

Package ‘kitesquare’

July 22, 2025

Title Visualize Contingency Tables Using Kite-Square Plots

Version 0.0.2

Description Create a kite-square plot for contingency tables using 'ggplot2', to display their relevant quantities in a single figure (marginal, conditional, expected, observed, chi-squared). The plot resembles a flying kite inside a square if the variables are independent, and deviates from this the more dependence exists.

License LGPL (>= 3)

Encoding UTF-8

URL <https://github.com/HUGLeipzig/kitesquare>

BugReports <https://github.com/HUGLeipzig/kitesquare/issues>

RoxygenNote 7.3.2

VignetteBuilder quarto

Imports tidyr, dplyr, ggplot2, ggh4x, scales, rlang

Suggests knitr, quarto

Config/Needs/website quarto

NeedsCompilation no

Author John Wiedenhöft [aut, cre, cph] (ORCID:
<<https://orcid.org/0000-0002-6935-1517>>)

Maintainer John Wiedenhöft <john.wiedenhoeft@medizin.uni-leipzig.de>

Repository CRAN

Date/Publication 2025-03-20 13:50:09 UTC

Contents

kitesquare	2
Index	5

kitesquare	<i>Create a kite-square plot</i>
------------	----------------------------------

Description

Given a data frame or tibble, creates a kite-square plot to visualize the contingency table of two discrete variables.

Usage

```
kitesquare(  
  df,  
  x,  
  y,  
  obs,  
  normalize = FALSE,  
  full_range = FALSE,  
  center_x = TRUE,  
  center_y = TRUE,  
  center = NULL,  
  fill_x = FALSE,  
  fill_y = FALSE,  
  fill = NULL,  
  kite = TRUE,  
  spars = TRUE,  
  square = TRUE,  
  chi2 = TRUE,  
  bars_x = TRUE,  
  bars_y = TRUE,  
  bars = NULL,  
  intersect_x = TRUE,  
  intersect_y = TRUE,  
  intersect = NULL,  
  color_x = "#e31a1c",  
  color_y = "#1f78b4",  
  kite_color = "black",  
  square_color = "black",  
  spars_color = "black",  
  chi2_color = "#bebebe",  
  border_color = "black",  
  alpha_fill = 0.3,  
  alpha_chi2 = 0.3,  
  alpha = NULL,  
  pointsize = 3,  
  linewidth = 1,  
  whisker_length = 0.05,  
  extend_whiskers = FALSE,
```

```
dodge_x = 2,
...
)
```

Arguments

<code>df</code>	A tibble or data frame of observations.
<code>x, y</code>	Name of the variable in <code>df</code> for x (columns) and y (rows), as string or expression.
<code>obs</code>	Name of observation counts in <code>df</code> . If a combination of y and x appears multiple times in <code>df</code> , <code>obs</code> are added together. If not provided, a value of 1 will be assumed for each line in <code>df</code> .
<code>normalize</code>	Should values be normalized to probabilities and expressed in percent?
<code>full_range</code>	If <code>normalize</code> is TRUE, should all axes limits be from 0 to 1?
<code>center_x, center_y, center</code>	Should a binary x or y variable be centered (axis reversed) so that the spars meet? <code>center</code> overrides both.
<code>fill_x, fill_y, fill</code>	Should the space between x or y bars and the axis be filled? <code>fill</code> overrides both.
<code>kite, square, spars, chi2</code>	Should the kite, square, spars and χ^2 patches be drawn?
<code>bars_x, bars_y, bars</code>	Should the bars for the x and y variables be drawn? <code>bars</code> overrides both.
<code>intersect_x, intersect_y, intersect</code>	Should the intersect positions for x and y variables with their axes be drawn? <code>intersect</code> overrides both.
<code>color_x, color_y</code>	Colors for x and y.
<code>kite_color, square_color, spars_color, chi2_color</code>	Color of the kite, square, spars and χ^2 patches.
<code>border_color</code>	Color for the border around each cell.
<code>alpha_fill, alpha_chi2, alpha</code>	Transparency for fill and χ^2 patches. <code>alpha</code> overrides both.
<code>pointsize</code>	The point size for intersects and spars.
<code>linewidth</code>	The line width for bars and spars.
<code>whisker_length</code>	The length of bar whiskers.
<code>extend_whiskers</code>	Should the bar whiskers be extended to wrap around the χ^2 patches?
<code>dodge_x</code>	The number of levels the x axis labels should dodge.
<code>...</code>	Further arguments passed to <code>ggplot2::facet_grid()</code> .

Value

A `ggplot` object, with an extra `$table` key. The latter contains the tibble of coordinates created internally for plotting.

Examples

```
df <- dplyr::tibble(  
  X=c('A', 'A', 'B', 'B', 'B'),  
  Y=c('U', 'V', 'U', 'V', 'V'),  
  count=c(30,15,30,70,65))  
kitesquare(df, X, Y, count)  
kitesquare(df, X, Y, count, normalize=TRUE, center_x=FALSE)
```

Index

kitesquare, [2](#)