

Package ‘egg’

July 13, 2019

Type Package

Title Extensions for 'ggplot2': Custom Geom, Custom Themes, Plot Alignment, Labelled Panels, Symmetric Scales, and Fixed Panel Size

Version 0.4.5

License GPL-3

Description Miscellaneous functions to help customise 'ggplot2' objects. High-level functions are provided to post-process 'ggplot2' layouts and allow alignment between plot panels, as well as setting panel sizes to fixed values. Other functions include a custom 'geom', and helper functions to enforce symmetric scales or add tags to faceted plots.

VignetteBuilder knitr

Depends gridExtra (>= 2.3), ggplot2

Imports gtable, grid, grDevices, utils

Suggests knitr, png

RoxygenNote 6.1.1

NeedsCompilation no

Author Baptiste Auguie [aut, cre]

Maintainer Baptiste Auguie <baptiste.auguie@gmail.com>

Repository CRAN

Date/Publication 2019-07-13 06:00:27 UTC

R topics documented:

| | |
|------------------------------|----|
| expose_layout | 2 |
| geom_custom | 2 |
| ggarrange | 3 |
| gtable_frame | 4 |
| set_panel_size | 5 |
| symmetric_range | 6 |
| tag_facet | 7 |
| tag_facet_outside | 8 |
| theme_article | 9 |
| theme_presentation | 10 |

Index**11**

| | |
|---------------|----------------------|
| expose_layout | <i>expose_layout</i> |
|---------------|----------------------|

Description

Schematic view of a ggplot object's layout.

Usage

```
expose_layout(p, draw = TRUE, newpage = TRUE)
```

Arguments

| | |
|---------|--------------------------|
| p | ggplot |
| draw | logical, draw the gtable |
| newpage | logical |

Value

gtable

Examples

```
p1 <- qplot(mpg, wt, data=mtcars, colour=cyl)
p2 <- qplot(mpg, data = mtcars) + ggtitle('title')
p3 <- qplot(mpg, data = mtcars, geom = 'dotplot')
p4 <- p1 + facet_wrap(~carb, nrow=1) + theme(legend.position='none') +
  ggtitle('faceted plot')
p1 <- lapply(list(p1,p2, p3, p4), expose_layout, FALSE, FALSE)
grid.arrange(grobs=p1, widths=c(1.2,1,1),
             layout_matrix = rbind(c(1, 2, 3),
                                   c(4, 4, 4)))
```

| | |
|-------------|--------------------|
| geom_custom | <i>geom_custom</i> |
|-------------|--------------------|

Description

Draw user-defined grobs, typically annotations, at specific locations.

Usage

```
geom_custom(mapping = NULL, data = NULL, inherit.aes = TRUE, ...)
```

Arguments

| | |
|-------------|--|
| mapping | mapping |
| data | data |
| inherit.aes | inherit.aes |
| ... | arguments passed to the geom's draw_group method |

Value

layer

Examples

```
library(grid)
d <- data.frame(x=rep(1:3, 4), f=rep(letters[1:4], each=3))
gl <- replicate(4, matrix(sample(palette(), 9, TRUE), 3, 3), FALSE)
dummy <- data.frame(f=letters[1:4], data = I(gl))
ggplot(d, aes(f,x)) +
  facet_wrap(~f)+
  theme_bw() +
  geom_point()+
  geom_custom(data = dummy, aes(data = data, y = 2),
              grob_fun = function(x) rasterGrob(x, interpolate = FALSE,
                                                width=unit(1,'cm'),
                                                height=unit(1,'cm')))
```

ggarrange

ggarrange

Description

Arrange multiple ggplot objects on a page, aligning the plot panels.

Usage

```
ggarrange(..., plots = list(...), nrow = NULL, ncol = NULL,
          widths = NULL, heights = NULL, byrow = TRUE, top = NULL,
          bottom = NULL, left = NULL, right = NULL, padding = unit(0.5,
          "line"), clip = "on", draw = TRUE, newpage = TRUE, debug = FALSE,
          labels = NULL, label.args = list(gp = grid::gpar(font = 4, cex =
          1.2)))
```

Arguments

| | |
|-------|-----------------|
| ... | ggplot objects |
| plots | list of ggplots |
| nrow | number of rows |

| | |
|-------------------------|---|
| <code>ncol</code> | number of columns |
| <code>widths</code> | list of requested widths |
| <code>heights</code> | list of requested heights |
| <code>byrow</code> | logical, fill by rows |
| <code>top</code> | optional string, or grob |
| <code>bottom</code> | optional string, or grob |
| <code>left</code> | optional string, or grob |
| <code>right</code> | optional string, or grob |
| <code>padding</code> | unit of length one, margin around annotations |
| <code>clip</code> | argument of gtable |
| <code>draw</code> | logical: draw or return a grob |
| <code>newpage</code> | logical: draw on a new page |
| <code>debug</code> | logical, show layout with thin lines |
| <code>labels</code> | character labels used for annotation of subfigures |
| <code>label.args</code> | label list of parameters for the formatting of labels |

Value

gtable of aligned plots

Examples

```
p1 <- ggplot(mtcars, aes(mpg, wt, colour = factor(cyl))) +
  geom_point()
p2 <- ggplot(mtcars, aes(mpg, wt, colour = factor(cyl))) +
  geom_point() + facet_wrap(~ cyl, ncol=2, scales = 'free') +
  guides(colour='none') +
  theme()
ggarrange(p1, p2, widths = c(2,1), labels = c('a', 'b'))
```

`gtable_frame`

gtable_frame

Description

Reformat the gtable associated with a ggplot object into a 3x3 gtable where the central cell corresponds to the plot panel(s).

Usage

```
gtable_frame(g, width = unit(1, "null"), height = unit(1, "null"),
  debug = FALSE)
```

Arguments

| | |
|---------------------|---------------------------|
| <code>g</code> | gtable |
| <code>width</code> | requested width |
| <code>height</code> | requested height |
| <code>debug</code> | logical draw gtable cells |

Value

3x3 gtable wrapping the plot

Examples

```
library(grid)
library(gridExtra)
p1 <- ggplot(mtcars, aes(mpg, wt, colour = factor(cyl))) +
  geom_point()

p2 <- ggplot(mtcars, aes(mpg, wt, colour = factor(cyl))) +
  geom_point() + facet_wrap(~ cyl, ncol=2, scales = 'free') +
  guides(colour='none') +
  theme()

p3 <- ggplot(mtcars, aes(mpg, wt, colour = factor(cyl))) +
  geom_point() + facet_grid(. ~ cyl, scales = 'free')

g1 <- ggplotGrob(p1);
g2 <- ggplotGrob(p2);
g3 <- ggplotGrob(p3);
fg1 <- gtable_frame(g1)
fg2 <- gtable_frame(g2)
fg12 <- gtable_frame(gtable_rbind(fg1,fg2), width=unit(2,'null'), height=unit(1,'null'))
fg3 <- gtable_frame(g3, width=unit(1,'null'), height=unit(1,'null'))
grid.newpage()
combined <- gtable_cbind(fg12, fg3)
grid.draw(combined)
```

set_panel_size

set_panel_size

Description

Set the panel width/height of a ggplot to a fixed value.

Usage

```
set_panel_size(p = NULL, g = ggplot2::ggplotGrob(p), file = NULL,
  margin = unit(1, "mm"), width = unit(4, "cm"), height = unit(4,
  "cm"))
```

Arguments

| | |
|--------|-----------------------------------|
| p | ggplot2 |
| g | gtable |
| file | optional output filename |
| margin | grid unit |
| width | grid unit, requested panel width |
| height | grid unit, requested panel height |

Value

gtable with fixed panel sizes

Examples

```
p1 <- qplot(mpg, wt, data=mtcars, colour=cyl)
p2 <- p1 + facet_wrap(~carb, nrow=1)
grid.arrange(grobs=lapply(list(p1,p2), set_panel_size))
```

symmetric_range *symmetric_range*

Description

Function to ensure that a position scale is symmetric about 0

Usage

```
symmetric_range(range)
```

Arguments

| | |
|-------|-------------------|
| range | range of the data |
|-------|-------------------|

Value

symmetric range

Examples

```
library(ggplot2)
ggplot(mpg, aes(cty, hwy)) +
  geom_point() +
  scale_x_continuous(limits = symmetric_range)
```

| | |
|-----------|------------------|
| tag_facet | <i>tag_facet</i> |
|-----------|------------------|

Description

Adds a dummy text layer to a ggplot to label facets and sets facet strips to blank. This is the typical formatting for some journals that consider facets as subfigures and want to minimise margins around figures.

Usage

```
tag_facet(p, open = "(", close = ")", tag_pool = letters, x = -Inf,
  y = Inf, hjust = -0.5, vjust = 1.5, fontface = 2, family = "",
  ...)
```

Arguments

| | |
|----------|---|
| p | ggplot |
| open | opening character, default: (|
| close | closing character, default:) |
| tag_pool | character vector to pick tags from |
| x | x position within panel, default: -Inf |
| y | y position within panel, default: Inf |
| hjust | hjust |
| vjust | vjust |
| fontface | fontface |
| family | font family |
| ... | further arguments passed to geom_text layer |

Value

plot with facet strips removed and replaced by in-panel tags

Examples

```
library(ggplot2)
mydf = data.frame(
  x = 1:90,
  y = rnorm(90),
  red = rep(letters[1:3], 30),
  blue = c(rep(1, 30), rep(2, 30), rep(3, 30)))

p <- ggplot(mydf) +
  geom_point(aes(x = x, y = y)) +
  facet_wrap(
    ~ red + blue)
tag_facet(p)
```

| | |
|-------------------|--------------------------|
| tag_facet_outside | <i>tag_facet_outside</i> |
|-------------------|--------------------------|

Description

Adds a dummy text layer to a ggplot to label facets and sets facet strips to blank. This is the typical formatting for some journals that consider facets as subfigures and want to minimise margins around figures.

Usage

```
tag_facet_outside(p, open = c("(", ""), close = c(")", "."),
  tag_fun_top = function(i) letters[i],
  tag_fun_right = utils::as.roman, x = c(0, 0), y = c(0.5, 1),
  hjust = c(0, 0), vjust = c(0.5, 1), fontface = c(2, 2),
  family = "", draw = TRUE, ...)
```

Arguments

| | |
|---------------|---|
| p | ggplot |
| open | opening character, default: (|
| close | closing character, default:) |
| tag_fun_top | labelling function |
| tag_fun_right | labelling function |
| x | x position within cell |
| y | y position within cell |
| hjust | hjust |
| vjust | vjust |
| fontface | fontface |
| family | font family |
| draw | logical: draw the resulting gtable |
| ... | further arguments passed to geom_text layer |

Value

plot with facet strips removed and replaced by in-panel tags

Examples

```
library(ggplot2)
d = data.frame(
  x = 1:90,
  y = rnorm(90),
  red = rep(letters[1:3], 30),
  blue = c(rep(1, 30), rep(2, 30), rep(3, 30)))

p <- ggplot(d) +
  geom_point(aes(x = x, y = y)) +
  facet_grid(red ~ blue)

tag_facet_outside(p)
```

| | |
|---------------|--|
| theme_article | <i>Theme with minimalistic (and opinionated) defaults suitable for publication</i> |
|---------------|--|

Description

Theme with minimalistic (and opinionated) defaults suitable for publication

Usage

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

Arguments

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

Examples

```
library(ggplot2)

d = data.frame(
  x = 1:90,
  y = rnorm(90),
  red = rep(letters[1:3], 30),
  blue = c(rep(1, 30), rep(2, 30), rep(3, 30)))

p <- ggplot(d) +
  geom_point(aes(x = x, y = y)) +
  facet_grid(red ~ blue)
tag_facet(p + theme_article())
p + theme_presentation()

# example of use with cairo device
```

```
# ggsave("fig_talk.pdf", p + theme_presentation("Source Sans Pro"),  
#       width=14, height=7, device = cairo_pdf, bg='transparent')
```

| | |
|--------------------|---|
| theme_presentation | <i>Theme with minimalistic (and opinionated) defaults suitable for presentation</i> |
|--------------------|---|

Description

Theme with minimalistic (and opinionated) defaults suitable for presentation

Usage

```
theme_presentation(base_size = 24, base_family = "")
```

Arguments

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

Index

`expose_layout`, 2

`geom_custom`, 2

`ggarrange`, 3

`gtable_frame`, 4

`set_panel_size`, 5

`symmetric_range`, 6

`tag_facet`, 7

`tag_facet_outside`, 8

`theme_article`, 9

`theme_presentation`, 10