Package 'productplots'

August 29, 2016

Title Product Plots for R

Description Framework for visualising tables of counts, proportions and probabilities. The framework is called product plots, alluding to the computation of area as a product of height and width, and the statistical concept of generating a joint distribution from the product of conditional and marginal distributions. The framework, with extensions, is sufficient to encompass over 20 visualisations previously described in fields of statistical graphics and 'infovis', including bar charts, mosaic plots, 'treemaps', equal area plots and fluctuation diagrams.

Version 0.1.1

Imports plyr, ggplot2

Suggests reshape2, testthat, covr

License GPL-2

LazyData true

RoxygenNote 5.0.1

URL https://github.com/hadley/productplots

BugReports https://github.com/hadley/productplots/issues

NeedsCompilation no

Author Hadley Wickham [aut, cre], Heike Hofmann [aut]

Maintainer Hadley Wickham <hadley@rstudio.com>

Repository CRAN

Date/Publication 2016-07-02 07:38:04

R topics documented:

`ddecker` *Template for a double decker plot. A double decker plot is composed of a sequence of spines in the same direction, with the final spine in the opposite direction.*

Description

Template for a double decker plot. A double decker plot is composed of a sequence of spines in the same direction, with the final spine in the opposite direction.

Usage

```
ddecker(direction = "h")
```

Arguments

`direction`	direction of first split

`find_col_level` *Find the first level which has columns.*

Description

Returns `NA` if no columns at any level.

Usage

```
find_col_level(df)
```

Arguments

df	data frame of rectangle positions

---

find_row_level *Find the first level which has rows.*

Description

Returns NA if no rows at any level.

Usage

```
find_row_level(df)
```

Arguments

df	data frame of rectangle positions

---

fluct *Fluctation partitioning.*

Description

Fluctation partitioning.

Usage

```
fluct(data, bounds, offset = 0.05, max = NULL)
```

Arguments

data	bounds data frame
bounds	bounds of space to partition
offset	space between spines
max	maximum value

flucts *Template for a fluctuation diagram.*

Description

Template for a fluctuation diagram.

Usage

```
flucts(direction = "h")
```

Arguments

direction	direction of first split

happy *Data related to happiness from the general social survey.*

Description

The data is a small sample of variables related to happiness from the general social survey (GSS). The GSS is a yearly cross-sectional survey of Americans, run from 1976. We combine data for 25 years to yield 51,020 observations, and of the over 5,000 variables, we select nine related to happiness:

Usage

```
data(happy)
```

Format

A data frame with 51020 rows and 10 variables

Details

- age. age in years: 18–89.
- degree. highest education: lt high school, high school, junior college, bachelor, graduate.
- finrela. relative financial status: far above, above average, average, below average, far below.
- happy. happiness: very happy, pretty happy, not too happy.
- health. health: excellent, good, fair, poor.
- marital. marital status: married, never married, divorced, widowed, separated.
- sex. sex: female, male.
- wtsall. probability weight. 0.43–6

hbar *Horizontal bar partition: width constant, height varies.*

Description

Horizontal bar partition: width constant, height varies.

Usage

```
hbar(data, bounds, offset = 0.02, max = NULL)
```

Arguments

data	bounds data frame
bounds	bounds of space to partition
offset	space between spines
max	maximum value

hspine *Horizontal spine partition: height constant, width varies.*

Description

Horizontal spine partition: height constant, width varies.

Usage

```
hspine(data, bounds, offset = 0.01, max = NULL)
```

Arguments

data	bounds data frame
bounds	bounds of space to partition
offset	space between spines
max	maximum value

`mosaic` *Template for a mosaic plot. A mosaic plot is composed of spines in alternating directions.*

Description

Template for a mosaic plot. A mosaic plot is composed of spines in alternating directions.

Usage

```
mosaic(direction = "v")
```

Arguments

`direction`	direction of first split

`nested` *Template for a nested barchart. A nested bar is just a sequence of bars in the same direction.*

Description

Template for a nested barchart. A nested bar is just a sequence of bars in the same direction.

Usage

```
nested(direction = "h")
```

Arguments

`direction`	direction of first split

`prodplot` *Create a product plot*

Description

Create a product plot

Usage

```
prodplot(data, formula, divider = mosaic(), cascade = 0, scale_max = TRUE,
  na.rm = FALSE, levels = -1L, ...)
```

Arguments

data	input data frame
formula	formula specifying display of plot
divider	divider function
cascade	cascading amount, per nested layer
scale_max	Logical vector of length 1. If TRUE maximum values within each nested layer will be scaled to take up all available space. If FALSE, areas will be comparable between nested layers.
na.rm	Logical vector of length 1 - should missing levels be silently removed?
levels	an integer vector specifying which levels to draw.
...	other arguments passed on to draw

Examples

```
if (require("ggplot2")) {
prodplot(happy, ~ happy, "hbar")
prodplot(happy, ~ happy, "hspine")

prodplot(happy, ~ sex + happy, c("vspine", "hbar"))
prodplot(happy, ~ sex + happy, stacked())

prodplot(happy, ~ happy + sex | health, mosaic("h")) + aes(fill=happy)
# The levels argument can be used to extract a given level of the plot
prodplot(happy, ~ sex + happy, stacked(), level = 1)
prodplot(happy, ~ sex + happy, stacked(), level = 2)
}
```

---

scale_x_product *Generate an x-scale for ggplot2 graphics.*

Description

Generate an x-scale for ggplot2 graphics.

Usage

```
scale_x_product(df)
```

Arguments

df	list of data frame produced by prodcalc, formula and divider

scale_y_product *Generate a y-scale for ggplot2 graphics.*

Description

Generate a y-scale for ggplot2 graphics.

Usage

```
scale_y_product(df)
```

Arguments

df	list of data frame produced by prodcalc, formula and divider

spine *Spine partition: divide longest dimesion.*

Description

Spine partition: divide longest dimesion.

Usage

```
spine(data, bounds, offset = 0.01, max = NULL)
```

Arguments

data	bounds data frame
bounds	bounds of space to partition
offset	space between spines
max	maximum value

stacked — *Template for a stacked bar chart. A stacked bar chart starts with a bar and then continues with spines in the opposite direction.*

Description

Template for a stacked bar chart. A stacked bar chart starts with a bar and then continues with spines in the opposite direction.

Usage

```
stacked(direction = "h")
```

Arguments

direction	direction of first split

tile — *Tree map partitioning.*

Description

Adapated from SquarifiedLayout in http://www.cs.umd.edu/hcil/treemap-history/Treemaps-Java-Algorithms.zip

Usage

```
tile(data, bounds, max = 1)
```

Arguments

data	bounds data frame
bounds	bounds of space to partition
max	maximum value

vbar *Vertical bar partition: height constant, width varies.*

Description

Vertical bar partition: height constant, width varies.

Usage

```
vbar(data, bounds, offset = 0.02, max = NULL)
```

Arguments

data	bounds data frame
bounds	bounds of space to partition
offset	space between spines
max	maximum value

vspine *Vertical spine partition: width constant, height varies.*

Description

Vertical spine partition: width constant, height varies.

Usage

```
vspine(data, bounds, offset = 0.01, max = NULL)
```

Arguments

data	bounds data frame
bounds	bounds of space to partition
offset	space between spines
max	maximum value

Index