Package ‘ggborderline’

August 9, 2021

URL https://github.com/wurli/ggborderline,
     https://wurli.github.io/ggborderline/

Type Package
Title Line Plots that Pop
Version 0.1.0
Author Jacob Scott
Maintainer Jacob Scott <jscott2718@gmail.com>
Description A set of geometries to make line plots a little bit nicer. Use along with ‘ggplot2’ to:
               - Improve the clarity of line plots with many overlapping lines
               - Draw more realistic worms.
License MIT + file LICENSE
Encoding UTF-8
LazyData false
Imports ggplot2, scales
RoxygenNote 7.1.1
NeedsCompilation no
Repository CRAN
Date/Publication 2021-08-09 07:00:02 UTC

`R` topics documented:

draw_key_borderpath .................................................. 2
GeomBorderpath .......................................................... 2
geom_borderpath ......................................................... 3
scale_bordercolour_continuous ....................................... 5

Index 7

1
draw_key_borderpath  Key glyphs for legends

Description
Key glyphs for legends

Usage
draw_key_borderpath(data, params, size)

Arguments
data, params, size
See ggplot2::draw_key_path() for usage

Value
A gTree object

GeomBorderpath  ggborderlines extensions to ggplot2

Description
ggborderlines makes use of the ggproto class system to extend the functionality of ggplot2. In general the actual classes should be of little interest to users as the standard ggplot2 api of using geom_* and stat_* functions for building up the plot is encouraged.

Usage
GeomBorderpath
GeomBorderline
GeomBorderstep

Format
An object of class GeomBorderpath (inherits from GeomPath, Geom, ggproto, gg) of length 4.
An object of class GeomBorderline (inherits from GeomBorderpath, GeomPath, Geom, ggproto, gg) of length 4.
An object of class GeomBorderstep (inherits from GeomBorderpath, GeomPath, Geom, ggproto, gg) of length 2.
Description

This set of geoms is very similar to `ggplot2::geom_path()`, `ggplot2::geom_line()` and `ggplot2::geom_step()`, with the only difference being that they accept two additional aesthetics, `bordercolour` and `bordersize`. For additional documentation, please refer to the `ggplot2` geoms.

Usage

```r
geom_borderpath(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  ...,
  lineend = "butt",
  linejoin = "round",
  linemitre = 10,
  arrow = NULL,
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE
)
```

```r
gem_borderline(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  ...,
  lineend = "butt",
  linejoin = "round",
  linemitre = 10,
  arrow = NULL,
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = NA
)
```

```r
gem_borderstep(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  direction = "hv",
  ...,
  lineend = "butt",
  linejoin = "round",
  linemitre = 10,
  arrow = NULL,
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE
)
```
Arguments

mapping Set of aesthetic mappings created by \texttt{aes()} or \texttt{aes()}. If specified and \texttt{inherit.aes} = \texttt{TRUE} (the default), it is combined with the default mapping at the top level of the plot. You must supply \texttt{mapping} if there is no plot mapping.

data The data to be displayed in this layer. There are three options:
If \texttt{NULL}, the default, the data is inherited from the plot data as specified in the call to \texttt{ggplot()}. A \texttt{data.frame}, or other object, will override the plot data. All objects will be fortified to produce a data frame. See \texttt{fortify()} for which variables will be created.
A function will be called with a single argument, the plot data. The return value must be a \texttt{data.frame}, and will be used as the layer data. A function can be created from a formula (e.g. \texttt{~ head(.x,10)}).

stat The statistical transformation to use on the data for this layer, as a string.

position Position adjustment, either as a string, or the result of a call to a position adjustment function.

... Other arguments passed on to \texttt{layer()}. These are often aesthetics, used to set an aesthetic to a fixed value, like \texttt{colour = "red"} or \texttt{size = 3}. They may also be parameters to the paired geom/stat.

lineend Line end style (round, butt, square).

linejoin Line join style (round, mitre, bevel).

linemitre Line mitre limit (number greater than 1).

arrow Arrow specification, as created by \texttt{grid::arrow()}.

na.rm If \texttt{FALSE}, the default, missing values are removed with a warning. If \texttt{TRUE}, missing values are silently removed.

show.legend logical. Should this layer be included in the legends? \texttt{NA}, the default, includes if any aesthetics are mapped. \texttt{FALSE} never includes, and \texttt{TRUE} always includes. It can also be a named logical vector to finely select the aesthetics to display.

inherit.aes If \texttt{FALSE}, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn’t inherit behaviour from the default plot specification, e.g. \texttt{borders()}.

direction direction of stairs: ‘vh’ for vertical then horizontal, ‘hv’ for horizontal then vertical, or ’mid’ for step half-way between adjacent x-values.

Value

A \texttt{ggproto} layer object
Examples

```r
require(ggplot2)

ggplot(economics_long, aes(date, value01, colour = variable)) +
  geom_borderline()

# You can control the size and colour of the border with the
# bordersize and bordercolour aesthetics:
  ggplot(economics_long, aes(date, value01, bordercolour = variable)) +
  geom_borderline(bordersize = .4, colour = "white")

# The background 'border' part of the geom is always solid, however this
# can be used to create some nice effects:
x <- seq(0, 4 * pi, length.out = 500)
test_data <- data.frame(
  x = rep(x, 2), y = c(sin(x), cos(x)),
  fun = rep(c("sin", "cos"), each = 500)
)
  ggplot(test_data, aes(x, y, colour = fun)) +
  geom_borderline(size = 1, linetype = "dashed", lineend = "round")
```

scale_bordercolour_continuous

Scales for borderlines

Description

These scales control the size and colour of the borders in borderlines. They work in much the same way as `ggplot2::scale_colour_continuous()`, `ggplot2::scale_size_discrete()`, etc.

Usage

```r
scale_bordercolour_continuous(..., aesthetics = "bordercolour")

scale_bordercolour_discrete(..., aesthetics = "bordercolour")

scale_bordersize_continuous(..., aesthetics = "bordersize")

scale_bordersize_discrete(..., aesthetics = "bordersize")
```

Arguments

- `...`: Passed to the corresponding `ggplot2` scales
- `aesthetics`: Character string or vector of character strings listing the name(s) of the aesthetic(s) that this scale works with. This can be useful, for example, to apply colour settings to the bordercolour and colour aesthetics at the same time, via aesthetics = c("bordercolour", "colour").
Value

A ggproto scale object
Index

* datasets
  - GeomBorderpath, 2

aes(). 4
aes_(). 4
borders(). 4
draw_key_borderpath, 2
fortify(). 4

geom_borderline (geom_borderpath), 3
geom_borderpath, 3
geom_borderstep (geom_borderpath), 3
GeomBorderline (GeomBorderpath), 2
GeomBorderpath, 2
GeomBorderstep (GeomBorderpath), 2
ggborderline-extensions
  - (GeomBorderpath), 2
ggplot(). 4
ggrid::arrow(), 4
layer(), 4

scale_bordercolour_continuous, 5
scale_bordercolour_discrete
  - (scale_bordercolour_continuous), 5
scale_bordersize_continuous
  - (scale_bordercolour_continuous), 5
scale_bordersize_discrete
  - (scale_bordercolour_continuous), 5