

Package: ggeasy (via r-universe)

August 27, 2024

Title Easy Access to 'ggplot2' Commands

Version 0.1.4

Maintainer Jonathan Carroll <rpkg@jcarroll.com.au>

Description Provides a series of aliases to commonly used but difficult to remember 'ggplot2' sequences.

License GPL (>=2)

Encoding UTF-8

Imports ggplot2, rlang

RoxygenNote 7.1.1

Suggests knitr, rmarkdown, testthat (>= 2.1.0), clipr, labelled, patchwork, vdiff, covr

VignetteBuilder knitr

URL <https://github.com/jonocarroll/ggeasy>

BugReports <https://github.com/jonocarroll/ggeasy/issues>

Repository <https://jonocarroll.r-universe.dev>

RemoteUrl <https://github.com/jonocarroll/ggeasy>

RemoteRef HEAD

RemoteSha 91c7a641d7522850301de397195a14a7f6823bc0

Contents

easy_add_legend_title	2
easy_change_legend	3
easy_change_text	4
easy_labs	6
easy_remove_axes	7
easy_remove_gridlines	8
easy_remove_legend	10
easy_remove_legend_title	11
easy_rotate_labels	11
ggeasy	12

easy_add_legend_title *Easily add legend title(s)*

Description

Update the title(s) of the specified aesthetic, or all aesthetics at once.

Usage

```
easy_add_legend_title(..., teach = FALSE)
```

Arguments

... A list of new name-value pairs. The name should be an aesthetic. If only a character value is provided then *all* legend titles will be changed to that.

teach print longer equivalent [ggplot2](#) expression?

Value

a [theme](#) object

Author(s)

Jonathan Carroll

Examples

```
library(ggplot2)

# Add legend title to a specific aesthetic
ggplot(mtcars, aes(wt, mpg, colour = cyl, size = hp)) +
  geom_point() + easy_add_legend_title(col = "Number of Cylinders")

# Add legend title to all aesthetics
ggplot(mtcars, aes(wt, mpg, colour = cyl)) +
  geom_point() + easy_add_legend_title("Number of Cylinders")
```

easy_change_legend *Easily modify legends*

Description

Change legend position, direction, or justification.

Usage

```
easy_change_legend(  
  what = c("position", "direction", "justification"),  
  to,  
  teach = FALSE  
)
```

```
easy_move_legend(  
  to = c("right", "none", "left", "bottom", "top"),  
  teach = FALSE  
)
```

```
easy_legend_at(to = c("right", "none", "left", "bottom", "top"), teach = FALSE)
```

```
easy_rotate_legend(to = c("vertical", "horizontal"), teach = FALSE)
```

```
easy_adjust_legend(to = c("left", "right", "center"), teach = FALSE)
```

Arguments

what	legend component to modify ("position", "direction", or "justification")
to	to what to set the legend component should be changed
teach	print longer equivalent ggplot2 expression?

Details

Due to limitations of [ggplot2](#) this will apply to all legends at once

Value

a [theme](#) object

Author(s)

Jonathan Carroll

Examples

```

library(ggplot2)

# Move legends to bottom
ggplot(mtcars, aes(wt, mpg, colour = cyl, size = hp)) +
  geom_point() + easy_move_legend("bottom")

# Make legends horizontal
ggplot(mtcars, aes(wt, mpg, colour = cyl, size = hp)) +
  geom_point() + easy_rotate_legend("horizontal")

# Justify legends to the bottom and justify to the right
ggplot(mtcars, aes(wt, mpg, colour = cyl, size = hp)) +
  geom_point() +
  easy_move_legend("bottom") +
  easy_adjust_legend("right")

```

easy_change_text *Easily change text appearance*

Description

Change any of the text parameters such as

Usage

```

easy_change_text(
  which = .all_theme_els,
  what = .all_element_text,
  to = NULL,
  teach = FALSE
)

.all_theme_els

.all_element_text

easy_all_text_size(size = NULL, teach = FALSE)

easy_all_text_color(color = NULL, teach = FALSE)

easy_all_text_colour(colour = NULL, teach = FALSE)

easy_text_size(which = .all_theme_els, size = NULL, teach = FALSE)

easy_text_color(which = .all_theme_els, color = NULL, teach = FALSE)

easy_text_colour(which = .all_theme_els, colour = NULL, teach = FALSE)

```

```
easy_x_axis_title_size(size, teach = FALSE)
easy_y_axis_title_size(size, teach = FALSE)
easy_x_axis_labels_size(size = NULL, teach = FALSE)
easy_y_axis_labels_size(size = NULL, teach = FALSE)
easy_plot_title_size(size = NULL, teach = FALSE)
easy_plot_subtitle_size(size = NULL, teach = FALSE)
easy_plot_caption_size(size = NULL, teach = FALSE)
easy_plot_legend_size(size = NULL, teach = FALSE)
easy_plot_legend_title_size(size = NULL, teach = FALSE)
easy_center_title(teach = FALSE)
```

Arguments

which	which element to change (see <code>ggeasy:::all_theme_els</code>)
what	what attribute of the element to change (see <code>ggeasy:::all_element_text</code>)
to	value to which the attribute should be set
teach	print longer equivalent <code>ggplot2</code> expression?
size	size to set attributes to
color	colo(u)r to set attributes to
colour	colo(u)r to set attributes to

Format

An object of class character of length 15.

Value

a `theme` object which can be used in `ggplot2` calls

Author(s)

Jonathan Carroll

Examples

```
library(ggplot2)
# make all text larger
```

```

ggplot(mtcars, aes(mpg, hp)) +
  geom_point() +
  easy_all_text_size(22)
# also works if accidentally using easy_text_size(n)

# make the x and y axis text larger
ggplot(mtcars, aes(mpg, hp)) +
  geom_point() +
  easy_text_size(c("axis.text.x", "axis.text.y"), 22)

# make the x axis labels larger
ggplot(mtcars, aes(mpg, hp)) +
  geom_point() +
  easy_x_axis_labels_size(22)

# make the plot title larger
ggplot(mtcars, aes(mpg, hp)) +
  geom_point() +
  labs(title = "My Plot") +
  easy_plot_title_size(22)

# make the legend title larger
ggplot(mtcars, aes(mpg, hp)) +
  geom_point(aes(fill = gear)) +
  easy_plot_legend_title_size(22)

# make all the text red
ggplot(mtcars, aes(mpg, hp)) +
  geom_point(aes(fill = gear)) +
  easy_all_text_color("red")

# make all the text 45 degrees, right-justified
ggplot(mtcars, aes(mpg, hp)) +
  geom_point(aes(fill = gear)) +
  easy_change_text(what = "angle", to = 45) +
  easy_change_text(what = "hjust", to = 1)

# make just x-axis text 45 degrees, right-justified
ggplot(mtcars, aes(mpg, hp)) +
  geom_point(aes(fill = gear)) +
  easy_change_text(which = "axis.text.x", what = "angle", to = 45) +
  easy_change_text(which = "axis.text.x", what = "hjust", to = 1)

```

easy_labs

Easily add ggplot labels using label attribute of 'data.frame' column

Description

Applies same logic as [labs](#) but uses as default the column label attribute if present as the variable label in the plot.

Usage

```
easy_labs(..., teach = FALSE)
```

Arguments

... A list of new name-value pairs. The name should either be an aesthetic, or one of "title", "subtitle", or "caption"

teach print longer equivalent `ggplot2` expression?

Examples

```
iris_labs <- iris

lbl <- c('Sepal Length', 'Sepal Width', 'Petal Length', 'Petal Width', 'Flower Species')

labelled::var_label(iris_labs) <- split(lbl, names(iris_labs))

p <- ggplot2::ggplot(iris_labs, ggplot2::aes(x=Sepal.Length, y=Sepal.Width))+
  ggplot2::geom_line(ggplot2::aes(colour=Species))

p

p + easy_labs()
p + easy_labs(title = "Plot Title", subtitle = 'Plot Subtitle', x = 'x axis label')

p + easy_labs(teach = TRUE)
```

 easy_remove_axes

Easily remove one or more axes

Description

`easy_remove_axes` by default removes both axes, but can remove only x or y if "x" or "y" is given to the 'which' argument

Usage

```
easy_remove_axes(
  which = c("both", "x", "y"),
  what = c("ticks", "title", "text", "line"),
  teach = FALSE
)

easy_remove_y_axis(what = c("ticks", "title", "text", "line"), teach = FALSE)

easy_remove_x_axis(what = c("ticks", "title", "text", "line"), teach = FALSE)
```

Arguments

which	which axis or axes to remove, by default "both"
what	axis components to remove ("ticks", "title", "text", and/or "line")
teach	print longer equivalent <code>ggplot2</code> expression?

Details

`easy_remove_x_axis` and `easy_remove_y_axis` remove just the x or y axis, respectively.

Value

a `theme` object which can be used in `ggplot2` calls

Author(s)

Alicia Schep

Examples

```
library(ggplot2)
# Remove all axes
ggplot(mtcars, aes(wt, mpg)) +
  geom_point() + easy_remove_axes()

# remove just x axis
ggplot(mtcars, aes(wt, mpg)) +
  geom_point() + easy_remove_x_axis()

# can also use:
ggplot(mtcars, aes(wt, mpg)) +
  geom_point() + easy_remove_axes("x")

# Remove y axis
ggplot(mtcars, aes(wt, mpg)) +
  geom_point() + easy_remove_y_axis()

# Remove just the ticks
# Remove y axis
ggplot(mtcars, aes(wt, mpg)) +
  geom_point() + easy_remove_y_axis(what = "ticks")
```

`easy_remove_gridlines` *Easily remove gridlines*

Description

Easily remove any gridlines from a `ggplot`.

Usage

```
easy_remove_gridlines(  
  axis = c("both", "x", "y"),  
  minor = TRUE,  
  major = TRUE,  
  teach = FALSE  
)  
  
easy_remove_x_gridlines(minor = TRUE, major = TRUE, teach = FALSE)  
  
easy_remove_y_gridlines(minor = TRUE, major = TRUE, teach = FALSE)
```

Arguments

axis	From which axis should grid lines be removed? By default "both".
minor	Should minor grid lines be removed? By default TRUE.
major	Should major grid lines be removed? By default TRUE.
teach	Should the equivalent ggplot2 expression be printed? By default FALSE.

Value

a [theme](#) object which can be used in [ggplot2](#) calls

Author(s)

Thomas Neitmann

Examples

```
library(ggplot2)  
p <- ggplot(mtcars, aes(hp, mpg))  
  
# remove all grid lines at once  
p + easy_remove_gridlines()  
  
# remove all minor grid lines  
p + easy_remove_gridlines(major = FALSE)  
  
# remove all major grid lines  
p + easy_remove_gridlines(minor = FALSE)  
  
# remove x gridlines  
p + easy_remove_x_gridlines()  
# or  
p + easy_remove_gridlines(axis = "x")  
  
# remove y gridlines  
p + easy_remove_y_gridlines()
```

easy_remove_legend *Easily remove legend(s)*

Description

With no argument, will remove all legends. Provide the name(s) of specific aesthetic to remove only certain legends.

Usage

```
easy_remove_legend(..., teach = FALSE)
```

Arguments

... optional name(s) specific aesthetics for which to remove the legend
teach print longer equivalent [ggplot2](#) expression?

Value

either a [theme](#) object or a [guides](#) object, both of which can be used in [ggplot2](#) calls

Author(s)

Alicia Schep

Examples

```
library(ggplot2)

# Remove all legends
ggplot(mtcars, aes(wt, mpg, colour = cyl, size = hp)) +
  geom_point() + easy_remove_legend()

# remove just size legend
ggplot(mtcars, aes(wt, mpg, colour = cyl, size = hp)) +
  geom_point() + easy_remove_legend("size")

# can also use:
ggplot(mtcars, aes(wt, mpg, colour = cyl, size = hp)) +
  geom_point() + easy_remove_legend(size)

# Remove more than one
ggplot(mtcars, aes(wt, mpg, colour = cyl, size = hp)) +
  geom_point() + easy_remove_legend(size, color)
```

`easy_remove_legend_title`*Easily remove legend title Remove the legend title*

Description

Easily remove legend title Remove the legend title

Usage

```
easy_remove_legend_title(teach = FALSE)
```

Arguments

`teach` print longer equivalent `ggplot2` expression?

Value

a `theme` object

Examples

```
library(ggplot2)
# remove legend title from all aesthetics
ggplot(mtcars, aes(wt, mpg, colour = cyl)) +
  geom_point() + easy_remove_legend_title()
```

`easy_rotate_labels`*Easily rotate x axis labels*

Description

A shortcut to

Usage

```
easy_rotate_labels(
  which = c("both", "x", "y"),
  angle = 90,
  side = c("left", "middle", "right"),
  teach = FALSE
)
```

```
easy_rotate_x_labels(
  angle = 90,
  side = c("left", "middle", "right"),
```

```
  teach = FALSE
)

easy_rotate_y_labels(
  angle = 90,
  side = c("left", "middle", "right"),
  teach = FALSE
)
```

Arguments

which	which axis or axes to rotate, by default "both"
angle	angle through which the text should be rotated. Can also be one of "startatop" or "startatbottom" to define where the text should start.
side	horizontal justification of the text before rotation
teach	print longer equivalent <code>ggplot2</code> expression?

Details

```
theme(axis.text.x = element_text(angle, hjust))
```

Value

a `theme` object which can be used in `ggplot2` calls.

Examples

```
library(ggplot2)
ggplot(mtcars, aes(mpg, hp)) +
  geom_point() +
  easy_rotate_labels()
```

ggeasy

ggeasy.

Description

Helper functions for making using `ggplot2` easier.

Index

- * **datasets**
 - easy_change_text, 4
 - .all_element_text (easy_change_text), 4
 - .all_theme_els (easy_change_text), 4
- easy_add_legend_title, 2
- easy_adjust_legend
 - (easy_change_legend), 3
- easy_all_text_color (easy_change_text), 4
- easy_all_text_colour
 - (easy_change_text), 4
- easy_all_text_size (easy_change_text), 4
- easy_center_title (easy_change_text), 4
- easy_change_legend, 3
- easy_change_text, 4
- easy_labs, 6
- easy_legend_at (easy_change_legend), 3
- easy_move_legend (easy_change_legend), 3
- easy_plot_caption_size
 - (easy_change_text), 4
- easy_plot_legend_size
 - (easy_change_text), 4
- easy_plot_legend_title_size
 - (easy_change_text), 4
- easy_plot_subtitle_size
 - (easy_change_text), 4
- easy_plot_title_size
 - (easy_change_text), 4
- easy_remove_axes, 7
- easy_remove_gridlines, 8
- easy_remove_legend, 10
- easy_remove_legend_title, 11
- easy_remove_x_axis (easy_remove_axes), 7
- easy_remove_x_gridlines
 - (easy_remove_gridlines), 8
- easy_remove_y_axis (easy_remove_axes), 7
- easy_remove_y_gridlines
 - (easy_remove_gridlines), 8
- easy_rotate_labels, 11
- easy_rotate_legend
 - (easy_change_legend), 3
- easy_rotate_x_labels
 - (easy_rotate_labels), 11
- easy_rotate_y_labels
 - (easy_rotate_labels), 11
- easy_text_color (easy_change_text), 4
- easy_text_colour (easy_change_text), 4
- easy_text_size (easy_change_text), 4
- easy_x_axis_labels_size
 - (easy_change_text), 4
- easy_x_axis_title_size
 - (easy_change_text), 4
- easy_y_axis_labels_size
 - (easy_change_text), 4
- easy_y_axis_title_size
 - (easy_change_text), 4
- ggeasy, 12
- ggplot2, 2, 3, 5, 7–12
- guides, 10
- labs, 6
- theme, 2, 3, 5, 8–12