

Package ‘treemapify’

November 18, 2017

Title Draw Treemaps in 'ggplot2'

Version 2.4.0

Author David Wilkins <david@wilcox.org>

Maintainer David Wilkins <david@wilcox.org>

URL <https://github.com/wilcox/treemapify>

BugReports <https://github.com/wilcox/treemapify/issues>

Description Provides 'ggplot2' geoms for drawing treemaps.

Depends R (>= 3.1)

Imports plyr (>= 1.3), ggplot2 (>= 2.2.1), ggfittext (>= 0.5.0), grid (>= 3.1)

Suggests knitr, rmarkdown, testthat

License GPL (>= 3)

LazyData true

RoxygenNote 6.0.1

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2017-11-18 22:29:57 UTC

R topics documented:

G20	2
geom_treemap	2
geom_treemap_subgroup_border	4
geom_treemap_subgroup_text	5
geom_treemap_text	6
treemapify	8

Index	10
--------------	-----------

G20

Statistics on the G-20 group of major world economies.

Description

A dataset containing economic and demographic statistics about members of the G-20 group of major world economies.

Usage

G20

Format

A data frame with 20 rows and five variables:

region the country's region

country the country

gdp_mil_usd the country's GDP, in millions of US dollars

hdi the country's Human Development Index

econ_classification the country's economic classification

Source

http://en.wikipedia.org/wiki/G-20_major_economies

geom_treemap

A 'ggplot2' geom to draw a treemap.

Description

'geom_treemap' provides a plot layer that divides the plot into tiles representing dataset observations. The relative area of each tile expresses an 'area' aesthetic.

Usage

```
geom_treemap(mapping = NULL, data = NULL, stat = "identity",  
             position = "identity", na.rm = FALSE, show.legend = NA,  
             inherit.aes = TRUE, fixed = F, ...)
```

Arguments

mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...

Standard geom arguments as for 'ggplot2::geom_rect'.

fixed If 'TRUE', the alternative 'fixed' tile layout algorithm will be used.

Details

'geom_treemap' requires an 'area' aesthetic. It will ignore any aesthetics relating to the x and y axes (e.g. 'xmin' or 'y'), as the x and y axes are not meaningful in a treemap. Several other standard 'ggplot2' aesthetics are supported (see Aesthetics). To add text labels to tiles, see 'geom_treemap_text'.

An optional 'subgroup' aesthetic will cause the tiles to be clustered in subgroups within the treemap. See 'geom_treemap_subgroup_border' and 'geom_treemap_subgroup_text' to draw borders around subgroups and label them, respectively.

Two algorithms for the tile layout are provided. With the default 'squarified' algorithm ('fixed = FALSE'), the priority is ensuring the tiles have an aesthetically pleasing aspect ratio; that is, they are not too narrow or too short. In this algorithm, tile placement proceeds from the bottom left corner, filling alternately rightwards then upwards until all tiles are placed. See Bruls et al. (1999) for the full algorithm.

With the alternative 'fixed' layout algorithm ('fixed = TRUE'), the plot area is divided into vertical columns, which are filled from left to right with an equal number of tiles beginning at the bottom of each column. Unlike the default 'squarified' algorithm, with the 'fixed' algorithm the relative positions of the tiles are fixed by their order in the input data frame. This can result in aesthetically unpleasing tiles, but it allows side-by-side comparisons or animations to be created.

All 'treemapify' geoms added to a plot should have the same value for 'fixed', or they will not share a common layout.

Aesthetics

- area (required)
- alpha
- colour
- fill
- linetype
- subgroup

References

Bruls, M., Huizing, K., & van Wijk, J. (1999). Squarified Treemaps (pp. 33-42). Proceedings of the Joint Eurographics and IEEE TCVG Symposium on Visualization. <http://www.win.tue.nl/~vanwijk/stm.pdf>

See Also

geom_treemap_text, geom_treemap_subgroup_border, geom_treemap_subgroup_text

Examples

```
ggplot2::ggplot(G20, ggplot2::aes(area = gdp_mil_usd, fill = region)) +  
  geom_treemap()
```

geom_treemap_subgroup_border

A 'ggplot2' geom to draw a border around a subgroup of treemap tiles.

Description

When 'geom_treemap' is used with the 'subgroup' aesthetic to subgroup treemap tiles, 'geom_treemap_subgroup_border' can be used to draw a border around each subgroup.

Usage

```
geom_treemap_subgroup_border(mapping = NULL, data = NULL,  
  stat = "identity", position = "identity", na.rm = FALSE,  
  show.legend = NA, inherit.aes = TRUE, fixed = F, ...)
```

Arguments

mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...
Standard geom arguments as for 'ggplot2::geom_rect'.

fixed If 'TRUE', the alternative 'fixed' tile layout algorithm will be used.

Details

'geom_treemap_subgroup_border' requires 'area' and 'subgroup' aesthetics. Several other standard 'ggplot2' aesthetics are supported (see Aesthetics).

All 'treemapify' geoms added to a plot should have the same value for 'fixed', or they will not share a common layout (see 'geom_treemap' for details on the layout algorithms).

Aesthetics

- area (required)
- subgroup (required)
- colour
- size
- linetype
- alpha

See Also

geom_treemap, geom_treemap_subgroup_text

Examples

```
ggplot2::ggplot(G20, ggplot2::aes(area = gdp_mil_usd, fill = hdi,
                                subgroup = region)) +
  geom_treemap() +
  geom_treemap_subgroup_border()
```

geom_treemap_subgroup_text

A 'ggplot2' geom to add text labels to treemap subgroups.

Description

When 'geom_treemap' is used with the 'subgroup' aesthetic to subgroup treemap tiles, 'geom_treemap_subgroup_text' can be used to add a text label to each subgroup.

Usage

```
geom_treemap_subgroup_text(mapping = NULL, data = NULL, stat = "identity",
  position = "identity", na.rm = FALSE, show.legend = FALSE,
  inherit.aes = TRUE, padding.x = grid::unit(1, "mm"),
  padding.y = grid::unit(1, "mm"), place = "bottom", min.size = 4,
  grow = F, reflow = F, fixed = F, ...)
```

Arguments

mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...

Standard geom arguments as for 'ggplot2::geom_text'.

padding.x, padding.y

'grid::unit' object, giving horizontal or vertical padding between text and edge of tile. Defaults to 1 mm.

place Where inside the box to place the text. Default is 'bottom'; other options are 'topleft', 'top', 'topright', etc.

min.size Minimum font size, in points. If provided, text that would need to be shrunk below this size to fit the box will not be drawn. Defaults to 4 pt.

grow If 'TRUE', text will be grown as well as shrunk to fill the box.

reflow If 'TRUE', text will be reflowed (wrapped) to better fit the box.

fixed If 'TRUE', the alternative 'fixed' tile layout algorithm will be used.

Details

'geom_treemap_subgroup_text' requires 'area', 'label' and 'subgroup' aesthetics. Several other standard 'ggplot2' aesthetics are supported (see Aesthetics).

'geom_treemap_subgroup_text' uses the 'ggfittext' package to fit text to the subgroup. All text drawing options available in 'ggfittext::geom_fit_text' (growing, reflowing, etc.) are also available here. For full details on how these options work, see the documentation for 'ggfittext::geom_fit_text'.

All 'treemapify' geoms added to a plot should have the same value for 'fixed', or they will not share a common layout (see 'geom_treemap' for details on the layout algorithms).

Aesthetics

- area (required)
- label (required)
- subgroup (required)
- colour
- size
- alpha
- family
- fontface
- angle

See Also

geom_treemap, geom_treemap_subgroup_border

Examples

```
ggplot2::ggplot(G20, ggplot2::aes(area = gdp_mil_usd, fill = hdi,
                                  subgroup = region, label = region)) +
  geom_treemap() +
  geom_treemap_subgroup_text()
```

geom_treemap_text

A 'ggplot2' geom to add text labels to treemap tiles

Description

'geom_treemap_text' can be used to add a text label to each tile in a treemap created with 'geom_treemap'.

Usage

```
geom_treemap_text(mapping = NULL, data = NULL, stat = "identity",
  position = "identity", na.rm = FALSE, show.legend = FALSE,
  inherit.aes = TRUE, padding.x = grid::unit(1, "mm"),
  padding.y = grid::unit(1, "mm"), place = "topleft", min.size = 4,
  grow = F, reflow = F, fixed = F, ...)
```

Arguments

mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...
Standard geom arguments as for 'ggplot2::geom_text'.

padding.x, padding.y
'grid::unit' object, giving horizontal or vertical padding between text and edge of tile. Defaults to 1 mm.

place
Where inside the box to place the text. Default is 'bottom'; other options are 'topleft', 'top', 'topright', etc.

min.size
Minimum font size, in points. If provided, text that would need to be shrunk below this size to fit the box will not be drawn. Defaults to 4 pt.

grow
If 'TRUE', text will be grown as well as shrunk to fill the box.

reflow
If 'TRUE', text will be reflowed (wrapped) to better fit the box.

fixed
If 'TRUE', the alternative 'fixed' tile layout algorithm will be used.

Details

'geom_treemap_text' requires 'area' and 'label' aesthetics. Several other standard 'ggplot2' aesthetics are supported (see Aesthetics).

'geom_treemap_text' uses the 'ggfittext' package to fit text to tiles. All text drawing options available in 'ggfittext::geom_fit_text' (growing, reflowing, etc.) are also available here. For full details on how these options work, see the documentation for 'ggfittext::geom_fit_text'.

All 'treemapify' geoms added to a plot should have the same value for 'fixed', or they will not share a common layout (see 'geom_treemap' for details on the layout algorithms).

Aesthetics

- area (required)
- label (required)
- subgroup
- colour
- size
- alpha
- family
- fontface
- angle

See Also

geom_treemap

Examples

```
ggplot2::ggplot(G20, ggplot2::aes(area = gdp_mil_usd,
                                fill = econ_classification,
                                label = country)) +
  geom_treemap() +
  geom_treemap_text()
```

treemapify

Generate a treemap layout.

Description

‘treemapify’ and ‘treemapify_fixed’ produce a set of raw coordinates used to draw a treemap from a data frame of observations. To draw a treemap with ‘ggplot2’, use ‘geom_treemap’ instead.

Usage

```
treemapify(data, area, fill, group, label, xlim = c(0, 100), ylim = c(0,
  100))
```

```
treemapify_fixed(data, area, fill, group, label, xlim = c(0, 100),
  ylim = c(0, 100))
```

Arguments

data	A tidy data frame.
area	Name of the variable (a column in ‘data’) to be mapped to the area of treemap tiles.
fill	Name of the variable (a column in ‘data’) to be mapped to the fill colour of treemap tiles.
group	Optionally, name of the variable (a column in ‘data’) by which the tiles will be grouped; that is, in the final treemap layout, these tiles will be kept together.
label	Optionally, name of the variable (a column in ‘data’) giving tile labels that will be preserved in the output data frame.
xlim, ylim	Optional two-element vectors specifying the x and y limits of the plot area into which the treemap is to be drawn.

Details

'treemapify' and 'treemapify_fixed' return a data frame of tile coordinates for the treemap. This is only useful if you wish to draw the treemap without 'ggplot2', or for some edge cases such as treemaps drawn in 'R Shiny' (see e.g. <https://stackoverflow.com/q/45021775>). The easiest way to draw a treemap with this package is to use the provided 'ggplot2' geoms, such as 'geom_treemap'.

Input data frame must be in tidy format, i.e. each row must represent a single observation and each column a single variable. You must provide, as arguments, the names of the variables that will be represented by the area and fill colour of treemap tiles. Optionally, you can also select a variable by which to group the tiles within the treemap layout, and a variable that will be preserved as a label for each tile.

Two algorithms for the tile layout are provided. With the default 'squarified' algorithm ('treemapify'), the priority is ensuring the tiles have an aesthetically pleasing aspect ratio; that is, they are not too narrow or too short. In this algorithm, tile placement proceeds from the bottom left corner, moving alternately rightwards and upwards until all tiles are placed. See Bruls et al. (1999) for the full algorithm.

With the alternative 'fixed' layout algorithm ('fixed = TRUE'), the plot area is divided into vertical columns, which are filled from left to right with an equal number of tiles beginning at the bottom of each column. Unlike the default 'squarified' algorithm, with the 'fixed' algorithm the relative positions of the tiles are fixed by their order in the input data frame. This can result in aesthetically unpleasing tiles, but it allows side-by-side comparisons or animations to be created.

References

Bruls, M., Huizing, K., & van Wijk, J. (1999). Squarified Treemaps (pp. 33-42). Proceedings of the Joint Eurographics and IEEE TCVG Symposium on Visualization. <http://www.win.tue.nl/~vanwijk/stm.pdf>

See Also

geom_treemap

Examples

```
treemapify(G20, area = "gdp_mil_usd", fill = "hdi", group = "region",  
           label = "country")
```

```
treemapify_fixed(G20, area = "gdp_mil_usd", fill = "hdi", group = "region",  
                 label = "country")
```

Index

*Topic **datasets**

G20, [2](#)

G20, [2](#)

geom_treemap, [2](#)

geom_treemap_subgroup_border, [4](#)

geom_treemap_subgroup_text, [5](#)

geom_treemap_text, [6](#)

treemapify, [8](#)

treemapify_fixed (treemapify), [8](#)