Package 'ggglyph'

June 8, 2021

Title Multivariate Data Visualization using Glyphs

Version 0.0.0.9000

Description Provides geoms for visualizing multivariate data using 'ggplot2'.

License GPL-2 | GPL-3

Encoding UTF-8

RoxygenNote 7.1.1

LazyData true

RdMacros mathjaxr,
      Rdpack

Depends R (>= 3.5.0)

Imports mathjaxr,
      Rdpack

R topics documented:

---

geom_starglyph   *Add Star Glyphs as a Scatterplot*

Description

The starglyph geom is used to plot multivariate data as star glyphs (Siegel et al. 1972; Chambers et al. 1983) in a scatterplot.

Usage

```
geom_starglyph(
  mapping = NULL,
  data,
  stat = "identity",
  position = "identity",
  ...,
```

```
    cols = character(0L),
    whisker = TRUE,
    contour = TRUE,
    linewidth = 1,
    show.legend = NA,
    inherit.aes = TRUE
)
```

Arguments

mapping	Set of aesthetic mappings created by [aes()](#) or [aes_()](#). If specified and inherit.aes = TRUE (the default), it is combined with the default mapping at the top level of the plot. You must supply mapping if there is no plot mapping.
data	The data to be displayed in this layer. There are three options:
	If NULL, the default, the data is inherited from the plot data as specified in the call to [ggplot()](#).
	A data.frame, or other object, will override the plot data. All objects will be fortified to produce a data frame. See [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 data.frame, and will be used as the layer data. A function can be created from a formula (e.g. ~ 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.
cols	Name of columns specifying the variables to be plotted in the glyphs as a character vector.
whisker	logical. If TRUE, plots the star glyph whiskers.
contour	logical. If TRUE, plots the star glyph contours. glyph.
linewidth	The line width.
show.legend	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes. It can also be a named logical vector to finely select the aesthetics to display.
inherit.aes	If 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. [borders()](#).

Value

A geom layer.

Aesthetics

geom_starglyph() understands the following aesthetics (required aesthetics are in bold):

- **x**
- **y**
- alpha
- colour

- fill
- group
- shape
- size
- stroke
- linetype

References

Chambers JM, Cleveland WS, Kleiner B, Tukey PA (1983). *Graphical Methods for Data Analysis*. Chapman and Hall/CRC, Boca Raton. ISBN 978-1-351-07230-4.

Siegel JH, Farrell EJ, Goldwyn RM, Friedman HP (1972). "The surgical implications of physiologic patterns in myocardial infarction shock." *Surgery*, **72**(1), 126–141.

See Also

[starglyphGrob](#)

Examples

```
# Scale the data
zs <- c("hp", "drat", "wt", "qsec", "vs", "am", "gear", "carb")
mtcars[ , zs] <- lapply(mtcars[ , zs], scales::rescale)

mtcars$cyl <- as.factor(mtcars$cyl)
mtcars$lab <- row.names(mtcars)

library(ggplot2)

# Both whiskers and contour
ggplot() +
  geom_starglyph(data = mtcars, aes(x = mpg, y = disp, fill = cyl),
                 cols = zs, whisker = TRUE, contour = TRUE,
                 size = 0.1, alpha =  0.5) +
  ylim(c(-0, 550))

# Only contours (polygon)
ggplot() +
  geom_starglyph(data = mtcars, aes(x = mpg, y = disp, fill = cyl),
                 cols = zs, whisker = FALSE, contour = TRUE,
                 size = 0.1, alpha =  0.5) +
  ylim(c(-0, 550))

# Only whiskers
ggplot() +
  geom_starglyph(data = mtcars, aes(x = mpg, y = disp, colour = cyl),
                 cols = zs, whisker = TRUE, contour = FALSE,
                 size = 0.1) +
  geom_point(data = mtcars, aes(x = mpg, y = disp, colour = cyl)) +
  ylim(c(-0, 550))

# With text annotations
ggplot() +
```

```
    geom_starglyph(data = mtcars, aes(x = mpg, y = disp, colour = cyl),
                   cols = zs, whisker = TRUE, contour = FALSE,
                   size = 0.1) +
    geom_point(data = mtcars, aes(x = mpg, y = disp, colour = cyl)) +
    geom_text(data = mtcars, aes(x = mpg, y = disp, label = lab), cex = 2) +
    ylim(c(-0, 550))

  # Faceted
  ggplot() +
    geom_starglyph(data = mtcars, aes(x = mpg, y = disp, fill = cyl),
                   cols = zs, whisker = TRUE, contour = TRUE,
                   size = 0.1, alpha =  0.5) +
    ylim(c(-0, 550)) +
    facet_grid(. ~ cyl)
```

---

starglyphGrob   *Draw a Star Glyph*

Description

Uses [Grid](#) graphics to draw a star glyph (Siegel et al. 1972; Chambers et al. 1983).

Usage

```
starglyphGrob(
  x = 0.5,
  y = 0.5,
  z,
  size = 1,
  col = "black",
  fill = NA,
  lwd = 1,
  alpha = 1,
  angle.start = 0,
  angle.stop = 2 * base::pi,
  whisker = TRUE,
  contour = TRUE
)
```

Arguments

x	The horizontal position.
y	The vertical position.
z	A numeric vector specifying the distance of star gylph points from the center.
size	The size of glyphs.
col	The colour of whisker and contours.
fill	The fill colour.
lwd	The line width.
alpha	The alpha transparency value.
angle.start	The start angle for the glyph in radians. Default is zero.

angle.stop	The stop anlge for the glyph in radians. Default is 2π
whisker	logical. If TRUE, plots the star glyph whiskers.
contour	logical. If TRUE, plots the star glyph contours. glyph.

Value

A [grobTree](#) object.

References

Chambers JM, Cleveland WS, Kleiner B, Tukey PA (1983). *Graphical Methods for Data Analysis*. Chapman and Hall/CRC, Boca Raton. ISBN 978-1-351-07230-4.

Siegel JH, Farrell EJ, Goldwyn RM, Friedman HP (1972). "The surgical implications of physiologic patterns in myocardial infarction shock." *Surgery*, **72**(1), 126–141.

See Also

[geom_starglyph](#)

Examples

```
stargylph <- starglyphGrob(x = 250, y = 250,
                           z = c(0.24, 0.3, 0.8, 1.4, 0.6, 0.33), size = 100)
grid::grid.newpage()
grid::grid.draw(stargylph)
```

Index