Package 'canvasXpress'

March 10, 2023

Version 1.42.9

Title Visualization Package for CanvasXpress in R

Description Enables creation of visualizations using the CanvasXpress framework in R. CanvasXpress is a standalone JavaScript library for reproducible research with complete tracking of data and end-user modifications stored in a single PNG image that can be played back. See <https://www.canvasxpress.org> for more information.

Type Package

License GPL-3

Encoding UTF-8

Language en-US

URL https://github.com/neuhausi/canvasXpress

BugReports https://github.com/neuhausi/canvasXpress/issues

Depends R (>= 3.5)

Imports htmlwidgets (>= 1.0), htmltools, httr, jsonlite, stats

RoxygenNote 7.2.3

Suggests shiny (>= 1.1.0), canvasXpress.data, dplyr, glue, grid, knitr, png, readr, rlang, rmarkdown, stringr, testthat, tibble, tidyr, limma, ggplot2

VignetteBuilder knitr

NeedsCompilation no

Author Isaac Neuhaus [aut], Connie Brett [aut, cre]

Maintainer Connie Brett <connie@aggregate-genius.com>

Repository CRAN

Date/Publication 2023-03-10 21:10:02 UTC

R topics documented:

`canvasXpress-package` *CanvasXpress Visualization Package*

Description

A package to assist in creating visualizations in CanvasXpress in R.

Details

CanvasXpress is a standalone JavaScript library for reproducible research with complete tracking of data and end-user modifications stored in a single PNG image that can be played back for an extensive set of visualizations.

More Information

https://www.canvasxpress.org

```
browseVignettes(package = "canvasXpress")
```

Author(s)

Maintainer: Connie Brett <connie@aggregate-genius.com>

Authors:

- Isaac Neuhaus <imnphd@gmail.com>

See Also

Useful links:

- https://github.com/neuhausi/canvasXpress
- Report bugs at https://github.com/neuhausi/canvasXpress/issues

canvasXpress *HTML Widget Creation*

Description

Custom HTML widget creation function based on widget YAML and JavaScript for use in any html-compatible context

Usage

```
canvasXpress(
  data = NULL,
  smpAnnot = NULL,
  varAnnot = NULL,
  graphType = "Scatter2D",
  events = NULL,
  afterRender = NULL,
  pretty = FALSE,
  digits = 4,
  width = 600,
  height = 400,
  destroy = FALSE,
  ...
)
```

Arguments

data	data.frame-, matrix-, or list- classed data object
smpAnnot	additional data that applies to samples (columns)
varAnnot	additional data that applies to variables (rows)
graphType	type of graph to be plotted - default = "Scatter2D"
events	user-defined events (e.g. mousemove, mouseout, click and dblclick)
afterRender	event triggered after rendering
pretty	print tagged code (JSON/HTML) nicely - default = FALSE
digits	display digits - default = 4
width	plot width (valid CSS units) - default = 600px
height	plot height (valid CSS units) - default = 400px
destroy	used to indicate removal of a plot - default = FALSE
...	additional parameters passed to canvasXpress

Value

htmlwidgets object

`canvasXpress.json` *HTML Widget Creation using JSON input*

Description

Custom HTML widget creation function based on widget YAML and JavaScript for use in any html-compatible context using raw JSON input. Validation of data and configuration is deferred completely to the canvasXpress JavaScript library.

Usage

```
canvasXpress.json(
  json,
  pretty = FALSE,
  digits = 4,
  width = 600,
  height = 400,
  destroy = FALSE
)
```

Arguments

`json`	JSON string or object
`pretty`	print tagged code (JSON/HTML) nicely - default = FALSE
`digits`	display digits - default = 4
`width`	plot width (valid CSS units) - default = 600px
`height`	plot height (valid CSS units) - default = 400px
`destroy`	used to indicate removal of a plot - default = FALSE

Details

For the formatting of the JSON input object see

Note: this function is intended for use by advanced users who are experimenting with or need to utilize the json-formatted input to canvasXpress and are comfortable debugging chart issues in a browser (JavaScript) context instead of in R.

Value

htmlwidgets object

More Information

https://www.canvasxpress.org

Examples

```
my_json <- '{ "data": {"y": { "vars": ["Performance"],
                              "smps": ["January"],
                              "data": [[85]] }},
              "config": { "graphType": "Meter",
                          "meterType": "gauge" }}'

canvasXpress.json(my_json)
```

---

canvasXpressOutput *Shiny UI function*

Description

Output creation function for canvasXpressOutput in Shiny applications and interactive Rmd documents

Usage

```
canvasXpressOutput(outputId, width = "100%", height = "400px")
```

Arguments

outputId	shiny unique ID
width	width of the element - default = 100%
height	height of the element - default = 400px

Value

Output function that enables the use of the widget in applications

See Also

renderCanvasXpress

cxShinyExample

cxHtmlPage *Stand-Alone HTML Page Creation*

Description

This function creates and returns a stand-alone HTML page containing the given canvasXpress object. Width and height can be inferred from the canvasXpress object (default) or overridden for the page output.

Usage

```
cxHtmlPage(chartObject, width = NULL, height = NULL)
```

Arguments

chartObject	a canvasXpress plot object
width	plot width override for the HTML page (valid CSS units) - default = NULL
height	plot height override for the HTML page (valid CSS units) - default = NULL

Value

a character string containing a self-contained html page

Examples

```
## Not run:
my_chart <- canvasXpress(data      = data.frame(Sample1 = c(33, 48),
                                                Sample2 = c(44, 59),
                                                Sample3 = c(55, 6)),
                         graphType = "Bar",
                         title     = "Example Bar Chart",
                         width     = "600px")

# create a page using the chart dimensions on my_chart
html_page <- cxHtmlPage(my_chart)

# or change the chart width/height for this page:
html_page <- cxHtmlPage(my_chart, width = "100%", height = "70vh")

# save page for viewing/sharing
writeLines(html_page, tempfile(fileext = ".html"))

## End(Not run)
```

cxShinyExample *Create Shiny Example Application*

Description

This function runs one of the available shiny example applications. To see the list of available example applications run the function with no inputs

Usage

```
cxShinyExample(example = NULL)
```

Arguments

example	character name of a valid example application.

Value

Launches a running shiny example application

See Also

canvasXpressOutput

renderCanvasXpress

renderCanvasXpress *Shiny Render function*

Description

Render function for canvasXpressOutput in Shiny applications and interactive Rmd documents

Usage

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

Arguments

expr	expression used to render the canvasXpressOutput
env	environment to use - default = parent.frame()
quoted	whether the expression is quoted - default = FALSE

Value

Render function that enables the use of the widget in applications

Destroy

When there exists a need to visually remove a plot from a Shiny application when it is not being immediately replaced with a new plot use the destroy option as in:

```
renderCanvasXpress({canvasXpress(destroy = TRUE)})
```

See Also

canvasXpressOutput

cxShinyExample

Index