title string. Use "" for no title.

alpha_transform  For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_prefix (e.g. "log10").

facet_axes  Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels  Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels  A function that takes the breaks as inputs (e.g. \( x \) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).
When the facet layout is "wrap," the position of the facet labels. Either "top", "right", "bottom" or "left".

When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

The number of columns of facets. Only applies to a facet layout of "wrap".

The number of rows of facets. Only applies to a facet layout of "wrap".

Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

Title string.

Subtitle string.

Caption title string.

A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

A ggplot object.

library(ggplot2)
library(dplyr)

data.frame(year = 1875:1972, level = as.vector(LakeHuron)) |>
  mutate(level_min = level - 1, level_max = level + 1) |>
  gg_ribbon(
    x = year,
    ymin = level_min,
    ymax = level_max,
    y_title = "Level",
    col_pal = scales::alpha(blue, 0),
  ) +
  geom_line(
    mapping = aes(x = year, y = level),
    colour = blue,
  )
Description

Create a rug ggplot with a wrapper around `ggplot2::ggplot()` + `geom_rug()`.

Usage

```r
gg_rug(
  data = NULL,
  ..., 
  stat = "identity", 
  position = "identity", 
  coord = ggplot2::coord_cartesian(clip = "off"), 
  mode = NULL, 
  x = NULL, 
  xmin = NULL, 
  xmax = NULL, 
  xend = NULL, 
  y = NULL, 
  ymin = NULL, 
  ymax = NULL, 
  yend = NULL, 
  z = NULL, 
  col = NULL, 
  alpha = NULL, 
  facet = NULL, 
  facet2 = NULL, 
  group = NULL, 
  subgroup = NULL, 
  label = NULL, 
  text = NULL, 
  sample = NULL, 
  mapping = NULL, 
  x_breaks = NULL, 
  x_expand = NULL, 
  x_expand_limits = NULL, 
  x_labels = NULL, 
  x_limits = NULL, 
  x_oob = scales::oob_keep, 
  x_position = "bottom", 
  x_title = NULL, 
  x_transform = NULL, 
  y_breaks = NULL, 
  y_expand = NULL, 
  y_expand_limits = NULL, 
  y_labels = NULL, 
  y_title = NULL, 
  y_transform = NULL, 
  ...) 
)```
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha-expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case)
Arguments

data
A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.

yend Unquoted yend aesthetic variable.

z Unquoted z aesthetic variable.

col Unquoted col aesthetic variable.

alpha Unquoted alpha aesthetic variable.

facet Unquoted facet aesthetic variable.

facet2 Unquoted facet2 aesthetic variable.

group Unquoted group aesthetic variable.

subgroup Unquoted subgroup aesthetic variable.

label Unquoted label aesthetic variable.

text Unquoted text aesthetic variable.

sample Unquoted sample aesthetic variable.

mapping Set of additional aesthetic mappings within ggplot2::aes() for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

x_expand, y_expand Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).
x_expand_limits, y_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels
A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

x_limits, y_limits
A vector of length 2 to determine the limits of the axis.

x_oob, y_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".

x_title, y_title
Axis title string. Use "" for no title.

x_transform, y_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_prefix (e.g. "log10").

col_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

col_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

col_pal
Colours to use. A character vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
For a continuous variable, a scales::rescale() function.

col_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title
Axis title string. Use "" for no title.

col_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_prefix (e.g. "log10").

alpha_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

alpha_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.
alpha_legend_ncol, alpha_legend_nrow
  The number of columns and rows for the legend guide.

alpha_legend_rev
  Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
  A vector of length 2 to determine the limits of the axis.

alpha_oob
  For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal
  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na
  Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title
  Axis title string. Use "" for no title.

alpha_transform
  For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes
  Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels
  Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels
  A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position
  When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch
  When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout
  Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol
  The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow
  The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales
  Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space
  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title
  Title string.

subtitle
  Subtitle string.

caption
  Caption title string.

titles_to_case
  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.
Value
A ggplot object.

Examples

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

penguins |>
  mutate(across(sex, \(x\) stringr::str_to_sentence(x))) |>
  gg_rug(
    x = flipper_length_mm,
    y = body_mass_g,
    col = sex,
  )
```

---

### gg_segment

**Segment ggplot**

**Description**
Create a segment ggplot with a wrapper around `ggplot2::ggplot()` + `geom_segment()`.

**Usage**

```
gg_segment(
  data = NULL,
  ..., stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
```
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

data A data frame or tibble.
...
Other arguments passed to within a params list in layer().
stat A statistical transformation to use on the data. A snakecase character string of a
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position A position adjustment. A snakecase character string of a ggproto Position sub-
class object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).
coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).
mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_.*.
x Unquoted x aesthetic variable.
xmin Unquoted xmin aesthetic variable.
xmax Unquoted xmax aesthetic variable.
xend Unquoted xend aesthetic variable.
y Unquoted y aesthetic variable.
ymin Unquoted ymin aesthetic variable.
ymax Unquoted ymax aesthetic variable.
yend Unquoted yend aesthetic variable.
z Unquoted z aesthetic variable.
col Unquoted col aesthetic variable.
alpha Unquoted alpha aesthetic variable.
**gg_segment**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>facet</td>
<td>Unquoted facet aesthetic variable.</td>
</tr>
<tr>
<td>facet2</td>
<td>Unquoted facet2 aesthetic variable.</td>
</tr>
<tr>
<td>group</td>
<td>Unquoted group aesthetic variable.</td>
</tr>
<tr>
<td>subgroup</td>
<td>Unquoted subgroup aesthetic variable.</td>
</tr>
<tr>
<td>label</td>
<td>Unquoted label aesthetic variable.</td>
</tr>
<tr>
<td>text</td>
<td>Unquoted text aesthetic variable.</td>
</tr>
<tr>
<td>sample</td>
<td>Unquoted sample aesthetic variable.</td>
</tr>
<tr>
<td>mapping</td>
<td>Set of additional aesthetic mappings within <code>ggplot2::aes()</code> for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.</td>
</tr>
<tr>
<td>x_breaks, y_breaks</td>
<td>A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_pretty()</code>), or a vector of breaks.</td>
</tr>
<tr>
<td>x_expand, y_expand</td>
<td>Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code>).</td>
</tr>
<tr>
<td>x_expand_limits, y_expand_limits</td>
<td>For a continuous variable, any values that the limits should encompass (e.g. 0).</td>
</tr>
<tr>
<td>x_labels, y_labels</td>
<td>A function that takes the breaks as inputs (e.g. ( x ) <code>stringr::str_to_sentence(x)</code> or <code>scales::label_comma()</code>) , or a vector of labels.</td>
</tr>
<tr>
<td>x_limits, y_limits</td>
<td>A vector of length 2 to determine the limits of the axis.</td>
</tr>
<tr>
<td>x_oob, y_oob</td>
<td>For a continuous scale variable, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code>.</td>
</tr>
<tr>
<td>x_position, y_position</td>
<td>The position of the axis (i.e. &quot;left&quot;, &quot;right&quot;, &quot;bottom&quot; or &quot;top&quot;). If using <code>y_position = &quot;top&quot;</code> with a <em><em>mode</em></em> theme, add <code>caption = &quot;&quot;</code> or <code>caption = &quot;\n&quot;</code>.</td>
</tr>
<tr>
<td>x_title, y_title</td>
<td>Axis title string. Use &quot;&quot; for no title.</td>
</tr>
<tr>
<td>x_transform, y_transform</td>
<td>For a numeric scale, a transformation object (e.g. <code>scales::transform_log10()</code>) or character string of this minus the transform_ prefix (e.g. &quot;log10&quot;).</td>
</tr>
<tr>
<td>col_breaks</td>
<td>A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_pretty()</code>), or a vector of breaks.</td>
</tr>
<tr>
<td>col_expand_limits</td>
<td>For a continuous variable, any values that the limits should encompass (e.g. 0).</td>
</tr>
<tr>
<td>col_labels</td>
<td>A function that takes the breaks as inputs (e.g. ( x ) <code>stringr::str_to_sentence(x)</code> or <code>scales::label_comma()</code>) , or a vector of labels.</td>
</tr>
<tr>
<td>col_legend_ncol, col_legend_nrow</td>
<td>The number of columns and rows for the legend guide.</td>
</tr>
<tr>
<td>col_legend_rev</td>
<td>Reverse the elements of the legend guide. Defaults to FALSE.</td>
</tr>
<tr>
<td>col_limits</td>
<td>A vector of length 2 to determine the limits of the axis.</td>
</tr>
</tbody>
</table>
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

Colours to use. A character vector of hex codes (or names).

Colour to use for NA values. A character vector of a hex code (or name).

For a continuous variable, a `scales::rescale()` function.

For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. `scales::transform_log10()`)
or character string of this minus the transform_ prefix (e.g. "log10").

A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

For a continuous variable, any values that the limits should encompass (e.g. 0).

A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)`)or `scales::label_comma()`), or a vector of labels.

The number of columns and rows for the legend guide.

Reverse the elements of the legend guide. Defaults to FALSE.

A vector of length 2 to determine the limits of the axis.

For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

Alpha value to use for the NA value. A integer between 0 and 1.

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. `scales::transform_log10()`)
or character string of this minus the transform_ prefix (e.g. "log10").

Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)`,or a named vector of labels (e.g. c("value1" = "label1", ...)).

When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".
facet_layout  Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol  The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow  The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales  Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title  Title string.

subtitle  Subtitle string.

caption  Caption title string.

titles_to_case  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value

A ggplot object.

Examples

```r
library(ggplot2)
library(dplyr)

data.frame(x1 = 2.62, x2 = 3.57, y1 = 21.0, y2 = 15.0) |>
gg_segment(
  x = x1,
  xend = x2,
  y = y1,
  yend = y2,
)
```

Description

Create a blank ggplot with a wrapper around `ggplot2::ggplot() + geom_sf()`.
Usage

gg_sf(
  data = NULL,
  ..., 
  stat = "sf",
  position = "identity",
  coord = ggplot2::coord_sf(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_position = "bottom",
  x_title = NULL,
  x_transform = NULL,
  y_breaks = NULL,
  y_expand = NULL,
  y_expand_limits = NULL,
  y_labels = NULL,
  y_limits = NULL,
  y_oob = scales::oob_keep,
  y_position = "left",
  y_title = NULL,
  y_transform = NULL,
  col_breaks = NULL,
  col_expand_limits = NULL,
  col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

data A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a gggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a gggproto Position subclass object (e.g. ggplot2::position_identity()).
coord  A coordinate system. A `coord_*()` function that outputs a constructed `ggproto Coord` subclass object (e.g. `ggplot2::coord_cartesian()`).

mode A `_mode_*` theme (e.g. `grey_mode_b()`, `grey_mode_r()`, or `dark_mode_r()`). This argument adds the theme with side-effects, as the `gg_*` function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of `gg_*`.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.

yend Unquoted yend aesthetic variable.

z Unquoted z aesthetic variable.

col Unquoted col aesthetic variable.

alpha Unquoted alpha aesthetic variable.

facet Unquoted facet aesthetic variable.

facet2 Unquoted facet2 aesthetic variable.

group Unquoted group aesthetic variable.

subgroup Unquoted subgroup aesthetic variable.

label Unquoted label aesthetic variable.

text Unquoted text aesthetic variable.

sample Unquoted sample aesthetic variable.

mapping Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

x_expand, y_expand Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

x_expand_limits, y_expand_limits For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels A function that takes the breaks as inputs (e.g. `(x)` `stringr::str_to_sentence(x)`) or `scales::label_comma()`), or a vector of labels.

x_limits, y_limits A vector of length 2 to determine the limits of the axis.

x_oob, y_oob For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`. 
x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using
y_position = "top" with a *mode_* theme, add caption = "" or caption = "\n".

x_title, y_title
Axis title string. Use "" for no title.

x_transform, y_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10())
or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector
of breaks.

col_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)
or scales::label_comma()), or a vector of labels.

col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a scales::oob_* function of how to handle
values outside of limits. Defaults to scales::oob_keep.

col_pal
Colours to use. A character vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
For a continuous variable, a scales::rescale() function.

col_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. De-
defaults to FALSE, which colours in a gradient.

col_title
Axis title string. Use "" for no title.

col_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10())
or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector
of breaks.

alpha_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)
or scales::label_comma()), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow
The number of columns and rows for the legend guide.

alpha_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
A vector of length 2 to determine the limits of the axis.

alpha_oob
For a continuous scale variable, a scales::oob_* function of how to handle
values outside of limits. Defaults to scales::oob_keep.
alpha_pal  Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na  Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title  Axis title string. Use "" for no title.

alpha_transform  For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes  Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels  Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels  A function that takes the breaks as inputs (e.g. \(x) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position  When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch  When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout  Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol  The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow  The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales  Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title  Title string.

subtitle  Subtitle string.

caption  Caption title string.

titles_to_case  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value  A ggplot object.
Examples

library(ggplot2)
library(dplyr)

if (requireNamespace("sf", quietly = TRUE)) {
  sf::st_read(system.file("shape/nc.shp", package = "sf")) |> 
    gg_sf(
      col = AREA,
    )
}

---

**gg_smooth**  
*Smooth ggplot*

### Description

Create a smooth ggplot with a wrapper around `ggplot2::ggplot()` + `geom_smooth()`.

### Usage

```r
GG_smooth(
  data = NULL,
  ..., ...
  stat = "smooth",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
)```

mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facets 

Arguments

data A data frame or tibble.
... Other arguments passed to within a params list in layer().
stat A statistical transformation to use on the data. A snakecase character string of a 
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).
coord A coordinate system. A coord_*() function that outputs a constructed ggproto 
Coord subclass object (e.g. ggplot2::coord_cartesian()).
mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()).

This argument adds the theme with side-effects, as the gg_* function will re-

moves selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme 
on to the output of gg_*.
x Unquoted x aesthetic variable.
xmin Unquoted xmin aesthetic variable.
xmax Unquoted xmax aesthetic variable.
xend Unquoted xend aesthetic variable.
y Unquoted y aesthetic variable.
ymin Unquoted ymin aesthetic variable.
ymax Unquoted ymax aesthetic variable.
yend Unquoted yend aesthetic variable.
z Unquoted z aesthetic variable.
col Unquoted col aesthetic variable.
alpha Unquoted alpha aesthetic variable.
facet Unquoted facet aesthetic variable.
facet2 Unquoted facet2 aesthetic variable.
group Unquoted group aesthetic variable.
subgroup Unquoted subgroup aesthetic variable.

titles_to_case = snakecase::to_sentence_case
)
**gg_smooth**

- **label**: Unquoted label aesthetic variable.
- **text**: Unquoted text aesthetic variable.
- **sample**: Unquoted sample aesthetic variable.
- **mapping**: Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

**x_breaks, y_breaks**
- A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

**x_expand, y_expand**
- Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

**x_expand_limits, y_expand_limits**
- For a continuous variable, any values that the limits should encompass (e.g. 0).

**x_labels, y_labels**
- A function that takes the breaks as inputs (e.g. \(x) stringr::str_to_sentence(x) or `scales::label_comma()`), or a vector of labels.

**x_limits, y_limits**
- A vector of length 2 to determine the limits of the axis.

**x_oob, y_oob**
- For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

**x_position, y_position**
- The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y_position = "top"` with a *_mode_* theme, add caption = "" or caption = "\n".

**x_title, y_title**
- Axis title string. Use "" for no title.

**x_transform, y_transform**
- For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_ prefix (e.g. "log10").

**col_breaks**
- A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

**col_expand_limits**
- For a continuous variable, any values that the limits should encompass (e.g. 0).

**col_labels**
- A function that takes the breaks as inputs (e.g. \(x) stringr::str_to_sentence(x) or `scales::label_comma()`), or a vector of labels.

**col_legend_ncol, col_legend_nrow**
- The number of columns and rows for the legend guide.

**col_legend_rev**
- Reverse the elements of the legend guide. Defaults to `FALSE`.

**col_limits**
- A vector of length 2 to determine the limits of the axis.

**col_oob**
- For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

**col_pal**
- Colours to use. A character vector of hex codes (or names).

**col_pal_na**
- Colour to use for NA values. A character vector of a hex code (or name).
For a continuous variable, a `scales::rescale()` function.

For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the `transform_` prefix (e.g. "log10").

A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

For a continuous variable, any values that the limits should encompass (e.g. 0).

A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

The number of columns and rows for the legend guide.

Reverse the elements of the legend guide. Defaults to FALSE.

A vector of length 2 to determine the limits of the axis.

For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

Alpha value to use for the NA value. A integer between 0 and 1.

Axis title string. Use "" for no title.

For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the `transform_` prefix (e.g. "log10").

Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)`), or a named vector of labels (e.g. c("value1" = "label1", ...)).

When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

The number of columns of facets. Only applies to a facet layout of "wrap".

The number of rows of facets. Only applies to a facet layout of "wrap".
facet_scales  Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title  Title string.
subtitle  Subtitle string.
caption  Caption title string.
titles_to_case  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value
A ggplot object.

Examples

library(ggplot2)
library(dplyr)
library(palmerpenguins)
penguins |> 
  mutate(across(sex, \(x\) stringr::str_to_sentence(x))) |> 
  tidyr::drop_na(sex) |> 
  gg_smooth(
    x = flipper_length_mm, 
    y = body_mass_g, 
    col = sex, 
    se = TRUE, 
  )
position = "identity",
coord = ggplot2::coord_cartesian(clip = "off"),
mode = NULL,
x = NULL,
xmin = NULL,
xmax = NULL,
xend = NULL,
y = NULL,
ymin = NULL,
ymax = NULL,
yend = NULL,
z = NULL,
col = NULL,
alpha = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_limits = NULL,
col_oob = scales::oob_keep,
Arguments

data A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will re-
moves selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x Unquoted x aesthetic variable.
xmin Unquoted xmin aesthetic variable.
xmax Unquoted xmax aesthetic variable.
xend Unquoted xend aesthetic variable.
y Unquoted y aesthetic variable.
ymin Unquoted ymin aesthetic variable.
ymax Unquoted ymax aesthetic variable.
yend Unquoted yend aesthetic variable.
z Unquoted z aesthetic variable.
col Unquoted col aesthetic variable.
alpha Unquoted alpha aesthetic variable.
facet Unquoted facet aesthetic variable.
facet2 Unquoted facet2 aesthetic variable.
group Unquoted group aesthetic variable.
subgroup Unquoted subgroup aesthetic variable.
label Unquoted label aesthetic variable.
text Unquoted text aesthetic variable.
sample Unquoted sample aesthetic variable.
mapping Set of additional aesthetic mappings within ggplot2::aes() for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks
A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

x_expand, y_expand
Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).

x_expand_limits, y_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels
A function that takes the breaks as inputs (e.g. \(x) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.

x_limits, y_limits
A vector of length 2 to determine the limits of the axis.

x_oob, y_oob For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".
x_title, y_title
    Axis title string. Use "" for no title.

x_transform, y_transform
    For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks
    A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

col_expand_limits
    For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
    A function that takes the breaks as inputs (e.g. \(\text{\texttt{\textbackslash{(x) stringr::str_to_sentence(x)}}}\) or scales::label_comma()), or a vector of labels.

col_legend_ncol, col_legend_nrow
    The number of columns and rows for the legend guide.

col_legend_rev
    Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
    A vector of length 2 to determine the limits of the axis.

col_oob
    For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

col_pal
    Colours to use. A character vector of hex codes (or names).

col_pal_na
    Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
    For a continuous variable, a scales::rescale() function.

col_steps
    For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.

col_title
    Axis title string. Use "" for no title.

col_transform
    For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks
    A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.

alpha_expand_limits
    For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
    A function that takes the breaks as inputs (e.g. \(\text{\texttt{\textbackslash{(x) stringr::str_to_sentence(x)}}}\) or scales::label_comma()), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow
    The number of columns and rows for the legend guide.

alpha_legend_rev
    Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
    A vector of length 2 to determine the limits of the axis.

alpha_oob
    For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal
    Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na
    Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title
    Axis title string. Use "" for no title.
alpha_transform
For a numeric scale, a transformation object (e.g. scales::transform_log10())
or character string of this minus the transform_ prefix (e.g. "log10").

facet_axes Whether to add interior axes and ticks with "margins", "all", "all_x", or
"all_y".

facet_axis_labels Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels A function that takes the breaks as inputs (e.g. \(x) stringr::str_to_sentence(x)),
or a named vector of labels (e.g. c("value1" = "label1", ...)).

facet_labels_position When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title Title string.

subtitle Subtitle string.
caption Caption title string.
titles_to_case A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value
A ggplot object.

Examples

library(ggplot2)
library(dplyr)
economics |>
  gg_step(
x = date,
y = unemploy,
coord = ggplot2::coord_cartesian(clip = "on"),
x_limits = c(lubridate::ymd("2010-01-01"), lubridate::NA_Date_),
y_expand_limits = 0,
y_title = "Unemployment",
)

---

**gg_text**  
**Text ggplot**

**Description**

Create a text plot with a wrapper around `ggplot2::ggplot() + geom_text()`.

**Usage**

```r
gg_text(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  ```
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legacy_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legacy_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",}
 facet_space = "fixed",
 title = NULL,
 subtitle = NULL,
 caption = NULL,
 titles_to_case = snakecase::to_sentence_case
)

Arguments

data A data frame or tibble.
...
Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggproto Position sub-
class object minus the Position prefix (e.g. "identity"), or a position_() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_() function that outputs a constructed ggproto
Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()).
This argument adds the theme with side-effects, as the gg_* function will re-
moves selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme
on to the output of gg_*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.

yend Unquoted yend aesthetic variable.

z Unquoted z aesthetic variable.

col Unquoted col aesthetic variable.

alpha Unquoted alpha aesthetic variable.

facet Unquoted facet aesthetic variable.

facet2 Unquoted facet2 aesthetic variable.

group Unquoted group aesthetic variable.

subgroup Unquoted subgroup aesthetic variable.

label Unquoted label aesthetic variable.

text Unquoted text aesthetic variable.

sample Unquoted sample aesthetic variable.
mapping
Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

x_expand, y_expand
Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

x_expand_limits, y_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels
A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

x_limits, y_limits
A vector of length 2 to determine the limits of the axis.

x_oob, y_oob
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y_position = "top"` with a `*_mode_*` theme, add `caption = ""` or `caption = \"n\"`.

x_title, y_title
Axis title string. Use "" for no title.

x_transform, y_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the `transform_` prefix (e.g. "log10").

col_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

col_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

col_pal
Colours to use. A character vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
For a continuous variable, a `scales::rescale()` function.

col_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.
col_title  
Axis title string. Use "" for no title.

col_transform  
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`)
or character string of this minus the transform_prefix (e.g. "log10").

alpha_breaks  
A scales::breaks_* function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

alpha_expand_limits  
For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels  
A function that takes the breaks as inputs (e.g. \( x \) stringr::str_to_sentence(x)
or `scales::label_comma()`), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow  
The number of columns and rows for the legend guide.

alpha_legend_rev  
Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits  
A vector of length 2 to determine the limits of the axis.

alpha_oob  
For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.

alpha_pal  
Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na  
Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title  
Axis title string. Use "" for no title.

alpha_transform  
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`)
or character string of this minus the transform_prefix (e.g. "log10").

facet_axes  
Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels  
Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels  
A function that takes the breaks as inputs (e.g. \( x \) stringr::str_to_sentence(x)),
or a named vector of labels (e.g. c(“value1” = “label1”, ...)).

facet_labels_position  
When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch  
When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout  
Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol  
The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow  
The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales  
Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space  When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title  Title string.

subtitle  Subtitle string.

caption  Caption title string.

titles_to_case  A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value
A ggplot object.

Examples

library(ggplot2)
library(dplyr)

bind_rows(
  mtcars |> slice_min(order_by = mpg),
  mtcars |> slice_max(order_by = mpg)) |> tibble::rownames_to_column("model") |> gg_text(
  x = model,
  y = mpg,
  col = mpg,
  label = model,
  size = 3.53,
  y_expand_limits = 0,
  y_title = "Miles per gallon",
  col_pal = c(orange, "white", teal),
)

---

**gg_tile**  
*Tile ggplot*

Description

Create a tile plot with a wrapper around ggplot2::ggplot() + geom_tile().

Usage

```r
gg_tile(
  data = NULL,
  ..., 
  stat = "identity",
)```
position = "identity",
coord = ggplot2::coord_cartesian(clip = "off"),
mode = NULL,
x = NULL,
xmin = NULL,
xmax = NULL,
xend = NULL,
y = NULL,
ymin = NULL,
ymax = NULL,
yend = NULL,
z = NULL,
col = NULL,
alpha = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpa_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

**data**  
A data frame or tibble.

**...**  
Other arguments passed to within a params list in layer().

**stat**  
A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

**position**  
A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).

**coord**  
A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

**mode**  
A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will re-
moves selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

- **x**: Unquoted x aesthetic variable.
- **xmin**: Unquoted xmin aesthetic variable.
- **xmax**: Unquoted xmax aesthetic variable.
- **xend**: Unquoted xend aesthetic variable.
- **y**: Unquoted y aesthetic variable.
- **ymin**: Unquoted ymin aesthetic variable.
- **ymax**: Unquoted ymax aesthetic variable.
- **yend**: Unquoted yend aesthetic variable.
- **z**: Unquoted z aesthetic variable.
- **col**: Unquoted col aesthetic variable.
- **alpha**: Unquoted alpha aesthetic variable.
- **facet**: Unquoted facet aesthetic variable.
- **facet2**: Unquoted facet2 aesthetic variable.
- **group**: Unquoted group aesthetic variable.
- **subgroup**: Unquoted subgroup aesthetic variable.
- **label**: Unquoted label aesthetic variable.
- **text**: Unquoted text aesthetic variable.
- **sample**: Unquoted sample aesthetic variable.

**mapping**

Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. `shape`, `linetype`, `linewidth`, or `size`) or for delayed evaluation.

**x_breaks, y_breaks**

A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

**xexpand, yexpand**

Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

**x_expand_limits, y_expand_limits**

For a continuous variable, any values that the limits should encompass (e.g. 0).

**x_labels, y_labels**

A function that takes the breaks as inputs (e.g. \(x) \) `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

**x_limits, y_limits**

A vector of length 2 to determine the limits of the axis.

**x_oob, y_oob**

For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

**x_position, y_position**

The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".
x_title, y_title
   Axis title string. Use "" for no title.

x_transform, y_transform
   For a numeric scale, a transformation object (e.g. scales::transform_log10())
   or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks
   A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector
   of breaks.

col_expand_limits
   For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
   A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)
   or scales::label_comma()), or a vector of labels.

col_legend_ncol, col_legend_nrow
   The number of columns and rows for the legend guide.

col_legend_rev
   Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
   A vector of length 2 to determine the limits of the axis.

col_oob
   For a continuous scale variable, a scales::oob_* function of how to handle
   values outside of limits. Defaults to scales::oob_keep.

col_pal
   Colours to use. A character vector of hex codes (or names).

col_pal_na
   Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
   For a continuous variable, a scales::rescale() function.

col_steps
   For a continuous variable, TRUE or FALSE of whether to colour in steps. De-
  faults to FALSE, which colours in a gradient.

col_title
   Axis title string. Use "" for no title.

col_transform
   For a numeric scale, a transformation object (e.g. scales::transform_log10())
   or character string of this minus the transform_ prefix (e.g. "log10").

alpha_breaks
   A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector
   of breaks.

alpha_expand_limits
   For a continuous variable, any values that the limits should encompass (e.g. 0).

alpha_labels
   A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)
   or scales::label_comma()), or a vector of labels.

alpha_legend_ncol, alpha_legend_nrow
   The number of columns and rows for the legend guide.

alpha_legend_rev
   Reverse the elements of the legend guide. Defaults to FALSE.

alpha_limits
   A vector of length 2 to determine the limits of the axis.

alpha_oob
   For a continuous scale variable, a scales::oob_* function of how to handle
   values outside of limits. Defaults to scales::oob_keep.

alpha_pal
   Alpha values to use. For a continuous variable, a vector of length 2 between 0
   and 1. For a discrete variable, a vector of integers between 0 and 1.

alpha_pal_na
   Alpha value to use for the NA value. A integer between 0 and 1.

alpha_title
   Axis title string. Use "" for no title.
alpha_transform

For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the `transform_` prefix (e.g. "log10").

facet_axes

Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".

facet_axis_labels

Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_labels

A function that takes the breaks as inputs (e.g. `\(x\) stringr::str_to_sentence(x)`), or a named vector of labels (e.g. `c("value1" = "label1", ...)`).

facet_labels_position

When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".

facet_labels_switch

When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".

facet_layout

Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".

facet_ncol

The number of columns of facets. Only applies to a facet layout of "wrap".

facet_nrow

The number of rows of facets. Only applies to a facet layout of "wrap".

facet_scales

Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space

When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title

Title string.

subtitle

Subtitle string.

caption

Caption title string.

titles_to_case

A function to format unspecified titles_to_case. Defaults to `snakecase::to_sentence_case`.

Value

A ggplot object.

Examples

```r
library(ggplot2)
library(dplyr)
library(palmerpenguins)

penguins |>
  mutate(across(sex, \(x\) stringr::str_to_sentence(x))) |>
  group_by(species, sex) |>
```
gg_violin

```r
summarise(across(flipper_length_mm, \(x\) mean(x, na.rm = TRUE))) |>
  gg_tile(
    x = sex,
    y = species,
    col = flipper_length_mm,
  )
```

---

**gg_violin**  
*Violin ggplot*

**Description**

Create a violin plot with a wrapper around `ggplot2::ggplot()` + `geom_violin()`.

**Usage**

```r
gg_violin(  
data = NULL,
  ...,  
  stat = "ydensity",
  position = "dodge",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  alpha = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
)```
x_limits = NULL,
x_oob = scales::oob_keep,
x_position = "bottom",
x_title = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_position = "left",
y_title = NULL,
y_transform = NULL,
col_breaks = NULL,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_pal = NULL,
col_pal_na = "darkgrey",
col_rescale = scales::rescale(),
col_steps = FALSE,
col_title = NULL,
col_transform = NULL,
alpha_breaks = NULL,
alpha_expand_limits = NULL,
alpha_labels = NULL,
alpha_legend_ncol = NULL,
alpha_legend_nrow = NULL,
alpha_legend_rev = FALSE,
alpha_limits = NULL,
alpha_oob = scales::oob_keep,
alpha_pal = NULL,
alpha_pal_na = NA,
alpha_title = NULL,
alpha_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_labels = NULL,
facet_labels_position = "top",
facet_labels_switch = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facetscales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
titles_to_case = snakecase::to_sentence_case
)

Arguments

data
A data frame or tibble.

... Other arguments passed to within a params list in layer().

stat A statistical transformation to use on the data. A snakecase character string of a
ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position A position adjustment. A snakecase character string of a ggrepproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_(*()) function that outputs a ggrepproto Position subclass object (e.g. ggplot2::position_identity()).

coord A coordinate system. A coord_(*()) function that outputs a constructed ggrepproto Coord subclass object (e.g. ggplot2::coord_cartesian()).

mode A *_mode_* theme (e.g. grey_mode_b(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects, as the gg_* function will removes selected gridlines/axis-line/ticks. To avoid these side-effects, + the theme on to the output of gg_*.

x Unquoted x aesthetic variable.

xmin Unquoted xmin aesthetic variable.

xmax Unquoted xmax aesthetic variable.

xend Unquoted xend aesthetic variable.

y Unquoted y aesthetic variable.

ymin Unquoted ymin aesthetic variable.

ymax Unquoted ymax aesthetic variable.

yend Unquoted yend aesthetic variable.

z Unquoted z aesthetic variable.

col Unquoted col aesthetic variable.

alpha Unquoted alpha aesthetic variable.

facet Unquoted facet aesthetic variable.

facet2 Unquoted facet2 aesthetic variable.

group Unquoted group aesthetic variable.

subgroup Unquoted subgroup aesthetic variable.

label Unquoted label aesthetic variable.

text Unquoted text aesthetic variable.

sample Unquoted sample aesthetic variable.
mapping  Set of additional aesthetic mappings within `ggplot2::aes()` for non-supported aesthetics (e.g. shape, linetype, linewidth, or size) or for delayed evaluation.

x_breaks, y_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

x_expand, y_expand
Padding to the limits with the `ggplot2::expansion()` function, or a vector of length 2 (e.g. `c(0, 0)`).

x_expand_limits, y_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

x_labels, y_labels
A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

x_limits, y_limits
A vector of length 2 to determine the limits of the axis.

x_oob, y_oob
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

x_position, y_position
The position of the axis (i.e. "left", "right", "bottom" or "top"). If using `y_position = "top"` with a *_mode_* theme, add caption = "" or caption = "\n".

x_title, y_title
Axis title string. Use "" for no title.

x_transform, y_transform
For a numeric scale, a transformation object (e.g. `scales::transform_log10()`) or character string of this minus the transform_ prefix (e.g. "log10").

col_breaks
A `scales::breaks_*` function (e.g. `scales::breaks_pretty()`), or a vector of breaks.

col_expand_limits
For a continuous variable, any values that the limits should encompass (e.g. 0).

col_labels
A function that takes the breaks as inputs (e.g. \(x\) `stringr::str_to_sentence(x)` or `scales::label_comma()`), or a vector of labels.

col_legend_ncol, col_legend_nrow
The number of columns and rows for the legend guide.

col_legend_rev
Reverse the elements of the legend guide. Defaults to FALSE.

col_limits
A vector of length 2 to determine the limits of the axis.

col_oob
For a continuous scale variable, a `scales::oob_*` function of how to handle values outside of limits. Defaults to `scales::oob_keep`.

col_pal
Colours to use. A character vector of hex codes (or names).

col_pal_na
Colour to use for NA values. A character vector of a hex code (or name).

col_rescale
For a continuous variable, a `scales::rescale()` function.

col_steps
For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE, which colours in a gradient.
col_title  Axis title string. Use "" for no title.
col_transform For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_prefix (e.g. "log10").
alpha_breaks A scales::breaks_* function (e.g. scales::breaks_pretty()), or a vector of breaks.
alpha_expand_limits For a continuous variable, any values that the limits should encompass (e.g. 0).
alpha_labels A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x) or scales::label_comma()), or a vector of labels.
alpha_legend_ncol, alpha_legend_nrow The number of columns and rows for the legend guide.
alpha_legend_rev Reverse the elements of the legend guide. Defaults to FALSE.
alpha_limits A vector of length 2 to determine the limits of the axis.
alpha_oob For a continuous scale variable, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
alpha_pal Alpha values to use. For a continuous variable, a vector of length 2 between 0 and 1. For a discrete variable, a vector of integers between 0 and 1.
alpha_pal_na Alpha value to use for the NA value. A integer between 0 and 1.
alpha_title Axis title string. Use "" for no title.
alpha_transform For a numeric scale, a transformation object (e.g. scales::transform_log10()) or character string of this minus the transform_prefix (e.g. "log10").
facet_axes Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y".
facet_axis_labels Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_labels A function that takes the breaks as inputs (e.g. \(x\) stringr::str_to_sentence(x)), or a named vector of labels (e.g. c("value1" = "label1", ...)).
facet_labels_position When the facet layout is "wrap", the position of the facet labels. Either "top", "right", "bottom" or "left".
facet_labels_switch When the facet layout is "grid", whether to switch the facet labels to the opposite side of the plot. Either "x", "y" or "both".
facet_layout Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
facet_ncol The number of columns of facets. Only applies to a facet layout of "wrap".
facet_nrow The number of rows of facets. Only applies to a facet layout of "wrap".
facet_scales Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title

Title string.

subtitle

Subtitle string.

caption

Caption title string.

titles_to_case

A function to format unspecified titles_to_case. Defaults to snakecase::to_sentence_case.

Value

A ggplot object.

Examples

library(ggplot2)
library(dplyr)
library(palmerpenguins)

penguins |> tidyr::drop_na(sex) |> mutate(across(sex, \(x) stringr::str_to_sentence(x))) |> gg_violin(
  x = sex,
  y = body_mass_g,
  col = sex,
  facet = species,
  mode = light_mode_b(),
)

---

greyness

The grey_mode_* theme colours

Description

A vector of colours used in the grey_mode_* themes.

Usage

greyness

Format

An object of class character of length 3.
greys

Value
A character vector.

Examples
scales::show_col(greyness)

greys Grey colours

Description
A vector of grey colours derived from the US Census Bureau’s Data Visualisation Standards.

Usage
greys

Format
An object of class character of length 7.

Value
A character vector.

Examples
scales::show_col(greys)

grey_mode_b Grey mode theme with bottom legend

Description
Grey mode theme for a ggplot visualisation with bottom legend. It uses the colours from greyness.

Usage
grey_mode_b(base_size = 11, base_family = "", x_title = TRUE, y_title = TRUE)

Arguments
base_size The base size of the text. Defaults to 11.
base_family The base family of the text. Defaults to "".
x_title TRUE or FALSE whether to have a x axis title. Defaults to TRUE.
y_title TRUE or FALSE whether to have a y axis title. Defaults to TRUE.
grey_mode_n

Value

A ggplot theme.

Examples

```r
library(palmerpenguins)
library(ggplot2)

penguins |>
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = species,
    mode = grey_mode_b()
  )
```

grey_mode_n
Grey mode theme with no legend

Description

Grey mode theme for a ggplot visualisation with no legend. It uses the colours from greyness.

Usage

```r
grey_mode_n(base_size = 11, base_family = "", x_title = TRUE, y_title = TRUE)
```

Arguments

- `base_size`: The base size of the text. Defaults to 11.
- `base_family`: The base family of the text. Defaults to "".
- `x_title`: TRUE or FALSE whether to have a x axis title. Defaults to TRUE.
- `y_title`: TRUE or FALSE whether to have a y axis title. Defaults to TRUE.

Value

A ggplot theme.

Examples

```r
library(palmerpenguins)
library(ggplot2)

penguins |>
  gg_jitter(
    x = species,
    y = body_mass_g,
  )
```
Grey mode theme with right legend

Description
Grey mode theme for a ggplot visualisation with legend at right. It uses the colours from greyness.

Usage
grey_mode_r(base_size = 11, base_family = "", x_title = TRUE, y_title = TRUE)

Arguments
- base_size: The base size of the text. Defaults to 11.
- base_family: The base family of the text. Defaults to "."
- x_title: TRUE or FALSE whether to have a x axis title. Defaults to TRUE.
- y_title: TRUE or FALSE whether to have a y axis title. Defaults to TRUE.

Value
A ggplot theme.

Examples
library(palmerpenguins)
library(ggplot2)
penguins |>
gg_point(
x = flipper_length_mm,
y = body_mass_g,
col = species,
mode = grey_mode_r()
)
grey_mode_t  
*Grey mode theme with top legend*

**Description**

Grey mode theme for a ggplot visualisation with top legend. It uses the colours from `greyness`.

**Usage**

```r
grey_mode_t(base_size = 11, base_family = "", x_title = TRUE, y_title = TRUE)
```

**Arguments**

- `base_size`: The base size of the text. Defaults to 11.
- `base_family`: The base family of the text. Defaults to "".
- `x_title`: TRUE or FALSE whether to have a x axis title. Defaults to TRUE.
- `y_title`: TRUE or FALSE whether to have a y axis title. Defaults to TRUE.

**Value**

A ggplot theme.

**Examples**

```r
library(palmerpenguins)
library(ggplot2)

penguins |>
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = species,
    mode = grey_mode_t()
  )
```

lightness  
*The light_mode_* theme colours*

**Description**

A vector of colours used in the `light_mode_*` themes.

**Usage**

```r
lightness
```
Format

An object of class character of length 3.

Value

A character vector.

Examples

scales::show_col(lightness)

Description

Light mode theme for a ggplot visualisation with bottom legend. It uses the colours from lightness.

Usage

light_mode_b(base_size = 11, base_family = "", x_title = TRUE, y_title = TRUE)

Arguments

base_size The base size of the text. Defaults to 11.
base_family The base family of the text. Defaults to "."

x_title TRUE or FALSE whether to have a x axis title. Defaults to TRUE.

y_title TRUE or FALSE whether to have a y axis title. Defaults to TRUE.

Value

A ggplot theme.

Examples

library(palmerpenguins)
library(ggplot2)

penguins |>  
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = species,
    mode = light_mode_b()
  )
### light_mode_n

**Light mode theme with no legend**

**Description**

Light mode theme for a ggplot visualisation with no legend. It uses the colours from `lightness`.

**Usage**

```r
light_mode_n(base_size = 11, base_family = "", x_title = TRUE, y_title = TRUE)
```

**Arguments**

- `base_size`: The base size of the text. Defaults to 11.
- `base_family`: The base family of the text. Defaults to "".
- `x_title`: TRUE or FALSE whether to have a x axis title. Defaults to TRUE.
- `y_title`: TRUE or FALSE whether to have a y axis title. Defaults to TRUE.

**Value**

A ggplot theme.

**Examples**

```r
library(palmerpenguins)
library(ggplot2)

penguins |>
  gg_jitter(
    x = species,
    y = body_mass_g,
    col = species,
    mode = light_mode_n()
  )
```

---

### light_mode_r

**Light mode theme with right legend**

**Description**

Light mode theme for a ggplot visualisation with legend at right. It uses the colours from `lightness`.

**Usage**

```r
light_mode_r(base_size = 11, base_family = "", x_title = TRUE, y_title = TRUE)
```
**Arguments**

- `base_size` The base size of the text. Defaults to 11.
- `base_family` The base family of the text. Defaults to "".
- `x_title` TRUE or FALSE whether to have a x axis title. Defaults to TRUE.
- `y_title` TRUE or FALSE whether to have a y axis title. Defaults to TRUE.

**Value**

A ggplot theme.

**Examples**

```r
library(palmerpenguins)
library(ggplot2)

penguins |>
    gg_point(
        x = flipper_length_mm,
        y = body_mass_g,
        col = species,
        mode = light_mode_r()
    )
```

---

**Description**

Light mode theme for a ggplot visualisation with top legend. It uses the colours from lightness.

**Usage**

```r
light_mode_t(base_size = 11, base_family = "", x_title = TRUE, y_title = TRUE)
```

**Arguments**

- `base_size` The base size of the text. Defaults to 11.
- `base_family` The base family of the text. Defaults to "".
- `x_title` TRUE or FALSE whether to have a x axis title. Defaults to TRUE.
- `y_title` TRUE or FALSE whether to have a y axis title. Defaults to TRUE.

**Value**

A ggplot theme.
Examples

```r
library(palmerpenguins)
library(ggplot2)

penguins |> gg_point(
  x = flipper_length_mm,
  y = body_mass_g,
  col = species,
  mode = light_mode_t()
)
```

---

**mode_set**  
*Set the mode*

---

**Description**

Set a theme for when the mode argument in a `gg_*` function in NULL. Use `mode_set(light_mode_r())` to unset a set mode. Note `ggplot2::theme_set()` sets globally a new theme that is added to the `gg_*` function output with no side-effects. Use `ggplot2::theme_set(theme_grey())` to unset the set theme.

**Usage**

```r
mode_set(new = grey_mode_r())
```

**Arguments**

- `new`  
  A new theme to add to the mode argument where NULL (e.g. `dark_mode_r()`).

**Value**

A set theme.

---

**navy**  
*A navy colour*

---

**Description**

A navy colour derived from the US Census Bureau’s Data Visualisation Standards. Note this colour is not accessible with dark modes.

**Usage**

```r
navy
```
orange  

**Format**

An object of class character of length 1.

**Value**

A character vector.

**Examples**

scales::show_col(navy)

---

| orange | A orange colour |

**Description**

A orange colour derived from the US Census Bureau’s Data Visualisation Standards.

**Usage**

orange

**Format**

An object of class character of length 1.

**Value**

A character vector.

**Examples**

scales::show_col(orange)
plum

A plum colour

Description
A plum colour inspired by the NZ tree fuchsia (kōtukutuku)

Usage
plum

Format
An object of class character of length 1.

Value
A character vector.

Examples
scales::show_col(plum)

replace_seq

Replace a sequence of elements in a vector

Description
Keep every nth element in a vector, and replace the rest with a value such as "".

Usage
replace_seq(x, ..., keep_nth = 2, offset = 0, replacement = "")

Arguments

x A vector.

... If numeric, other arguments passed to the scales::comma function.

keep_nth The increment of elements to keep as is. Defaults to 2.

offset An offset to start at the intended offset. Defaults to 0. Possible replaces are -1 to (keep_nth - 2)

replacement The replacement value to replace non-kept elements with. Defaults to "".

Value
A vector.
Examples

```
replace_seq(seq(1000, 7000, 1000))
replace_seq(seq(1000, 7000, 1000), offset = -1)
replace_seq(seq(1000, 7000, 1000), keep_nth = 3)
replace_seq(LETTERS[1:12])
```

---

**teal**

*A teal colour*

---

**Description**

A teal colour derived from the US Census Bureau’s Data Visualisation Standards.

**Usage**

teaI

**Format**

An object of class character of length 1.

**Value**

A character vector.

**Examples**

```
scales::show_col(teal)
```
Index

* datasets
  blue, 4
  darkness, 5
greyness, 216
greys, 217
lightness, 220
navy, 224
orange, 225
plum, 226
teal, 227

aes_contrast, 3
blue, 4
dark_mode_b, 5
dark_mode_n, 6
dark_mode_r, 7
dark_mode_r(), 11, 16, 21, 27, 32, 38, 43, 49, 54, 60, 65, 70, 76, 81, 87, 92, 98, 103, 109, 114, 120, 125, 131, 136, 142, 147, 153, 158, 164, 169, 175, 180, 186, 191, 196, 202, 207, 213, 224
dark_mode_t, 8
darkness, 5

gem_area(), 8
gem_bar(), 14
gem_bar2d(), 19
gem_blank(), 25
gem_boxplot(), 30
gem_col(), 36
gem_contour(), 41
gem_contour_filled(), 46
gem_crossbar(), 52
gem_density(), 57
gem_density2d(), 63
gem_density2d_filled(), 68
gem_errorbar(), 74

geom_area(), 79
geom_function(), 85
geom_hex(), 90
geom_histogram(), 95
geom_jitter(), 101
gem_label(), 107
geom_line(), 112
geom_linerange(), 118
geom_path(), 123
geom_point(), 129
geom_pointrange(), 134
geom_polygon(), 140
geom_quantile(), 151
geom_raster(), 156
geom_rect(), 161
geom_ribbon(), 167
geom_rug(), 173
geom_segment(), 178
geom_sf(), 183
geom_smooth(), 189
geom_step(), 194
geom_text(), 200
geom_tile(), 205
geom_violin(), 211
gg_area, 8
gg_bar, 14
gg_bar2d, 19
gg_blanket, 25
gg_boxplot, 30
gg_col, 36
gg_contour, 41
gg_contour_filled, 46
gg_crossbar, 52
gg_density, 57
gg_density2d, 63
gg_density2d_filled, 68
gg_errorbar, 74
gg_freqpoly, 79
INDEX

gg_polygon, 95
gg_histogram, 101
gg_label, 107
gg_line, 112
gg_linerange, 118
gg_path, 123
gg_point, 129
gg_pointrange, 134
gg_polygon, 140
gg_qq, 145
gg_quantile, 151
gg_raster, 156
gg_rect, 161
gg_ribbon, 167
gg_rug, 173
gg_segment, 178
gg_sf, 183
gg_smooth, 189
gg_step, 194
gg_text, 200
gg_tile, 205
gg_violin, 211
ggplot2::aes, 3
ggplot2::aes(), 11, 17, 22, 27, 33, 38, 44,
49, 55, 60, 66, 71, 76, 82, 87, 93, 98,
104, 109, 115, 120, 126, 131, 137,
142, 148, 153, 159, 164, 170, 175,
181, 186, 192, 197, 203, 208, 214

ggplot2::coord_cartesian(), 11, 16, 21,
27, 32, 38, 43, 49, 54, 60, 65, 70, 76,
81, 87, 92, 98, 103, 109, 114, 120,
125, 131, 136, 142, 147, 153, 158,
164, 169, 175, 180, 186, 191, 196,
202, 207, 213

ggplot2::expansion(), 11, 17, 22, 28, 33,
38, 44, 49, 55, 60, 66, 71, 76, 82, 87,
93, 98, 104, 109, 115, 120, 126, 131,
137, 142, 148, 154, 159, 164, 170,
175, 181, 186, 192, 197, 203, 208,
214

ggplot2::ggplot(), 8, 14, 19, 25, 30, 36, 41,
46, 52, 57, 63, 68, 74, 79, 85, 90, 95,
101, 107, 112, 118, 123, 129, 134,
140, 145, 151, 156, 161, 167, 173,
178, 183, 189, 194, 200, 205, 211

ggplot2::theme_set(), 224
grey_mode_b, 217
grey_mode_b(), 11, 16, 21, 27, 32, 38, 43, 49,
54, 60, 65, 70, 76, 81, 87, 92, 98,
103, 109, 114, 120, 125, 131, 136,
142, 147, 153, 158, 164, 169, 175,
180, 186, 191, 196, 202, 207, 213
grey_mode_n, 218
grey_mode_r, 219
grey_mode_r(), 11, 16, 21, 27, 32, 38, 43, 49,
54, 60, 65, 70, 76, 81, 87, 92, 98,
103, 109, 114, 120, 125, 131, 136,
142, 147, 153, 158, 164, 169, 175,
180, 186, 191, 196, 202, 207, 213
grey_mode_t, 220
greyness, 216
greys, 217

light_mode_b, 221
grey_mode_n, 222
light_mode_r, 222
light_mode_t, 223
lightness, 220

mode_set, 224

navy, 224
orange, 225
plum, 226

replace_seq, 226

scales::breaks_pretty(), 11, 12, 17, 18,
22, 23, 28, 33, 34, 38, 39, 44, 45,
49, 50, 55, 60, 61, 66, 67, 71, 72,
76, 77, 82, 83, 87, 88, 93, 94, 98, 99,
104, 105, 109, 110, 115, 116, 120,
121, 126, 127, 131, 132, 137, 138,
142, 143, 148, 149, 154, 155, 159,
160, 164, 165, 170, 171, 175, 176,
181, 182, 186, 187, 192, 193, 197,
198, 203, 204, 208, 209, 214, 215

scales::label_comma(), 11, 12, 17, 18, 22,
23, 28, 29, 33, 34, 39, 44, 45, 49, 50,
55, 56, 60, 61, 66, 67, 71, 72, 77, 82,
83, 88, 93, 94, 98, 99, 104, 105, 110,
115, 116, 121, 126, 127, 132, 137,
138, 143, 148, 149, 154, 155, 159,
160, 164, 165, 170, 171, 176, 181,
182, 186, 187, 192, 193, 197, 198, 203, 204, 208, 209, 214, 215
teal, 227