Package 'figuRes2'

October 13, 2022

Type Package

Title Support for a Variety of Figure Production Tasks

Version 1.0.0

Date 2022-08-29

URL https://github.com/gcicc/figures2

Maintainer Greg Cicconetti <greg.cicconetti@gmail.com>

Description We view a figure as a collection of graphs/tables assembled on a page and optionally annotated with metadata
(titles, headers and footers). Functions and supporting documentation are offered to streamline a variety of figure production task.

License GPL-2

Encoding UTF-8

LazyLoad no

Depends R (>= 3.5.0)

Imports survival, ggplot2, scales, stringr, plyr, grid, gridExtra, gtable, reshape2, grDevices, utils

Suggests RColorBrewer, knitr, rmarkdown, tidyverse, latex2exp

ByteCompile TRUE

VignetteBuilder knitr

RoxygenNote 7.2.1

NeedsCompilation no

Author Greg Cicconetti [aut, cre] (<https://orcid.org/0000-0002-5380-7516>), David Wade [aut]

Repository CRAN

Date/Publication 2022-09-09 08:02:55 UTC

R topics documented:

`all_in_one` *all_in_one*

Description

Produces a single pdf file with based on rows in the outputplan whose UseSubset column is equals 'Y'. A progress bar is displayed.

Usage

```
all_in_one(UseSubset = "SAC", filename = "SAC.pdf", reportNR = TRUE)
```

Arguments

`UseSubset`	Corresponds to a column name in outputplan holding flags (all_in_one)
`filename`	common_root.pdf or common_root.csv
`reportNR`	If TRUE, a plot with missing figure numbers and titles is produced

Details

Prerequisites: You need to have output, code, data directory paths defined in your workspace. These should take variable names od, cd, dd, respectively. This can be done by running a personalized set of the following commands:

Code directory needs to hold the .r files associated with the subset of figures to be produced.

Suggest running outputplan.report() first. A progress bar also helps to see run is incomplete. A manual check on the total number of pages in the final pdf should be made.

Value

This function creates a pdf file holding all figures produced based on a subset of the outputplan.

Value

A .pdf file called filename.pdf is deposited in the output directory.

Author(s)

Greg Cicconetti

annotate.page *annotate.page*

Description

Optionally adds up to 4 lines for titles, 3 lines for right and left headers, and 5 lines of footnotes

Usage

```
annotate.page(
  page.height = 8.5,
  page.width = 11,
  top.margin = 1 - 0.5,
  bottom.margin = 1 - 0.5,
  right.margin = 0.75,
  left.margin = 0.75,
  foot.size = 10,
  head.size = 10,
  title.size = 14,
  add.fignum = TRUE,
  fnote.buffer = 0,
  header.buffer = 0,
  fignum.buffer = 1,
  title.buffer = 2,
  fignum = "1.100",
 title = list("If ggplot populates title, annotate.page's title argument gets a ",
    "list of whitespace text strings. If annotate.page is populating titles,",
    "use whitespaces and newline escape characters in ggplot titles",
    "to ensure ggplot object is shrunken titles do not stamp over your graphs"),
 ulh = list("Upper Left Header 1", "Upper Left Header 2", "Upper Left Header 3"),
 urh = list("Upper Right Header 1", "Upper Right Header 2", "Upper Right Header 3"),
  fnote = list("Footnote1: Up to five lines of footnotes can be annotated.",
    "Footnote2: Graphic region height can be flexed.", "Footnote3", "Footnote4",
   "Footnote5: In large-scale production, this may hold file name, time stamp, etc."),
  override = "",
  addTime = TRUE
)
```

Arguments

page.height	used by build.page and annotate.page; presumed to be inches
page.width	used by build.page and annotate.page; presumed to be inches
top.margin	used by build.page and annotate.page; presumed to be inches
bottom.margin	used by build.page and annotate.page
right.margin	used by build.page and annotate.page; presumed to be inches
left.margin	used by build.page and annotate.page; presumed to be inches

foot.size	default: 10; passed to grid.text via gp (annotate.page)
head.size	default: 10 (anotate.page)
title.size	default: 14; passed to grid.text via gp (annotate.page)
add.fignum	logical (annotate.page)
fnote.buffer	fine-control of vertical position (annotate.page)
header.buffer	fine-control of vertical position (anotate.page)
fignum.buffer	fine-control of vertical position (annotate.page)
title.buffer	fine-control of vertical position (annotate.page)
fignum	figure number (annotate.page)
title	vector of title lines (annotate.page)
ulh	vector for upper left headers (annotate.page)
urh	vector for upper right headers (annotate.page)
fnote	vector of 5 footnotes. 5th row is traditionally reserved for filepath, table reference and time stamp. Populate from bottom up. (annotate.page)
override	override
addTime	logical for ading time stamp (annotate.page)

Value

Following an application of build.page, this function stamps on meta-data.

Author(s)

Greg Cicconetti

---

bar.plot *bar.plot*

Description

A function for creating harmonized ggplot2 bar charts

Usage

```
bar.plot(
  parent.df,
  category.col = "TRTGRP",
  category.label = "Treatment Group",
  x.label = "",
  y.col = "GWHRT",
  y.label = "Percentage of Subjects",
  y.limits = c(0, 0.7),
  y.ticks = seq(0, 0.3, 0.05),
```

```
  bar.position = "dodge",
  category.palette = c("red", "blue"),
  text.size = 3,
  text.buffer = 0.05,
  killMissing = TRUE
)
```

Arguments

parent.df	data.frame used by ggplot
category.col	data.frame column associated with categorical variable (bar.plot, box.plot, cdf.plot, dot.plot, km.plot)
category.label	passed to x-axis label
x.label	value gets passed to labs
y.col	parent.df column associated with response vairable
y.label	value gets passed to labs
y.limits	passed to scale_y_continuous
y.ticks	passed to scale_y_continuous
bar.position	passed to geom_bar (bar.plot)
category.palette	colors assoicated with categorical variable
text.size	value gets passed to geom_text
text.buffer	used by bar.plot to control text placement
killMissing	logical used by bar.plot

Value

A ggplot object is returned.

Author(s)

Greg Cicconetti

Examples

```
{
# Access dummy demography dataset
data(demog.data)
levels(demog.data$SEX) <- c("Female", "Male")

# A ggplot object is returned
p1 <- bar.plot(parent.df = demog.data, y.col = "SEX",
x.label= "Gender", y.label = "Percentage of Subjects",
category.col = "REGION", category.label = "Region",
y.limits = c(0, 0.35), y.ticks = seq(0, 0.5, 0.05),
bar.position= "dodge",
category.palette = RColorBrewer::brewer.pal(n=5, name = "Dark2"),
```

```
text.size =4, text.buffer=.025, killMissing = TRUE)
print(p1)
}
```

`benrisk2.data` *This is a dataset structured for building figures using forest.plot*

Description

This is a dataset structured for building figures using forest.plot

Author(s)

Greg Cicconetti

`box.plot` *box.plot*

Description

A function for creating harmonized ggplot2 boxplots

Usage

```
box.plot(
  parent.df,
  y.col = "AGE",
  y.label = "AGE",
  category.col = "TRTGRP",
  category.label = "Treatment Group",
  y.limits = NULL,
  y.ticks = NULL,
  y.digits = 0,
  shape.palette = c(21, 22),
  category.palette = c(2, 3),
  text.size = 4
)
```

Arguments

`parent.df`	data.frame used by ggplot
`y.col`	parent.df column associated with response vairable
`y.label`	value gets passed to labs
`category.col`	data.frame column associated with categorical variable (bar.plot, box.plot, cdf.plot, dot.plot, km.plot)

category.label	passed to x-axis label
y.limits	passed to scale_y_continuous
y.ticks	passed to scale_y_continuous
y.digits	passed to scale_y_continuous label's, fmt (box.plot, line.plot)
shape.palette	values passed to scale_shape_manual
category.palette	colors assoicated with categorical variable
text.size	value gets passed to geom_text

Value

A ggplot object is returned.

Author(s)

Greg Cicconetti

Examples

```
{
data(demog.data)
# pre-processing

levels(demog.data$SEX) <- c("Female", "Male")

 p1 <- box.plot(parent.df = demog.data,
     y.col = "BMI",
     y.label = expression(paste("BMI (m/kg",phantom()^2,")")),
     category.col = "SEX",
     category.label = "Gender",
      y.limits = c(0, 70),
      y.ticks = seq(0, 100, 10),
      y.digits = 0,
      shape.palette = c(20, 20),
      category.palette = rainbow(6),
      text.size = 4)
 print(p1)
}
```

---

boxplot.driver *This holds lines to a driver file created by the large-scale vignette*

Description

This holds lines to a driver file created by the large-scale vignette

Author(s)

Greg Cicconetti

build.page *build.page*

Description

Takes page dimensions, figure layout dimensions and an ordered list of grobs/ggplot objects orients them on a page

Usage

```
build.page(
  interior.h = c(1),
  interior.w = c(1),
  ncol = 1,
  nrow = 1,
  interior,
  test.dim = FALSE,
  page.height = 8.5,
  page.width = 11,
  right.margin = 0.75,
  left.margin = 0.75,
  top.margin = 1.4 - 0.5,
  bottom.margin = 1.75 - 0.5,
  pos = 1,
  envir = as.environment(pos)
)
```

Arguments

interior.h	a vector summing to 1 to indicate how to partition the heights (build.page)
interior.w	a vector summing to 1 to indicate how to partition the widths (build.page)
ncol	number of columns for the grid of graphics being built by build.page
nrow	number of rows for the grid of graphics being built by build.page
interior	a list of nrow*ncol grobs/ggplot objects to be displayed in the grid, ordered by row then col (build.page)
test.dim	logical. Assists with figure development. If TRUE it makes a call to grid.show.layout.
page.height	used by build.page and annotate.page; presumed to be inches
page.width	used by build.page and annotate.page; presumed to be inches
right.margin	used by build.page and annotate.page; presumed to be inches
left.margin	used by build.page and annotate.page; presumed to be inches
top.margin	used by build.page and annotate.page; presumed to be inches
bottom.margin	used by build.page and annotate.page
pos	used internally by some functions
envir	used internally by some functions

Value

This writes graphics/grobs to a device.

Author(s)

Greg Cicconetti

Examples

```
{
# Commenting out calls to pdf and dev.off.
# pdf(file = "demonstrating build.page.pdf", width = 11, height = 8.5)
build.page(test.dim= TRUE)
build.page(interior.w = c(.5, .5), ncol=2, nrow=1, test.dim= TRUE)
build.page(interior.h = c(.5, .5), ncol=1, nrow=2, test.dim= TRUE)
build.page(interior.h = c(.5, .5), interior.w = c(.5, .5), ncol=2, nrow=2, test.dim= TRUE)
build.page(interior.h=c(1/3,1/3,1/3),
           interior.w=c(1),
           ncol=1, nrow=3,
           test.dim=TRUE)
build.page(interior.h=c(2, 1, 3)/6,
           interior.w=c(.6, .4),
           ncol=2, nrow=3,
           test.dim=TRUE)
build.page(interior.h=c(1/3,1/3,1/3),
           interior.w=c(.5, .5),
           ncol=2, nrow=3,
           test.dim=TRUE,
           top.margin=.1,
           bottom.margin=.1,
           right.margin=.1,
           left.margin=.1)

parabola.up <- ggplot2::ggplot(data.frame(x=-10:10, y=(-10:10)^2), ggplot2::aes(x=x,y=y))+
ggplot2::geom_line()
parabola.down <- ggplot2::ggplot(data.frame(x=-10:10, y=-(-10:10)^2), ggplot2::aes(x=x,y=y))+
ggplot2::geom_line()
cubic.up <- ggplot2::ggplot(data.frame(x=-10:10, y=(-10:10)^3), ggplot2::aes(x=x,y=y))+
ggplot2::geom_line()
cubic.down <- ggplot2::ggplot(data.frame(x=-10:10, y=-(-10:10)^3), ggplot2::aes(x=x,y=y))+
ggplot2::geom_line()

red.parabola.up <- ggplot2::ggplot(data.frame(x=-10:10, y=(-10:10)^2), ggplot2::aes(x=x,y=y))+
ggplot2::geom_line(color="red")
red.parabola.down <- ggplot2::ggplot(data.frame(x=-10:10, y=-(-10:10)^2), ggplot2::aes(x=x,y=y))+
ggplot2::geom_line(color="red")
red.cubic.up <- ggplot2::ggplot(data.frame(x=-10:10, y=(-10:10)^3), ggplot2::aes(x=x,y=y))+
ggplot2::geom_line(color="red")
red.cubic.down <- ggplot2::ggplot(data.frame(x=-10:10, y=-(-10:10)^3), ggplot2::aes(x=x,y=y))+
ggplot2::geom_line(color="red")
```

```
build.page(interior.h = c(.5, .5), nrow=2, ncol=1,
           test.dim= FALSE, interior = list(parabola.up,
                                    parabola.down))

build.page(interior.w = c(.5, .5), nrow=1, ncol=2,
           test.dim= FALSE, interior = list(parabola.up,
                                    parabola.down))

build.page(interior.w = c(.5, .5), interior.h = c(.5, .5), nrow=2, ncol=2,
           test.dim= FALSE, interior = list(parabola.up,
                                    red.parabola.up,
                                    parabola.down,
                                    red.parabola.down
                                    ))

build.page(interior.h=c(1/3,1/3,1/3),
           interior.w=c(1),
           ncol=1, nrow=3,
           interior = list(parabola.up,
                           parabola.down,
                           cubic.up
           ))

build.page(interior.w=c(1/3,1/3,1/3),
           interior.h=c(1),
           ncol=3, nrow=1,
           interior = list(parabola.up,
                           parabola.down,
                           cubic.up
           ))

build.page(interior.h=c(2, 1, 3)/6,
           interior.w=c(.6, .4),
           ncol=2, nrow=3,
           interior = list(parabola.up,
                parabola.down,
                cubic.up,
                cubic.down,
                red.parabola.down,
                red.cubic.down)
           )

build.page(interior.h=c(1/3, 1/3, 1/3),
           interior.w=c(.5, .5),
           ncol=2, nrow=3,
           interior = list(parabola.up,
                           parabola.down,
                           cubic.up,
                           cubic.down,
                           red.parabola.down,
                           red.cubic.down)
)
```

```
build.page(interior.h=c(1/3,1/3,1/3),
           interior.w=c(.5, .5),
           ncol=2, nrow=3,
           top.margin=.1,
           bottom.margin=.1,
           right.margin=.1,
           left.margin=.1,
           interior = list(parabola.up,
                           parabola.down,
                           cubic.up,
                           cubic.down,
                           red.parabola.down,
                           red.cubic.down))


# dev.off()

}
```

`category_by_visit` *This is a dataset that would need some pre-processing ahead of using line.plot*

Description

This is a dataset that would need some pre-processing ahead of using line.plot

Author(s)

Greg Cicconetti

`cdf.data` *This is a dataset structured for building figures using cdf.plot*

Description

This is a dataset structured for building figures using cdf.plot

Author(s)

Greg Cicconetti

cdf.plot *cdf.plot*

Description

A function for creating harmonized ggplot2 cumulative distribution plots. Statistics computed by stat_ecdf().

Usage

```
cdf.plot(
  parent.df,
  category.col,
  category.label,
  response.col,
  x.label = "",
  x.limits = NULL,
  x.ticks = NULL,
  y.label = "",
  y.limits = c(0, 1),
  y.ticks = seq(0, 1, 0.2),
  line.size = 0.75,
  category.palette = c("red", "blue")
)
```

Arguments

parent.df	data.frame used by ggplot
category.col	data.frame column associated with categorical variable (bar.plot, box.plot, cdf.plot, dot.plot, km.plot)
category.label	passed to x-axis label
response.col	used by cdf.plottttt
x.label	value gets passed to labs
x.limits	value gets passed to scale_x_continuous
x.ticks	value gets passed to scale_x_continuous
y.label	value gets passed to labs
y.limits	passed to scale_y_continuous
y.ticks	passed to scale_y_continuous
line.size	value gets passed to size within geom_line, geom_step
category.palette	colors assoicated with categorical variable

Value

A ggplot object is returned.

Author(s)

Greg Cicconetti

Examples

```
{
data(demog.data)
cdf.plot(parent.df= demog.data,
  category.col = "SEX",
  category.label   = "Gender",
  response.col = "BMI",
  x.label = expression(paste("BMI (m/kg",phantom()^2,")")),
  x.limits=c(0,60),
  x.ticks=seq(0,60,5),
  y.label = "Percentage of Subjects",
  y.limits= c(0,1),
  y.ticks = seq(0,1,.2),
  line.size =.75,
  category.palette =c("red", "blue")
  )
}
```

---

check.ggplot.outliers *check.ggplot.outliers*

Description

Reports via cat statements when ggplot windows truncate data

Usage

```
check.ggplot.outliers(plot.object = NULL)
```

Arguments

plot.object	the ggplot object to check

Details

Used in conjunction with log files created with start_session_log

Author(s)

David Wade

default.settings *default.settings*

Description

Global Defaults

Usage

```
default.settings(
  pos = 1,
  envir = as.environment(pos),
  my.path = getwd(),
  main.theme = "theme_bw",
  page.width = 11,
  page.height = 8.5,
  right.margin = 0.75,
  left.margin = 0.75,
  top.margin = 1.4 - 0.5,
  bottom.margin = 1.75 - 0.5
)
```

Arguments

pos	used internally by some functions
envir	used internally by some functions
my.path	path to main directory,
main.theme	text string name of theme to be called by theme_set,
page.width	used by build.page and annotate.page; presumed to be inches
page.height	used by build.page and annotate.page; presumed to be inches
right.margin	used by build.page and annotate.page; presumed to be inches
left.margin	used by build.page and annotate.page; presumed to be inches
top.margin	used by build.page and annotate.page; presumed to be inches
bottom.margin	used by build.page and annotate.page

Details

Global Defaults

Value

This function assigns character string objects to the global environment.

Value

The following are assigned to global environment upon calling:

my.path

- dd
- cd
- od
- blankPanel
- page.width
- page.height
- right.margin
- left.margin
- top.margin
- bottom.margin
- graph.region.h
- graph.region.w

Author(s)

Greg Cicconetti

`demog.data` *This is a dataset structured for building figures using bar.plot, box.plot, and cdf.plot*

Description

This is a dataset structured for building figures using bar.plot, box.plot, and cdf.plot

Author(s)

Greg Cicconetti

`dot.plot` *dot.plot*

Description

A function for creating harmonized ggplot2 dot plots with compatiability with table.plot and forest.plot.

Usage

```
dot.plot(
  parent.df = dot.df.melt,
  category.col = "Treatment",
  y.rank.col = "rank",
  y.label.rank.col = "label.rank",
  y.label.col = "subgroup",
  Point.Est = "percent",
  x.limits = c(0, 1),
  x.ticks = seq(0, 1, 0.2),
  y.limits = NULL,
  shape.palette = c(16, 17),
  x.label = "Estimate",
  y.label = "Item",
  category.palette = c("red", "blue")
)
```

Arguments

parent.df	data.frame used by ggplot
category.col	data.frame column associated with categorical variable (bar.plot, box.plot, cdf.plot, dot.plot, km.plot)
y.rank.col	column holding ranks for line items in forest/dot/table plots
y.label.rank.col	column holding ranks for labels in forest/dot/table plots
y.label.col	column holding labels for forest/dot/table plots
Point.Est	point estimate
x.limits	value gets passed to scale_x_continuous
x.ticks	value gets passed to scale_x_continuous
y.limits	passed to scale_y_continuous
shape.palette	values passed to scale_shape_manual
x.label	value gets passed to labs
y.label	value gets passed to labs
category.palette	colors assoicated with categorical variable

Value

A ggplot object is returned.

Author(s)

Greg Cicconetti

`driver1` *This holds lines to a driver file created by the large-scale vignette*

Description

This holds lines to a driver file created by the large-scale vignette

Author(s)

Greg Cicconetti

`driver10` *This holds lines to a driver file created by the large-scale vignette*

Description

This holds lines to a driver file created by the large-scale vignette

Author(s)

Greg Cicconetti

`driver2` *This holds lines to a driver file created by the large-scale vignette*

Description

This holds lines to a driver file created by the large-scale vignette

Author(s)

Greg Cicconetti

`driver3` *This holds lines to a driver file created by the large-scale vignette*

Description

This holds lines to a driver file created by the large-scale vignette

Author(s)

Greg Cicconetti

`driver4` *This holds lines to a driver file created by the large-scale vignette*

Description

This holds lines to a driver file created by the large-scale vignette

Author(s)

Greg Cicconetti

`driver5` *This holds lines to a driver file created by the large-scale vignette*

Description

This holds lines to a driver file created by the large-scale vignette

Author(s)

Greg Cicconetti

`driver6` *This holds lines to a driver file created by the large-scale vignette*

Description

This holds lines to a driver file created by the large-scale vignette

Author(s)

Greg Cicconetti

`driver7` *This holds lines to a driver file created by the large-scale vignette*

Description

This holds lines to a driver file created by the large-scale vignette

Author(s)

Greg Cicconetti

driver8 *This holds lines to a driver file created by the large-scale vignette*

Description

This holds lines to a driver file created by the large-scale vignette

Author(s)

Greg Cicconetti

driver9 *This holds lines to a driver file created by the large-scale vignette*

Description

This holds lines to a driver file created by the large-scale vignette

Author(s)

Greg Cicconetti

facetAdjust *FacetLabelAdjuster*

Description

This function takes a 'facet wrapped' ggplot and adds axis labels when a rxc grid is incomplete

Usage

```
facetAdjust(x, pos = c("up", "down"), newpage = is.null(vp), vp = NULL)
```

Arguments

x	a ggplot object
pos	maintain default
newpage	maintain default
vp	maintain default

Value

This function returns a ggplot object.

References

<http://stackoverflow.com/questions/13297155/add-floating-axis-labels-in-facet-wrap-plot>

figuRes2 *figuRes2: A package for building and annotating mult-panel figures with application to large scale figure production*

Description

This package takes the view that a figure is a collection of graphs/tables assembled on a page and optionally annotated with metadata (titles, headers and footers). The steps to figure building can then be chunked as follows:

1. Data importation
2. Data pre-processing
3. Graph/table building (with subsequent processing necessary)
4. Assembling graph/tables on a page
5. Optional annotation to complete the figure

The figuRes2 package provides a suite of functions for producing harmonized figures using the ggplot2 packages. Additional ggplot themes are included. The package provides functions to assist with assembling multiple graphics on a page and annotating the page with headers and footnotes. Functions to facilitate data processing and mass figure production are included. Data sets are included to demonstrate how the functions work and this document contains a section that walks through the workflow for large scale figure production.

Details

All graphing functions in this package presume a data.frame is supplied with a specific data structure. In practice these can be either imported (e.g., as a .csv file) or generated with R (e.g., output of simulation or call to a probability distirbution function).

Data pre-processing of imported files may be required to ensure the data.frames are organized properly, factors are properly organized and labeled appropriatel, etc. To handle this, the user may wish to author functions to assist with this pre-processing. The demog.data data set and related process.bslchar function provide an example.

The the build.page function is designed to help visualize how graphics are organized on a page, as well as execute the task. The graphics passed to this function can be created with the functions in this package or by the user. With the former, keep in mind that these are merely functions that facilitate the construction of ggplot objects.

In the simplest case a figure will consist of a single graphic.

Some figures call for augmenting a graphic with a table (e.g., forest plots, Kaplan-Meier curves). In these cases, the tables are built using either table.plot or nsubj.plot (or again, the user coded ggplot text table). In the case of Kaplan-Meier curves, it is standard practice to arrange the KM curve on top of a table reporting the Number at Risk. Other figures call for juxtaposing two figures. In these cases, the task is either to arrange 2 graphics in a 1 (row) x 2 (col) or a 2 x 1 grid. More generally, the task is to arrange a dashboard of graphics/tables on an nrow x ncol grid and place them on page with predefined margins.

Once the individual graphs/tables have been created for a figure, pre-processing may be required. E.g., there may be a need to align the y-axes when stacking graphics: if Graph A has the longest y-axis tick label, Graph B will need to be adjusted so graphics are aligned when arranging them on a 2 x 1 grid.

When the collection of graphs/tables have been pre-processed, they can be passed to the build.page function. This function requires the user to specify how the row widths and column heights should be specified as well as the order in which to populate the cells of the grid of graphics.

The defaults presume figures are being displayed on an 8.5 inch x 11 inch page, with landscape orientation and margins of 1.5 inches at the top and bottom and 1 inch margins at the left and right. These dimensions provide sufficient room for 2 lines of headers, 4 lines of footnotes and a effective central region for graphs and tables of size (8.5 - 3) inch x (11 - 2) inch. Generalizing from the defaults is straightforward. Trial and error will be required to fine tune aesthetic aspects.

The function annotate.page has been coded to optionally populate with blank entries (helpful when building graphics that don't require annotation and where margins are minimized), dummy entries (helpful in developement phases) or entries coming from a data.frame called outputplan (helpful for mass figure production).

Author(s)

Greg Cicconetti

`fmt` *fmt*

Description

A function to control number of digits used in graphics.

Usage

```
fmt(digits = 2)
```

Arguments

`digits`	number of digits displayed

Details

This function is used within ggplot, e.g. (scale_y_continuous(labels=fmt(digits=3))) to control the number of digits presented. By default, axis labels will truncate zeros so that labels might read: 0, 2.5, 5, 7.5. Using this will result in labels: 0.0, 2.5, 5.0, 7.5.

Author(s)

Greg Cicconetti

forest.data *This is a dataset structured for building figures using forest.plot*

Description

This is a dataset structured for building figures using forest.plot

Author(s)

Greg Cicconetti

forest.plot *forest.plot*

Description

A function for creating harmonized forest.plots via ggplot2 offering compatiability with table.plot and dot.plot.

Usage

```
forest.plot(
  parent.df,
  y.rank.col = "rank",
  Point.Est = "hr",
  lower.lim = "low",
  upper.lim = "high",
  y.label.rank.col = "rank",
  y.label.col = "subcategory",
  x.label = "Estimate",
  y.label = "Item",
  log.trans = TRUE,
  x.limits = c(0.21, 5),
  x.ticks = 2^(-2:2),
  y.limits = NULL,
  category.color = "category",
  background.palette = c("red", "blue"),
  category.palette = c("red", "blue"),
  shape.palette = c(16, 16),
  flip.palette = FALSE
)
```

Arguments

parent.df	data.frame used by ggplot
y.rank.col	column holding ranks for line items in forest/dot/table plots
Point.Est	point estimate
lower.lim	column holding lower limit of CI
upper.lim	column holding upper limit of CI (forest.plot)
y.label.rank.col	column holding ranks for labels in forest/dot/table plots
y.label.col	column holding labels for forest/dot/table plots
x.label	value gets passed to labs
y.label	value gets passed to labs
log.trans	Logical; if TRUE log transformation is applied to x axis (ensure x.limits are positive!) (forest.plot)
x.limits	value gets passed to scale_x_continuous
x.ticks	value gets passed to scale_x_continuous
y.limits	passed to scale_y_continuous
category.color	data.frame column assocated with aes color mapping (forest.plot, line.plot, nsubj.plot, table.plot)
background.palette	palette gets passed to scale_fill_manual (forest.plot)
category.palette	colors assoicated with categorical variable
shape.palette	values passed to scale_shape_manual
flip.palette	logical; if TRUE it reverse the order of colors used for background (forest.plot)

Value

A ggplot object is returned.

Author(s)

Greg Cicconetti

gcurve *gcurve*

Description

A function to exploit base R's curve function. This returns a data.frame holding x and y values returned from a call to curve, but suppress the plotting of that function

Usage

```
gcurve(
  expr,
  from = NULL,
  to = NULL,
  n = 101,
  add = FALSE,
  type = "l",
  xname = "x",
  xlab = xname,
  ylab = NULL,
  log = NULL,
  xlim = NULL,
  category = NULL,
  ...
)
```

Arguments

expr	inherited from curve
from	inherited from curve
to	the range over which the function will be plotted.
n	inherited from curve
add	inherited from curve
type	inherited from curve
xname	inherited from curve
xlab	inherited from curve
ylab	inherited from curve
log	inherited from curve
xlim	inherited from curve
category	option to add a column populated with a factor (by gcurve)
...	inherited from curve

Value

A data.frame is returned. Columns include x, y, and optionally category.

Author(s)

Greg Cicconetti

See Also

graphics::curve

Examples

```
{
require(ggplot2)
curve(dnorm(x, mean=0, sd=1), from=-4, to = 4, n= 1001)
ggplot(gcurve(expr = dnorm(x, mean=0, sd=1),from=-4, to = 4, n= 1001,
category= "Standard Normal"), aes(x=x, y=y)) + geom_line()
}
```

---

get.top.xaxis *get.top.xaxis*

Description

This takes two ggplot objects, steals the bottom x-axis from 2nd object and returns a gtable object with that bottom x-axis per object 1 and top x-axis per object 2

Usage

```
get.top.xaxis(bottom.axis.version, top.axis.version)
```

Arguments

`bottom.axis.version`
: ggplot object with bottom x-axis (get.top.xaxis)

`top.axis.version`
: ggplot object with intended top x-axis in bottom position (get.top.xaxis)

Value

This function returns a ggplot object.

Author(s)

Greg Cicconetti

graphic.params *Standard graphics names*

Description

This is a dummy function whose purpose is to serve as repository for arguments used by figuRes2 functions.

Usage

```
graphic.params(
  add.fignum,
  addBars,
  addTime,
  at.risk.palette,
  background.palette,
  bar.position,
  bar.width,
  base_family,
  base_size,
  bottom.axis.version,
  bottom.margin,
  category,
  category.color,
  category.col,
  category.label,
  category.symbol.col,
  category.palette,
  cd,
  censor.col,
  centime.col,
  dd,
  envir,
  fignum,
  fignum.buffer,
  filename,
  flip.palette,
  fnote,
  fnote.buffer,
  foot.size,
  fromthetop,
  gg.list,
  head.size,
  header.buffer,
  interior,
  interior.h,
  interior.w,
```

killMissing,
left.margin,
linetype.col,
line.size,
linetype.palette,
loadplan,
logd,
log.trans,
lower.lim,
main.theme,
my.path,
ncol,
nrow,
nsubj.plot.label,
od,
outfile,
override,
page.height,
page.width,
parent.df,
pdval,
Point.Est,
pos,
reportNR,
response.col,
right.margin,
shape.label,
shape.palette,
source.code,
text.buffer,
test.dim,
text.col,
text.col1,
text.col2,
text.col3,
text.col4,
text.size,
title,
title.buffer,
title.size,
toBMP,
toEPS,
toJPEG,
top.axis.version,
top.margin,
toPDF,
toPNG,
toWMF,

```
  ulh,
  upper.lim,
  urh,
  UseSubset,
  x.col,
  x.label,
  x.limits,
  x.ticks,
  x.ticks.labels,
  y.col,
  y.digits,
  y.label,
  y.label.col,
  y.label.rank.col,
  y.limits,
  y.rank.col,
  y.ticks,
  ymax.col,
  ymin.col
)
```

Arguments

add.fignum	logical (annotate.page)
addBars	logical to add error bars (line.plot)
addTime	logical for ading time stamp (annotate.page)
at.risk.palette	colors to be assocated with categorical variable in accompanying km.plot generated at.risk table
background.palette	palette gets passed to scale_fill_manual (forest.plot)
bar.position	passed to geom_bar (bar.plot)
bar.width	used by line.plot
base_family	used in set_theme calls
base_size	used in set_theme calls
bottom.axis.version	ggplot object with bottom x-axis (get.top.xaxis)
bottom.margin	used by build.page and annotate.page
category	option to add a column populated with a factor (by gcurve)
category.color	data.frame column assocated with aes color mapping (forest.plot, line.plot, nsubj.plot, table.plot)
category.col	data.frame column associated with categorical variable (bar.plot, box.plot, cdf.plot, dot.plot, km.plot)
category.label	passed to x-axis label

`category.symbol.col`
: used by line.plot

`category.palette`
: colors assoicated with categorical variable

`cd`
: directory where driver (code) files are stored

`censor.col`
: name of parent.df column associated with censor variable

`centime.col`
: name of parent.df column associated with censored time

`dd`
: directory where data is stored

`envir`
: used internally by some functions

`fignum`
: figure number (annotate.page)

`fignum.buffer`
: fine-control of vertical position (annotate.page)

`filename`
: common_root.pdf or common_root.csv

`flip.palette`
: logical; if TRUE it reverse the order of colors used for background (forest.plot)

`fnote`
: vector of 5 footnotes. 5th row is traditionally reserved for filepath, table reference and time stamp. Populate from bottom up. (annotate.page)

`fnote.buffer`
: fine-control of vertical position (annotate.page)

`foot.size`
: default: 10; passed to grid.text via gp (annotate.page)

`fromthetop`
: logical. If TRUE KM curve decends from 1, if FALSE KM curve ascends from 0 Ensure you have an appropriate censor.col passed above!

`gg.list`
: a list of ggplot objects (sync.ylab.widths)

`head.size`
: default: 10 (anotate.page)

`header.buffer`
: fine-control of vertical position (anotate.page)

`interior`
: a list of nrow*ncol grobs/ggplot objects to be displayed in the grid, ordered by row then col (build.page)

`interior.h`
: a vector summing to 1 to indicate how to partition the heights (build.page)

`interior.w`
: a vector summing to 1 to indicate how to partition the widths (build.page)

`killMissing`
: logical used by bar.plot

`left.margin`
: used by build.page and annotate.page; presumed to be inches

`linetype.col`
: name of parent.df column associated with linetype

`line.size`
: value gets passed to size within geom_line, geom_step

`linetype.palette`
: values passed to scale_linetype_manual

`loadplan`
: logical; if TRUE then it loads from the filename

`logd`
: directory where log files are sent

`log.trans`
: Logical; if TRUE log transformation is applied to x axis (ensure x.limits are positive!) (forest.plot)

`lower.lim`
: column holding lower limit of CI

`main.theme`
: text string name of theme to be called by theme_set,

`my.path`
: path to main directory,

ncol	number of columns for the grid of graphics being built by build.page
nrow	number of rows for the grid of graphics being built by build.page
nsubj.plot.label	used in km.plot
od	directory where output files are sent
outfile	If (toPDF== TRUE & outfile == "") a .pdf file with root name taken from outputplan$outfile[which(outputplan$rcode ==source.code)]. Otherwise a .pdf will be created the value of outfile. The pdf is stored in mypath/od defined in setpaths.r.
override	override
page.height	used by build.page and annotate.page; presumed to be inches
page.width	used by build.page and annotate.page; presumed to be inches
parent.df	data.frame used by ggplot
pdval	value passed to position_dodge (lineplot)
Point.Est	point estimate
pos	used internally by some functions
reportNR	If TRUE, a plot with missing figure numbers and titles is produced
response.col	used by cdf.plottttt
right.margin	used by build.page and annotate.page; presumed to be inches
shape.label	value sets passed to labs
shape.palette	values passed to scale_shape_manual
source.code	This is intended to be a darapladib graphics driver file returning a graphic possibly with complete headers and footers.
text.buffer	used by bar.plot to control text placement
test.dim	logical. Assists with figure development. If TRUE it makes a call to grid.show.layout.
text.col	used by nsubj.plot
text.col1	name of column holding text for column 1 (table.plot)
text.col2	name of column holding text for column 2; can be NULL (table.plot)
text.col3	name of column holding text for column 3; can be NULL (table.plot)
text.col4	name of column holding text for column 4; can be NULL (table.plot)
text.size	value gets passed to geom_text
title	vector of title lines (annotate.page)
title.buffer	fine-control of vertical position (annotate.page)
title.size	default: 14; passed to grid.text via gp (annotate.page)
toBMP	Logical. If TRUE a .bmp file will be created. (run.specific)
toEPS	Logical. If TRUE a .eps file will be created. (run.specific)
toJPEG	Logical. If TRUE a .jpeg file will be created. (run.specific)
top.axis.version	ggplot object with intended top x-axis in bottom position (get.top.xaxis)

top.margin	used by build.page and annotate.page; presumed to be inches
toPDF	Logical. If TRUE a .pdf file will be created. If FALSE graphic is sent to screen. (run.specific)
toPNG	Logical. If TRUE a .png file will be created. (run.specific)
toWMF	Logical. If TRUE a .wmf file will be created. (run.specific)
ulh	vector for upper left headers (annotate.page)
upper.lim	column holding upper limit of CI (forest.plot)
urh	vector for upper right headers (annotate.page)
UseSubset	Corresponds to a column name in outputplan holding flags (all_in_one)
x.col	parent.df column associated with response vairable (line.plot, nsubj.plot)
x.label	value gets passed to labs
x.limits	value gets passed to scale_x_continuous
x.ticks	value gets passed to scale_x_continuous
x.ticks.labels	passed to scale_x_continuous
y.col	parent.df column associated with response vairable
y.digits	passed to scale_y_continuous label's, fmt (box.plot, line.plot)
y.label	value gets passed to labs
y.label.col	column holding labels for forest/dot/table plots
y.label.rank.col	column holding ranks for labels in forest/dot/table plots
y.limits	passed to scale_y_continuous
y.rank.col	column holding ranks for line items in forest/dot/table plots
y.ticks	passed to scale_y_continuous
ymax.col	name of parent.df column associated with ymax (line.plot errorbars)
ymin.col	name of parent.df column associated with ymin (line.plot errorbars)

Value

This function is just a convenient location to store argument names.

Author(s)

Greg Cicconetti

---

km.data *This is a dataset structured for building figures using km.plot*

Description

This is a dataset structured for building figures using km.plot

Author(s)

Greg Cicconetti

km.plot *km.plot*

Description

A function for creating harmonized Kaplan-Meier plots and accompanying At Risk table.

Usage

```
km.plot(
  parent.df,
  censor.col = "CENSOR",
  centime.col = "CENTIME.DAY",
  category.col = "REGION",
  category.palette = rainbow(5),
  at.risk.palette = rainbow(5),
  category.label = "Treatment Group",
  nsubj.plot.label = "Number at Risk",
  linetype.palette = 1:6,
  x.label = "Time Since Randomization",
  y.label = "Percetage of Subjects",
  x.limits = c(0, 48),
  x.ticks = seq(0, 48, 3),
  y.ticks = seq(0, 0.01, 0.005),
  y.limits = c(0, 0.01),
  line.size = 0.75,
  fromthetop = FALSE,
  text.size = 4
)
```

Arguments

parent.df	data.frame used by ggplot
censor.col	name of parent.df column associated with censor variable
centime.col	name of parent.df column associated with censored time
category.col	data.frame column associated with categorical variable (bar.plot, box.plot, cdf.plot, dot.plot, km.plot)
category.palette	colors assoicated with categorical variable
at.risk.palette	colors to be assocated with categorical variable in accompanying km.plot generated at.risk table
category.label	passed to x-axis label
nsubj.plot.label	used in km.plot

linetype.palette	values passed to scale_linetype_manual
x.label	value gets passed to labs
y.label	value gets passed to labs
x.limits	value gets passed to scale_x_continuous
x.ticks	value gets passed to scale_x_continuous
y.ticks	passed to scale_y_continuous
y.limits	passed to scale_y_continuous
line.size	value gets passed to size within geom_line, geom_step
fromthetop	logical. If TRUE KM curve decends from 1, if FALSE KM curve ascends from 0 Ensure you have an appropriate censor.col passed above!
text.size	value gets passed to geom_text

Value

A ggplot object is returned.

Author(s)

Greg Cicconetti

See Also

sync.ylab.widths, nsubj.plot

Examples

```
{
require(ggplot2); require(gridExtra)
data(km.data)
working.df <- km.data
head(working.df)
km.M <- km.plot(parent.df = subset(working.df, SEX== "M"),
      centime.col = "CENTIME.DAY",
      category.col = "TRTGRP",
      category.palette = c("red", "blue"),
      at.risk.palette = c("red","blue"),
      linetype.palette = c("solid","dotted"),
      y.limits=c(0,.01),
      y.ticks=seq(0,.01,.005),
      x.limits=c(-3,48),
      x.ticks=seq(0,48,6))
print(km.M[[1]])
print(km.M[[2]])
grid.arrange(km.M[[1]] + theme(legend.position= "bottom"), km.M[[2]], ncol=1)
comeback <- sync.ylab.widths(list(km.M[[1]]+ theme(legend.position= "bottom"), km.M[[2]]))
grid.arrange(comeback[[1]] , comeback[[2]], ncol=1)
build.page(interior.h = c(.8, .2),
      interior.w = c(1),
```

```
        ncol=1, nrow=2,
        interior = list(comeback[[1]],
            comeback[[2]]))
}
```

`line.plot` *line.plot*

Description

A function for creating harmonized line plots with optional errorbars.

Usage

```
line.plot(
  parent.df,
  category.palette = c("red", "blue"),
  linetype.palette = c("dotted", "blank", "solid", "blank"),
  line.size = 0.75,
  shape.palette = c(24, 21),
  x.label = "Visit",
  y.label = "Response",
  category.label = "Treatment Group",
  x.limits = NULL,
  x.ticks = NULL,
  x.ticks.labels = NULL,
  addBars = TRUE,
  bar.width = 1,
  pdval = 0.25,
  x.col = "XVALUES",
  y.col = "YVALUES",
  y.limits = NULL,
  y.ticks = NULL,
  category.color = "CATEGORY.COLOR",
  category.symbol.col = "CATEGORY.SYMBOL",
  y.digits = 0,
  ymin.col = "YMIN",
  ymax.col = "YMAX",
  linetype.col = "LTYPE"
)
```

Arguments

`parent.df`	data.frame used by ggplot
`category.palette`	colors assoicated with categorical variable

linetype.palette	values passed to scale_linetype_manual
line.size	value gets passed to size within geom_line, geom_step
shape.palette	values passed to scale_shape_manual
x.label	value gets passed to labs
y.label	value gets passed to labs
category.label	passed to x-axis label
x.limits	value gets passed to scale_x_continuous
x.ticks	value gets passed to scale_x_continuous
x.ticks.labels	passed to scale_x_continuous
addBars	logical to add error bars (line.plot)
bar.width	used by line.plot
pdval	value passed to position_dodge (lineplot)
x.col	parent.df column associated with response vairable (line.plot, nsubj.plot)
y.col	parent.df column associated with response vairable
y.limits	passed to scale_y_continuous
y.ticks	passed to scale_y_continuous
category.color	data.frame column assocated with aes color mapping (forest.plot, line.plot, nsubj.plot, table.plot)
category.symbol.col	used by line.plot
y.digits	passed to scale_y_continuous label's, fmt (box.plot, line.plot)
ymin.col	name of parent.df column associated with ymin (line.plot errorbars)
ymax.col	name of parent.df column associated with ymax (line.plot errorbars)
linetype.col	name of parent.df column associated with linetype

Value

A ggplot object is returned.

Author(s)

Greg Cicconetti/David Wade

lineplot.data *lineplot.data*

Description

lineplot.data

Usage

```
data("lineplot.data")
```

Format

A data frame with 190 observations on the following 17 variables.

`Analysis.Visit..N.` a numeric vector

`Analysis.Visit` a factor with levels `BASELINE` `DAY 1 SCREEN` `WEEK -2 SCREEN` `WEEK -4` `WEEK 10` `WEEK 12` `WEEK 14` `WEEK 16` `WEEK 18` `WEEK 2` `WEEK 20` `WEEK 21` `WEEK 22` `WEEK 23` `WEEK 24` `WEEK 28` `FOLLOW-UP` `WEEK 4` `WEEK 6` `WEEK 8`

`tt_segorder` a numeric vector

`X.` a factor with levels `Analysis` `Value`

`Order.of.Statistical.List` a numeric vector

`X..1` a factor with levels `25th Percentile` `75th Percentile` `LCLM` `Max.` `Mean` `Median` `Min.` `n` `SD` `UCLM`

`Summary.Level.Variable.Added.by.TU_STATSWITHTOTAL` a numeric vector

`NAME.OF.FORMER.VARIABLE` a factor with levels `TT_RESULT`

`LABEL.OF.FORMER.VARIABLE` a factor with levels `Result - formatted`

`Not.Assigned..N.3.` a numeric vector

`Control..N.10.` a numeric vector

`X4.mg..N.11.` a numeric vector

`X6.mg..N.16.` a numeric vector

`X8.mg..N.8.` a numeric vector

`X10.mg..N.11.` a numeric vector

`X12.mg..N.6.` a numeric vector

`X.N.1.` a numeric vector

Details

No details.

Examples

```
data(lineplot.data)
## maybe str(lineplot.data) ; plot(lineplot.data) ...
```

nsubj.plot *nsubj.plot*

Description

A function to create tables to accompany KMs and lineplots

Usage

```
nsubj.plot(
  parent.df,
  category.palette = c("red", "blue"),
  x.label = "Number of Subjects",
  y.label = "Treatment\nGroup",
  text.size = 4,
  x.col = "XVALUES",
  text.col = "N",
  category.color = "CATEGORY",
  x.limits = c(0.5, 18),
  x.ticks = unique(parent.df$XVALUES),
  x.ticks.labels = unique(parent.df$XVALUES)
)
```

Arguments

parent.df	data.frame used by ggplot
category.palette	colors assoicated with categorical variable
x.label	value gets passed to labs
y.label	value gets passed to labs
text.size	value gets passed to geom_text
x.col	parent.df column associated with response vairable (line.plot, nsubj.plot)
text.col	used by nsubj.plot
category.color	data.frame column assocated with aes color mapping (forest.plot, line.plot, nsubj.plot, table.plot)
x.limits	value gets passed to scale_x_continuous
x.ticks	value gets passed to scale_x_continuous
x.ticks.labels	passed to scale_x_continuous

Value

A ggplot object is returned.

Author(s)

Greg Cicconetti/David Wade

outputplan — *This is a dataset structured to facilitate mass figure production*

Description

This is a dataset structured to facilitate mass figure production

Author(s)

Greg Cicconetti

raw.lineplot.data — *This is a dataset that would need some pre-processing ahead of using line.plot*

Description

This is a dataset that would need some pre-processing ahead of using line.plot

Author(s)

Greg Cicconetti

refresh.outputplan — *Refresh the Output Plan*

Description

Reloads outputplan_study.csv file and applies canonical formatting changes.

Usage

```
refresh.outputplan(
  loadplan = TRUE,
  filename = "outputplan.csv",
  pos = 1,
  envir = as.environment(pos)
)
```

Arguments

loadplan	logical; if TRUE then it loads from the filename
filename	common_root.pdf or common_root.csv
pos	used internally by some functions
envir	used internally by some functions

Details

Ensure all columns are read in as character vectors. Ensure all missing entries are replaced with blank character string. Ensure all escape characters for carrige returns are respected. Grabs the 'modified time' from file attributes associated with .csv files named in the outputplan.

Value

This function returns a data.frame.

Author(s)

Greg Cicconetti

`run.specific` *run.specific*

Description

This function sources a .r driver file and sends its product to a newly opened 8.5in x 11in screen or a pdf file with 8.5in x 11in dimensions.

Usage

```
run.specific(
  source.code = "g_AErr2.r",
  outfile = "",
  toPDF = FALSE,
  toWMF = FALSE,
  toJPEG = FALSE,
  toPNG = FALSE,
  toBMP = FALSE,
  toEPS = FALSE,
  dpires = 600,
  use.log = FALSE
)
```

Arguments

`source.code`	This is intended to be a darapladib graphics driver file returning a graphic possibly with complete headers and footers.
`outfile`	If (toPDF== TRUE & outfile == "") a .pdf file with root name taken from outputplan$outfile[which(outputplan$rcode ==source.code)]. Otherwise a .pdf will be created the value of outfile. The pdf is stored in mypath/od defined in setpaths.r.
`toPDF`	Logical. If TRUE a .pdf file will be created. If FALSE graphic is sent to screen. (run.specific)
`toWMF`	Logical. If TRUE a .wmf file will be created. (run.specific)

toJPEG	Logical. If TRUE a .jpeg file will be created. (run.specific)
toPNG	Logical. If TRUE a .png file will be created. (run.specific)
toBMP	Logical. If TRUE a .bmp file will be created. (run.specific)
toEPS	Logical. If TRUE a .eps file will be created. (run.specific)
dpires	passed to devices
use.log	logical to write a log file

Value

This function passes output to a device, be it the computer screen or to file.

Author(s)

David wade

`start_session_log` *start_session_log*

Description

A function to start logging the session history for a graphic driver run

Usage

```
start_session_log(
  x,
  outputfile = "example.PDF",
  pos = 1,
  envir = as.environment(pos),
  ...
)
```

Arguments

x	used internally
outputfile	passed to name the session history log file
pos	used internally by some functions
envir	used internally by some functions
...	additional params

Details

Note that the stop_session_log function is used to stop the logging and save the log file.

Value

This function works in conjunction with stop_session_log to create a log file.

Value

No objects are returned by this function.

Author(s)

David Wade

`stop_session_log` *stop_session_log*

Description

A function to stop logging the session history for a graphic driver run and save the session history file

Usage

```
stop_session_log()
```

Details

Note that the start_session_log function is used to start the logging, and it must be called first.

Value

This function works in conjunction with start_session_log to create a log file.

Value

No objects are returned by this function.

Author(s)

David Wade

summary.lineplot.data *This is a dataset that would need some pre-processing ahead of using line.plot*

Description

This is a dataset that would need some pre-processing ahead of using line.plot

Author(s)

Greg Cicconetti

sync.ylab.widths *sync.ylab.widths*

Description

Aligns the widths of ggplot objects to ensure common plot regions. The maximum length required for y-axis labels among the list is determined and applied to the other plots. This assists in syncing the widths of ggplot objects for the purpose of align figures on a page.

Usage

```
sync.ylab.widths(gg.list, default.length = 2)
```

Arguments

gg.list	a list of ggplot objects (sync.ylab.widths)
default.length	set to 2

Value

A ggplot object is returned.

Author(s)

Greg Cicconetti

table.plot *table.plot*

Description

A function for creating harmonized table plots with A function for plotting columns of text in a figure offering compatiability with forest.plot and dot.plot.

Usage

```
table.plot(
  parent.df,
  y.rank.col = "Subcategory",
  category.color = "Treatment",
  text.col1 = "Point_Est",
  text.col2 = NULL,
  text.col3 = NULL,
  text.col4 = NULL,
  text.size = 12,
  xtick.labs = c("", "", ""),
  x.limits = NULL,
  y.limits = NULL,
  x.label = "Text",
  y.label = "Item",
  y.label.rank.col = "rank",
  y.label.col = "subcategory",
  category.palette = c("red", "blue")
)
```

Arguments

parent.df	data.frame used by ggplot
y.rank.col	column holding ranks for line items in forest/dot/table plots
category.color	data.frame column assocated with aes color mapping (forest.plot, line.plot, nsubj.plot, table.plot)
text.col1	name of column holding text for column 1 (table.plot)
text.col2	name of column holding text for column 2; can be NULL (table.plot)
text.col3	name of column holding text for column 3; can be NULL (table.plot)
text.col4	name of column holding text for column 4; can be NULL (table.plot)
text.size	value gets passed to geom_text
xtick.labs	xtick labels
x.limits	value gets passed to scale_x_continuous
y.limits	passed to scale_y_continuous
x.label	value gets passed to labs

y.label	value gets passed to labs
y.label.rank.col	column holding ranks for labels in forest/dot/table plots
y.label.col	column holding labels for forest/dot/table plots
category.palette	colors assoicated with categorical variable

Value

A ggplot object is returned.

Author(s)

Greg Cicconetti

`theme_grey2_nomargins` *figuRes2 themes*

Description

Adapts theme_grey() found in ggplot2

Usage

```
theme_grey2_nomargins(base_size = 12, base_family = "")

theme_grey2_default_margins(base_size = 12, base_family = "")

theme_bw2_nomargins(base_size = 12, base_family = "")

theme_bw2_default_margins(base_size = 12, base_family = "")

theme_table_nomargins(base_size = 12, base_family = "")
```

Arguments

base_size	used in set_theme calls
base_family	used in set_theme calls

Details

axis.text colour changed from "grey50" to "black"; legend.position changed from "right" to "bottom"; legend.direction changed to "horizontal"; plot.margin changed from default unit(c(1, 1, 0.5, 0.5), "lines") to unit(c(0, 0, 0, 0), "in")

Value

The returns a function that can be passed to ggplot2::theme_set

Functions

- `theme_grey2_default_margins()`: Same as theme_grey2_nomargins but with margins set to ggplot defaults, unit(c(1, 1, 0.5, 0.5), "lines")
- `theme_bw2_nomargins()`: Similar to theme_grey2
- `theme_bw2_default_margins()`: Similar to theme_bw_nomargins but with margins set to ggplot defaults, unit(c(1, 1, 0.5, 0.5), "lines")
- `theme_table_nomargins()`: alteration to theme_grey

Author(s)

Greg Cicconetti

Examples

```
{
ggplot2::theme_set(theme_grey2_nomargins())
}
```

Index