Package: ckbplotr (via r-universe)

October 16, 2026

Title Create Plots

Description Create publication-quality plots for epidemiology.

Version 0.12.3

Encoding UTF-8

LazyData true

Depends ggplot2 (>= 3.4.0), R (>= 4.3.0)

Imports dplyr, tibble, rlang, purrr, tidyr, utils, stringi, grid, gridExtra, marquee, ragg, ggh4x, legendry, lifecycle, glue, cli

RoxygenNote 7.3.3

Roxygen list(markdown = TRUE)

Suggests gridtext (>= 0.1.5), ggtext, patchwork, rmarkdown, knitr, testthat

VignetteBuilder knitr

URL https://neilstats.github.io/ckbplotr/, https://doi.org/10.5281/zenodo.6382217

BugReports https://github.com/neilstats/ckbplotr/issues

License GPL (>= 3)

Config/testthat/edition 3

Config/pak/sysreqs libfontconfig1-dev libfreetype6-dev libfribidi-dev libharfbuzz-dev libicu-dev libjpeg-dev libpng-dev libtiff-dev libwebp-dev

Repository https://neilstats.r-universe.dev

Date/Publication 2026-01-17 13:05:25 UTC

RemoteUrl https://github.com/neilstats/ckbplotr

RemoteRef HEAD

RemoteSha 81b639858e56f75cc9cd42896b900dcffe7895f8

Contents

ckb_style 2
forest_plot 3
geom_text_move 9
ggpreview 11
plot_like_ckb 12
prepare_figure 12
save_figure 14
shape_plot 14
theme_ckb 18

Index **19**

`ckb_style` *Make a ggplot into CKB style*

Description

Make a ggplot into CKB style

Usage

```
ckb_style(
  xlims = NULL,
  ylims = NULL,
  gap = c(0.025, 0.025),
  ext = c(0, 0),
  ratio = 1.5,
  width = NULL,
  height = NULL,
  base_size = 11,
  base_line_size = base_size/22,
  ink = "black",
  paper = "white",
  colour = NULL,
  axis.title.margin = 1,
  plot.margin = margin(0.5, 1.5, 0.5, 0.5, "lines"),
  clip = "on"
)
```

Arguments

`xlims`	A numeric vector of length two. The limits of the x-axis.
`ylims`	A numeric vector of length two. The limits of the y-axis.
`gap`	A numeric vector of length two. The gap between plotting area and axis to the left and bottom of the plot, as a proportion of the x-axis length. (Default: c(0.025, 0.025))

ext	A numeric vector of length two. The extensions to add to the right and top of the plot, as a proportion of the x-axis length. (Default: c(0, 0))
ratio	The ratio (y-axis:x-axis) to use for the plot. Ignored if both width and height are set. (Default: 1.5)
width	A grid::unit object to set the width of the plot (not including the gap or extension).
height	A grid::unit object to set the height of the plot (not including the gap or extension).
base_size	Base font size, given in pts. (Default: 11)
base_line_size	Base size for line elements. (Deault: base_size/22)
ink, paper	Colour for foreground and background elements. (Defaults: "black" and "white")
colour	Deprecated. Use ink instead.
axis.title.margin	Margin between axis titles and plot. (Default: 1)
plot.margin	Margin around entire plot (Default: margin(0.5, 0, 0.5, 0, "lines"))
clip	Passed to clip argument of ggplot2::coord_cartesian(). (Default: "on")

---

forest_plot *Make a forest plot with ggplot2*

Description

Creates a forest plot with ggplot

Usage

```
forest_plot(
  panels,
  row.labels = NULL,
  row.labels.levels = NULL,
  rows = NULL,
  row.labels.heading = NULL,
  row.labels.space = c(0, 1, 0, 0),
  exponentiate = TRUE,
  logscale = exponentiate,
  panel.names = NULL,
  panel.headings = NULL,
  panel.headings.align = c("panel", "plot"),
  col.key = "key",
  col.estimate = c("estimate", "est", "beta", "loghr"),
  col.stderr = c("stderr", "std.error", "std.err", "se"),
  col.lci = NULL,
  col.uci = NULL,
  col.left = NULL,
```

```
col.right = NULL,
left.parse = FALSE,
right.parse = FALSE,
left.heading = "",
right.heading = as.list(xlab),
left.pos = NULL,
right.pos = NULL,
left.hjust = 1,
right.hjust = 0,
left.gap = c("I", "W"),
right.gap = c("I", "W"),
heading.space = 0,
heading.rule = FALSE,
estcolumn = TRUE,
col.keep = NULL,
ci.delim = ", ",
digits = 2,
title = "",
xlab = "HR (95% CI)",
xlim = NULL,
xticks = NULL,
nullval = NULL,
col.diamond = NULL,
diamond = NULL,
col.bold = NULL,
bold.labels = NULL,
scalepoints = FALSE,
minse = NULL,
pointsize = 3 * base_size/11,
shape = 15,
plotcolour = "black",
colour = plotcolour,
cicolour = colour,
fill = colour,
ciunder = NULL,
addtext = NULL,
bottom.space = 0.7,
left.space = NULL,
right.space = NULL,
mid.space = unit(5, "mm"),
plot.margin = margin(2, 8, 2, 8, "mm"),
panel.width = NULL,
panel.height = NULL,
base_size = 11,
base_line_size = base_size/22,
stroke = 0,
diamonds.linewidth = base_line_size,
row.labels.element = "marquee::element_marquee",
```

```
  quiet = FALSE,
  printplot = !quiet,
  showcode = !quiet,
  data.function = NULL,
  addaes = NULL,
  addarg = NULL,
  add = NULL,
  envir = NULL,
  blankrows = NULL,
  col.right.parse = NULL,
  col.left.heading = NULL,
  col.right.heading = NULL,
  col.left.pos = NULL,
  col.right.pos = NULL,
  col.left.hjust = NULL,
  col.right.hjust = NULL,
  col.left.gap = NULL,
  col.right.gap = NULL,
  col.heading.space = NULL,
  col.heading.rule = NULL
)
```

Arguments

panels	A list of data frames. These should include columns or point estimates, and standard errors or confidence interval limits. If you specify a row.labels data frame, then they must also all contain a key column with the same name (which can be specified by col.key).
row.labels	A data frame that contains the labels to be used for the rows of the plot. Use NA if a lower level heading is not required for a given row.
row.labels.levels	A character vector. The names of columns in row.labels to use as headings/subheadings/labels for labelling rows.
rows	If set, then only rows matching these labels (at the first level) will be included.
row.labels.heading	Title to be placed above row labels.
row.labels.space	A numeric vector specifying the space after a row label heading, at the end of a row label heading 'section'. (Default: c(0, 1, 0, 0))
exponentiate	Exponentiate estimates (and CIs) before plotting. (Default: TRUE)
logscale	Use log scale on the axis, and add a line at null effect. (Default: exponentiate)
panel.names	A character vector. The names to be used for each forest plot panel. If none provided, then they will be numbered 1, 2, 3 ...
panel.headings	Titles to be placed above each forest plot.
panel.headings.align	Panel headings are by default centred over the plotting area ("panel"). Set to "plot" to centre over plotting area and text columns.

`col.key` Name of column that links the results given in each data frame provided in panels and the labels given in row.labels. If row.labels data frame is not given, then this column will be used as row labels. (Default: "key")

`col.estimate`, `col.stderr`, `col.lci`, `col.uci`
Names of columns for: point estimates, standard errors, lower and upper limits of confidence intervals.

`col.left`, `col.right`
Names of columns to be printed to the left/right of the plot.

`left.parse`, `right.parse`
A character vector, the same length as col.left/col.right (+ 1 if estcolumn = TRUE). Should the contents of the columns be parsed into expressions. Use "col", "heading" or "both" to parse the column content only, heading or both.

`left.heading`, `right.heading`
Headings for columns.

`left.pos`, `right.pos`
A unit vector to position col.left/col.right columns.

`left.hjust`, `right.hjust`
A numeric vector. The horizontal justification of col.left/col.right columns. (Default: 1)

`left.gap`, `right.gap`
A character vector of length two. The two characters control the gaps between the first text column and the panel, and successive text columns. (Default: c("I", "W"))

`heading.space` Position of the titles given by left.heading and right.heading. Increase to move them up. (Default: 0)

`heading.rule` Include a horizontal rule below column headings? (Default: FALSE)

`estcolumn` Include column of estimates and confidence intervals to the right of each plot. (Default: TRUE)

`col.keep` Names of additional columns to be kept in returned data frame.

`ci.delim` Character string to separate lower and upper limits of confidence interval. (Default: ", ")

`digits` Number of digits after decimal point to show for estimates and confidence intervals. (Default: 2)

`title` Title to appear at the top of the plot.

`xlab` Label to appear below the x-axis. (Default: "HR (95% CI)")

`xlim` A numeric vector. The limits of the x axis.

`xticks` A numeric vector. The tick points of the x axis.

`nullval` Add a vertical reference line at this value. (If logscale == TRUE then by default it will be added at 1, but use NA not to plot this line.)

`col.diamond` Plot estimates and CIs as diamonds. Name of a column of logical values.

`diamond` Alternative to col.diamond. A character vectors identify the rows (using the key values) for which the estimate and CI should be plotted using a diamond.

`col.bold` Plot text as bold. Name of a column of logical values.

`bold.labels` A character vector identifying row labels (using key values) which should additionally be bold. (Default: NULL)

`scalepoints` Should the points be scaled by inverse of the standard error? (Default: FALSE)

`minse` Minimum standard error to use when scaling point size. (Default will use minimum in the data.)

`pointsize` The (largest) size of box to use for plotting point estimates. (Default: 3 * base_size / 11)

`shape` Shape of points. An integer, or name of a column of integers. (Default: 15 (square))

`plotcolour` Colour for all parts of the plot. (Default: "black")

`colour` Colour of points. Name of a colour, or name of a column of colour names. Use a list of colour names for different colours on each panel. (Default will use plotcolour.)

`cicolour` Colour of CI lines. Name of a colour, or name of a column of colour names. Use a list of colour names for different colours on each panel. Use a vector (or list of vectors) in conjunction with panel.width to set a different colour for narrow confidence interval lines. (Default will use colour.)

`fill` Fill colour of points. Name of a colour, or name of a column of colour names. Use a list of colour names for different colours on each panel. (Default will use colour.)

`ciunder` Plot CI lines before points. A logical value, or name of a column of logical values. (Default will plot CI lines after points.)

`addtext` A list of data frames. List must be the same length as panels. Data frames should contain a column with the name specified in col.key, and one or more of:

1. a column named 'text' containing character strings
2. columns named 'het_dof', 'het_stat', and 'het_p' containing character strings
3. columns names 'trend_stat' and 'trend_p' containing character strings

The character strings, heterogeneity test, and trend test results will be plotted in the column of estimates and CIs, below the row with the key given in the col.key column.

`bottom.space` Space between bottom row and axis. (Default: 0.7)

`left.space`, `right.space`, `mid.space`
Space to the left/right/between panels. (Default mid.space: unit(5, "mm"))

`plot.margin` Plot margin, given as margin(top, right, bottom, left, units). (Default: margin(2, 8, 2, 8, "mm"))

`panel.width`, `panel.height`
Set width/height of panels. A grid::unit object, if a numeric is given assumed to be in mm. If panel.width is used, will also apply different formatting to narrow CIs.

`base_size` base font size, given in pts.

`base_line_size` base size for line elements

`stroke` Size of outline of shapes. (Default: 0)

`diamonds.linewidth`
: Line width for diamonds. (Default: base_line_size)

`row.labels.element`
: A character string giving the name of the theme element to use for the row labels (axis.text.y). (Default: "marquee::element_marquee")

`quiet`
: Set to TRUE to not print the plot nor show generated code in the RStudio 'Viewer' pane. (Default: FALSE)

`printplot`
: Print the plot. (Default: !quiet)

`showcode`
: Show the ggplot2 code to generate the plot in RStudio 'Viewer' pane. (Default: !quiet)

`data.function`
: Name of a function to apply to data frame before plotting.

`addaes`, `addarg`, `add`
: Methods for customising the plot. See documentation for details.

`envir`
: Environment in which to evaluate the plot code. May be useful when calling this function inside another function.

`blankrows`
: DEPRECATED

`col.right.parse`
: Deprecated. Use `right.parse` instead.

`col.left.heading`
: Deprecated. Use `left.heading` instead.

`col.right.heading`
: Deprecated. Use `right.heading` instead.

`col.left.pos`
: Deprecated. Use `left.pos` instead.

`col.right.pos`
: Deprecated. Use `right.pos` instead.

`col.left.hjust`
: Deprecated. Use `left.hjust` instead.

`col.right.hjust`
: Deprecated. Use `right.hjust` instead.

`col.left.gap`
: Deprecated. Use `left.gap` instead.

`col.right.gap`
: Deprecated. Use `right.gap` instead.

`col.heading.space`
: Deprecated. Use `heading.space` instead.

`col.heading.rule`
: Deprecated. Use `heading.rule` instead.

Details

The function returns the plot and ggplot2 code to create the plot. In RStudio, the ggplot2 code will be shown in the viewer.

Value

A list:

plot the plot

code ggplot2 code to generate the plot

Notes

Saving forest plots and row labels:

From version 0.12.0, the row labels in forest plots are drawn using `marquee::element_marquee` to allow for styling using markdown. When saving it is best to use a graphics device that supports 'glyph' features (see https://marquee.r-lib.org/reference/marquee_grob.html#rendering). For example `pdf()` or the devices supplied by the ragg package, which are used as default by `ggsave()` and `save_figure()`.

To return to the old method which used the ggtext package, set `row.labels.element = "ggtext::element_markdown"`.

Spacing:

The function attempts to set the positions of columns of text and spacing automatically. Where this does not produce a satisfactory layout, you can use the arguments `left.pos`, `right.pos`, `left.space`, `right.space`, and `mid.space` to control positions and spacing manually.

The plot will fill the vertical space available. Use `plot.margin` to change the top and bottom margins as needed.

Confidence intervals:

When standard errors are supplied to the `shape_plot()` and `forest_plot()` functions, confidence intervals are calculated as 95\ using the Normal approximation method (with critical value 1.96).

Stroke:

The `stroke` argument sets the stroke aesthetic for plotted shapes. See https://ggplot2.tidyverse.org/articles/ggplot2-specs.html for more details. The stroke size adds to the total size of a shape, so unless `stroke = 0` the scaling of size by inverse variance will be very slightly inaccurate (but there are probably more important things to worry about).

`geom_text_move` *Text that can be moved*

Description

This geom adds a fixed horizontal and/or vertical move to ggplot2::geom_text()

Usage

```
geom_text_move(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "nudge",
  ...,
  parse = FALSE,
  move_x = unit(0, "pt"),
  move_y = unit(0, "pt"),
  check_overlap = FALSE,
  na.rm = FALSE,
```

```
  show.legend = NA,
  inherit.aes = TRUE
)
```

Arguments

mapping	Set of aesthetic mappings created by aes(). If specified and inherit.aes = TRUE (the default), it is combined with the default mapping at the top level of the plot. You must supply mapping if there is no plot mapping.
data	The data to be displayed in this layer. There are three options: If NULL, the default, the data is inherited from the plot data as specified in the call to ggplot(). A data.frame, or other object, will override the plot data. All objects will be fortified to produce a data frame. See fortify() for which variables will be created. A function will be called with a single argument, the plot data. The return value must be a data.frame, and will be used as the layer data. A function can be created from a formula (e.g. ~ head(.x, 10)).
stat	The statistical transformation to use on the data for this layer. When using a geom_*() function to construct a layer, the stat argument can be used to override the default coupling between geoms and stats. The stat argument accepts the following: • A Stat ggproto subclass, for example StatCount. • A string naming the stat. To give the stat as a string, strip the function name of the stat_ prefix. For example, to use stat_count(), give the stat as "count". • For more information and other ways to specify the stat, see the layer stat documentation.
position	A position adjustment to use on the data for this layer. This can be used in various ways, including to prevent overplotting and improving the display. The position argument accepts the following: • The result of calling a position function, such as position_jitter(). This method allows for passing extra arguments to the position. • A string naming the position adjustment. To give the position as a string, strip the function name of the position_ prefix. For example, to use position_jitter(), give the position as "jitter". • For more information and other ways to specify the position, see the layer position documentation.
...	Other arguments passed on to layer()'s params argument. These arguments broadly fall into one of 4 categories below. Notably, further arguments to the position argument, or aesthetics that are required can *not* be passed through .... Unknown arguments that are not part of the 4 categories below are ignored. • Static aesthetics that are not mapped to a scale, but are at a fixed value and apply to the layer as a whole. For example, colour = "red" or linewidth = 3. The geom's documentation has an **Aesthetics** section that lists the available options. The 'required' aesthetics cannot be passed on to the

params. Please note that while passing unmapped aesthetics as vectors is technically possible, the order and required length is not guaranteed to be parallel to the input data.

- When constructing a layer using a stat_*() function, the ... argument can be used to pass on parameters to the geom part of the layer. An example of this is stat_density(geom = "area", outline.type = "both"). The geom's documentation lists which parameters it can accept.
- Inversely, when constructing a layer using a geom_*() function, the ... argument can be used to pass on parameters to the stat part of the layer. An example of this is geom_area(stat = "density", adjust = 0.5). The stat's documentation lists which parameters it can accept.
- The key_glyph argument of layer() may also be passed on through .... This can be one of the functions described as key glyphs, to change the display of the layer in the legend.

parse	If TRUE, the labels will be parsed into expressions and displayed as described in ?plotmath.
move_x	Unit value to move text horizontally (Default: unit(0, "pt"))
move_y	Unit value to move text vertically (Default: unit(0, "pt"))
check_overlap	If TRUE, text that overlaps previous text in the same layer will not be plotted. check_overlap happens at draw time and in the order of the data. Therefore data should be arranged by the label column before calling geom_text(). Note that this argument is not supported by geom_label().
na.rm	If FALSE, the default, missing values are removed with a warning. If TRUE, missing values are silently removed.
show.legend	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes. It can also be a named logical vector to finely select the aesthetics to display. To include legend keys for all levels, even when no data exists, use TRUE. If NA, all levels are shown in legend, but unobserved levels are omitted.
inherit.aes	If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. annotation_borders().

Aesthetics

geom_text_move() understands the same aesthetics as ggplot2::geom_text()

---

ggpreview *Create a plot preview and display it in the Viewer pane.*

Description

This function saves a ggplot2 plot to a temporary PNG file and then embeds it in an HTML page, which is opened in the Viewer pane.

Usage

```
ggpreview(...)
```

Arguments

...	Arguments passed to ggsave and the PNG device function.

Device

The plot is saved using ggsave with the png device, regardless of what is specified in the call, so any arguments not used by ggsave or png are ignored.

---

plot_like_ckb *Make a ggplot into CKB style*

Description

Make a ggplot into CKB style

Usage

```
plot_like_ckb(plot, ...)
```

Arguments

plot	A ggplot2 plot
...	Arguments passed to ckb_style()

Value

A ggplot2 plot.

---

prepare_figure *Prepare figure for saving*

Description

Prepare figure for saving

Usage

```
prepare_figure(
  figure,
  title = "",
 title.pos = grid::unit.c(unit(1.27/2, "cm"), unit(1, "npc") - unit(1.27/2, "cm")),
  title.just = c(0, 1),
  title.gpar = list(fontsize = 12, fontface = "bold"),
  footer = "",
  footer.pos = grid::unit.c(unit(1.27/2, "cm"), unit(1.27/3, "cm")),
  footer.just = c(0, 0),
  footer.gpar = list(fontsize = 9),
  margin = unit(c(2.27, 1.27, 1.27, 1.27), units = "cm"),
  size = NULL,
  valign = 0.5,
  halign = 0.5,
  pagesize = c("A4", "A5"),
  landscape = FALSE,
  pagedim = NULL
)
```

Arguments

figure	Plot (or graphical object).
title	Title to be added to the page. (Default: "")
title.pos	Position of the title text. Default is 1/4 inch from top left of page. (Default: unit.c(unit(1.27/2, "cm"), unit(1, "npc") - unit(1.27/2, "cm")))
title.just	Justification of the title text. (Default: c(0, 1))
title.gpar	Graphical parameters for title. (Default: list(fontsize = 12, fontface = "bold"))
footer	Footer to be added to the page. (Default: "")
footer.pos	Position of the footer text. Default is 1/6 inch from bottom and 1/4 inch from left of page. (Default: unit.c(unit(1.27/2, "cm"), unit(1.27/3, "cm")))
footer.just	Justification of the footer text. (Default: c(0, 0))
footer.gpar	Graphical parameters for footer. (Default: list(fontsize = 9))
margin	Margin to be placed around the plot. Default is 2.27cm top, 1.27cm (1/2 inch) other sides. (Default: unit(c(2.27, 1.27, 1.27, 1.27), units = "cm"))
size	A unit vector of length two (width, height). Size of plot (a width/height larger than page weight/height minus margins will be ignored), centred within margins. By default, plot will fill the space within margins.
valign	If size is set, where to place figure within margins. 1 = top, 0.5 = middle, 0 = bottom. (Default: 0.5)
halign	If size is set, where to place figure within margins. 1 = right, 0.5 = middle, 0 = left (Default: 0.5)
pagesize	Page size of output: "A4" or "A5". (Default: "A4")
landscape	Landscape page orientation? (Default: False)

pagedim	Dimensions (width, height) of output. Overrides pagesize and landscape arguments if used.

`save_figure` *Output plots as files*

Description

Output plots as files

Usage

```
save_figure(
  figure,
  filename,
  cropped = NULL,
  args = NULL,
  args_cropped = NULL,
  preview = FALSE,
  ...
)
```

Arguments

figure	Plot (or graphical object).
filename	Name of file to create.
cropped	Name of second output file of the figure without margins or title.
args	List of arguments passed to `ggplot2::ggsave()` for the main figure.
args_cropped	List of arguments passed to `ggplot2::ggsave()` for the cropped figure.
preview	Preview the output in the RStudio Viewer pane. (Default: False)
...	Other arguments passed to prepare_figure.

`shape_plot` *Make a shape plot with ggplot2*

Description

Make a shape plot with ggplot2

Usage

```
shape_plot(
  data,
  col.x = "x",
  col.estimate = c("estimate", "est", "beta", "loghr"),
  col.stderr = c("stderr", "std.error", "std.err", "se"),
  col.lci = NULL,
  col.uci = NULL,
  col.n = NULL,
  exponentiate = FALSE,
  logscale = exponentiate,
  xlogscale = FALSE,
  scalepoints = FALSE,
  digits = 2,
  minse = NA,
  pointsize = 3 * base_size/11,
  col.group = NULL,
  shape = 15,
  plotcolour = "black",
  colour = plotcolour,
  cicolour = colour,
  fill = colour,
  ciunder = NULL,
  lines = c("none", "lmw", "lm", "connect"),
  xlims = NULL,
  ylims = NULL,
  height = NULL,
  width = NULL,
  gap = c(0.025, 0.025),
  ext = c(0.025, 0.025),
  ratio = 1.5,
  plot.margin = margin(0.5, 1.5, 0.5, 0.5, "lines"),
  clip = "off",
  base_size = 11,
  base_line_size = base_size/22,
  stroke = base_size/22,
  axis.title.margin = 1,
  xbreaks = NULL,
  ybreaks = NULL,
  xlab = "Risk factor",
  ylab = "Estimate (95% CI)",
  legend.name = "",
  legend.position = "top",
  title = NULL,
  quiet = FALSE,
  printplot = !quiet,
  showcode = !quiet,
  addaes = NULL,
```

```
  addarg = NULL,
  add = NULL,
  envir = NULL
)
```

Arguments

data	The data frame containing estimates to be plotted.
col.x	Name of column that provides the x-axis value (e.g. exposure, risk factor, dependent variable). (Default: "x")
col.estimate	Name of column that provides point estimates. (Default: "estimate")
col.stderr	Name of column that provides standard errors. (Default: "stderr")
col.lci	Name of column that provides lower limit of confidence intervals.
col.uci	Name of column that provides upper limit of confidence intervals.
col.n	Name of column that provides number to be plotted below CIs.
exponentiate	Exponentiate estimates (and CIs) before plotting, use log scale on the axis. (Default: FALSE)
logscale	Use log scale for the y axis. (Default: exponentiate)
xlogscale	Use log scale for the x axis. (Default: FALSE)
scalepoints	Should the points be scaled by inverse of the standard error? (Default: FALSE)
digits	Number of digits to use in text of estimates.
minse	Minimum standard error to use when scaling point size. (Default will use minimum in the data.)
pointsize	The (largest) size of box to use for plotting point estimates. (Default: 3 * base_size / 11)
col.group	Name of column that groups the estimates. (Default: NULL)
shape	Shape of points. An integer, or name of a column of integers. (Default: 15)
plotcolour	Colour for non-data aspects of the plot. (Default: "black")
colour	Colour of points. Name of a colour, or name of a column of colour names. (Default will use plotcolour)
cicolour	Colour of CI lines. Colour of CI lines. Name of a colour, or name of a column of colour names. (Default will use plotcolour)
fill	Fill colour of points. Fill colour of points. Name of a colour, or name of a column of colour names. (Default will use plotcolour)
ciunder	Plot CI lines before points. A logical value, or name of a column of logical values. (Default will plot CI lines after points.)
lines	Add lines to the plot. "lmw" = Linear fit through estimates, weighted by inverse variance. "lm" = Unweighted linear fit through estimates. "connect" = Lines connecting each estimate. (Default: "none")
xlims	A numeric vector of length two. The limits of the x-axis.
ylims	A numeric vector of length two. The limits of the y-axis.

height	Panel height to use and apply different formatting to short CIs. A grid::unit() object, or if numeric is assumed to be in mm.
width	Panel width.A grid::unit() object, or if numeric is assumed to be in mm.
gap	A numeric vector of length two. The gap between plotting area and axis to the left and bottom of the plot, as a proportion of the x-axis length. (Default: c(0.025, 0.025))
ext	A numeric vector of length two. The extensions to add to the right and top of the plot, as a proportion of the x-axis length. (Default: c(0.025, 0.025))
ratio	The ratio (y-axis:x-axis) to use for the plot. (Default: 1.5)
plot.margin	Plot margin, given as margin(top, right, bottom, left, units). (Default: margin(0.5, 1.5, 0.5, 0.5, "lines"))
clip	Passed to clip argument of ggplot2::coord_cartesian(). (Default: "on")
base_size	Base font size, given in pts. (Default: 11)
base_line_size	Base size for line elements. (Deault: base_size/22)
stroke	Size of outline of shapes. (Default: base_size/22)
axis.title.margin	Margin between axis titles and plot. (Default: 1)
xbreaks	Breaks for the x axis. Passed to ggplots::scale_x_continuous. (Default: NULL)
ybreaks	Breaks for the y axis. Passed to ggplots::scale_y_continuous. (Default: NULL)
xlab	Label for x-axis. (Default: "Risk factor")
ylab	Label for y-axis. (Default: "Estimate (95% CI)")
legend.name	The name of the colour scale/legend for groups. (Default: "")
legend.position	Position of the legend for groups ("none", "left", "right", "bottom", "top", or two-element numeric vector). (Default: "top")
title	Plot title. (Default: NULL)
quiet	Set to TRUE to not print the plot nor show generated code in the RStudio 'Viewer' pane. (Default: FALSE)
printplot	Print the plot. (Default: !quiet)
showcode	Show the ggplot2 code to generate the plot in RStudio 'Viewer' pane. (Default: !quiet)
addaes, addarg, add	Methods for customising the plot. See documentation for details.
envir	Environment in which to evaluate the plot code. May be useful when calling this function inside another function.

Value

A list:

plot the plot

code ggplot2 code to generate the plot

Notes

Confidence intervals:

When standard errors are supplied to the `shape_plot()` and `forest_plot()` functions, confidence intervals are calculated as 95\ using the Normal approximation method (with critical value 1.96).

Stroke:

The `stroke` argument sets the stroke aesthetic for plotted shapes. See https://ggplot2.tidyverse.org/articles/ggplot2-specs.html for more details. The stroke size adds to total size of a shape, so unless `stroke = 0` the scaling of size by inverse variance will be slightly inaccurate (but there are probably more important things to worry about).

`theme_ckb` *CKB ggplot theme*

Description

A ggplot2 theme, based on theme_bw.

Usage

```
theme_ckb(
  base_size = 11,
  base_line_size = base_size/22,
  ink = "black",
  paper = "white",
  colour = NULL,
  axis.title.margin = 1,
  plot.margin = margin(0.5, 1.5, 0.5, 0.5, "lines"),
  ...
)
```

Arguments

`base_size`	Base font size, given in pts. (Default: 11)
`base_line_size`	Base size for line elements. (Deault: base_size/22)
`ink, paper`	Colour for foreground and background elements. (Defaults: "black" and "white")
`colour`	Deprecated. Use `ink` instead.
`axis.title.margin`	Margin between axis titles and plot. (Default: 1)
`plot.margin`	Margin around entire plot (Default: margin(0.5, 0, 0.5, 0, "lines"))
`...`	Arguments passed to `ggplot2::theme_bw()` if using ggplot2 4.0.0 or later.

Index

∗ **datasets**
    geom_text_move, 9

aes(), *10*
annotation_borders(), *11*

ckb_style, 2

forest_plot, 3
fortify(), *10*

geom_text_move, 9
GeomTextMove *(*geom_text_move*)*, 9
ggplot(), *10*
ggpreview, 11

key glyphs, *11*

layer position, *10*
layer stat, *10*
layer(), *10, 11*

plot_like_ckb, 12
prepare_figure, 12, *14*

save_figure, 14
shape_plot, 14

theme_ckb, 18