Package ‘c3’

May 29, 2018

Type Package
Title 'C3.js' Chart Library
Description Create interactive charts with the 'C3.js' charting library. All plot types in 'C3.js' are available and include line, bar, scatter, and mixed geometry plots. Plot annotations, labels and axis are highly adjustable. Interactive web based charts can be embedded in R Markdown documents or Shiny web applications.

Version 0.2.0
Maintainer Matt Johnson <mrjoh3@gmail.com>
Depends R (>= 3.2.2)
Imports jsonlite, data.table, lazyeval, htmlwidgets, dplyr, viridis
URL https://github.com/mrjoh3/c3
License GPL (>= 3)
LazyData TRUE
Encoding UTF-8
RoxygenNote 6.0.1
Suggests testthat, RColorBrewer, knitr, rmarkdown, webshot, purrr
VignetteBuilder knitr
NeedsCompilation no
Author Matt Johnson [aut, cre]
Repository CRAN
Date/Publication 2018-05-29 10:20:57 UTC

R topics documented:

  c3 ....................................................... 2
  c3-shiny ............................................ 3
  c3_bar ............................................... 4
  c3_chart_size ...................................... 5
Description

An ‘R’ wrapper, or htmlwidget, for the c3 javascript charting library by Masayuki Tanaka.

Usage

c3(data, x = NULL, y = NULL, group = NULL, width = NULL, height = NULL, axes = NULL, labels = NULL, hide = NULL, onclick = NULL, onmouseover = NULL, onmouseout = NULL, ...)

Arguments

data        data.frame or tibble
x            character column name
y            character column name
group        character column name
width        integer htmlwidget width (separate from plot width)
height       integer htmlwidget height (separate from plot height)
c3-shiny

Shiny bindings for c3

Description
Output and render functions for using c3 within Shiny applications and interactive Rmd documents.

Usage

```r
c3Output(outputId, width = "100\%", height = "100\%")
renderC3(expr, env = parent.frame(), quoted = FALSE)
```

See Also
Other c3: `RColorBrewer`, `grid`, `legend`, `region`, `subchart`, `tooltip`, `xAxis`, `zoom`
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 c3
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.

c3_bar Bar Plot

Description

Add bars to a C3 plot

Usage

c3_bar(c3, stacked = FALSE, rotated = FALSE, bar_width = 0.6, zerobased = TRUE)

Arguments

c3 c3 htmlwidget object
stacked boolean place bars on top of each other
rotated boolean use to make x-axis vertical
bar_width numeric pixel width of bars
zerobased boolean

Value

c3

Examples

data.frame(a = c(1, 2, 3, 2), b = c(2, 3, 1, 5)) %>%
c3() %>%
c3_bar(stacked = TRUE)
Description

Modify the size of the chart within the htmlwidget area. Generally charts size to the div in which they are placed. These options enable finer scale sizing with the div.

Usage

c3_chart_size(c3, left = NULL, right = NULL, top = NULL, bottom = NULL, width = NULL, height = NULL, ...)

Arguments

c3 c3 htmlwidget object
left integer padding pixels
right integer padding pixels
top integer padding pixels
bottom integer padding pixels
width integer pixels
height integer pixels
... additional options passed to the padding and size objects

Value

c3

Examples

data.frame(a = c(1,2,3,2), b = c(2,4,1,5)) %>%
c3() %>%
c3_chart_size(width = 600, height = 200)
c3_color

Description
Manually assign colors

Usage
c3_color(c3, colors)

Arguments
- c3: c3 htmlwidget object
- colors: character vector of colors

Value
c3

Examples
data.frame(a = c(1,2,3,2), b = c(2,4,1,5)) %>%
c3() %>%
c3_color(c('red','black'))

c3_colour

Description
Manually assign colours

Usage
c3_colour(c3, colours)

Arguments
- c3: c3 htmlwidget object
- colours: character vector of colours

Value
c3
Examples

data.frame(a = c(1L, 2, 3, 2), b = c(2, 4, 1, 5)) %>%
c3() %>%
c3_colour(c('red', 'black'))

data.frame(red = 20, green = 45, blue = 10) %>%
c3() %>%
c3_donut(title = 'Colors')

Description

Create simple Donut charts

Usage

c3_donut(c3, expand = TRUE, title = NULL, width = NULL, show = TRUE,
threshold = NULL, format = NULL, ...)

Arguments

c3 c3 htmlwidget object
expand boolean expand segment on hover
title character
width integer pixels width of donut
show boolean show labels
threshold numeric proportion of segment to hide label
format character label js function, wrap character or character vector in JS()
... additional values passed to the donut label object

Value

c3

Examples

data.frame(a = c(1L, 2, 3, 2), b = c(2, 4, 1, 5)) %>%
c3() %>%
c3_colour(c('red', 'black'))

data.frame(red = 20, green = 45, blue = 10) %>%
c3() %>%
c3_donut(title = 'Colors')
c3\_gauge

Gauge Charts

Description

Create simple Guage Charts

Usage

c3\_gauge(c3, label = NULL, min = 0, max = 100, units = NULL, width = NULL, pattern = c("#FF0000", ","#F97600", ","#F6C600", ","#60B044"), threshold = list(unit = "value", max = 100, values = c(30, 60, 90, 100)), height = NULL, ...)

Arguments

c3 c3 htmlwidget object
label list with options:
  • show: boolean
  • format: function, wrap in JS()
min numeric
max numeric
units character appended to numeric value
width integer pixel width of the arc
pattern character vector or pallete of colors
threshold list with options:
  • unit: character one of 'percent', 'value'
  • max: numeric
  • values: numeric vector of threshold values for color change
height integer pixel height of the chart. Proprtion of guage never changes so height scales with width despite this setting.
... additional values passed to the guage, color and size objects

Value

c3

Examples

data.frame(data=10) %>%
c3() %>%
c3\_gauge(title = 'Colors')
**Description**

Add lines to a C3 plot

**Usage**

```r
c3_line(c3, type, stacked = FALSE, connectNull = FALSE, step_type = NULL)
```

**Arguments**

- **c3**: c3 htmlwidget object
- **type**: character type of line plot. Must be one of:
  - line
  - spline
  - step
  - area
  - area-step
- **stacked**: boolean
- **connectNull**: boolean connect null (missing) data points
- **step_type**: character, one of:
  - step
  - step-after
  - step-before

**Value**

`c3`

**Examples**

```r
data.frame(a=c(1,2,3,2),b=c(2,3,1,5)) %>%
c3() %>%
c3_line('spline')
```
c3_mixedGeom

Mixed Geometry Plots

Description

Use multiple geometry types in a single plot.

Usage

```r
c3_mixedGeom(c3, types, type = "line", stacked = NULL)
```

Arguments

- **c3**: c3 htmlwidget object.
- **types**: list containing key value pairs of column header and plot type.
- **type**: character default plot type where not defined.
- **stacked**: character vector of column headers to stack.

Value

- **c3**

Examples

```r
data <- data.frame(a = abs(rnorm(20) * 10),
                  b = abs(rnorm(20) * 10),
                  c = abs(rnorm(20) * 10),
                  d = abs(rnorm(20) * 10))
data %>%
c3() %>%
c3_mixedGeom(type = 'bar',
              stacked = c('b', 'd'),
              types = list(a='area',
                            c='spline'))
```

c3_pie

Pie Charts

Description

C3 Pie Charts.

Usage

```r
c3_pie(c3, show = TRUE, threshold = NULL, format = NULL, expand = TRUE, ...
```
c3_scatter

Arguments

- `c3`: c3 htmlwidget object
- `show`: boolean show labels
- `threshold`: numeric proportion of segment to hide label
- `format`: character label js function, wrap character or character vector in JS()
- `expand`: boolean expand segment on hover
- `...`: additional values passed to the pie label object

Value

c3

Examples

data.frame(red = 20, green = 45, blue = 10) %>%
c3() %>%
c3_pie()

---

c3_scatter  Scatter Plots

Description

For scatter plots options are defined in the `c3` function. Options are limited to x, y and groups

Usage

c3_scatter(c3)

Arguments

- `c3`: c3 htmlwidget object

Value

c3

Examples

`iris %>%
c3(x = 'Sepal_Length',
y = 'Sepal_Width',
    group = 'Species') %>%
c3_scatter()`
c3_selection  

**Description**

Define options for selecting data within the plot area

**Usage**

```r
c3_selection(c3, enabled = FALSE, grouped = FALSE, multiple = FALSE, 
draggable = FALSE, isselectable = JS("function () { return true; }")), ...)
```

**Arguments**

- `c3`: c3 htmlwidget object
- `enabled`: boolean
- `grouped`: boolean
- `multiple`: boolean
- `draggable`: boolean
- `isselectable`: character js function, wrap character or character vector in JS()
- `...`: additional options passed to data selection object

**Value**

`c3`

**Examples**

```r
data.frame(a = c(1,2,3,2), b = c(2,3,1,5)) %>%
c3() %>%
c3_selection(enabled = TRUE, 
multiple = TRUE)
```

c3_viridis  

**Description**

Use Viridis palette options

**Usage**

```r
c3_viridis(c3, pal = "D")
```
check_stacked

Arguments

- c3: c3 htmlwidget object
- pal: character palette options

Value

- c3

Examples

```r
data.frame(a = c(1,2,3,2), b = c(2,4,1,5)) %>%
c3() %>%
c3_viridis()
```

Description

For plots where stacking is required this function will define the columns to be stacked based on column headers.

Usage

```r
check_stacked(c3, stacked)
```

Arguments

- c3: c3 htmlwidget object
- stacked: boolean

Value

- c3 object
grid

C3 Grid

Description
Modify grid and line elements on both x and y axis

Usage
grid(c3, axis, show = TRUE, lines = NULL, ticks = NULL, ...)

## S3 method for class 'c3'
grid(c3, axis, show = TRUE, lines = NULL, ticks = NULL, ...)

Arguments
c3 c3 htmlwidget object
axis character 'x' or 'y'
show boolean
lines dataframe with options:
  • value: numeric, character or date depending on axis
  • text: character (optional)
  • class: character css class (optional)
  • position: character one of 'start', 'middle', 'end' (optional)
ticks boolean placeholder. Not yet implemented in C3.js
... additional options passed to the grid object

Value
c3

See Also
Other c3: RColorBrewer, c3, legend, region, subchart, tooltip, xAxis, zoom

Examples
iris %>%
c3(x = 'Sepal_Length', y = 'Sepal_Width', group = 'Species') %>%
c3_scatter() %>%
gridscatter('y') %>%
gridscatter('x', show = FALSE, lines = data.frame(value=c(5, 6),
        text = c('Line 1', 'Line 2'))
### legend

**C3 Legend Options**

#### Description

Modify plot elements that relate to the legend. The c3 legend is on by default, this function allows the legend to be removed, or other legend attributes to be set.

#### Usage

```r
legend(c3, hide = FALSE, position = NULL, inset = NULL, item = NULL, ...)
```

```r
## S3 method for class 'c3'
legend(c3, hide = FALSE, position = NULL, inset = NULL, item = NULL, ...)
```

#### Arguments

- **c3**: c3 htmlwidget object
- **hide**: boolean or character of parameters to hide
- **position**: character one of 'bottom', 'right', 'inset'
- **inset**: list with options:
  - anchor: character one of 'top-left', 'top-right', 'bottom-left', 'bottom-right'
  - x: integer pixels
  - y: integer pixels
  - step: numeric
- **item**: list with options:
  - onclick: character js function, wrap character or character vector in JS()
  - onmouseover: character js function, wrap character or character vector in JS()
  - onmouseout: character js function, wrap character or character vector in JS()
  - ... additional options passed to the legend object

#### Value

- **c3**

#### See Also

Other c3: [RColorBrewer](#), [c3.grid](#), [region](#), [subchart](#), [tooltip](#), [xAxis](#), [zoom](#)
Examples

```r
iris %>%
c3(x='Sepal.Length', y='Sepal.Width', group = 'Species') %>%
c3_scatter() %>%
legend(position = 'right')
```

---

<table>
<thead>
<tr>
<th>point_options</th>
<th>Point Options</th>
</tr>
</thead>
</table>

Description

Modify point options

Usage

```r
point_options(c3, show = TRUE, r = 2.5, expand = TRUE, expand.r = 1.75,
select.r = 4)
```

Arguments

- `c3`: c3 htmlwidget object
- `show`: boolean
- `r`: numeric radius of point
- `expand`: boolean
- `expand.r`: numeric multiplier for radius expansion
- `select.r`: numeric multiplier for radius expansion

Value

- `c3`

Examples

```r
data.frame(a = c(1,2,3,2), b = c(2,4,1,5)) %>%
c3() %>%
point_options(r = 5, expand.r = 2)
```
**RColorBrewer RColorBrewer Palette**

**Description**
Use RColorBrewer palettes

**Usage**
```
RColorBrewer(c3, pal = "Spectral")
```

---

**Arguments**
- `c3`: c3 htmlwidget object
- `pal`: character palette must match ‘RColorBrewer::brewer.pal.info’

**Value**
- `c3`

**See Also**
Other c3: `c3, grid, legend, region, subchart, tooltip, xAxis, zoom`

**Examples**
```
data.frame(a = c(1,2,3,2), b = c(2,4,1,5), c = c(5,3,4,1)) %>%
c3() %>%
RColorBrewer()
```

---

**region**

Modify region elements on both x and y axis

**Description**
Regions are defined in multiple axis by passing a single ‘data.frame’

**Usage**
```
region(c3, regions)
```

---

---
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>c3</td>
<td>c3 htmlwidget object</td>
</tr>
<tr>
<td>regions</td>
<td>data.frame with columns listed below. Any columns can be missing but results may be unexpected.</td>
</tr>
<tr>
<td></td>
<td>• axis: character one of 'x', 'y', 'y2'</td>
</tr>
<tr>
<td></td>
<td>• start: numeric but must match defined axis type</td>
</tr>
<tr>
<td></td>
<td>• end: numeric but must match defined axis type</td>
</tr>
<tr>
<td></td>
<td>• class: character css class</td>
</tr>
</tbody>
</table>

Value

c3

See Also

Other c3: RColorBrewer, c3, grid, legend, subchart, tooltip, xAxis, zoom

Examples

iris %>%
c3(x = 'Sepal_Length', y = 'Sepal_Width', group = 'Species') %>%
c3_scatter() %>%
region(data.frame(axis = 'x',
                start = 5,
                end = 6))

Description

Subcharts are defined in multiple axis by passing a single `data.frame`. Subcharts are listed as an experimental feature in the C3 documentation.

Usage

subchart(c3, height = 20, onbrush = NULL)

## S3 method for class 'c3'
subchart(c3, height = 20, onbrush = NULL)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>c3</td>
<td>c3 htmlwidget object</td>
</tr>
<tr>
<td>height</td>
<td>integer pixels</td>
</tr>
<tr>
<td>onbrush</td>
<td>character js function, wrap character or character vector in JS()</td>
</tr>
</tbody>
</table>
**tickAxis**

**Value**

c3

**See Also**

Other c3: RColorBrewer, c3, grid, legend, region, tooltip, xAxis, zoom

**Examples**

data.frame(a = abs(rnorm(20) * 10),
           b = abs(rnorm(20) * 10),
           date = seq(as.Date("2014-01-01"), by = "month", length.out = 20))
c3(x = "date")
subchart(height = 20, onbrush = 'function (domain) { console.log(domain) }')

---

tickAxis

**Axis Tick Options**

**Description**

Modify axis tick formatting options

**Usage**

tickAxis(c3, axis, centered = TRUE, format = NULL, culling = NULL,
      count = NULL, fit = TRUE, values = NULL, rotate = 0, outer = TRUE,
      ...)

**Arguments**

c3 c3 htmlwidget object
axis character 'x', 'y' or 'y2' axis
centered boolean (x-axis only)
format character js function, wrap character or character vector in JS()
culling boolean or list defining number of ticks 'list(max = 5)' this option effects tick labels (x-axis only)
count integer number of ticks to display. This effects tick lines and labels
fit boolean position ticks evenly or set to values (x-axis only)
values vector. Must match axis format type
rotate integer degrees to rotate labels (x-axis only)
outer boolean show axis outer tick
... additional options passed to axis tick object
Value

c3

Examples

data.frame(a = c(1,2,3,2), b = c(2,4,1,5)) %>%
c3() %>%
tickAxis('y', values = c(1,3))

tooltip  C3 Tooltips

Description

Modify plot elements that relate to tooltips. C3.js documentation contains an extended example.

Usage

tooltip(c3, show = TRUE, grouped = TRUE, format = NULL, position = NULL, contents = NULL, ...)

## S3 method for class 'c3'
tooltip(c3, show = TRUE, grouped = TRUE, format = NULL, position = NULL, contents = NULL, ...)

Arguments

c3  c3 htmlwidget object
show  boolean show or hide tooltips
grouped  boolean
format  list with options:
  • title: character js function, wrap character or character vector in JS()
  • name: character js function, wrap character or character vector in JS()
  • value: character js function, wrap character or character vector in JS()

position  character js function, wrap character or character vector in JS()
contents  character js function, wrap character or character vector in JS()
...
  addition options passed to the tooltip object

Value

c3

See Also

Other c3: RColorBrewer, c3, grid, legend, region, subchart, xAxis, zoom
Examples

data <- data.frame(a = abs(rnorm(20) * 10),
b = abs(rnorm(20) * 10),
c = abs(rnorm(20) * 10),
d = abs(rnorm(20) * 10))
data %>%
c3() %>%
tooltip(format = list(title = htmlwidgets::js("function (x) { return 'Data ' + x; }")),
    name = htmlwidgets::js("function (name, ratio, id, index)",
        '{ return name; }")
    value = htmlwidgets::js("function (value, ratio, id, index)",
        '{ return ratio; }")

xAxis

C3 Axis

Description

Modify plot elements that relate to the axis.

Usage

xAxis(c3, show = TRUE, type = "indexed", localtime = NULL,
categories = NULL, max = NULL, min = NULL, padding = list(),
height = NULL, extent = NULL, label = NULL, ...)

## S3 method for class 'c3'
xAxis(c3, show = TRUE, type = "indexed", localtime = NULL,
categories = NULL, max = NULL, min = NULL, padding = list(),
height = NULL, extent = NULL, label = NULL, ...)

yAxis(c3, show = TRUE, inner = NULL, max = NULL, min = NULL,
padding = NULL, inverted = NULL, center = NULL, label = NULL, ...)

## S3 method for class 'c3'
yAxis(c3, show = TRUE, inner = NULL, max = NULL,
min = NULL, padding = NULL, inverted = NULL, center = NULL,
label = NULL, ...)

y2Axis(c3, show = TRUE, inner = NULL, max = NULL,
padding = NULL, inverted = NULL, center = NULL, label = NULL, ...)

## S3 method for class 'c3'
y2Axis(c3, show = TRUE, inner = NULL, max = NULL,
min = NULL, padding = NULL, inverted = NULL, center = NULL,
label = NULL, ...)
Arguments

- c3: c3 htmlwidget object
- show: boolean
- type: character on of 'indexed', 'timeseries' or 'category'
- localtime: boolean
- categories: character vector. Can be used to modify axis labels. Not needed if already defined in data
- max: numeric set value of axis range
- min: numeric set value of axis range
- padding: list with options:
  - left: numeric pixels
  - right: numeric pixels
- height: integer pixels to set height of axis
- extent: vector or character function (wrapped in JS()) that returns a vector of values
- label: can be character or list with options (see c3 axis-x-label):
  - text: character
  - position: character
  
  label position options for horizontal axis are:
  - inner-right
  - inner-center
  - inner-left
  - outer-right
  - outer-center
  - outer-left

  label position options for vertical axis are:
  - inner-top
  - inner-middle
  - inner-bottom
  - outer-top
  - outer-middle
  - outer-bottom

- ... additional options passed to the axis object
- inner: boolean show axis inside chart (Y and Y2 axis only)
- inverted: boolean TRUE will reverse the direction of the axis (Y and Y2 axis only)
- center: integer or numeric value for center line (Y and Y2 axis only)

Value

- c3
See Also
Other c3: RColorBrewer, c3, grid, legend, region, subchart, tooltip, zoom

Examples

data.frame(a=c(1,2,3,2),b=c(2,3,1,5)) %>%
c3(axes = list(a = 'y',
    b = 'y2')) %>%
xAxis(label = list(text = 'testing',
    position = 'inner-center')) %>%
y2Axis()

Description
Enable chart Zoom.

Usage

zoom(c3, enabled = TRUE, rescale = NULL, extent = NULL, onzoom = NULL,
onzoomstart = NULL, onzoomend = NULL, ...)

## S3 method for class 'c3'
zoom(c3, enabled = TRUE, rescale = NULL, extent = NULL,
onzoom = NULL, onzoomstart = NULL, onzoomend = NULL, ...)

Arguments

c3      c3 htmlwidget object
enabled  boolean default is TRUE
rescale  boolean rescale axis when zooming
extent   numeric vector
onzoom  character js function, wrap character or character vector in JS()
onzoomstart  character js function, wrap character or character vector in JS()
onzoomend   character js function, wrap character or character vector in JS()
...       additional options passed to the zoom object

Value

c3
See Also

Other c3: RColorBrewer, c3.grid.legend, region, subchart, tooltip, xAxis

Examples

data.frame(a = abs(rnorm(20) * 10),
            b = abs(rnorm(20) * 10)) %>%
c3() %>%
zoom()

Description

Imports the pipe operator from magrittr.

Usage

lhs %>% rhs

Arguments

lhs a c3 object
rhs a pie settings function

Examples

data.frame(a=c(1,2,3,2),b=c(2,3,1,5)) %>%
c3()
Index

\%

\texttt{c3}, 2, 14, 15, 17–20, 23, 24
\texttt{c3\_shiny}, 3
\texttt{c3\_bar}, 4
\texttt{c3\_chart\_size}, 5
\texttt{c3\_color}, 6
\texttt{c3\_colour}, 6
\texttt{c3\_donut}, 7
\texttt{c3\_gauge}, 8
\texttt{c3\_line}, 9
\texttt{c3\_mixedGeom}, 10
\texttt{c3\_pie}, 10
\texttt{c3\_scatter}, 11
\texttt{c3\_selection}, 12
\texttt{c3\_viridis}, 12
\texttt{c3Output (c3\_shiny)}, 3
\texttt{check\_stacked}, 13

\texttt{grid}, 3, 14, 15, 17–20, 23, 24

\texttt{legend}, 3, 14, 15, 17–20, 23, 24

\texttt{point\_options}, 16

\texttt{RCColorBrewer}, 3, 14, 15, 17, 18–20, 23, 24
\texttt{region}, 3, 14, 15, 17, 19, 20, 23, 24
\texttt{renderC3 (c3\_shiny)}, 3

\texttt{subchart}, 3, 14, 15, 17, 18, 18, 20, 23, 24

\texttt{tickAxis}, 19
\texttt{tooltip}, 3, 14, 15, 17–19, 20, 23, 24

\texttt{xAxis}, 3, 14, 15, 17–20, 21, 24

\texttt{y2Axis (xAxis)}, 21
\texttt{yAxis (xAxis)}, 21

\texttt{zoom}, 3, 14, 15, 17–20, 23, 23

25