

Package ‘ezplot’

January 30, 2019

Type Package

Title Functions for Common Chart Types

Version 0.2.2

Author Wojtek Kostecki

Maintainer Wojtek Kostecki <wojtek.kostecki@gmail.com>

Description Wrapper for the ggplot2 package that creates a variety of common charts (e.g. bar, line, area, ROC, waterfall, pie) while aiming to reduce typing.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R (>= 2.10)

Imports dplyr, forcats, ggplot2, magrittr, rlang

RoxygenNote 6.1.1

Suggests knitr, lubridate, methods, rmarkdown, ROCR, testthat, tibble, tidyr, covr

NeedsCompilation no

Repository CRAN

Date/Publication 2019-01-30 18:03:17 UTC

R topics documented:

| | |
|-------------------------|---|
| agg_data | 2 |
| area_plot | 3 |
| bar_plot | 4 |
| ez_col | 5 |
| ez_data | 6 |
| ez_jet | 7 |
| ez_labels | 7 |
| ez_log_breaks | 8 |
| ez_png | 8 |

| | |
|--------------------------|----|
| get_incr | 9 |
| line_plot | 9 |
| mean_plot | 10 |
| model_plot | 11 |
| nameifnot | 12 |
| na_plot | 12 |
| not_numeric | 13 |
| no_null | 13 |
| pie_plot | 14 |
| quick_facet | 15 |
| rel_plot | 15 |
| reorder_levels | 16 |
| roc | 16 |
| roc_plot | 17 |
| save_png | 18 |
| side_plot | 18 |
| text_contrast | 19 |
| theme_ez | 20 |
| tile_plot | 20 |
| unpack_cols | 21 |
| waterfall_plot | 22 |

| | |
|--------------|-----------|
| Index | 23 |
|--------------|-----------|

| | |
|----------|------------------------|
| agg_data | <i>Aggregates data</i> |
|----------|------------------------|

Description

Aggregates data

Usage

```
agg_data(data, cols = names(data), group_by = NULL,
         agg_fun = function(x) sum(x, na.rm = TRUE), group_by2 = NULL)
```

Arguments

| | |
|-----------|--|
| data | A data.frame. |
| cols | Named character vector of column names. |
| group_by | Vector of grouping column names. |
| agg_fun | Function to use for aggregating. |
| group_by2 | Vector of grouping column names to use for delayed (post aggregation) calculation. |

Value

An aggregated data.frame.

Examples

```
df = ez_data()
agg_data(df, c(Units = "units", Value = "value"))
agg_data(df["fct"])
agg_data(df[c("fct", "value")])
agg_data(df, "value", "fct")
agg_data(df, "value", c("fct", "year"))
agg_data(df, c(x = "year", y = "value"), c(x = "year"))
```

area_plot

area_plot

Description

Aggregates a data.frame and creates a stacked area chart.

Usage

```
area_plot(data, x, y, group = NULL, facet_x = NULL, facet_y = NULL,
  size = 14, reorder = c("group", "facet_x", "facet_y"),
  palette = ez_col, labels_y = if (position == "fill") {
    function(x) ez_labels(100 * x, append = "%") } else { ez_labels },
  labels_x = identity, use_theme = theme_ez, position = c("stack",
  "fill"), facet_scales = "fixed", facet_ncol = NULL)
```

Arguments

| | |
|-----------|--|
| data | A data.frame. |
| x | A named character value. Evaluates to a column. |
| y | A named character value. Evaluates to a column. |
| group | A character value. Evaluates to a column. |
| facet_x | A character value. Evaluates to a column. |
| facet_y | A character. Evaluates to a column. |
| size | theme size for use_theme(). Default is 14. |
| reorder | A character vector specifying the group variables to reorder. Default is c("group", "facet_x", "facet_y"). |
| palette | Colour function. |
| labels_y | label formatting function |
| labels_x | label formatting function |
| use_theme | ggplot theme function |
| position | Either "stack" (default) or "fill" |

| | |
|--------------|---|
| facet_scales | Option passed to scales argument in facet_wrap or facet_grid. Default is "fixed". |
| facet_ncol | Option passed to ncol argument in facet_wrap or facet_grid. Default is NULL. |

Value

A ggplot object.

Examples

```
df = ez_data()
area_plot(df, "year2", "units", size = 10)
area_plot(df, "year2", "units", "fct", use_theme = ggplot2::theme_bw)
area_plot(df, "year2", "units", "fct", reorder = NULL)
area_plot(df, "year2", "units", "fct", position = "fill")
area_plot(df, "year2", c("Unit Sales" = "units"), size = 12)
area_plot(df, "year2", c("Unit Sales" = "units"), "fct", "char")
area_plot(df, "year2", "units", "fct", "char", "num", position = "fill")
area_plot(df, "as.character(year)", "units")
```

bar_plot

bar_plot

Description

bar_plot

Usage

```
bar_plot(data, x, y, group = NULL, facet_x = NULL, facet_y = NULL,
  size = 14, width = NULL, reorder = c("group", "facet_x",
  "facet_y"), palette = ez_col, labels_y = if (position == "fill") {
  function(x) ez_labels(100 * x, append = "%") } else { ez_labels },
  label_pos = c("auto", "inside", "top", "both"), rescale_y = 1.1,
  label_cutoff = 0.12, use_theme = theme_ez, position = "stack",
  facet_scales = "fixed")
```

Arguments

| | |
|---------|---|
| data | A data.frame. |
| x | A named character value. Evaluates to a column. |
| y | A named character value. Evaluates to a column. |
| group | A character value. Evaluates to a column. |
| facet_x | A character value. Evaluates to a column. |
| facet_y | A character. Evaluates to a column. |

| | |
|--------------|--|
| size | theme size for use_theme(). Default is 14. |
| width | Width of bar. |
| reorder | A character vector specifying the group variables to reorder. Default is c("group", "facet_x", "facet_y"). |
| palette | Colour function. |
| labels_y | label formatting function |
| label_pos | Position of labels. Can be "auto", "inside", "top" or "both". |
| rescale_y | Rescaling factor for y-axis limits |
| label_cutoff | Cutoff size (proportion of limit range) for excluding labels |
| use_theme | ggplot theme function |
| position | Either "stack" (default) or "fill" |
| facet_scales | Option passed to scales argument in facet_wrap or facet_grid. Default is "fixed". |

Value

A ggplot object.

Examples

```
df = ez_data()
bar_plot(df, "year", "units")
bar_plot(df, "year", c("Share of Units" = "units"), "fct", position = "fill")
bar_plot(df, "year", "units", "fct", use_theme = ggplot2::theme_bw)
bar_plot(df, "year", "units", "fct", reorder = NULL, label_pos = "both")
bar_plot(df, "year", c(Units = "units"), "fct", "char")
bar_plot(df, "year", "units", "fct", "char", "num")
bar_plot(df, "year", "ifelse(fct == 'X', units, -units)", "fct", label_pos = "both")
```

 ez_col

Color palette interpolation

Description

Color palette interpolation

Usage

```
ez_col(n = 50, palette = NULL)
```

Arguments

| | |
|---------|-----------------------------|
| n | number of colours |
| palette | palette to interpolate from |

Value

rgb

Examples

```
ez_col(15)
ez_col(2, c("blue", "red"))
ez_col(3, c("blue", "red"))
```

ez_data

ez_data

Description

Creates example data using `expand.grid()` on `char`, `fct` and `num` and then adding random units and value columns.

Usage

```
ez_data(start_date = as.Date("2012-01-01"),
        end_date = as.Date("2016-12-31"), char = c("A", "B"),
        fct = factor(c("X", "Y", "Other"), c("X", "Y", "Other")), num = c(10,
        20), seed = 9)
```

Arguments

| | |
|-------------------------|-----------------------------|
| <code>start_date</code> | A start date value. |
| <code>end_date</code> | An end date value. |
| <code>char</code> | Vector of character values. |
| <code>fct</code> | Vector of factor values. |
| <code>num</code> | Vector of numeric values. |
| <code>seed</code> | A random seed. |

Value

A data.frame.

Examples

```
df = ez_data()
summary(df)
```

| | |
|--------|---------------|
| ez_jet | <i>ez_jet</i> |
|--------|---------------|

Description

ez_jet

Usage

```
ez_jet(n = 100, palette = c("dodgerblue4", "steelblue2", "olivedrab3",  
"darkgoldenrod1", "brown"))
```

Arguments

| | |
|---------|------------------------------|
| n | Number of colours to return. |
| palette | Vector of colours. |

| | |
|-----------|---|
| ez_labels | <i>Function for formatting numeric labels</i> |
|-----------|---|

Description

Function for formatting numeric labels

Usage

```
ez_labels(x, prepend = "", append = "", as_factor = FALSE,  
round = Inf, signif = Inf)
```

Arguments

| | |
|-----------|----------------------------|
| x | numeric |
| prepend | character |
| append | character |
| as_factor | logical |
| round | numeric passed to round() |
| signif | numeric passed to signif() |

Value

y

Examples

```
ez_labels(10^(0:10))
ez_labels(2000, append = " apples")
ez_labels(0:10, append = " apples", as_factor = TRUE)
ez_labels(c(0, 0.1, 0.01, 0.001, 0.0001))
```

| | |
|---------------|----------------------|
| ez_log_breaks | <i>ez_log_breaks</i> |
|---------------|----------------------|

Description

Creates break vector for a log axis

Usage

```
ez_log_breaks(limits)
```

Arguments

| | |
|--------|----------------|
| limits | Limits of axis |
|--------|----------------|

Examples

```
ez_log_breaks(limits = c(0.9, 11))
```

| | |
|--------|---------------|
| ez_png | <i>ez_png</i> |
|--------|---------------|

Description

Saves ggplot or ezplot objects to png (with useful defaults).

Usage

```
ez_png(g, file, width = 1200, height = 600, res = 72, resx = 1,
  ..., vp = NULL, dir.create = FALSE, check = TRUE)
```


Arguments

| | |
|------------|---|
| g | A ggplot or ezplot object. |
| file | A png file path. |
| width | Image width (in pixels). Default is 1200. |
| height | Image height (in pixels). Default is 600. |
| res | Resolution (PPI) of output image. Default is 72. |
| resx | Resolution multiplier. Default is 1. |
| ... | Further arguments to pass to png(). |
| vp | A viewport object created with grid::viewport. |
| dir.create | Logical. If TRUE, creates the directory to save into. Default is FALSE. |
| check | Logical. If TRUE, opens png file after saving. Default is TRUE. |

get_incr

get_incr

Description

returns the minimum increment between sorted unique values of a vector

Usage

```
get_incr(x)
```

Arguments

| | |
|---|--------------------------|
| x | A numeric or date vector |
|---|--------------------------|

line_plot

line_plot

Description

Creates line plots.

Usage

```
line_plot(data, x, y, group = NULL, facet_x = NULL, facet_y = NULL,
  yoy = FALSE, linesize = 1, size = 14, palette = ez_col,
  labels_y = ez_labels, use_theme = theme_ez, facet_scales = "fixed")
```

Arguments

| | |
|--------------|--|
| data | A data.frame. |
| x | A named character value. Evaluates to a column. |
| y | A named character value. Evaluates to a column. |
| group | A character value. Evaluates to a column. |
| facet_x | A character value. Evaluates to a column. |
| facet_y | A character. Evaluates to a column. |
| yoy | Logical used to indicate whether a YOY grouping should be created. Default is FALSE. |
| linesize | width of line for geom_line(). Default is 1. |
| size | theme size for use_theme(). Default is 14. |
| palette | Colour function. |
| labels_y | label formatting function |
| use_theme | ggplot theme function |
| facet_scales | Option passed to scales argument in facet_wrap or facet_grid. Default is "fixed". |

Value

A ggplot object.

Examples

```
df = ez_data()
line_plot(df, "week", "value", use_theme = ggplot2::theme_bw)
line_plot(df, "week", c("Sales ($) = "value"))

line_plot(df, "week", "value", "char")
line_plot(df, "week", "value", "char", "fct")
line_plot(df, "week", "value", "char", "fct", "num", facet_scales = "free_y")
line_plot(df, "year2", "~ value / units", "char", "fct", "num")
line_plot(df, "week", c("value", "units"))
line_plot(df, "week", "value", yoy = TRUE)
```

mean_plot

mean_plot

Description

Chart to compare the means across groups using a bar chart.

Usage

```
mean_plot(data, x, y, size = 14, labels = ez_labels)
```

Arguments

| | |
|--------|---|
| data | data.frame |
| x | quoted expression (required) |
| y | quoted expression (required) |
| size | base_size for ggplot2 theme (default is 20) |
| labels | function for formatting labels (default is ez_labels) |

Examples

```
library(dplyr)
mean_plot(mtcars, c("Number of Cylinders" = "factor(cyl)"), "hp > 110",
          labels = function(x) ez_labels(100 * x, append = "%"))
```

| | |
|------------|-------------------|
| model_plot | <i>model_plot</i> |
|------------|-------------------|

Description

model_plot

Usage

```
model_plot(data, x, actual, fitted, facet_x = NULL, point_size = 2,
           size = 14)
```

Arguments

| | |
|------------|---|
| data | A data.frame. |
| x | A named character value. Evaluates to a column. |
| actual | A character value. Evaluates to a column. |
| fitted | A character value. Evaluates to a column. |
| facet_x | A character value. Evaluates to a column. |
| point_size | Numeric. Default is 2. |
| size | theme size for use_theme(). Default is 14. |

Value

A ggplot object.

Examples

```

y = rnorm(26)
df = data.frame(ID = 1:26, actual = y + rnorm(26), fitted = y, id = letters)
model_plot(df, "ID", "actual", "fitted")
model_plot(df, "id", "actual", "fitted")

```

| | |
|-----------|------------------|
| nameifnot | <i>nameifnot</i> |
|-----------|------------------|

Description

Names unnamed elements of a character vector.

Usage

```
nameifnot(x, make.names = FALSE)
```

Arguments

| | |
|------------|--|
| x | A character vector. |
| make.names | Logical. Whether to force names of x to be valid variable names. Default is FALSE. |

Value

A named vector.

| | |
|---------|----------------|
| na_plot | <i>na_plot</i> |
|---------|----------------|

Description

Visual representation of the NAs in a data.frame

Usage

```
na_plot(data)
```

Arguments

| | |
|------|---------------|
| data | A data.frame. |
|------|---------------|

Value

A ggplot object.

Examples

```
na_plot(airquality)
```

| | |
|-------------|--------------------|
| not_numeric | <i>not_numeric</i> |
|-------------|--------------------|

Description

Returns names of non-numeric columns.

Usage

```
not_numeric(x)
```

Arguments

x A data.frame.

Value

A character vector.

| | |
|---------|----------------|
| no_null | <i>no_null</i> |
|---------|----------------|

Description

Converts "NULL" character to NULL.

Usage

```
no_null(x)
```

Arguments

x A character vector.

Value

y

Examples

```
no_null(NULL)
no_null("NULL")
no_null("NOPE")
```

pie_plot *pie_plot*

Description

pie_plot

Usage

```
pie_plot(data, x, y = "1", facet_x = NULL, facet_y = NULL,
  labels_y = function(x) ez_labels(x * 100, append = "%", round = round,
  signif = signif), size = 14, label_cutoff = 0.04, round = Inf,
  signif = 3, palette = ez_col, reorder = c("x", "facet_x",
  "facet_y"), label_x = 0.8)
```

Arguments

| | |
|--------------|--|
| data | A data.frame. |
| x | A named character value. Evaluates to a column. |
| y | A named character value. Evaluates to a column. |
| facet_x | A character value. Evaluates to a column. |
| facet_y | A character. Evaluates to a column. |
| labels_y | label formatting function |
| size | theme size for use_theme(). Default is 14. |
| label_cutoff | Cutoff size (proportion of limit range) for excluding labels |
| round | Option for rounding label. |
| signif | Option for retaining significant figures in label. |
| palette | Colour function. |
| reorder | A character vector specifying the group variables to reorder. Default is c("group", "facet_x", "facet_y"). |
| label_x | Position of label from centre of pie. 0 is the centre of the pie and 1 is the outer edge. |

Value

ggplot object

Examples

```
df = ez_data()
pie_plot(df, "fct", "units")
pie_plot(df, "fct", "units", reorder = NULL, label_x = 0.5)
pie_plot(df, "fct", "units", "year", reorder = NULL, label_x = 0.5)
pie_plot(df, "fct", "units", "year", "char")
pie_plot(df, "fct", "units", "year", "char", reorder = NULL)
```

| | |
|-------------|--------------------|
| quick_facet | <i>Quick facet</i> |
|-------------|--------------------|

Description

Applies faceting to ggplot objects when `g[["data"]]` has a `facet_x` or `facet_y` column.

Usage

```
quick_facet(g, ncol = NULL, ...)
```

Arguments

| | |
|-------------------|---|
| <code>g</code> | A ggplot object. |
| <code>ncol</code> | Number of facet columns. |
| <code>...</code> | Arguments to pass to <code>facet_grid</code> or <code>facet_wrap</code> . |

| | |
|----------|--------------------------|
| rel_plot | <i>relationship plot</i> |
|----------|--------------------------|

Description

Examine the relationship between two vectors from data

Usage

```
rel_plot(data, x, y, group = NULL, size = 14, point_size = 2.5)
```

Arguments

| | |
|-------------------------|--|
| <code>data</code> | A data.frame. |
| <code>x</code> | A named character value. Evaluates to a column. |
| <code>y</code> | A named character value. Evaluates to a column. |
| <code>group</code> | A character value. Evaluates to a column. |
| <code>size</code> | theme size for <code>use_theme()</code> . Default is 14. |
| <code>point_size</code> | Numeric. Default is 2. |

Examples

```
rel_plot(mtcars, "wt", "hp")
rel_plot(mtcars, "wt", "hp", "factor(cyl)")
rel_plot(mtcars, "factor(cyl)", "hp")
```

| | |
|----------------|---|
| reorder_levels | <i>Order levels of factor columns using fct_reorder</i> |
|----------------|---|

Description

Order levels of factor columns using `fct_reorder`

Usage

```
reorder_levels(data, cols = c("group", "facet_x", "facet_y"), y = "y",
  .desc = rep(TRUE, length(cols)))
```

Arguments

| | |
|--------------------|--|
| <code>data</code> | A data.frame. |
| <code>cols</code> | Names of columns to reorder. |
| <code>y</code> | Numeric column for order priority. |
| <code>.desc</code> | A logical vector of length 1 or <code>ncol(data)</code> . Default is TRUE for all columns in <code>cols</code> . |

Value

A data.frame.

Examples

```
library(magrittr)
ezplot::reorder_levels(mtcars, "cyl", "1") %>% str
ezplot::reorder_levels(mtcars, "cyl", "1", FALSE) %>% str
ezplot::reorder_levels(mtcars, "cyl", "mpg") %>% str
```

| | |
|-----|------------|
| roc | <i>roc</i> |
|-----|------------|

Description

Calculates ROC and AUC

Usage

```
roc(actual, fitted)
```

Arguments

| | |
|---------------------|------------------------------------|
| <code>actual</code> | Vector with two levels |
| <code>fitted</code> | Vector with values between 0 and 1 |

Examples

```
ezplot::roc(sample(c(TRUE, FALSE), 1, replace = TRUE), runif(1))
ezplot::roc(sample(c(TRUE, FALSE), 3, replace = TRUE), runif(3))
```

| | |
|----------|-----------------|
| roc_plot | <i>roc_plot</i> |
|----------|-----------------|

Description

roc_plot

Usage

```
roc_plot(data, actual, fitted, group = NULL, facet_x = NULL,
         facet_y = NULL, size = 14)
```

Arguments

| | |
|---------|--|
| data | A data.frame. |
| actual | Vector of actuals values |
| fitted | Vector of fitted values |
| group | A character value. Evaluates to a column. |
| facet_x | A character value. Evaluates to a column. |
| facet_y | A character. Evaluates to a column. |
| size | theme size for use_theme(). Default is 14. |

Examples

```
library(ggplot2)
n = 10000
df = data.frame(actual = sample(c(FALSE, TRUE), n, replace = TRUE),
               runif = runif(n))
df[["fitted"]] = runif(n) ^ ifelse(df[["actual"]] == 1, 0.5, 2)

ggplot(df) +
  geom_density(aes(fitted, fill = actual), alpha = 0.5)

roc_plot(df, "actual", "actual")
roc_plot(df, "actual", "fitted")
roc_plot(df, "actual", "runif")

roc_plot(df, "actual", "fitted", "sample(c(1, 2), n(), TRUE)")

roc_plot(df, "actual", "fitted",
         "sample(c(1, 2), n(), TRUE)",
         "sample(c(3, 4), n(), TRUE)")
```

```
roc_plot(df, "actual", "fitted",
         "sample(c(1, 2), n(), TRUE)",
         "sample(c(3, 4), n(), TRUE)",
         "sample(c(5, 6), n(), TRUE)")
```

save_png

save_png

Description

Saves ggplot or ezplot objects to png.

Usage

```
save_png(g, file, width, height, res, ..., vp = NULL)
```

Arguments

| | |
|--------|--|
| g | A ggplot or ezplot object. |
| file | A png file path. |
| width | Width of output image. |
| height | Height of output image. |
| res | Resolution of output image. |
| ... | Further arguments to pass to png(). |
| vp | A viewport object created with grid::viewport. |

side_plot

side_plot

Description

side_plot

Usage

```
side_plot(data, x, y, labels_y = ez_labels, size = 14,
          palette = ez_col, signif = 3, reorder = TRUE, rescale_y = 1.25)
```

Arguments

| | |
|-----------|--|
| data | A data.frame. |
| x | A named character value. Evaluates to a column. |
| y | A named character value. Evaluates to a column. |
| labels_y | label formatting function |
| size | theme size for use_theme(). Default is 14. |
| palette | Colour function. |
| signif | Number of significant digits. |
| reorder | A character vector specifying the group variables to reorder. Default is c("group", "facet_x", "facet_y"). |
| rescale_y | Rescaling factor for y-axis limits |

Examples

```
side_plot(mtcars, "gear", "1")
side_plot(mtcars, "cyl", c("Cars with <120 HP" = "hp < 120"))
side_plot(mtcars, "cyl", c(count = "ifelse(cyl == 4, 1, -1)", "hp <= 120"))
side_plot(mtcars, "cyl", c("hp <= 120", "~ - wt / cyl"))
side_plot(mtcars, "cyl", c("1", "-1"))
```

| | |
|---------------|----------------------|
| text_contrast | <i>text_contrast</i> |
|---------------|----------------------|

Description

text_contrast

Usage

```
text_contrast(x)
```

Arguments

| | |
|---|--------------------|
| x | Vector of colours. |
|---|--------------------|

Value

Vector indicating whether black or white should be used for text overlaid on x.

Examples

```
text_contrast("#000000")
text_contrast("black")
```

| | |
|----------|----------------------|
| theme_ez | <i>Default theme</i> |
|----------|----------------------|

Description

Default theme

Usage

```
theme_ez(base_size = 11, base_family = "")
```

Arguments

| | |
|-------------|------------------|
| base_size | base font size |
| base_family | base fond family |

Value

theme

Examples

```
library(ggplot2)
ggplot(mtcars) + geom_point(aes(cyl, mpg)) + theme_ez()
```

| | |
|-----------|------------------|
| tile_plot | <i>tile_plot</i> |
|-----------|------------------|

Description

Creates tile plots.

Usage

```
tile_plot(data, x, y, z, facet_x = NULL, facet_y = NULL, size = 14,
  facet_ncol = NULL, labels_x = NULL, labels_y = NULL,
  labels_z = ez_labels, palette = ez_jet, reorder = c("facet_x",
  "facet_y", "x", "y"))
```

Arguments

| | |
|------------|--|
| data | A data.frame. |
| x | A named character value. Evaluates to a column. |
| y | A named character value. Evaluates to a column. |
| z | A named character. Evaluates to a column and is mapped to the fill colour of the tiles. |
| facet_x | A character value. Evaluates to a column. |
| facet_y | A character. Evaluates to a column. |
| size | theme size for use_theme(). Default is 14. |
| facet_ncol | Option passed to ncol argument in facet_wrap or facet_grid. Default is NULL. |
| labels_x | label formatting function |
| labels_y | label formatting function |
| labels_z | label formatting function |
| palette | Colour function. |
| reorder | A character vector specifying the group variables to reorder. Default is c("group", "facet_x", "facet_y"). |

Examples

```
tile_plot(mtcars, "factor(cyl)", "factor(am)", "mpg")
tile_plot(ez_data(), "year", "char", "value", "fct", "num", reorder = NULL)
```

unpack_cols

Unpack cols argument to agg_data

Description

Unpack cols argument to agg_data

Usage

```
unpack_cols(x)
```

Arguments

x cols

Value

list

Examples

```
ezplot::unpack_cols("x")
ezplot::unpack_cols(c(x = "x", y = "x + y", expr = "~ x + y"))
```

| | |
|----------------|-----------------------|
| waterfall_plot | <i>waterfall_plot</i> |
|----------------|-----------------------|

Description

function for creating waterfall charts

Usage

```
waterfall_plot(data, x, y, group, size = 14, labels = ez_labels,
  label_rescale = 1, y_min = "auto", rescale_y = 1.1, n_signif = 3,
  rotate_xlabel = FALSE, bottom_label = TRUE, ingroup_label = FALSE,
  n_x = 2)
```

Arguments

| | |
|---------------|--|
| data | A data.frame. |
| x | A named character value. Evaluates to a column. |
| y | A named character value. Evaluates to a column. |
| group | A character value. Evaluates to a column. |
| size | theme size for use_theme(). Default is 14. |
| labels | Function for formatting labels. |
| label_rescale | Scaling factor for chart labels (relative to axis labels). |
| y_min | Minimum limit of y axis. |
| rescale_y | Scaling factor to extend y_max. |
| n_signif | Number of significant figures in labels. |
| rotate_xlabel | Logical. |
| bottom_label | Logical. |
| ingroup_label | Logical. Shows in-group percentage change. |
| n_x | Number of x levels to show in chart. |

Examples

```
df = ez_data()
waterfall_plot(df, "year", "units", "fct")
waterfall_plot(df, "year", "units", "char")
waterfall_plot(df, "year", "units", "fct", n_x = 3)
waterfall_plot(df, "year", "units", "fct",
  label_rescale = 0.5,
  ingroup_label = TRUE,
  bottom_label = FALSE,
  n_x = 3,
  size = 20,
  y_min = 0,
  rotate_xlabel = TRUE)
```

Index

agg_data, 2
area_plot, 3

bar_plot, 4

ez_col, 5
ez_data, 6
ez_jet, 7
ez_labels, 7
ez_log_breaks, 8
ez_png, 8

get_incr, 9

line_plot, 9

mean_plot, 10
model_plot, 11

na_plot, 12
nameifnot, 12
no_null, 13
not_numeric, 13

pie_plot, 14

quick_facet, 15

rel_plot, 15
reorder_levels, 16
roc, 16
roc_plot, 17

save_png, 18
side_plot, 18

text_contrast, 19
theme_ez, 20
tile_plot, 20

unpack_cols, 21

waterfall_plot, 22