

Package ‘highcharter’

January 15, 2019

Type Package

Version 0.7.0

Title A Wrapper for the 'Highcharts' Library

Description A wrapper for the 'Highcharts' library including shortcut functions to plot R objects. 'Highcharts' <<http://www.highcharts.com/>> is a charting library offering numerous chart types with a simple configuration syntax.

URL <http://jkunst.com/highcharter>

BugReports <https://github.com/jbkunst/highcharter/issues>

License MIT + file LICENSE

RoxygenNote 6.1.1

Encoding UTF-8

Depends R (>= 2.10)

Imports htmlwidgets, magrittr, purrr, rlist, assertthat, zoo, dplyr (>= 0.7.0), tibble (>= 1.1), stringr, broom, xts, quantmod, tidy, htmltools, jsonlite, igraph, lubridate, crosstalk, yaml, rlang (>= 0.1.1), whisker

Suggests knitr, rmarkdown, survival, ggplot2, httr, viridisLite, shiny, MASS, gapminder

VignetteBuilder knitr

LazyData true

NeedsCompilation no

Author Joshua Kunst [aut, cre],
Nuno Agostinho [ctb] (hchart.survfit, densities and hc_add_series_scatter),
Danton Noriega [ctb] (hcaes_),
Martin John Hadley [ctb] (hc_add_event_point improvement),
Eduardo Flores [ctb] (First version hc_add_series_df_tidy),
Dean Kilfoyle [ctb] (First version hc_add_series_boxplot),
Adline Dsilva [ctb] (First version Matrix heatmap),
Kamil Slowikowski [ctb] (<<https://orcid.org/0000-0002-2843-6370>>),

Christian Minich [ctb] (hcaes mutate_mapping improvement),
Jonathan Armond [ctb] (mutate_mapping bugfix)

Maintainer Joshua Kunst <jbkunst@gmail.com>

Repository CRAN

Date/Publication 2019-01-15 16:50:03 UTC

R topics documented:

citytemp	5
colorize	5
color_classes	6
color_stops	6
create_yaxis	7
dash_styles	7
datetime_to_timestamp	8
download_map_data	8
export_hc	9
favorite_bars	10
favorite_pies	10
fix_l_length_data	11
get_data_from_map	11
get_hc_series_from_df	12
globaltemp	12
hcaes	13
hcaes_string	13
hcbars	14
hcboxplot	14
hcdensity	15
hchart	15
hchart.survfit	16
hchist	17
hciconarray	17
hcmmap	18
hccords	19
hccpie	20
hccspark	20
hccreemap	21
hccreemap2	22
hccsts	23
hc_accessibility	24
hc_add_dependency	24
hc_add_dependency_fa	25
hc_add_event_point	26
hc_add_series	27
hc_add_series.character	27
hc_add_series.data.frame	28
hc_add_series.density	28

hc_add_series.forecast	29
hc_add_series.geo_json	29
hc_add_series.lm	30
hc_add_series.numeric	30
hc_add_series.ts	31
hc_add_series.xts	31
hc_add_series_boxplot	32
hc_add_series_df	32
hc_add_series_flags	34
hc_add_series_labels_values	35
hc_add_series_list	36
hc_add_series_map	37
hc_add_series_ohlc	38
hc_add_series_scatter	39
hc_add_series_times_values	40
hc_add_series_treemap	41
hc_add_series_ts	42
hc_add_series_xts	43
hc_add_theme	44
hc_annotations	44
hc_annotationsOptions	45
hc_boost	45
hc_chart	46
hc_colorAxis	47
hc_colors	48
hc_credits	48
hc_defs	49
hc_drilldown	49
hc_elementId	51
hc_exporting	52
hc_legend	53
hc_mapNavigation	53
hc_motion	54
hc_navigator	54
hc_pane	55
hc_plotOptions	55
hc_rangeSelector	56
hc_responsive	57
hc_rm_series	57
hc_scrollbar	58
hc_series	58
hc_size	59
hc_theme	60
hc_theme_538	61
hc_theme_chalk	61
hc_theme_darkunica	62
hc_theme_db	62
hc_theme_economist	63

hc_theme_elementary	63
hc_theme_ffx	64
hc_theme_flat	64
hc_theme_flatdark	65
hc_theme_ft	65
hc_theme_ggplot2	66
hc_theme_google	66
hc_theme_gridlight	67
hc_theme_handdrawn	67
hc_theme_merge	68
hc_theme_monokai	68
hc_theme_null	69
hc_theme_sandsignika	69
hc_theme_smpl	70
hc_theme_sparkline	70
hc_theme_superheroes	71
hc_theme_tufte	71
hc_title	72
hc_tooltip	73
hc_xAxis	73
hex_to_rgba	74
highchart	75
highchart2	76
highcharter	76
highcharter-exports	77
highchartOutput	77
highcharts_demo	77
highchart_ct	78
hw_grid	78
is.hexcolor	79
is.highchart	79
list_parse	80
mutate_mapping	80
pokemon	81
random_id	82
renderHighchart	82
stars	83
str_to_id	83
tooltip_chart	84
tooltip_table	85
unemployment	86
uscountygeojson	87
usgeojson	87
vaccines	88
weather	88
worldgeojson	89

citytemp	<i>City temperatures from a year</i>
----------	--------------------------------------

Description

This data comes from the <http://www.highcharts.com/> examples.

Usage

```
citytemp
```

Format

A data frame with 12 observations and 5 variables.

Variables

- month: The months.
- tokyo: Tokyo's temperatures.
- new_york: New York's temperatures.
- berlin: Berlin's temperatures.
- london: London's temperatures.

colorize	<i>Create vector of color from vector</i>
----------	---

Description

Create vector of color from vector

Usage

```
colorize(x, colors = c("#440154", "#21908C", "#FDE725"))
```

Arguments

x	A numeric, character or factor object.
colors	A character string of colors (ordered) to colorize x

Examples

```
colorize(runif(10))
```

```
colorize(LETTERS[rbinom(20, 5, 0.5)], c("#FF0000", "#00FFFF"))
```

color_classes *Function to create dataClasses argument in hc_colorAxis*

Description

Function to create dataClasses argument in hc_colorAxis

Usage

```
color_classes(breaks = NULL, colors = c("#440154", "#21908C",
"#FDE725"))
```

Arguments

breaks A numeric vector
colors A character string of colors (ordered)

Examples

```
color_classes(c(0, 10, 20, 50))
```

color_stops *Function to create stops argument in hc_colorAxis*

Description

Function to create stops argument in hc_colorAxis

Usage

```
color_stops(n = 10, colors = c("#440154", "#21908C", "#FDE725"))
```

Arguments

n A numeric indicating how much quantiles generate.
colors A character string of colors (ordered)

Examples

```
color_stops(5)
```

create_yaxis *Creating multiples yAxis t use with highcharts*

Description

Creating multiples yAxis t use with highcharts

Usage

```
create_yaxis(naxis = 2, heights = 1, sep = 0.01, offset = 0,
             turnopposite = TRUE, ...)
```

Arguments

naxis	Number of axis an integer.
heights	A numeric vector. This values will be normalized.
sep	A numeric value for the separation (in percentage) for the panes.
offset	A numeric value (in percentage).
turnopposite	A logical value to turn the side of each axis or not.
...	Arguments defined in http://api.highcharts.com/highcharts/yAxis .

Examples

```
highchart() %>%
  hc_yAxis_multiples(create_yaxis(naxis = 2, heights = c(2, 1))) %>%
  hc_add_series(data = c(1,3,2), yAxis = 0) %>%
  hc_add_series(data = c(20, 40, 10), yAxis = 1)

highchart() %>%
  hc_yAxis_multiples(create_yaxis(naxis = 3, lineWidth = 2, title = list(text = NULL))) %>%
  hc_add_series(data = c(1,3,2)) %>%
  hc_add_series(data = c(20, 40, 10), yAxis = 1) %>%
  hc_add_series(data = c(200, 400, 500), type = "column", yAxis = 2) %>%
  hc_add_series(data = c(500, 300, 400), type = "column", yAxis = 2)
```

dash_styles *Get dash styles*

Description

Get dash style to use on highcharts objects.

Usage

```
dash_styles()
```

`datetime_to_timestamp` *Date to timestamps*

Description

Turn a date time vector to timestamp format

Usage

```
datetime_to_timestamp(dt)
```

Arguments

`dt` Date or datetime vector

Examples

```
datetime_to_timestamp(  
  as.Date(c("2015-05-08", "2015-09-12"),  
  format = "%Y-%m-%d"))
```

`download_map_data` *Helper function to download the map data form a url*

Description

The urls are listed in <https://code.highcharts.com/mapdata/>.

Usage

```
download_map_data(url = "custom/world.js", showinfo = FALSE)
```

Arguments

`url` The map's url.
`showinfo` Show the properties of the downloaded map to know how are the keys to add data in hcmmap.

See Also

[hcmmap](#)

Examples

```
## Not run:
mpdta <- download_map_data("https://code.highcharts.com/mapdata/countries/us/us-ca-all.js")
str(mpdta, 1)

## End(Not run)
```

export_hc*Function to export js file the configuration options*

Description

Function to export js file the configuration options

Usage

```
export_hc(hc, filename = NULL, as = "is", name = NULL)
```

Arguments

hc	A Highcharts object.
filename	String of the exported file.
as	String to define how to save the configuration options. One of 'is', 'container', 'variable'.
name	A variable used to put as name of the generated object if as is 'variable' and the css/js selector if is as is container.

Examples

```
fn <- "function(){
  console.log('Category: ' + this.category);
  alert('Category: ' + this.category);
}"

hc <- highcharts_demo() %>%
  hc_plotOptions(
    series = list(
      cursor = "pointer",
      point = list(
        events = list(
          click = JS(fn)
        )
      )
    )
  )
)

## Not run:
```

```

export_hc(hc, filename = "~/hc_is.js", as = "is")
export_hc(hc, filename = "~/hc_vr.js", as = "variable", name = "objectname")
export_hc(hc, filename = "~/hc_ct.js", as = "container", name = "#selectorid")

## End(Not run)

```

favorite_bars	<i>Marshall's Favorite Bars</i>
---------------	---------------------------------

Description

Data from How I met Your Mother: Marshall's Favorite Bars.

Usage

```
favorite_bars
```

Format

A data frame with 5 observations and 2 variables.

Variables

- bar: Bar's name.
- percent: In percentage of awesomeness

favorite_pies	<i>Marshall's Favorite Pies</i>
---------------	---------------------------------

Description

Data from How I met Your Mother: Marshall's Favorite Pies

Usage

```
favorite_pies
```

Format

A data frame with 5 observations and 2 variables.

Variables

- pie: Bar's name.
- percent: In percentage of tastiness

fix_1_length_data *Function to avoid the jsonlite::auto_unbox default*

Description

Function to avoid the jsonlite::auto_unbox default

Usage

```
fix_1_length_data(x)
```

Arguments

x And element, numeric or character

get_data_from_map *Helper function to get the data inside the map data The urls are listed in <https://code.highcharts.com/mapdata/>.*

Description

Helper function to get the data inside the map data The urls are listed in <https://code.highcharts.com/mapdata/>.

Usage

```
get_data_from_map(mapdata)
```

Arguments

mapdata A list obtained from [download_map_data](#).

See Also

[download_map_data](#)

Examples

```
dta <- download_map_data("https://code.highcharts.com/mapdata/countries/us/us-ca-all.js")
get_data_from_map(dta)
```

`get_hc_series_from_df` *Auxiliar function to get series and options from tidy frame for hchart.data.frame*

Description

This function is used in `hchart.data.frame` and `hc_add_series_df`

Usage

```
get_hc_series_from_df(data, type = NULL, ...)
```

Arguments

<code>data</code>	A <code>data.frame</code> object.
<code>type</code>	The type of chart. Possible values are line, scatter, point, columnn.
<code>...</code>	Aesthetic mappings as <code>x y group color low high</code> .

Examples

```
get_hc_series_from_df(iris, type = "point", x = Sepal.Width)
```

`globaltemp`

globaltemp

Description

Temperature information by years.

Usage

```
globaltemp
```

Format

A data frame with 1992 observations and 4 variables.

Variables

- `date`: Date.
- `lower`: Minimum temperature.
- `median`: Median temperature.
- `upper`: Maximum temperature.

Source

<http://www.climate-lab-book.ac.uk/2016/spiralling-global-temperatures/>

hcaes	<i>Define aesthetic mappings. Similar in spirit to ggplot2::aes</i>
-------	---

Description

Define aesthetic mappings. Similar in spirit to ggplot2::aes

Usage

```
hcaes(x, y, ...)
```

Arguments

<code>x, y, ...</code>	List of name value pairs giving aesthetics to map to variables. The names for x and y aesthetics are typically omitted because they are so common; all other aesthetics must be named.
------------------------	--

Examples

```
hcaes(x = xval, color = colorvar, group = grvar)
```

hcaes_string	<i>Define aesthetic mappings using strings. Similar in spirit to ggplot2::aes_string</i>
--------------	--

Description

Define aesthetic mappings using strings. Similar in spirit to ggplot2::aes_string

Usage

```
hcaes_string(x, y, ...)
```

```
hcaes_(x, y, ...)
```

Arguments

<code>x, y, ...</code>	List of name value pairs giving aesthetics to map to variables. The names for x and y aesthetics are typically omitted because they are so common; all other aesthetics must be named.
------------------------	--

Examples

```
hchart(mtcars, "point", hcaes_string("hp", "mpg", group = "cyl"))
```

```
hcaes_string(x = 'xval', color = 'colorvar', group = 'grvar')
```

hcbbar	<i>Shortcut to make a bar chart</i>
--------	-------------------------------------

Description

Shortcut to make a bar chart

Usage

```
hcbbar(x, ...)
```

Arguments

x	A character or factor vector.
...	Additional arguments for the data series http://api.highcharts.com/highcharts#series .

hcbboxplot	<i>Shortcut to make a boxplot</i>
------------	-----------------------------------

Description

Shortcut to make a boxplot

Usage

```
hcbboxplot(x = NULL, var = NULL, var2 = NULL, outliers = TRUE, ...)
```

Arguments

x	A numeric vector.
var	A string vector same length of x.
var2	A string vector same length of x.
outliers	A boolean value to show or not the outliers.
...	Additional arguments for the data series http://api.highcharts.com/highcharts#series .

Examples

```
hcbboxplot(x = iris$Sepal.Length, var = iris$Species, color = "red")
```

hcdensity	<i>Shortcut to make density charts</i>
-----------	--

Description

Shortcut to make density charts

Usage

```
hcdensity(x, ...)
```

Arguments

x	A numeric vector or a density object.
...	Additional arguments for the data series http://api.highcharts.com/highcharts#series .

hchart	<i>Create a highchart object from a particular data type</i>
--------	--

Description

hchart uses highchart to draw a particular plot for an object of a particular class in a single command. This defines the S3 generic that other classes and packages can extend.

Usage

```
hchart(object, ...)
```

Arguments

object	A R object.
...	Additional arguments for the data series (http://api.highcharts.com/highcharts#series).

Details

Run `methods(hchart)` to see what objects are supported.

hchart.survfit *Plot survival curves using Highcharts*

Description

Plot survival curves using Highcharts

Usage

```
## S3 method for class 'survfit'
hchart(object, ..., fun = NULL, markTimes = TRUE,
        symbol = "plus", markerColor = "black", ranges = FALSE,
        rangesOpacity = 0.3)
```

Arguments

object	A survfit object as returned from the survfit function
...	Extra parameters to pass to hc_add_series function
fun	Name of function or function used to transform the survival curve: log will put y axis on log scale, event plots cumulative events ($f(y) = 1-y$), cumhaz plots the cumulative hazard function ($f(y) = -\log(y)$), and cloglog creates a complimentary log-log survival plot ($f(y) = \log(-\log(y))$) along with log scale for the x-axis.
markTimes	Label curves marked at each censoring time? TRUE by default
symbol	Symbol to use as marker (plus sign by default)
markerColor	Color of the marker ("black" by default); use NULL to use the respective color of each series
ranges	Plot interval ranges? FALSE by default
rangesOpacity	Opacity of the interval ranges (0.3 by default)

Value

Highcharts object to plot survival curves

Examples

```
# Plot Kaplan-Meier curves
require("survival")
leukemia.surv <- survfit(Surv(time, status) ~ x, data = aml)
hchart(leukemia.surv)

# Plot the cumulative hazard function
lsurv2 <- survfit(Surv(time, status) ~ x, aml, type='fleming')
hchart(lsurv2, fun="cumhaz")
```



```
# Plot the fit of a Cox proportional hazards regression model
fit <- coxph(Surv(futime, fustat) ~ age, data = ovarian)
ovarian.surv <- survfit(fit, newdata=data.frame(age=60))
hchart(ovarian.surv, ranges = TRUE)
```

hchist *Shortcut to make an histogram*

Description

Shortcut to make an histogram

Usage

```
hchist(x, ...)
```

Arguments

x	A numeric vector.
...	Additional arguments for the data series http://api.highcharts.com/highcharts#series .

hciconarray *Shortcut to make icon arrays charts*

Description

Shortcut to make icon arrays charts

Usage

```
hciconarray(labels, counts, rows = NULL, icons = NULL, size = 4, ...)
```

Arguments

labels	A character vector
counts	A integer vector
rows	A integer to set
icons	A character vector same length (o length 1) as labels
size	Font size
...	Additional arguments for the data series http://api.highcharts.com/highcharts#series .

Examples

```

hcmmap(c("nice", "good"), c(10, 20))

hcmmap(c("nice", "good"), c(10, 20), size = 10)

hcmmap(c("nice", "good"), c(100, 200), icons = "child")

hcmmap(c("car", "truck", "plane"), c(75, 30, 20), icons = c("car", "truck", "plane")) %>%
  hc_add_theme(
    hc_theme_merge(
      hc_theme_flatdark(),
      hc_theme_null(chart = list(background-color = "#34495e"))
    )
  )

```

hcmmap	<i>Shortcut for create map from https://code.highcharts.com/mapdata/ collection.</i>
--------	---

Description

Shortcut for create map from <https://code.highcharts.com/mapdata/> collection.

Usage

```

hcmmap(map = "custom/world",
  download_map_data = getOption("highcharter.download_map_data"),
  data = NULL, value = NULL, joinBy = NULL, ...)

```

Arguments

map	String indicating what map to chart, a list from https://code.highcharts.com/mapdata/ . See examples.
download_map_data	A logical value whether to download (add as a dependency) the map. Default TRUE via <code>getOption("highcharter.download_map_data")</code> .
data	Optional data to make a choropleth, in case of use the <code>joinBy</code> and <code>value</code> are needed.
value	A string value with the name of the column to chart.
joinBy	What property to join the map and df.
...	Additional shared arguments for the data series (http://api.highcharts.com/highcharts#series).

Examples

```
## Not run:

hcmmap(nullColor = "#DADADA")
hcmmap(nullColor = "#DADADA", download_map_data = FALSE)

require(dplyr)
data("USArrests", package = "datasets")
USArrests <- mutate(USArrests, "woe-name" = rownames(USArrests))

hcmmap(map = "countries/us/us-all", data = USArrests,
       joinBy = "woe-name", value = "UrbanPop", name = "Urban Population")

# download_map_data = FALSE
hcmmap(map = "countries/us/us-all", data = USArrests,
       joinBy = "woe-name", value = "UrbanPop", name = "Urban Population",
       download_map_data = FALSE)

## End(Not run)
```

hpcarcords

Shortcut for create parallel coordinates

Description

Shortcut for create parallel coordinates

Usage

```
hpcarcords(df, ...)
```

Arguments

df	A data frame object.
...	Additional shared arguments for the data series (http://api.highcharts.com/highcharts#series) for the <code>hchar.data.frame</code> function.

Examples

```
require(viridisLite)

n <- 15

hpcarcords(head(mtcars, n), color = hex_to_rgba(magma(n), 0.5))

require(dplyr)
data(iris)
```

```
set.seed(123)
iris <- sample_n(iris, 60)
hccords(iris, color = colorize(iris$Species))
```

hccpie *Shortcut to make a pie chart*

Description

Shortcut to make a pie chart

Usage

```
hccpie(x, ...)
```

Arguments

x A character or factor vector.

... Additional arguments for the data series <http://api.highcharts.com/highcharts#series>.

hccspark *Shortcut to make sparklines*

Description

Shortcut to make sparklines

Usage

```
hccspark(x = NULL, type = NULL, ...)
```

Arguments

x A numeric vector.

type Type sparkline: line, bar, etc.

... Additional arguments for the data series <http://api.highcharts.com/highcharts#series>.


```
GNI: {point.valuecolor:,.0f}")
```

```
## End(Not run)
```

hctreemap2

Shortcut to create treemaps.

Description

This function helps create highcharts treemaps from data frames.

Usage

```
hctreemap2(data, group_vars, size_var, color_var = NULL, ...)
```

Arguments

data	data frame containing variables to organize each level of the treemap on
group_vars	vector of strings containing column names of variables to generate treemap levels from. the first listed column will specify the top level of the treemap. the unique values in each of these columns must have no intersection (including NAs).
size_var	string name of column containing numeric data to aggregate by
color_var	string name of column containing numeric data to color by. defaults to same column as size_var
...	additional shared arguments for the data series (http://api.highcharts.com/highcharts#series).

Value

highchart plot object

Examples

```
## Not run:

library(tidyverse)
library(highcharter)
library(RColorBrewer)

data_frame(
  index1 = sample(LETTERS[1:5], 500, replace = T),
  index2 = sample(LETTERS[6:10], 500, replace = T),
  index3 = sample(LETTERS[11:15], 500, replace = T),
  value = rpois(500, 5),
```

```

    color_value = rpois(500, 5)
  ) %>%
  hctreemap2(
    group_vars = c("index1", "index2", "index3"),
    size_var = "value",
    color_var = "color_value",
    layoutAlgorithm = "squarified",
    levelIsConstant = FALSE,
    levels = list(
      list(level = 1, dataLabels = list(enabled = TRUE)),
      list(level = 2, dataLabels = list(enabled = FALSE)),
      list(level = 3, dataLabels = list(enabled = FALSE))
    )
  ) %>%
  hc_colorAxis(minColor = brewer.pal(7, "Greens")[1],
              maxColor = brewer.pal(7, "Greens")[7]) %>%
  hc_tooltip(pointFormat = "<b>{point.name}</b>:<br>
    Value: {point.value:,.0f}<br>
    Color Value: {point.colorValue:,.0f}")

## End(Not run)

```

hcts

Shortcut to make time series or line charts

Description

Shortcut to make time series or line charts

Usage

```
hcts(x, ...)
```

Arguments

x	A numeric vector or a time series object.
...	Additional arguments for the data series http://api.highcharts.com/highcharts#series .

hc_accessibility *Setting accessibility options to highcharts objects*

Description

Options for configuring accessibility for the chart. Requires the accessibility module to be loaded.

Usage

```
hc_accessibility(hc, ...)
```

Arguments

hc	A highchart htmlwidget object.
...	Options defined in http://api.highcharts.com/highcharts/accessibility .

hc_add_dependency *Add modules or plugin dependencies to highcharts objects*

Description

Add modules or plugin dependencies to highcharts objects

Usage

```
hc_add_dependency(hc, name = "plugins/annotations.js")
```

Arguments

hc	A highchart htmlwidget object.
name	The partial path to the plugin or module, example: "plugins/annotations.js"

Examples

```
highchart() %>%
  hc_title(text = "I'm a pirate looking chart") %>%
  hc_xAxis(categories = month.abb) %>%
  hc_defs(
    patterns = list(
      list(
        id = "custom-pattern",
        path = list(d = "M 0 0 L 10 10 M 9 -1 L 11 1 M -1 9 L 1 11",
                    stroke = "black", strokeWidth = 1
        )
      )
    )
  )
```



```

    )
  ) %>%
  hc_add_series(data = 10 * dt(1 + 1:12 - mean(1:12), df = 2),
               type = "area",
               fillColor = "url(#custom-pattern)") %>%
  hc_add_theme(hc_theme_handdrawn()) %>%
  hc_add_dependency(name = "plugins/pattern-fill-v2.js")

data(mpg, package = "ggplot2")

hchart(mpg, "point", hcaes(displ, hwy), regression = TRUE,
       regressionSettings = list(type = "polynomial", order = 5, hideInLegend = TRUE)) %>%
  hc_add_dependency("plugins/highcharts-regression.js")

hchart(mpg, "point", hcaes(displ, hwy, group = drv), regression = TRUE) %>%
  hc_colors(c("#d35400", "#2980b9", "#2ecc71")) %>%
  hc_add_dependency("plugins/highcharts-regression.js")

```

hc_add_dependency_fa *Helpers functions to get FontAwesome icons code*

Description

Helpers functions to get FontAwesome icons code

Usage

```

hc_add_dependency_fa(hc)

fa_icon(iconname = "circle")

fa_icon_mark(iconname = "circle")

```

Arguments

hc	A highchart htmlwidget object.
iconname	The icon's name

Examples

```

dcars <- data.frame(x = runif(10), y = runif(10))
dtrck <- data.frame(x = rexp(10), y = rexp(10))

highchart() %>%
  hc_chart(zoomType = "xy") %>%
  hc_tooltip(

```

```

    useHTML = TRUE,
    pointFormat = paste0("<span style=\"color:{series.color};\">{series.options.icon}</span>",
      "{series.name}: <b>[{point.x}, {point.y}]</b><br/>")
      ) %>%
hc_add_series(dcars, "scatter", marker = list(symbol = fa_icon_mark("car")),
  icon = fa_icon("car"), name = "car") %>%
hc_add_series(dtrck, "scatter", marker = list(symbol = fa_icon_mark("plane")),
  icon = fa_icon("plane"), name = "plane") %>%
hc_add_dependency_fa()

fa_icon("car")
fa_icon_mark("car")
fa_icon_mark(iconname = c("car", "plane", "car"))

```

hc_add_event_point *Helpers to use highcharter as input in shiny apps*

Description

When you use highcharter in a shiny app, for example `renderHighcharter('my_chart')`, you can access to the actions of the user using `input$my_chart` and then use the `hc_add_event_point` via the `my_chart` input (`input$my_chart`). That's a way you can use a chart as an input.

Usage

```

hc_add_event_point(hc, series = "series", event = "click")

hc_add_event_series(hc, series = "series", event = "click")

```

Arguments

hc	A highchart htmlwidget object.
series	The name of type of series to apply the event.
event	The name of event: click, mouseOut, mouseOver. See http://api.highcharts.com/highcharts/plotOptions.area.splinerange.point.events.select for more details.

Note

Event details are accessible from `hc_name_EventType`, i.e. if a highchart is rendered against `output$my_hc` and we wanted the coordinates of the user-clicked point we would use `input$my_hc_click`

hc_add_series *Adding and removing series from highchart objects*

Description

Adding and removing series from highchart objects

Usage

```
hc_add_series(hc, data = NULL, ...)
```

Arguments

hc A highchart htmlwidget object.
data An R object like numeric, list, ts, xts, etc.
... Arguments defined in <http://api.highcharts.com/highcharts/chart>.

Examples

```
highchart() %>%
  hc_add_series(data = abs(rnorm(5)), type = "column") %>%
  hc_add_series(data = purrr::map(0:4, function(x) list(x, x)), type = "scatter", color = "blue")
```

hc_add_series.character
 hc_add_series for character and factor objects

Description

hc_add_series for character and factor objects

Usage

```
## S3 method for class 'character'
hc_add_series(hc, data, ...)

## S3 method for class 'factor'
hc_add_series(hc, data, ...)
```

Arguments

hc A highchart htmlwidget object.
data A character or factor object.
... Arguments defined in <http://api.highcharts.com/highcharts/chart>.

`hc_add_series.data.frame`*hc_add_series for data frames objects*

Description

hc_add_series for data frames objects

Usage

```
## S3 method for class 'data.frame'  
hc_add_series(hc, data, type = NULL,  
              mapping = hcaes(), ...)
```

Arguments

hc	A highchart htmlwidget object.
data	A data.frame object.
type	The type of the series: line, bar, etc.
mapping	The mapping, same idea as ggplot2.
...	Arguments defined in http://api.highcharts.com/highcharts#chart .

`hc_add_series.density` *hc_add_series for density objects*

Description

hc_add_series for density objects

Usage

```
## S3 method for class 'density'  
hc_add_series(hc, data, ...)
```

Arguments

hc	A highchart htmlwidget object.
data	A density object.
...	Arguments defined in http://api.highcharts.com/highcharts/chart .

hc_add_series.forecast

hc_add_series for forecast objects

Description

hc_add_series for forecast objects

Usage

```
## S3 method for class 'forecast'  
hc_add_series(hc, data, addOriginal = FALSE,  
              addLevels = TRUE, fillOpacity = 0.1, name = NULL, ...)
```

Arguments

hc	A highchart htmlwidget object.
data	A forecast object.
addOriginal	Logical value to add the original series or not.
addLevels	Logical value to show predictions bands.
fillOpacity	The opacity of bands.
name	The name of the series.
...	Arguments defined in http://api.highcharts.com/highcharts#chart .

hc_add_series.geo_json

hc_add_series for geo_json & geo_list objects

Description

hc_add_series for geo_json & geo_list objects

Usage

```
## S3 method for class 'geo_json'  
hc_add_series(hc, data, type = NULL, ...)
```

```
## S3 method for class 'geo_list'  
hc_add_series(hc, data, type = NULL, ...)
```

Arguments

hc	A highchart htmlwidget object.
data	A geo_json or geo_list object.
type	Type of series. Can be 'mapline', 'mapoint'.
...	Arguments defined in http://api.highcharts.com/highcharts/chart .

hc_add_series.lm *hc_add_series for lm and loess objects*

Description

hc_add_series for lm and loess objects

Usage

```
## S3 method for class 'lm'
hc_add_series(hc, data, type = "line", color = "#5F83EE",
  fillOpacity = 0.1, ...)

## S3 method for class 'loess'
hc_add_series(hc, data, type = "line",
  color = "#5F83EE", fillOpacity = 0.1, ...)
```

Arguments

hc	A highchart htmlwidget object.
data	A lm or loess object.
type	The type of the series: line, spline.
color	A stringr color.
fillOpacity	fillOpacity to the confidence interval.
...	Arguments defined in http://api.highcharts.com/highcharts#chart .

hc_add_series.numeric *hc_add_series for numeric objects*

Description

hc_add_series for numeric objects

Usage

```
## S3 method for class 'numeric'
hc_add_series(hc, data, ...)
```

Arguments

hc	A highchart htmlwidget object.
data	A numeric object
...	Arguments defined in http://api.highcharts.com/highcharts/chart .

hc_add_series.ts *hc_add_series for time series objects*

Description

hc_add_series for time series objects

Usage

```
## S3 method for class 'ts'
hc_add_series(hc, data, ...)
```

Arguments

hc	A highchart htmlwidget object.
data	A time series ts object.
...	Arguments defined in http://api.highcharts.com/highcharts/chart .

hc_add_series.xts *hc_add_series for xts objects*

Description

hc_add_series for xts objects

Usage

```
## S3 method for class 'xts'
hc_add_series(hc, data, ...)

## S3 method for class 'ohlcv'
hc_add_series(hc, data, type = "candlestick", ...)
```

Arguments

hc	A highchart htmlwidget object.
data	A xts object.
...	Arguments defined in http://api.highcharts.com/highcharts/chart .
type	The way to show the xts object. Can be 'candlestick' or 'ohlcv'.

hc_add_series_boxplot *Shortcut for create boxplot*

Description

Shortcut for create boxplot

Usage

```
hc_add_series_boxplot(hc, x, by = NULL, outliers = TRUE, ...)
```

Arguments

hc	A highchart htmlwidget object.
x	A numeric vector
by	A string vector same length of x
outliers	A boolean value to show or not the outliers
...	Additional arguments for the data series (http://api.highcharts.com/highcharts#series).

Examples

```
## Not run:  
highchart() %>%  
  hc_add_series_boxplot(x = iris$Sepal.Length, by = iris$Species, name = "length")  
  
## End(Not run)
```

hc_add_series_df *Shortcut for tidy data frame a la ggplot2/qplot*

Description

Function to create chart from tidy data frames. As same as qplot you can use aesthetic including the group variable

Usage

```
hc_add_series_df(hc, data, type = NULL, ...)
```


Arguments

hc	A highchart htmlwidget object.
data	A data.frame object.
type	The type of chart. Possible values are line, scatter, point, column, columnrange, etc. See http://api.highcharts.com/highcharts#series .
...	Aesthetic mappings, x y group color low high.

Details

The types supported are line, column, point, polygon, columnrange, spline, areaspline among others.

Automatically parsed de data frame (to a list o series). You you can use the default parameters of highcharts such as x, y, z, color, name, low, high for each series, for example check <http://api.highcharts.com/highcharts#series<bubble>.data>.

Examples

```
## Not run:
require("dplyr")
n <- 50
df <- data_frame(
  x = rnorm(n),
  y = x * 2 + rnorm(n),
  w = x^2
)

hc_add_series_df(highchart(), data = df, type = "point", x = x, y = y)
hc_add_series_df(highchart(), data = df, type = "point", color = w)
hc_add_series_df(highchart(), data = df, type = "point", color = w, size = y)

m <- 50
s <- cumsum(rnorm(m))
e <- 2 + rbeta(m, 2, 2)

df2 <- data_frame(
  var = seq(m),
  l = s - e,
  h = s + e,
  n = paste("I'm point ", var)
)

hc_add_series_df(highchart(), data = df2, type = "columnrange",
  x = var, low = l, high = h, name = n, color = var)

hc_add_series_df(highchart(), iris, "point",
  x = Sepal.Length, y = Sepal.Width, group = Species)

data(mpg, package = "ggplot2")
```

```

# point and scatter is the same
hc_add_series_df(highchart(), mpg, "scatter", x = displ, y = cty)
hc_add_series_df(highchart(), mpg, "point", x = displ, y = cty,
                  group = manufacturer)

mpgman <- count(mpg, manufacturer)
hc_add_series_df(highchart(), mpgman, "column", x = manufacturer, y = n) %>%
  hc_xAxis(type = "category")

mpgman2 <- count(mpg, manufacturer, year)
hc_add_series_df(highchart(), mpgman2, "bar", x = manufacturer, y = n, group = year) %>%
  hc_xAxis(type = "category")

data(economics, package = "ggplot2")

hc_add_series_df(highchart(), economics, "line", x = date, y = unemploy) %>%
  hc_xAxis(type = "datetime")

data(economics_long, package = "ggplot2")

economics_long2 <- filter(economics_long,
                          variable %in% c("pop", "uempmed", "unemploy"))

hc_add_series_df(highchart(), economics_long2, "line", x = date,
                  y = value01, group = variable) %>%
  hc_xAxis(type = "datetime")

## End(Not run)

```

hc_add_series_flags *Shortcut for add flags to highstock chart*

Description

This function helps to add flags highstock charts created from xts objects.

Usage

```

hc_add_series_flags(hc, dates, title = LETTERS[seq(length(dates))],
                    text = title, id = NULL, ...)

```

Arguments

hc	A highchart htmlwidget object.
dates	Date vector.

title	A character vector with titles.
text	A character vector with the description.
id	The name of the series to add the flags. A previous series must be added with this id.
...	Additional shared arguments for the <i>flags</i> data series (http://api.highcharts.com/highstock#plotOptions.flags)

Examples

```
## Not run:

library("quantmod")

usdjpy <- getSymbols("USD/JPY", src="oanda", auto.assign = FALSE)

dates <- as.Date(c("2015-05-08", "2015-09-12"), format = "%Y-%m-%d")
highchart(type = "stock") %>%
  hc_add_series_xts(usdjpy, id = "usdjpy") %>%
  hc_add_series_flags(dates,
                     title = c("E1", "E2"),
                     text = c("This is event 1", "This is the event 2"),
                     id = "usdjpy")

## End(Not run)
```

hc_add_series_labels_values

Shortcut for add series for pie, bar and columnn charts

Description

This function add data to plot pie, bar and columnn charts.

Usage

```
hc_add_series_labels_values(hc, labels, values, colors = NULL, ...)
```

Arguments

hc	A highchart htmlwidget object.
labels	A vector of labels.
values	A numeric vector. Same length of labels.
colors	A not required color vector (hexadecimal format). Same length of labels.
...	Additional shared arguments for the data series (http://api.highcharts.com/highcharts#series).

Examples

```
## Not run:
data("favorite_bars")
data("favorite_pies")

highchart() %>%
  hc_title(text = "This is a bar graph describing my favorite pies
              including a pie chart describing my favorite bars") %>%
  hc_subtitle(text = "In percentage of tastiness and awesomeness") %>%
  hc_add_series_labels_values(favorite_pies$pie, favorite_pies$percent, name = "Pie",
                             colorByPoint = TRUE, type = "columnn") %>%
  hc_add_series_labels_values(favorite_bars$bar, favorite_bars$percent,
                             colors = substr(terrain.colors(5), 0 , 7), type = "pie",
                             name = "Bar", colorByPoint = TRUE, center = c('35%', '10%'),
                             size = 100, dataLabels = list(enabled = FALSE)) %>%
  hc_yAxis(title = list(text = "percentage of tastiness"),
           labels = list(format = "{value}%"), max = 100) %>%
  hc_xAxis(categories = favorite_pies$pie) %>%
  hc_legend(enabled = FALSE) %>%
  hc_tooltip(pointFormat = "{point.y}%")

## End(Not run)
```

hc_add_series_list *Shortcut for data series from a list of data series*

Description

Shortcut for data series from a list of data series

Usage

```
hc_add_series_list(hc, x)
```

Arguments

hc A highchart htmlwidget object.
x A list or a data.frame of series.

Examples

```
ds <- lapply(seq(5), function(x){
  list(data = cumsum(rnorm(100, 2, 5)), name = x)
})

highchart() %>%
  hc_plotOptions(series = list(marker = list(enabled = FALSE))) %>%
```

```
hc_add_series_list(ds)
```

```
hc_add_series_map      Add a map series
```

Description

Add a map series

Usage

```
hc_add_series_map(hc, map, df, value, joinBy, ...)
```

Arguments

hc	A highchart htmlwidget object.
map	A list object loaded from a geojson file.
df	A data.frame object with data to chart. Code region and value are required.
value	A string value with the name of the column to chart.
joinBy	What property to join the map and df
...	Additional shared arguments for the data series (http://api.highcharts.com/highcharts#series).

Details

This function force the highchart object to be map type.

Examples

```
library("dplyr")

data("USArrests", package = "datasets")
data("usgeojson")

USArrests <- mutate(USArrests, state = rownames(USArrests))

highchart() %>%
  hc_title(text = "Violent Crime Rates by US State") %>%
  hc_subtitle(text = "Source: USArrests data") %>%
  hc_add_series_map(usgeojson, USArrests, name = "Murder arrests (per 100,000)",
                  value = "Murder", joinBy = c("woename", "state"),
                  dataLabels = list(enabled = TRUE,
                                    format = '{point.properties.postalcode}')) %>%
  hc_colorAxis(stops = color_stops()) %>%
  hc_legend(valueDecimals = 0, valueSuffix = "%") %>%
  hc_mapNavigation(enabled = TRUE)
```

```
## Not run:

data(worldgeojson, package = "highcharter")
data("GNI2014", package = "treemap")

highchart(type = "map") %>%
  hc_add_series_map(map = worldgeojson, df = GNI2014, value = "GNI", joinBy = "iso3") %>%
  hc_colorAxis(stops = color_stops()) %>%
  hc_tooltip(useHTML = TRUE, headerFormat = "",
    pointFormat = "this is {point.name} and have {point.population} people with gni of {point.GNI}")

## End(Not run)
```

hc_add_series_ohlc *Shortcut for create candlestick charts*

Description

This function helps to create candlestick from xts objects obtaining by getSymbols function from the **quantmod**.

Usage

```
hc_add_series_ohlc(hc, x, type = "candlestick", ...)
```

Arguments

hc	A highchart htmlwidget object.
x	A ohlc object from the quantmod package.
type	The type of chart. Can be candlestick or ohlc.
...	Additional shared arguments for the data series (http://api.highcharts.com/highcharts#series).

Examples

```
## Not run:

library("xts")

data(sample_matrix)

matrix_xts <- as.xts(sample_matrix, dateFormat = "Date")
```

```
head(matrix_xts)

class(matrix_xts)

highchart() %>%
  hc_add_series_ohlc(matrix_xts)

library("quantmod")

x <- getSymbols("AAPL", auto.assign = FALSE)
y <- getSymbols("SPY", auto.assign = FALSE)

highchart() %>%
  hc_add_series_ohlc(x) %>%
  hc_add_series_ohlc(y)

## End(Not run)
```

hc_add_series_scatter *Shortcut for create scatter plots*

Description

This function helps to create scatter plot from two numeric vectors. Options arguments like size, color and label for points are added.

Usage

```
hc_add_series_scatter(hc, x, y, z = NULL, color = NULL, label = NULL,
  showInLegend = FALSE, ...)
```

Arguments

hc	A highchart htmlwidget object.
x	A numeric vector.
y	A numeric vector. Same length of x.
z	A numeric vector for size. Same length of x.
color	A vector to color the points.
label	A vector to put names in the dots if you enable the datalabels.
showInLegend	Logical value to show or not the data in the legend box.
...	Additional shared arguments for the data series (http://api.highcharts.com/highcharts#series).

Examples

```
## Not run:
hc <- highchart()

hc_add_series_scatter(hc, mtcars$wt, mtcars$mpg)
hc_add_series_scatter(hc, mtcars$wt, mtcars$mpg, mtcars$drat)
hc_add_series_scatter(hc, mtcars$wt, mtcars$mpg, mtcars$drat, mtcars$am)
hc_add_series_scatter(hc, mtcars$wt, mtcars$mpg, mtcars$drat, mtcars$qsec)
hc_add_series_scatter(hc, mtcars$wt, mtcars$mpg, mtcars$drat, mtcars$qsec, rownames(mtcars))

# Add named attributes to data (attributes length needs to match number of rows)
hc_add_series_scatter(hc, mtcars$wt, mtcars$mpg, mtcars$drat, mtcars$qsec,
                      name = rownames(mtcars), gear = mtcars$gear) %>%
  hc_tooltip(pointFormat = "<b>{point.name}</b><br/>Gear: {point.gear}")

## End(Not run)
```

hc_add_series_times_values

Shortcut for create/add time series from times and values

Description

This function add a time series to a highchart object.

Usage

```
hc_add_series_times_values(hc, dates, values, ...)
```

Arguments

hc	A highchart htmlwidget object.
dates	A date vector (same length as values)
values	A numeric vector
...	Additional arguments for the data series (http://api.highcharts.com/highcharts#series).

Details

This function **modify** the type of chart to datetime

Examples

```
## Not run:

require("ggplot2")
data(economics, package = "ggplot2")

hc_add_series_times_values(hc = highchart(),
                           dates = economics$date,
                           values = economics$psavert,
                           name = "Personal Savings Rate")

## End(Not run)
```

hc_add_series_treemap *Shortcut for create treemaps*

Description

This function helps to create highcharts treemaps from treemap objects from the package treemap.

Usage

```
hc_add_series_treemap(hc, tm, ...)
```

Arguments

hc	A highchart htmlwidget object.
tm	A treemap object from the treemap package.
...	Additional shared arguments for the data series (http://api.highcharts.com/highcharts#series).

Examples

```
## Not run:

library("treemap")
library("viridis")

data(GNI2014)
head(GNI2014)

tm <- treemap(GNI2014, index = c("continent", "iso3"),
              vSize = "population", vColor = "GNI",
              type = "comp", palette = rev(viridis(6)),
              draw = FALSE)
```

```

highchart(height = 800) %>%
  hc_add_series_treemap(tm, allowDrillToNode = TRUE,
                      layoutAlgorithm = "squarified",
                      name = "tmdata") %>%
  hc_title(text = "Gross National Income World Data") %>%
  hc_tooltip(pointFormat = "<b>{point.name}</b>:<br>
                          Pop: {point.value:,.0f}<br>
                          GNI: {point.valuecolor:,.0f}")

## End(Not run)

```

hc_add_series_ts *Shortcut for create/add time series charts from a ts object*

Description

This function add a time series to a highchart object from a ts object.

Usage

```
hc_add_series_ts(hc, ts, ...)
```

Arguments

hc	A highchart htmlwidget object.
ts	A time series object.
...	Additional arguments for the data series (http://api.highcharts.com/highcharts#series).

Details

This function **modify** the type of chart to datetime

Examples

```

## Not run:
highchart() %>%
  hc_title(text = "Monthly Airline Passenger Numbers 1949-1960") %>%
  hc_subtitle(text = "The classic Box and Jenkins airline data") %>%
  hc_add_series_ts(AirPassengers, name = "passengers") %>%
  hc_tooltip(pointFormat = '{point.y} passengers')

highchart() %>%
  hc_title(text = "Monthly Deaths from Lung Diseases in the UK") %>%
  hc_add_series_ts(fdeaths, name = "Female") %>%

```

```
hc_add_series_ts(mdeaths, name = "Male")  
  
## End(Not run)
```

hc_add_series_xts *Shortcut for create highstock chart from xts object*

Description

This function helps to create highstock charts from xts objects obtaining by getSymbols function from the **quantmod**.

Usage

```
hc_add_series_xts(hc, x, ...)
```

Arguments

hc	A highchart htmlwidget object.
x	A xts object from the quantmod package.
...	Additional shared arguments for the data series (http://api.highcharts.com/highcharts#series).

Examples

```
## Not run:  
  
library("quantmod")  
  
usdjpy <- getSymbols("USD/JPY", src="oanda", auto.assign = FALSE)  
eurkpw <- getSymbols("EUR/KPW", src="oanda", auto.assign = FALSE)  
  
highchart(type = "stock") %>%  
  hc_add_series_xts(usdjpy, id = "usdjpy") %>%  
  hc_add_series_xts(eurkpw, id = "eurkpw")  
  
## End(Not run)
```

hc_add_theme	<i>Add themes to a highchart object</i>
--------------	---

Description

Add highcharts themes to a highchart object.

Usage

```
hc_add_theme(hc, hc_thm)
```

Arguments

hc	A highchart object
hc_thm	A highchart theme object ("hc_theme" class)

Examples

```
highchart() %>%
  hc_add_series(data = c(7.0, 6.9, 9.5, 14.5, 18.2, 21.5, 25.2,
                        26.5, 23.3, 18.3, 13.9, 9.6),
               type = "column") %>%
  hc_add_theme(hc_theme_sandsignika())
```

hc_annotations	<i>Setting annotations to highcharts objects</i>
----------------	--

Description

Helper function to add annotations to highcharts library.

Usage

```
hc_annotations(hc, ...)
hc_add_annotation(hc, ...)
hc_add_annotations(hc, x)
```

Arguments

hc	A highchart htmlwidget object.
...	Arguments defined in https://api.highcharts.com/highcharts/annotations .
x	A list or a data.frame of annotations.

Details

The x elements must have xValue and yValue elements

hc_annotatationsOptions *Setting annotations options to highcharts objects*

Description

Setting annotations options to highcharts objects

Usage

```
hc_annotatationsOptions(hc, ...)
```

Arguments

hc	A highchart htmlwidget object.
...	Options defined in https://www.highcharts.com/products/plugin-registry/single/17/Annotations .

hc_boost *Setting boost module options to highcharts objects*

Description

Options for the Boost module. The Boost module allows certain series types to be rendered by WebGL instead of the default SVG. This allows hundreds of thousands of data points to be rendered in milliseconds. In addition to the WebGL rendering it saves time by skipping processing and inspection of the data wherever possible.

Usage

```
hc_boost(hc, ...)
```

Arguments

hc	A highchart htmlwidget object.
...	Options defined in https://api.highcharts.com/highcharts/boost .

`hc_chart`*Setting chart options to highchart objects*

Description

Options regarding the chart area and plot area as well as general chart options.

Usage

```
hc_chart(hc, ...)
```

Arguments

`hc` A highchart htmlwidget object.
`...` Arguments defined in <http://api.highcharts.com/highcharts/chart>.

Examples

```
data(citytemp)

hc <- highchart() %>%
  hc_xAxis(categories = citytemp$month) %>%
  hc_add_series(name = "Tokyo", data = citytemp$tokyo) %>%
  hc_add_series(name = "London", data = citytemp$london)

hc %>%
  hc_chart(type = "column",
           options3d = list(enabled = TRUE, beta = 15, alpha = 15))

hc %>%
  hc_chart(borderColor = '#EBBA95',
           borderRadius = 10,
           borderWidth = 2,
           backgroundColor = list(
             linearGradient = c(0, 0, 500, 500),
             stops = list(
               list(0, 'rgb(255, 255, 255)'),
               list(1, 'rgb(200, 200, 255)')
             )
           )))
```

hc_colorAxis	<i>Setting color Axis options to highchart objects Function to set the axis color to highcharts objects.</i>
--------------	--

Description

Setting color Axis options to highchart objects Function to set the axis color to highcharts objects.

Usage

```
hc_colorAxis(hc, ...)
```

Arguments

hc	A highchart htmlwidget object.
...	Arguments are defined in http://api.highcharts.com/highmaps/colorAxis .

Examples

```
nyears <- 5

df <- expand.grid(seq(12) - 1, seq(nyears) - 1)
df$value <- abs(seq(nrow(df)) + 10 * rnorm(nrow(df))) + 10
df$value <- round(df$value, 2)
ds <- list_parse2(df)

hc <- highchart() %>%
  hc_chart(type = "heatmap") %>%
  hc_title(text = "Simulated values by years and months") %>%
  hc_xAxis(categories = month.abb) %>%
  hc_yAxis(categories = 2016 - nyears + seq(nyears)) %>%
  hc_add_series(name = "value", data = ds)

hc_colorAxis(hc, minColor = "#FFFFFF", maxColor = "#434348")

hc_colorAxis(hc, minColor = "#FFFFFF", maxColor = "#434348",
             type = "logarithmic")

require("viridisLite")

n <- 4
stops <- data.frame(q = 0:n/n,
                  c = substring(viridis(n + 1), 0, 7),
                  stringsAsFactors = FALSE)
stops <- list_parse2(stops)
```

```
hc_colorAxis(hc, stops = stops, max = 75)
```

hc_colors *Setting color options to highchart objects*

Description

An array containing the default colors for the chart's series. When all colors are used, new colors are pulled from the start again.

Usage

```
hc_colors(hc, colors)
```

Arguments

hc A highchart htmlwidget object.
colors A vector of colors.

Examples

```
library("viridisLite")  
  
cols <- viridis(3)  
cols <- substr(cols, 0, 7)  
  
highcharts_demo() %>%  
  hc_colors(cols)
```

hc_credits *Setting credits options to highchart objects*

Description

highcharters by default don't put credits in the chart. You can add credits using these options.

Usage

```
hc_credits(hc, ...)
```


Arguments

hc A highchart htmlwidget object.
 ... Arguments defined in <http://api.highcharts.com/highcharts/credits>.

Examples

```
data("citytemp")

highchart() %>%
  hc_xAxis(categories = citytemp$month) %>%
  hc_add_series(name = "Tokyo", data = citytemp$tokyo, type = "bar") %>%
  hc_credits(enabled = TRUE, text = "htmlwidgets.org",
            href = "http://www.htmlwidgets.org")
```

hc_defs	<i>Setting patterns to be used in highcharts series</i>
---------	---

Description

Helper function to use the fill patten plugin <http://www.highcharts.com/plugin-registry/single/9/Pattern-Fill>.

Usage

```
hc_defs(hc, ...)
```

Arguments

hc A highchart htmlwidget object.
 ... Arguments defined in <http://www.highcharts.com/plugin-registry/single/9/Pattern-Fill>.

hc_drilldown	<i>Setting drilldown options for highcharts objects</i>
--------------	---

Description

Options for drill down, the concept of inspecting increasingly high resolution data through clicking on chart items like columns or pie slices.

Usage

```
hc_drilldown(hc, ...)
```

Arguments

hc A highchart htmlwidget object.
... Arguments defined in <http://api.highcharts.com/highcharts/drilldown>.

Examples

```
library("dplyr")
library("purrr")

df <- data_frame(
  name = c("Animals", "Fruits", "Cars"),
  y = c(5, 2, 4),
  drilldown = tolower(name)
)

df

ds <- list_parse(df)
names(ds) <- NULL
str(ds)

hc <- highchart() %>%
  hc_chart(type = "columnn") %>%
  hc_title(text = "Basic drilldown") %>%
  hc_xAxis(type = "category") %>%
  hc_legend(enabled = FALSE) %>%
  hc_plotOptions(
    series = list(
      boderWidth = 0,
      dataLabels = list(enabled = TRUE)
    )
  ) %>%
  hc_add_series(
    name = "Things",
    colorByPoint = TRUE,
    data = ds
  )

dfan <- data_frame(
  name = c("Cats", "Dogs", "Cows", "Sheep", "Pigs"),
  value = c(4, 3, 1, 2, 1)
)

dffru <- data_frame(
  name = c("Apple", "Organes"),
  value = c(4, 2)
)

dfcar <- data_frame(
  name = c("Toyota", "Opel", "Volkswage"),
  value = c(4, 2, 2)
)
```

```
)  
  
second_el_to_numeric <- function(ls){  
  
  map(ls, function(x){  
    x[[2]] <- as.numeric(x[[2]])  
    x  
  })  
  
}  
  
dsan <- second_el_to_numeric(list_parse2(dfan))  
  
dsfru <- second_el_to_numeric(list_parse2(dffru))  
  
dscar <- second_el_to_numeric(list_parse2(dfcar))  
  
hc <- hc %>%  
  hc_drilldown(  
    allowPointDrilldown = TRUE,  
    series = list(  
      list(  
        id = "animals",  
        data = dsan  
      ),  
      list(  
        id = "fruits",  
        data = dsfru  
      ),  
      list(  
        id = "cars",  
        data = dscar  
      )  
    )  
  )  
  
hc
```

hc_elementId	<i>Setting</i> elementId
--------------	--------------------------

Description

Function to modify the id for the container.

Usage

```
hc_elementId(hc, id = NULL)
```

Arguments

hc	A highchart htmlwidget object.
id	A string

Examples

```
hchart(rnorm(10)) %>%
  hc_elementId("newid")
```

hc_exporting	<i>Setting exporting options for highcharts objects</i>
--------------	---

Description

Exporting options for highcharts objects. You can define the file's name or the output format.

Usage

```
hc_exporting(hc, ...)
```

Arguments

hc	A highchart htmlwidget object.
...	Arguments defined in http://api.highcharts.com/highcharts/exporting .

Examples

```
require("dplyr")

data("citytemp")

highchart() %>%
  hc_xAxis(categories = citytemp$month) %>%
  hc_add_series(name = "Tokyo", data = citytemp$tokyo) %>%
  hc_add_series(name = "London", data = citytemp$london) %>%
  hc_exporting(enabled = TRUE,
              filename = "custom-file-name")
```

hc_legend *Setting legend options to highchart objects*

Description

Function to modify styles for the box containing the symbol, name and color for each item or point item in the chart.

Usage

```
hc_legend(hc, ...)
```

Arguments

hc A highchart htmlwidget object.
... Arguments are defined in <http://api.highcharts.com/highcharts#legend>.

Examples

```
data(citytemp)

highchart() %>%
  hc_xAxis(categories = citytemp$month) %>%
  hc_add_series(name = "Tokyo", data = citytemp$tokyo) %>%
  hc_add_series(name = "London", data = citytemp$london) %>%
  hc_legend(align = "left", verticalAlign = "top",
            layout = "vertical", x = 0, y = 100)
```

hc_mapNavigation *Setting mapNavigation options to highmaps charts*

Description

Options regarding the mapNavigation: A collection of options for zooming and panning in a map.

Usage

```
hc_mapNavigation(hc, ...)
```

Arguments

hc A highchart htmlwidget object.
... Arguments defined in <http://api.highcharts.com/highmaps#mapNavigation>.

hc_motion	<i>Setting Motion options to highcharts objects</i>
-----------	---

Description

The Motion Highcharts Plugin adds an interactive HTML5 player to any Highcharts chart (Highcharts, Highmaps and Highstock).

Usage

```
hc_motion(hc, enabled = TRUE, startIndex = 0, ...)
```

Arguments

hc	A highchart htmlwidget object.
enabled	Enable the motion plugin.
startIndex	start index, default to 0.
...	Arguments defined in https://github.com/larsac07/Motion-Highcharts-Plugin/wiki .

hc_navigator	<i>Setting navigator options to highstock charts Options regarding the navigator: The miniseries below chart in a highstock chart.</i>
--------------	--

Description

Setting navigator options to highstock charts Options regarding the navigator: The miniseries below chart in a highstock chart.

Usage

```
hc_navigator(hc, ...)
```

Arguments

hc	A highchart htmlwidget object.
...	Arguments defined in http://api.highcharts.com/highstock#navigator .

hc_pane	<i>Setting panes options to highchart objects</i>
---------	---

Description

Applies only to polar charts and angular gauges. This configuration object holds general options for the combined X and Y axes set. Each xAxis or yAxis can reference the pane by index.

Usage

```
hc_pane(hc, ...)
```

Arguments

hc	A highchart htmlwidget object.
...	Arguments defined in http://api.highcharts.com/highcharts/pane .

hc_plotOptions	<i>Setting plot options to highchart objects</i>
----------------	--

Description

The plotOptions is a wrapper object for config objects for each series type. The configuration objects for each series can also be overridden for each series item as given in the series array.

Usage

```
hc_plotOptions(hc, ...)
```

Arguments

hc	A highchart htmlwidget object.
...	Arguments are defined in http://api.highcharts.com/highcharts#plotOptions .

Details

Configuration options for the series are given in three levels. Options for all series in a chart are given with the hc_plotOptions function. Then options for all series of a specific type are given in the plotOptions of that type, for example hc_plotOptions(line = list(...)). Next, options for one single series are given in the series array.

Examples

```

data(citytemp)

hc <- highchart() %>%
  hc_plotOptions(line = list(color = "blue",
                             marker = list(
                               fillColor = "white",
                               lineWidth = 2,
                               lineColor = NULL
                             )
                )) %>%
  hc_add_series(name = "Tokyo", data = citytemp$tokyo) %>%
  hc_add_series(name = "London", data = citytemp$london,
                marker = list(fillColor = "black"))

hc

# override the `blue` option with the explicit parameter
hc %>%
  hc_add_series(name = "London",
                data = citytemp$new_york,
                color = "red")

```

 hc_rangeSelector

Setting scrollbar options to highstock charts

Description

Options to edit the range selector which is The range selector is a tool for selecting ranges to display within the chart. It provides buttons to select preconfigured ranges in the chart, like 1 day, 1 week, 1 month etc. It also provides input boxes where min and max dates can be manually input.

Usage

```
hc_rangeSelector(hc, ...)
```

Arguments

hc A highchart htmlwidget object.

... Arguments defined in <http://api.highcharts.com/highstock#rangeSelector>.

hc_responsive	<i>Setting responsive options to highchart objects</i>
---------------	--

Description

Allows setting a set of rules to apply for different screen or chart sizes. Each rule specifies additional chart options.

Usage

```
hc_responsive(hc, ...)
```

Arguments

hc	A highchart htmlwidget object.
...	Arguments defined in http://api.highcharts.com/highcharts/responsive .

Examples

```
leg_500_opts <- list(enabled = FALSE)
leg_900_opts <- list(align = "right", verticalAlign = "middle", layout = "vertical")

highcharts_demo() %>%
  hc_responsive(
    rules = list(
      # remove legend if there is no much space
      list(
        condition = list(maxWidth = 500),
        chartOptions = list(legend = leg_500_opts)
      ),
      # put legend on the right when there is much space
      list(
        condition = list(minWidth = 900),
        chartOptions = list(legend = leg_900_opts)
      )
    )
  )
)
```

hc_rm_series	<i>Removing series to highchart objects</i>
--------------	---

Description

Removing series to highchart objects

Usage

```
hc_rm_series(hc, names = NULL)
```

Arguments

hc	A highchart htmlwidget object.
names	The series's names to delete.

hc_scrollbar	<i>Setting scrollbar options to highstock objects</i>
--------------	---

Description

Options regarding the scrollbar which is a means of panning over the X axis of a chart.

Usage

```
hc_scrollbar(hc, ...)
```

Arguments

hc	A highchart htmlwidget object.
...	Arguments defined in http://api.highcharts.com/highstock#scrollbar .

hc_series	<i>Setting series/data options from highchart objects</i>
-----------	---

Description

Setting series/data options from highchart objects

Usage

```
hc_series(hc, ...)
```

Arguments

hc	A highchart htmlwidget object.
...	Arguments defined in http://api.highcharts.com/highcharts/series .

Examples

```
highchart() %>%
  hc_series(
    list(
      name = "Tokyo",
      data = c(7.0, 6.9, 9.5, 14.5, 18.4, 21.5, 25.2, 26.5, 23.3, 18.3, 13.9, 9.6)
    ),
    list(
      name = "London",
      data = c(3.9, 4.2, 5.7, 8.5, 11.9, 15.2, 17.0, 16.6, 14.2, 10.3, 6.6, 4.8)
    )
  )
)
```

hc_size

Changing the size of a highchart object

Description

Changing the size of a highchart object

Usage

```
hc_size(hc, width = NULL, height = NULL)
```

Arguments

hc	A highchart htmlwidget object.
width	A numeric input in pixels.
height	A numeric input in pixels.

Examples

```
hc_size(hcts(rnorm(100)), 400, 200)
```

`hc_theme`*Highchart theme constructor*

Description

Function to create highcharts themes.

Usage

```
hc_theme(...)
```

Arguments

... A named parameters.

Details

More examples are in <http://www.highcharts.com/docs/chart-design-and-style/themes>.

Examples

```
hc <- highcharts_demo()

hc

thm <- hc_theme(
  colors = c('red', 'green', 'blue'),
  chart = list(
    backgroundColor = "#15C0DE"
  ),
  title = list(
    style = list(
      color = '#333333',
      fontFamily = "Erica One"
    )
  ),
  subtitle = list(
    style = list(
      color = '#666666',
      fontFamily = "Shadows Into Light"
    )
  ),
  legend = list(
    itemStyle = list(
      fontFamily = 'Tangerine',
      color = 'black'
    ),
    itemHoverStyle = list(
      color = 'gray'
    )
  )
)
```

```
    )  
  )  
)  
  
hc %>% hc_add_theme(thm)
```

hc_theme_538

Fivethirtyeight theme for highcharts

Description

Fivethirtyeight theme for highcharts

Usage

```
hc_theme_538(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_538())  
  
highcharts_demo() %>%  
  hc_add_theme(  
    hc_theme_538(  
      colors = c("red", "blue", "green"),  
      chart = list(backgroundColor = "white")  
    )  
  )
```

hc_theme_chalk

Chalk theme for highcharts

Description

Chalk theme for highcharts. Inspired by <https://www.amcharts.com/inspiration/hand-drawn/>.

Usage

```
hc_theme_chalk(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_chalk())
```

hc_theme_darkunica *Dark Unica theme for highcharts*

Description

Dark Unica theme for highcharts

Usage

```
hc_theme_darkunica(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_darkunica())
```

hc_theme_db *Dotabuff theme for highcharts*

Description

Dotabuff theme for highcharts

Usage

```
hc_theme_db(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_db())
```

hc_theme_economist *Economist theme for highcharts*

Description

Economist theme for highcharts

Usage

```
hc_theme_economist(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_economist())
```

hc_theme_elementary *Elementary (OS) theme for highcharts*

Description

Elementary (OS) theme for highcharts based on <https://elementary.io>

Usage

```
hc_theme_elementary(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_elementary())
```

hc_theme_ffx	<i>Firefox theme for highcharts</i>
--------------	-------------------------------------

Description

Theme inspired by <https://www.mozilla.org/en-US/styleguide/>.

Usage

```
hc_theme_ffx(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_ffx())
```

hc_theme_flat	<i>Flat theme for highcharts</i>
---------------	----------------------------------

Description

Base16 inspired theme <https://github.com/chriskempson/base16> and <https://github.com/cttobin/ggthemr#flat>

Usage

```
hc_theme_flat(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_flat())
```

hc_theme_flatdark *Flatdark theme for highcharts*

Description

Base16 inspired theme <https://github.com/chriskempson/base16> and <https://github.com/cttobin/ggthemr#flat>

Usage

```
hc_theme_flatdark(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_flatdark())
```

hc_theme_ft *Financial Times theme for highcharts*

Description

Financial Times theme for highcharts

Usage

```
hc_theme_ft(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_ft())
```

hc_theme_ggplot2 *ggplot2 theme for highcharts*

Description

Based on <https://ggplot2.tidyverse.org/>.

Usage

```
hc_theme_ggplot2(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_ggplot2())
```

hc_theme_google *Google theme for highcharts*

Description

Google theme for highcharts based on <https://books.google.com/ngrams/>.

Usage

```
hc_theme_google(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_google())
```

hc_theme_gridlight *Grid Light theme for highcharts*

Description

Grid Light theme for highcharts

Usage

```
hc_theme_gridlight(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_gridlight())
```

hc_theme_handdrawn *Hand Drawn theme for highcharts*

Description

Hand Drawn theme for highcharts. Inspired by <https://www.amcharts.com/inspiration/hand-drawn/>.

Usage

```
hc_theme_handdrawn(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_handdrawn())
```

hc_theme_merge *Merge themes*

Description

Function to combine hc_theme objects.

Usage

```
hc_theme_merge(...)
```

Arguments

... hc_theme objects.

Examples

```
thm <- hc_theme_merge(  
  hc_theme_darkunica(),  
  hc_theme(  
    chart = list(  
      backgroundColor = "transparent",  
      divBackgroundImage = "http://cdn.wall-pix.net/albums/art-3Dview/00025095.jpg"  
    ),  
    title = list(  
      style = list(  
        color = 'white',  
        fontFamily = "Erica One"  
      )  
    )  
  )  
)
```

hc_theme_monokai *Monokai theme for highcharts*

Description

A well know text editor theme

Usage

```
hc_theme_monokai(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_monokai())
```

hc_theme_null *Null theme for highcharts*

Description

Null theme for highcharts. Axis are removed (`visible = FALSE`).

Usage

```
hc_theme_null(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_null())
```

hc_theme_sandsignika *Sand Signika theme for highcharts*

Description

Sand Signika theme for highcharts

Usage

```
hc_theme_sandsignika(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_sandsignika())
```

hc_theme_smpl	<i>Simple theme for highcharts</i>
---------------	------------------------------------

Description

Design inspired by <https://github.com/hrbrmstr/hrbrmisc/blob/master/R/themes.r> and color by <https://www.materialui.co/flatuicolors>

Usage

```
hc_theme_smpl(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_smpl())
```

hc_theme_sparkline	<i>Sparkline theme for highcharts</i>
--------------------	---------------------------------------

Description

Based on <http://www.highcharts.com/demo/sparkline>.

Usage

```
hc_theme_sparkline(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_sparkline())
```

hc_theme_superheroes *Superheroes theme for highcharts*

Description

Inspired by <https://public.tableau.com/profile/ryansmith#!/vizhome/HeroesofNewYork/SuperheroesinNewYork>

Usage

```
hc_theme_superheroes(...)
```

Arguments

... Named argument to modify the theme

Examples

```
highcharts_demo() %>%  
  hc_add_theme(hc_theme_superheroes())
```

hc_theme_tufte *Tufte theme for highcharts*

Description

Design inspired by Edward Tufte style.

Usage

```
hc_theme_tufte(...)  
  
hc_theme_tufte2(...)
```

Arguments

... Named argument to modify the theme

Examples

```
n <- 15
dta <- dplyr::data_frame(
  x = rnorm(n),
  y = 1.5 * x + rnorm(n))
highchart() %>%
  hc_chart(type = "scatter") %>%
  hc_add_series(data = list_parse(dta)) %>%
  hc_add_theme(hc_theme_tufte())

values <- 1 + abs(rnorm(12))
highchart() %>%
  hc_chart(type = "column") %>%
  hc_add_series(data = values) %>%
  hc_xAxis(categories = month.abb) %>%
  hc_add_theme(hc_theme_tufte2())
```

 hc_title

Setting title and subtitle options to highchart objects

Description

Function to add and change title and subtitle's style.

Usage

```
hc_title(hc, ...)

hc_subtitle(hc, ...)
```

Arguments

hc A highchart htmlwidget object.
 ... Arguments are defined in <http://api.highcharts.com/highcharts#title>.

Examples

```
highchart() %>%
  hc_add_series(data = c(7.0, 6.9, 9.5, 14.5, 18.2, 21.5, 25.2,
    26.5, 23.3, 18.3, 13.9, 9.6),
    type = "column") %>%
  hc_title(text = "This is a title with <i>margin</i> and <b>Strong or bold text</b>",
    margin = 20, align = "left",
    style = list(color = "#90ed7d", useHTML = TRUE)) %>%
  hc_subtitle(text = "And this is a subtitle with more information",
```



```
align = "left", style = list(color = "#2b908f", fontWeight = "bold"))
```

hc_tooltip

Setting tooltip options to highchart objects

Description

Options for the tooltip that appears when the user hovers over a series or point.

Usage

```
hc_tooltip(hc, ..., sort = FALSE, table = FALSE)
```

Arguments

hc	A highchart htmlwidget object.
...	Arguments are defined in http://api.highcharts.com/highcharts#tooltip .
sort	Logical value to implement sort according this.point http://stackoverflow.com/a/16954666/829971 .
table	Logical value to implement table in tooltip: http://stackoverflow.com/a/22327749/829971 .

Examples

```
highcharts_demo() %>%  
  hc_tooltip(crosshairs = TRUE, borderWidth = 5, sort = TRUE, table = TRUE)
```

hc_xAxis

Setting axis options to highchart objects

Description

Change axis labels or style. Add lines or band to charts.

Usage

```
hc_xAxis(hc, ...)  
  
hc_yAxis(hc, ...)  
  
hc_yAxis_multiples(hc, ...)  
  
hc_zAxis(hc, ...)
```

Arguments

hc A highchart htmlwidget object.
 ... Arguments defined in <http://api.highcharts.com/highcharts/xAxis>.

Examples

```
highchart() %>%
  hc_add_series(data = c(7.0, 6.9, 9.5, 14.5, 18.2, 21.5, 25.2,
                        26.5, 23.3, 18.3, 13.9, 9.6),
               type = "spline") %>%
  hc_xAxis(title = list(text = "x Axis at top"),
           opposite = TRUE,
           plotLines = list(
             list(label = list(text = "This is a plotLine"),
                  color = "#FF0000",
                  width = 2,
                  value = 5.5))) %>%
  hc_yAxis(title = list(text = "y Axis at right"),
           opposite = TRUE,
           minorTickInterval = "auto",
           minorGridLineDashStyle = "LongDashDotDot",
           showFirstLabel = FALSE,
           showLastLabel = FALSE,
           plotBands = list(
             list(from = 25, to = 80, color = "rgba(100, 0, 0, 0.1)",
                  label = list(text = "This is a plotBand")))))

highchart() %>%
  hc_yAxis_multiples(
    list(top = "0%", height = "30%", lineWidth = 3),
    list(top = "30%", height = "70%", offset = 0,
         showFirstLabel = FALSE, showLastLabel = FALSE)
  ) %>%
  hc_add_series(data = rnorm(10)) %>%
  hc_add_series(data = rexp(10), type = "spline", yAxis = 1)
```

 hex_to_rgba

Transform colors from hexadecimal format to rgba hc notation

Description

Transform colors from hexadecimal format to rgba hc notation

Usage

```
hex_to_rgba(x, alpha = 1)
```

Arguments

x	colors in hexadecimal format
alpha	alpha

Examples

```
hex_to_rgba(x <- c("#440154", "#21908C", "#FDE725"))
```

highchart	<i>Create a Highcharts chart widget</i>
-----------	---

Description

This function creates a Highchart chart using **htmlwidgets**. The widget can be rendered on HTML pages generated from R Markdown, Shiny, or other applications.

Usage

```
highchart(hc_opts = list(), theme = getOption("highcharter.theme"),  
  type = "chart", width = NULL, height = NULL, elementId = NULL)
```

Arguments

hc_opts	A list object containing options defined as http://api.highcharts.com/highcharts .
theme	A hc_theme class object-
type	A character value to set if use Highchart, Highstock or Highmap. Options are "chart", "stock" and "map".
width	A numeric input in pixels.
height	A numeric input in pixels.
elementId	Use an explicit element ID for the widget.

highchart2 *Create a Highcharts chart widget*

Description

This widgets don't support options yet.

Usage

```
highchart2(hc_opts = list(), theme = NULL, width = NULL,
           height = NULL, elementId = NULL, debug = FALSE)
```

```
highchartzero(hc_opts = list(), theme = NULL, width = NULL,
              height = NULL, elementId = NULL)
```

Arguments

hc_opts	A list object containing options defined as http://api.highcharts.com/highcharts .
theme	A hc_theme class object
width	A numeric input in pixels.
height	A numeric input in pixels.
elementId	Use an explicit element ID for the widget.
debug	A boolean value if you want to print in the browser console the parameters given to highchart.

Details

This function creates a Highchart chart using **htmlwidgets**. The widget can be rendered on HTML pages generated from R Markdown, Shiny, or other applications.

highcharter *An htmlwidget interface to the Highcharts javascript chart library*

Description

Highcharts <http://www.highcharts.com/> is a mature javascript charting library. Highcharts provide a various type of charts, from scatters to heatmaps or treemaps.

Author(s)

Joshua Kunst (@jbkunst)

highcharter-exports *highcharter exported operators and S3 methods*

Description

The following functions are imported and then re-exported from the highcharter package to avoid listing the magrittr as Depends of highcharter.

highchartOutput *Widget output function for use in Shiny*

Description

Widget output function for use in Shiny

Usage

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

```
highchartOutput2(outputId, width = "100%", height = "400px")
```

Arguments

outputId	The name of the input.
width	A numeric input in pixels.
height	A numeric input in pixels.

highcharts_demo *Chart a demo for testing themes*

Description

Chart a demo for testing themes

Usage

```
highcharts_demo()
```

Examples

```
highcharts_demo()
```

highchart_ct	<i>Highcharter Crosstalk Widget</i>
--------------	-------------------------------------

Description

Highcharter Crosstalk Widget

Usage

```
highchart_ct(data = NULL, hc_opts = list(),
             theme = getOption("highcharter.theme"), width = NULL,
             height = NULL, elementId = NULL)
```

Arguments

data	Data frame
hc_opts	A list object containing options defined as http://api.highcharts.com/highcharts .
theme	A hc_theme class object.
width	A numeric input in pixels.
height	A numeric input in pixels.
elementId	Use an explicit element ID for the widget.

hw_grid	<i>Lays out highchart widgets into a "grid", similar to grid.arrange from gridExtra</i>
---------	---

Description

Lays out highchart widgets into a "grid", similar to grid.arrange from gridExtra

Usage

```
hw_grid(..., ncol = NULL, rowheight = NULL)
```

Arguments

...	either individual highchart objects or a mixture of individual highchart objects and lists of highchart objects.
ncol	how many columns in the grid
rowheight	Height in px.

is.hexcolor	<i>Check if a string vector is in hexadecimal color format</i>
-------------	--

Description

Check if a string vector is in hexadecimal color format

Usage

```
is.hexcolor(x)
```

Arguments

x	A string vectors
---	------------------

Examples

```
x <- c("#f0f0f0", "#FFF", "#99990000", "#00FFFFFF")
is.hexcolor(x)
```

is.highchart	<i>Reports whether x is a highchart object</i>
--------------	--

Description

Reports whether x is a highchart object

Usage

```
is.highchart(x)
```

Arguments

x	An object to test
---	-------------------

list_parse	<i>Convert an object to list with identical structure</i>
------------	---

Description

This functions are similar to `rlist::list.parse` but this removes names.

Usage

```
list_parse(df)
list_parse2(df)
```

Arguments

df A data frame to parse to list

Examples

```
x <- data.frame(a=1:3, type=c('A','C','B'), stringsAsFactors = FALSE)
list_parse(x)
list_parse2(x)
```

mutate_mapping	<i>Modify data frame according to mapping</i>
----------------	---

Description

Modify data frame according to mapping

Usage

```
mutate_mapping(data, mapping, drop = FALSE)
```

Arguments

data A data frame object.
mapping A mapping from hcaes function.
drop A logical argument to you drop variables or not. Default is FALSE

Examples

```
df <- head(mtcars)
mutate_mapping(data = df, mapping = hcaes(x = cyl, y = wt + cyl, group = gear))
mutate_mapping(data = df, mapping = hcaes(x = cyl, y = wt), drop = TRUE)
```

pokemon

pokemon

Description

Information about 718 pokemon.

Usage

pokemon

Format

A data frame with 718 observations and 20 variables.

Variables

- id:
- pokemon:
- species_id:
- height:
- weight:
- base_experience:
- type_1:
- type_2:
- attack:
- defense:
- hp:
- special_attack:
- special_defense:
- speed:
- color_1:
- color_2:
- color_f:
- egg_group_1:
- egg_group_2:
- url_image:

Source

jkunst.com/r/pokemon-visualize-em-all/

random_id	<i>Function to generate iids</i>
-----------	----------------------------------

Description

Function to generate iids

Usage

```
random_id(n = 1, length = 10)
```

Arguments

n	Number of ids
length	Length of ids

renderHighchart	<i>Widget render function for use in Shiny</i>
-----------------	--

Description

Widget render function for use in Shiny

Usage

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

```
renderHighchart2(expr, env = parent.frame(), quoted = FALSE)
```

Arguments

expr	A highchart expression.
env	A environment.
quoted	A boolean value.

stars	<i>stars</i>
-------	--------------

Description

A sample using by Nadieh Bremer blocks. <http://bl.ocks.org/nbremer/eb0d1fd4118b731d069e2ff98dfadc47>.

Usage

```
stars
```

Format

A data frame with 404 observations and 6 variables.

Variables

- bv: BV
- absmag: Magnitude
- lum: Luminosity
- temp: Temperature
- radiussun: Radius
- distance: Distance

str_to_id	<i>String to 'id' format</i>
-----------	------------------------------

Description

Turn a string to id format used in treemaps.

Usage

```
str_to_id(x)
```

Arguments

x A vector string.

Examples

```
str_to_id(" A string _ with sd / sdg    Underscores \    ")
```

tooltip_chart	<i>Helper to create charts in tooltips.</i>
---------------	---

Description

Helper to create charts in tooltips.

Usage

```
tooltip_chart(accessor = NULL, hc_opts = NULL, width = 250,  
             height = 150)
```

Arguments

accessor	A string indicating the name of the column where the data is.
hc_opts	A list of options using the http://api.highcharts.com/highcharts syntax.
width	A numeric input in pixels indicating the width of the tooltip.
height	A numeric input in pixels indicating the height of the tooltip.

Details

This function needs to be used in the `pointFormatter` argument inside of `hc_tooltip` function and `useHTML = TRUE` option.

Examples

```
## Not run:  
require(dplyr)  
require(purrr)  
require(tidyr)  
require(gapminder)  
data(gapminder, package = "gapminder")  
  
gp <- gapminder %>%  
  arrange(desc(year)) %>%  
  distinct(country, .keep_all = TRUE)  
  
gp2 <- gapminder %>%  
  nest(-country) %>%  
  mutate(data = map(data, mutate_mapping, hcaes(x = lifeExp, y = gdpPercap), drop = TRUE),  
         data = map(data, list_parse)) %>%  
  rename(ttdata = data)  
  
gptot <- left_join(gp, gp2)  
  
hc <- hchart(
```

```

        gptot,
        "point",
        hcaes(
            lifeExp,
            gdpPercap,
            name = country,
            size = pop,
            group = continent
        )
    ) %>%
    hc_yAxis(type = "logarithmic")

hc %>%
  hc_tooltip(useHTML = TRUE, pointFormatter = tooltip_chart(accessor = "ttdata"))

hc %>%
  hc_tooltip(useHTML = TRUE, pointFormatter = tooltip_chart(
    accessor = "ttdata",
    hc_opts = list(chart = list(type = "column"))
  ))

hc %>%
  hc_tooltip(
    useHTML = TRUE,
    positioner = JS("function () { return { x: this.chart.plotLeft + 10, y: 10}; }"),
    pointFormatter = tooltip_chart(
      accessor = "ttdata",
      hc_opts = list(
        title = list(text = "point.country"),
        xAxis = list(title = list(text = "lifeExp")),
        yAxis = list(title = list(text = "gdpPercap"))
      )
    )
  )

hc %>%
  hc_tooltip(
    useHTML = TRUE,
    pointFormatter = tooltip_chart(
      accessor = "ttdata",
      hc_opts = list(
        legend = list(enabled = TRUE),
        series = list(list(color = "gray", name = "point.name"))
      )
    )
  )

## End(Not run)

```

Description

Helper to make table in tooltips for the pointFormat parameter in hc_tooltip

Usage

```
tooltip_table(x, y, title = NULL, img = NULL, ...)
```

Arguments

x	A string vector with description text
y	A string with accessors example: point.series.name, point.x
title	A title tag with accessors or string
img	Image tag
...	html attributes for the table element

Examples

```
x <- c("Income:", "Genre", "Runtime")
y <- c("${point.y}", "{point.series.options.extra.genre}",
      "{point.series.options.extra.runtime}")

tooltip_table(x, y)
```

unemployment

US Counties unemployment rate

Description

This data comes from the <https://www.highcharts.com/samples/data/jsonp.php?filename=us-counties-unemployment.json> and is used in highmaps examples.

Usage

```
unemployment
```

Format

A data.frame with 3 variables and 3.216 observations.

Variables

- code: The county code.
- name: The county name.
- value: The unemployment.

Source

<https://www.highcharts.com/samples/data/jsonp.php?filename=us-counties-unemployment.json>

uscountygeojson	<i>US Counties map in Geojson format (list)</i>
-----------------	---

Description

This data comes from the <https://code.highcharts.com/mapdata/countries/us/us-all-all.js> and is used in highmaps examples.

Usage

uscountygeojson

Format

A list in geojson format.

usgeojson	<i>US States map in Geojson format (list)</i>
-----------	---

Description

This data comes from the <https://code.highcharts.com/mapdata/countries/us/us-all.js> and is used in highmaps examples.

Usage

usgeojson

Format

A list in geojson format.

vaccines	<i>Vaccines</i>
----------	-----------------

Description

The number of infected people by Measles, measured over 70-some years and across all 50 states. From the WSJ analysis: <http://graphics.wsj.com/infectious-diseases-and-vaccines/>

Usage

vaccines

Format

A data frame with 3,876 observations and 3 variables.

Variables

- year: Year
- state: Name of the state
- count: Number of cases per 100,000 people. If the value is NA the count was 0.

weather	<i>Weather</i>
---------	----------------

Description

Temperature information of San Francisco.

Usage

weather

Format

A data frame with 365 observations and 4 variables.

Variables

- date: Day in date format.
- min_temperaturec: Minimum temperature.
- max_temperaturec: Maximun temperature.
- mean_temperaturec: Mean temperature.

worldgeojson

World map in Geojson format (list)

Description

This data comes from the <https://code.highcharts.com/mapdata/custom/world.js> and is used in `highmaps examples.#'`

Usage

worldgeojson

Format

A list in geojson format.

Index

*Topic **datasets**

- citytemp, 5
- favorite_bars, 10
- favorite_pies, 10
- globaltemp, 12
- pokemon, 81
- stars, 83
- unemployment, 86
- uscountygeojson, 87
- usgeojson, 87
- vaccines, 88
- weather, 88
- worldgeojson, 89
- %>% (highcharter-exports), 77
- citytemp, 5
- color_classes, 6
- color_stops, 6
- colorize, 5
- create_yaxis, 7
- dash_styles, 7
- datetime_to_timestamp, 8
- download_map_data, 8, 11
- export_hc, 9
- fa_icon (hc_add_dependency_fa), 25
- fa_icon_mark (hc_add_dependency_fa), 25
- favorite_bars, 10
- favorite_pies, 10
- fix_1_length_data, 11
- get_data_from_map, 11
- get_hc_series_from_df, 12
- globaltemp, 12
- hc_accessibility, 24
- hc_add_annotation (hc_annotations), 44
- hc_add_annotations (hc_annotations), 44
- hc_add_dependency, 24
- hc_add_dependency_fa, 25
- hc_add_event_point, 26
- hc_add_event_series
 - (hc_add_event_point), 26
- hc_add_series, 27
- hc_add_series.character, 27
- hc_add_series.data.frame, 28
- hc_add_series.density, 28
- hc_add_series.factor
 - (hc_add_series.character), 27
- hc_add_series.forecast, 29
- hc_add_series.geo_json, 29
- hc_add_series.geo_list
 - (hc_add_series.geo_json), 29
- hc_add_series.lm, 30
- hc_add_series.loess (hc_add_series.lm), 30
- hc_add_series.numeric, 30
- hc_add_series.ohlcv (hc_add_series.xts), 31
- hc_add_series.ts, 31
- hc_add_series.xts, 31
- hc_add_series_boxplot, 32
- hc_add_series_df, 32
- hc_add_series_flags, 34
- hc_add_series_labels_values, 35
- hc_add_series_list, 36
- hc_add_series_map, 37
- hc_add_series_ohlcv, 38
- hc_add_series_scatter, 39
- hc_add_series_times_values, 40
- hc_add_series_treemap, 41
- hc_add_series_ts, 42
- hc_add_series_xts, 43
- hc_add_theme, 44
- hc_annotations, 44
- hc_annotationsOptions, 45
- hc_boost, 45
- hc_chart, 46

hc_colorAxis, 47
hc_colors, 48
hc_credits, 48
hc_defs, 49
hc_drilldown, 49
hc_elementId, 51
hc_exporting, 52
hc_legend, 53
hc_mapNavigation, 53
hc_motion, 54
hc_navigator, 54
hc_pane, 55
hc_plotOptions, 55
hc_rangeSelector, 56
hc_responsive, 57
hc_rm_series, 57
hc_scrollbar, 58
hc_series, 58
hc_size, 59
hc_subtitle (hc_title), 72
hc_theme, 60
hc_theme_538, 61
hc_theme_chalk, 61
hc_theme_darkunica, 62
hc_theme_db, 62
hc_theme_economist, 63
hc_theme_elementary, 63
hc_theme_ffx, 64
hc_theme_flat, 64
hc_theme_flatdark, 65
hc_theme_ft, 65
hc_theme_ggplot2, 66
hc_theme_google, 66
hc_theme_gridlight, 67
hc_theme_handdrawn, 67
hc_theme_merge, 68
hc_theme_monokai, 68
hc_theme_null, 69
hc_theme_sandsignika, 69
hc_theme_smpl, 70
hc_theme_sparkline, 70
hc_theme_superheroes, 71
hc_theme_tufte, 71
hc_theme_tufte2 (hc_theme_tufte), 71
hc_title, 72
hc_tooltip, 73
hc_xAxis, 73
hc_yAxis (hc_xAxis), 73
hc_yAxis_multiples (hc_xAxis), 73
hc_zAxis (hc_xAxis), 73
hcaes, 13
hcaes_ (hcaes_string), 13
hcaes_string, 13
hcbars, 14
hcbboxplot, 14
hcdensity, 15
hchart, 15
hchart.survfit, 16
hchart, 17
hciconarray, 17
hcmmap, 8, 18
hccparcords, 19
hccpie, 20
hccspark, 20
hcctreemap, 21
hcctreemap2, 22
hcts, 23
hex_to_rgba, 74
highchart, 75
highchart2, 76
highchart_ct, 78
highcharter, 76
highcharter-exports, 77
highcharter-package (highcharter), 76
highchartOutput, 77
highchartOutput2 (highchartOutput), 77
highcharts_demo, 77
highchartzero (highchart2), 76
hw_grid, 78

is.hexcolor, 79
is.highchart, 79

JS (highcharter-exports), 77

list_parse, 80
list_parse2 (list_parse), 80

mutate_mapping, 80

pokemon, 81

random_id, 82
renderHighchart, 82
renderHighchart2 (renderHighchart), 82

stars, 83
str_to_id, 83

tags (highcharter-exports), [77](#)

tooltip_chart, [84](#)

tooltip_table, [85](#)

unemployment, [86](#)

uscountygeojson, [87](#)

usgeojson, [87](#)

vaccines, [88](#)

weather, [88](#)

worldgeojson, [89](#)