

Package ‘scatterD3’

March 10, 2020

Type Package

Title D3 JavaScript Scatterplot from R

Version 0.9.1

Date 2020-03-10

Maintainer Julien Barnier <julien.barnier@cnrs.fr>

Description Creates 'D3' 'JavaScript' scatterplots from 'R' with interactive features : panning, zooming, tooltips, etc.

License GPL (>= 3)

VignetteBuilder knitr

Encoding UTF-8

URL <https://juba.github.io/scatterD3/>

BugReports <https://github.com/juba/scatterD3/issues>

LazyData TRUE

Enhances shiny

Imports htmlwidgets, digest, ellipse

Suggests knitr, rmarkdown

RoxygenNote 7.0.2

NeedsCompilation no

Author Julien Barnier [aut, cre],
Kent Russell [aut, ctb],
Mike Bostock [aut, cph] (d3.js library, <http://d3js.org>),
Susie Lu [aut, cph] (d3-legend library, <http://d3-legend.susielu.com/>),
Speros Kokenes [aut, cph] (d3-lasso-plugin library,
<https://github.com/skokenes/D3-Lasso-Plugin>),
Evan Wang [aut, cph] (d3-labeler plugin,
<https://github.com/tinker10/D3-Labeler>)

Repository CRAN

Date/Publication 2020-03-10 12:30:02 UTC

R topics documented:

scatterD3	2
scatterD3-shiny	6

Index	7
--------------	----------

scatterD3	<i>Scatter plot HTML widget</i>
-----------	---------------------------------

Description

Generates an interactive scatter plot based on d3.js. Interactive features include zooming, panning, text labels moving, tooltips, fading effects in legend. Additional handlers are provided to change label size, point opacity or export the figure as an SVG file via HTML form controls.

Usage

```
scatterD3(
  x,
  y,
  data = NULL,
  lab = NULL,
  x_log = FALSE,
  y_log = FALSE,
  point_size = 64,
  labels_size = 10,
  labels_positions = NULL,
  point_opacity = 1,
  opacities = NULL,
  hover_size = 1,
  hover_opacity = NULL,
  fixed = FALSE,
  col_var = NULL,
  col_continuous = NULL,
  colors = NULL,
  ellipses = FALSE,
  ellipses_level = 0.95,
  symbol_var = NULL,
  symbols = NULL,
  size_var = NULL,
  size_range = c(10, 300),
  sizes = NULL,
  col_lab = NULL,
  symbol_lab = NULL,
  size_lab = NULL,
  key_var = NULL,
  type_var = NULL,
```

```

opacity_var = NULL,
unit_circle = FALSE,
url_var = NULL,
tooltips = TRUE,
tooltip_text = NULL,
tooltip_position = "bottom right",
xlab = NULL,
ylab = NULL,
html_id = NULL,
width = NULL,
height = NULL,
legend_width = 150,
left_margin = 30,
xlim = NULL,
ylim = NULL,
dom_id_reset_zoom = "scatterD3-reset-zoom",
dom_id_svg_export = "scatterD3-svg-export",
dom_id_lasso_toggle = "scatterD3-lasso-toggle",
transitions = FALSE,
menu = TRUE,
lasso = FALSE,
lasso_callback = NULL,
click_callback = NULL,
init_callback = NULL,
zoom_callback = NULL,
zoom_on = NULL,
zoom_on_level = NULL,
disable_wheel = FALSE,
lines = data.frame(slope = c(0, Inf), intercept = c(0, 0), stroke_dasharray = c(5, 5)),
axes_font_size = "100%",
legend_font_size = "100%",
caption = NULL
)

```

Arguments

x	numerical vector of x values, or variable name if data is not NULL
y	numerical vector of y values, or variable name if data is not NULL
data	default dataset to use for plot.
lab	optional character vector of text labels, or variable name if data is not NULL
x_log	if TRUE, set x scale as logarithmic
y_log	if TRUE, set y scale as logarithmic
point_size	points size. Ignored if size_var is not NULL.
labels_size	text labels size
labels_positions	Either a data frame, as created by the "Export labels positions" menu entry, giving each label x and y position, or the value "auto" to use an automatic labeler.

<code>point_opacity</code>	points opacity, as an integer (same opacity for all points).
<code>opacities</code>	named list or named vector of opacities. Each opacity will be associated by their name within <code>'opacity_var'</code> .
<code>hover_size</code>	factor for changing size when hovering points
<code>hover_opacity</code>	points opacity when hovering
<code>fixed</code>	force a 1:1 aspect ratio
<code>col_var</code>	optional vector for points color mapping, or variable name if data is not NULL
<code>col_continuous</code>	specify if the color scale must be continuous. By default, if <code>col_var</code> is numeric, not a factor, and has more than 6 unique values, it is considered as continuous.
<code>colors</code>	vector of custom points colors. Colors must be defined as a hexadecimal string (eg "#FF0000"). If <code>colors</code> is a named list or a named vector, then the colors will be associated with their name within <code>col_var</code> . For a continuous color scale, can be a string giving the interpolate function name from <code>d3-scale-chromatic</code> (for example, "interpolatePurples")
<code>ellipses</code>	draw confidence ellipses for points or the different color mapping groups
<code>ellipses_level</code>	confidence level for ellipses (0.95 by default)
<code>symbol_var</code>	optional vector for points symbol mapping, or variable name if data is not NULL
<code>symbols</code>	vector of custom points symbols. Symbols must be defined as character strings with the following possible values : "circle", "cross", "diamond", "square", "star", "triangle", and "wye". If <code>symbols</code> is a named list or a named vector, then the symbols will be associated with their name within <code>symbol_var</code> .
<code>size_var</code>	optional vector for points size mapping, or variable name if data is not NULL
<code>size_range</code>	numeric vector of length 2, giving the minimum and maximum point sizes when mapping with <code>size_var</code>
<code>sizes</code>	named list or named vector of sizes. Each size will be associated by their name within <code>'size_var'</code> .
<code>col_lab</code>	color legend title. Set to NA to remove color legend entirely.
<code>symbol_lab</code>	symbols legend title. Set to NA to remove symbol legend entirely.
<code>size_lab</code>	size legend title. Set to NA to remove size legend entirely.
<code>key_var</code>	optional vector of rows ids, or variable name if data is not NULL. This is passed as a key to d3, and is only added in shiny apps where displayed rows are filtered interactively.
<code>type_var</code>	optional vector of points type : "point" for a dot (default), "arrow" for an arrow starting from the origin.
<code>opacity_var</code>	optional vector of points opacity (values between 0 and 1)
<code>unit_circle</code>	set tot TRUE to draw a unit circle
<code>url_var</code>	optional vector of URLs to be opened when a point is clicked
<code>tooltips</code>	logical value to display tooltips when hovering points
<code>tooltip_text</code>	optional character vector of tooltips text
<code>tooltip_position</code>	the tooltip position relative to its point. Must a combination of "top" or "bottom" with "left" or "right" (default is "bottom right").

xlab	x axis label
ylab	y axis label.
html_id	manually specify an HTML id for the svg root node. A random one is generated by default.
width	figure width, computed when displayed
height	figure height, computed when displayed
legend_width	legend area width, in pixels. Set to 0 to disable legend completely.
left_margin	margin on the left of the plot, in pixels
xlim	numeric vector of length 2, manual x axis limits
ylim	numeric vector of length 2, manual y axis limits
dom_id_reset_zoom	HTML DOM id of the element to bind the "reset zoom" control to.
dom_id_svg_export	HTML DOM id of the element to bind the "svg export" control to.
dom_id_lasso_toggle	HTML DOM id of the element to bind the "toggle lasso" control to.
transitions	if TRUE, data updates are displayed with smooth transitions, if FALSE the whole chart is redrawn. Only used within shiny apps.
menu	wether to display the tools menu (gear icon)
lasso	logical value to add https://github.com/skokenes/D3-Lasso-Plugin feature
lasso_callback	the body of a JavaScript callback function with the argument sel to be applied to a lasso plugin selection
click_callback	the body of a JavaScript callback function whose inputs are html_id, and the index of the clicked element.
init_callback	the body of a JavaScript callback function applied to the scatter object at init time.
zoom_callback	the body of a JavaScript callback function whose inputs are the new xmin, xmax, ymin and ymax after a zoom action is triggered.
zoom_on	coordinates where to center zoom on plot draw or update.
zoom_on_level	zoom level on plot draw or update. Ignored if 'zoom_on' is NULL.
disable_wheel	if TRUE, disable zooming with mousewheel.
lines	a data frame with at least the slope and intercept columns, and as many rows as lines to add to scatterplot. Style can be added with stroke, stroke_width and stroke_dasharray columns. To draw a vertical line, pass Inf as slope value.
axes_font_size	font size for axes text (any CSS compatible value)
legend_font_size	font size for legend text (any CSS compatible value)
caption	caption to be displayed when clicking on the corresponding icon. Either a character string, or a list with title, subtitle and text elements.

Details

Interactive scatter plots based on htmlwidgets and d3.js

Author(s)

Julien Barnier <julien.barnier@ens-lyon.fr>

Source

D3.js was created by Michael Bostock. See <http://d3js.org/>

Examples

```
scatterD3(x = mtcars$wt, y = mtcars$mpg, data=NULL, lab = rownames(mtcars),
          col_var = mtcars$cyl, symbol_var = mtcars$am,
          xlab = "Weight", ylab = "Mpg", col_lab = "Cylinders",
          symbol_lab = "Manual transmission", html_id = NULL)
```

scatterD3-shiny

Shiny bindings for scatterD3 widgets

Description

Output and render functions for using scatterD3 widgets within Shiny applications and interactive Rmd documents.

Usage

```
scatterD3Output(outputId, width = "100%", height = "600px")
```

```
renderScatterD3(expr, env = parent.frame(), quoted = FALSE)
```

Arguments

outputId	output variable to read from
width, height	Must be a valid CSS unit (like "100%", "400px", "auto") or a number, which will be coerced to a string and have "px" appended.
expr	An expression that generates a scatterD3 scatter plot.
env	The environment in which to evaluate expr.
quoted	Is expr a quoted expression (with quote())? This is useful if you want to save an expression in a variable.

Index

`renderScatterD3` (`scatterD3-shiny`), [6](#)

`scatterD3`, [2](#)

`scatterD3-shiny`, [6](#)

`scatterD3output` (`scatterD3-shiny`), [6](#)