

# Package ‘forplo’

October 13, 2022

**Type** Package

**Title** Flexible Forest Plots

**Version** 0.2.0

**Description** Simplifies the creation and customization of forest plots (alternatively called dot-and-whisker plots). Input classes accepted by 'forplo' are data.frame, matrix, lm, glm, and coxph. 'forplo' was written in base R and does not depend on other packages.

**License** GPL-3

**Encoding** UTF-8

**Language** en-US

**Depends** R (>= 3.5.0)

**VignetteBuilder** knitr, rmarkdown, lattice

**RoxygenNote** 7.1.2

**Suggests** knitr, rmarkdown, lattice, MASS, meta, survival, extrafont

**NeedsCompilation** no

**Author** Vincent ten Cate, PhD [cre, aut]

**Maintainer** ``Vincent ten Cate, PhD" <vincent10kd@gmail.com>

**Repository** CRAN

**Date/Publication** 2022-04-27 15:30:02 UTC

## R topics documented:

forplo . . . . .	2
<b>Index</b>	<b>7</b>

---

forplo

*forplo - flexible forest plots with R*

---

## Description

forplo is an R package meant to simplify the creation and customization of forest plots (alternatively called dot-and-whisker plots). Input classes accepted by forplo are `data.frame`, `matrix`, `lm`, `glm`, and `coxph`. forplo was written in base R and does not depend on other packages. For extensive examples and how to use all arguments for customization, please refer to the package vignette.

## Usage

```
forplo(  
  mat,  
  em = "OR",  
  row.labels = NULL,  
  linreg = FALSE,  
  prop = FALSE,  
  pval = NULL,  
  xlim = xlimits,  
  fliprow = NULL,  
  flipbelow1 = FALSE,  
  flipsymbol = "*",  
  ci.sep = "-",  
  ci.lwd = 1.5,  
  ci.edge = TRUE,  
  font = "sans",  
  groups = NULL,  
  grouplabs = NULL,  
  group.space = 1,  
  group.italics = FALSE,  
  indent.groups = NULL,  
  left.align = FALSE,  
  favorlabs = NULL,  
  add.arrow.left = FALSE,  
  add.arrow.right = FALSE,  
  arrow.left.length = 3,  
  arrow.right.length = 3,  
  arrow.vadj = 0,  
  sort = FALSE,  
  char = 20,  
  size = 1.5,  
  col = 1,  
  insig.col = "gray",  
  scaledot.by = NULL,  
  scaledot.factor = 0.75,  
  diamond = NULL,  
)
```

```

diamond.col = col,
diamond.line = TRUE,
add.columns = NULL,
add.colnames = NULL,
right.bar = FALSE,
rightbar.ticks = 0,
left.bar = TRUE,
leftbar.ticks = 0,
shade.every = NULL,
shade.col = "red",
shade.alpha = 0.05,
fill.by = NULL,
fill.colors = NULL,
fill.labs = NULL,
legend = FALSE,
legend.vadj = 0,
legend.hadj = 0,
legend.spacing = 1,
margin.left = NULL,
margin.top = 0,
margin.bottom = 2,
margin.right = 10,
horiz.bar = FALSE,
title = NULL,
save = FALSE,
save.path = NULL,
save.name = NULL,
save.type = "png",
save.width = 9,
save.height = 4.5
)

```

### Arguments

<code>mat</code>	An $n \times 3$ data.frame or matrix, or a regression model of class <code>lm</code> , <code>glm</code> or <code>coxph</code> .
<code>em</code>	Effect measure to be displayed (e.g. OR, RR, HR).
<code>row.labels</code>	Labels to display as variable names (character vector of length <code>nrow(mat)</code> ).
<code>linreg</code>	Set to <code>TRUE</code> if the estimates are from a linear regression model.
<code>prop</code>	Set to <code>TRUE</code> if the estimates are proportions.
<code>pval</code>	A numeric or character vector of same length as <code>nrow(mat)</code> , with p-values.
<code>xlim</code>	A numeric vector of length 2 indicating the limits of the x-axis.
<code>fliprow</code>	A numeric vector indicating which estimates should be inverted (only for ratios).
<code>flipbelow1</code>	Set to <code>TRUE</code> to invert all ratios below 1.
<code>flipsymbol</code>	A symbol to display besides inverted estimates. Asterisk by default.
<code>ci.sep</code>	The separator between confidence intervals. Dash by default.
<code>ci.lwd</code>	Line width for the confidence interval 'whiskers'.

<code>ci.edge</code>	Set to FALSE to remove the 90 degree edges at the end of the CI whiskers.
<code>font</code>	Controls the font family. 'Calibri' by default. Note: monospaced fonts work poorly.
<code>groups</code>	A numeric vector of length <code>nrow(mat)</code> indicating group membership of each element.
<code>grouplabs</code>	A character vector of equal length to the number of groups, with the labels of each group.
<code>group.space</code>	A single numeric value to indicate how much empty rows should be between grouped estimates.
<code>group.italics</code>	Set to TRUE to italicize the group labels.
<code>indent.groups</code>	A numeric vector indicating which groups to indent (works only when <code>left.align==TRUE</code> )
<code>left.align</code>	Set to TRUE to left align variable and group labels.
<code>favorlabs</code>	A character vector of length 2, providing labels for underneath the x-axis (e.g. <code>c('favors control','favors intervention')</code> ).
<code>add.arrow.left</code>	Adds an arrow pointing left underneath the x-axis.
<code>add.arrow.right</code>	Adds an arrow pointing right underneath the x-axis.
<code>arrow.left.length</code>	Controls the length of the arrow pointing left.
<code>arrow.right.length</code>	Controls the length of the arrow pointing right.
<code>arrow.vadj</code>	Allows to adjust the vertical placement of the arrows.
<code>sort</code>	Set to TRUE to sort the rows by effect size (not compatible with groups or diamond).
<code>char</code>	Controls the character to display for the dots. Equivalent to <code>pch</code> in the base R plot function.
<code>size</code>	Controls the size of the dots. Equivalent to <code>cex</code> in the base R plot function.
<code>col</code>	Controls the color of the dots. Equivalent to <code>col</code> in the base R plot function.
<code>insig.col</code>	Controls the color of the CI whiskers when crossing the null line. Gray by default.
<code>scaledot.by</code>	Numeric vector of length <code>nrow(mat)</code> to indicate relative importance of each variable (e.g. sample size, weight).
<code>scaledot.factor</code>	Scaling factor (scalar) for <code>scaledot.by</code> , to adapt the size of all scaled dots at once.
<code>diamond</code>	Numeric vector indicating the rows that should be displayed as diamonds (e.g. for meta-analytic estimates).
<code>diamond.col</code>	Controls the color of the diamonds.
<code>diamond.line</code>	Shows a dotted vertical line through the last diamond. Set to FALSE to disable.
<code>add.columns</code>	A data.frame of <code>nrow(mat)</code> with additional columns to add to the right of the plot.
<code>add.colnames</code>	A character vector of length <code>ncol(add.columns)</code> with column labels for these columns.

<code>right.bar</code>	Set to TRUE to show a vertical bar directly to the left of the estimates.
<code>rightbar.ticks</code>	Controls the tick marks on the right axis.
<code>left.bar</code>	Set to FALSE to remove the horizontal bar on the left axis.
<code>leftbar.ticks</code>	Controls the tick marks on the left axis.
<code>shade.every</code>	Controls row shading option. A value of 1 colors every other row, a value of 2 per blocks of 2, etc. Non-integer values also allowed.
<code>shade.col</code>	Controls the default row shading color. Default is 'red'.
<code>shade.alpha</code>	Controls the transparency of the row shading color. Default is 0.05.
<code>fill.by</code>	Numeric vector of length <code>nrow(mat)</code> indicating color group membership of each element.
<code>fill.colors</code>	Character vector of length <code>unique(fill.by)</code> , with colors for each color group.
<code>fill.labs</code>	Character vector of length <code>fill.colors</code> , specifying the legend labels.
<code>legend</code>	Set to TRUE to display a legend if <code>fill.colors</code> is not NULL.
<code>legend.vadj</code>	Controls the vertical placement of the legend.
<code>legend.hadj</code>	Controls the horizontal placement of the legend.
<code>legend.spacing</code>	Controls the spacing between legend items.
<code>margin.left</code>	Controls size of left margin.
<code>margin.top</code>	Controls size of top margin.
<code>margin.bottom</code>	Controls size of bottom margin.
<code>margin.right</code>	Controls size of right margin.
<code>horiz.bar</code>	Set to TRUE to display a horizontal bar below the plot.
<code>title</code>	Title to display above the plot. Equivalent to <code>title</code> in the base R plot function.
<code>save</code>	Set to TRUE to save the plot (also requires <code>save.name</code> and <code>save.path</code> ) in 300 dpi resolution.
<code>save.path</code>	Indicates folder where the plot should be saved.
<code>save.name</code>	Name of the plot (should not include filetype extension).
<code>save.type</code>	Filetype of the saved plot. Default is <code>.png</code> , but also supports <code>.wmf</code> on Windows.
<code>save.width</code>	Width of the saved plot in inches. Default is 9.
<code>save.height</code>	Height of the saved plot in inches. Default is 4.5.

### Value

The function plots in the user's plot window, but does not return anything.

### Examples

```
#==== Create some regression models =====
mod1 <- lm(Sepal.Length~Sepal.Width+Species+Petal.Width+Petal.Length,iris)

#==== Example forest plots=====
# default plot for linear regression model
forplo(mod1)
```

```
# customized plot for linear regression model
forplo(mod1,
  row.labels=c('Sepal width', 'Versicolor', 'Virginica', 'Petal width', 'Petal length'),
  groups=c(1,2,2,3,3),
  grouplabs=c('Sepal traits', 'Species', 'Petal traits'),
  shade.every=1,
  shade.col='gray',
  left.align=TRUE,
  xlim=c(-2,2),
  title='Linear regression with grouped estimates')

## More examples are given in the vignette.
```

# Index

forplo, [2](#)