Package ‘simplevis’

May 9, 2023

Type Package

Title Wrappers to Simplify 'leaflet' Visualisation

Version 7.1.0

Description Wrapper functions around the amazing 'leaflet' package that aims to simplify 'leaflet' visualisation.
See the 'ggblanket' package for 'ggplot2' wrappers.

License MIT + file LICENSE


BugReports https://github.com/StatisticsNZ/simplevis/issues/

Encoding UTF-8

LazyData true

Depends R (>= 3.5.0)

Imports dplyr, htmlwidgets, leaflet, leafem, leafpop, magrittr, rlang, scales, sf, shiny, snakecase, stars, stringr, tidyr, tidyselect, viridis, ggplot2

Suggests glue, gt, knitr, pals, palmerpenguins, patchwork, rgdal, rgeos, rmarkdown, s2, tibble, tidytext, plotly

VignetteBuilder knitr

RoxygenNote 7.2.3

NeedsCompilation no

Author David Hodge [aut, cre] (<https://orcid.org/0000-0002-3868-7501>), Kate Lee [ctb] (<https://orcid.org/0000-0002-0886-3746>), Xavier Miles [ctb] (<https://orcid.org/0000-0002-1727-5110>), Statistics New Zealand [cph]

Maintainer David Hodge <davidhodge931@gmail.com>

Repository CRAN

Date/Publication 2023-05-09 04:00:02 UTC
### add_tooltip

**Description**

Add a tooltip column of united variable names and values.

**Usage**

```r
add_tooltip(data, ..., titles = snakecase::to_sentence_case, name = "tooltip")
```

**Arguments**

- `data`: A data frame or tibble.
- `...`: Arguments passed to select (i.e unquoted variables, tidyselect helpers etc). If no arguments provided, uses all columns.
- `titles`: A function to format the variable names, including in rlang lambda format.
- `name`: The name of the column created. Defaults to "tooltip".

**Value**

A data frame or tibble with a column of text

**Examples**

```r
library(ggplot2)
iris %>%
  add_tooltip() %>%
  head(1)
```

```r
iris %>%
  ```
```r
add_tooltip(tidyselect::contains("Sepal"), Species) %>%
head(1)

p <- iris %>%
dplyr::mutate(Species = stringr::str_to_sentence(Species)) |>
add_tooltip(tidyselect::contains("Sepal"), Species) |>
ggplot() +
geom_point(aes(x = Sepal.Width, y = Sepal.Length, col = Species, text = tooltip))

p

if (requireNamespace("plotly", quietly = TRUE)) {
  plotly::ggplotly(p, tooltip = "text")
}
```

---

**example_borders**

*Example sf object of the New Zealand coastline.*

**Description**

Example sf object of the New Zealand coastline used to demonstrate adding borders to maps.

**Usage**

```r
example_borders
```

**Format**

An *sf* object.

---

**example_point**

*Example sf point object.*

**Description**

Example sf point object.

**Usage**

```r
example_point
```

**Format**

An *sf* object.
example_polygon

Example sf polygon object.

Usage
example_polygon

Format
An sf object.

description
example_stars

Example stars object.

Usage
example_stars

Format
A stars object.

leaf_basemap

Basemap stack in leaflet.

Description
Make a stack of leaflet baselayers for use in shiny apps.

Usage
leaf_basemap(bounds = NULL, basemap = "light")
Arguments

bounds A bbox object or numeric vector of length four, with xmin, ymin, xmax and ymax values in WGS84 (epsg 4326).

basemap The first layer to start in the basemap stack. Either "light", "dark", "street", "satellite", or "ocean". Defaults to "light".

Value

A leaflet object.

Examples

leaf_basemap(basemap = "dark")

leaf_basemap(bounds = c(166.70047,-34.45676, 178.52966,-47.06345))

leaf_clear

In shiny, clear all features, images and legends.

Description

In shiny, clear all features, images and legends.

Usage

leaf_clear(map_id = "leaf")

Arguments

map_id The map id for a leaflet map. Defaults to "leaf".

Value

A map object.
Description

Map of simple features in leaflet that is not coloured.

Usage

```r
leaf_sf(
  data, 
  popup = TRUE, 
  popup_vars_vctr = NULL, 
  popup_numeric_format = function(x) prettyNum(x, big.mark = "", scientific = FALSE), 
  popup_vars_rename = snakecase::to_sentence_case, 
  pal = pal_viridis_mix(1), 
  size_point = 2, 
  size_line = 2, 
  alpha_point = NULL, 
  alpha_line = NULL, 
  alpha_fill = NULL, 
  basemap = "light", 
  layer_id_var = NULL, 
  group_id = NULL, 
  map_id = "leaf"
)
```

Arguments

- **data**: An sf object of geometry type point/multipoint, linestring/multilinestring or polygon/multipolygon geometry type. Required input.
- **popup**: TRUE or FALSE of whether to have a popup.
- **popup_vars_vctr**: Vector of quoted variable names to include in the popup. If NULL, defaults to making a leafpop::popupTable of all columns.
- **popup_numeric_format**: A function to format all numeric variables within the popup column. Defaults to non-scientific. Use function(x) x to leave as is.
- **popup_vars_rename**: Function to rename column names for the popup. Defaults to snakecase::to_sentence_case. Use function(x) x to leave column names untransformed.
- **pal**: Character vector of hex codes.
- **size_point**: Size of points (i.e. radius). Defaults to 2.
- **size_line**: Size of lines around features (i.e. weight). Defaults to 2.
- **alpha_point**: The opacity of the points.
leaf_sf_col

alpha_line  The opacity of the outline.
alpha_fill   The opacity of the fill.
basemap      The underlying basemap. Either "light", "dark", "satellite", "street", or "ocean". Defaults to "light". Only applicable where shiny equals FALSE.
layer_id_var Unquoted variable to be used in shiny, so that in the event where a feature is clicked on, the value of this is returned for that feature (e.g. input$map_marker_click$id).
group_id     The id name for the sf group.
map_id       The map id for the leaflet map. Defaults to "leaf".

Value
A leaflet object.

Examples
## Not run:
leaf_sf(example_point)

leaf_sf(example_polygon)

## End(Not run)

leaf_sf_col

Simple feature leaflet map that is coloured.

Description
Map of simple features in leaflet that is coloured.

Usage
leaf_sf_col(
data,
col_var,
label_var = NULL,
popup = TRUE,
popup_vars_vctr = NULL,
popup_numeric_format = function(x) prettyNum(x, big.mark = "", scientific = FALSE),
popup_vars_rename = snakecase::to_sentence_case,
pal = NULL,
pal_na = "#7F7F7F",
pal_rev = FALSE,
alpha_point = NULL,
alpha_line = NULL,
alpha_fill = NULL,
size_point = 2,
size_line = 2,
basemap = "light",
col_breaks_n = 4,
col_cuts = NULL,
col_intervals_left = TRUE,
col_labels = NULL,
col_legend_none = FALSE,
col_method = NULL,
col_na_rm = FALSE,
col_title = NULL,
label_numeric_format = function(x) prettyNum(x, big.mark = ",", scientific = FALSE),
layer_id_var = NULL,
group_id = NULL,
legend_id = NULL,
map_id = "leaf"
)

Arguments

data An sf object of geometry type point/multipoint, linestring/multilinestring or polygon/multipolygon geometry type. Required input.

col_var Unquoted variable to colour the features by. Required input.

label_var Unquoted variable to label the features by. If NULL, defaults to using the colour variable.

popup TRUE or FALSE of whether to have a popup.

popup_vars_vctr Vector of quoted variable names to include in the popup. If NULL, defaults to making a leafpop::popupTable of all columns.

popup_numeric_format A function to format all numeric variables within the popup column. Defaults to non-scientific. Use function(x) x to leave as is.

popup_vars_rename Function to rename column names for the popup. Defaults to snakecase::to_sentence_case. Use function(x) x to leave column names untransformed.

pal Character vector of hex codes.

pal_na The hex code or name of the NA colour to be used.

pal_rev Reverses the palette. Defaults to FALSE.

alpha_point The opacity of the points.

alpha_line The opacity of the outline.

alpha_fill The opacity of the fill.

size_point Size of points (i.e. radius). Defaults to 2.

size_line Size of lines around features (i.e. weight). Defaults to 2.

basemap The underlying basemap. Either "light", "dark", "satellite", "street", or "ocean". Defaults to "light". Only applicable where shiny equals FALSE.
leaf_sf_col

**col_breaks_n**  For a numeric colour variable, the desired number of intervals on the colour scale.

**col_cuts**  A vector of cuts to colour a numeric variable. If "bin" is selected, the first number in the vector should be either -Inf or 0, and the final number Inf. If "quantile" is selected, the first number in the vector should be 0 and the final number should be 1. Defaults to quartiles.

**col_intervals_left**  For a numeric colour variable, TRUE or FALSE of whether bins or quantiles are to be cut left-closed. Defaults to TRUE.

**col_labels**  A function or named vector to modify the colour scale labels. Defaults to snake-case::to_sentence_case if categorical, and scales::label_comma() if numeric. Use function(x) x to keep labels untransformed.

**col_legend_none**  TRUE or FALSE of whether to remove the legend.

**col_method**  The method of colouring features, either "bin", "quantile", "continuous", or "category." If numeric, defaults to "bin".

**col_na_rm**  TRUE or FALSE of whether to include col_var NA values. Defaults to FALSE.

**col_title**  A title string that will be wrapped into the legend.

**label_numeric_format**  A function to format the numeric labels. Defaults to adding a comma separator. Use function(x) x to leave as is.

**layer_id_var**  Unquoted variable to be used in shiny, so that in the event where a feature is clicked on, the value of this is returned for that feature (e.g. input$map_marker_click$id).

**group_id**  The id name for the sf group.

**legend_id**  The id name for the layerId of the legend.

**map_id**  The map id for the leaflet map. Defaults to "leaf".

### Value

A leaflet object.

### Examples

```r
# Not run:
leaf_sf_col(example_point, 
  col_var = trend_category)

leaf_sf_col(example_polygon, 
  col_var = density)

leaf_sf_col(example_polygon, 
  col_var = density, 
  col_method = "bin", 
  col_breaks_n = 5)

leaf_sf_col(example_polygon, 
  col_var = density, 
  col_title = "Density")
```

leaf_stars

Stars leaflet map.

Description
Map of stars in leaflet that is not coloured.

Usage

leaf_stars(
  data,
  pal = pal_viridis_mix(1),
  alpha_fill = 0.5,
  basemap = "light",
  group_id = NULL,
  map_id = "map"
)

Arguments

data A stars object. Required input.
pal Character vector of hex codes.
alpha_fill The opacity of the fill. Defaults to 0.5.
basemap The underlying basemap. Either "light", "dark", "satellite", "street", or "ocean". Defaults to "light". Only applicable where shiny equals FALSE.
group_id The id name for the stars group.
map_id The map id for the leaflet map. Defaults to "map".

Value
A leaflet object.
Examples

```r
## Not run:
library(simplevis)

leaf_stars(example_stars)

## End(Not run)
```

---

**leaf_stars_col**  
*Stars leaflet map that is coloured.*

Description

Map of stars in leaflet that is coloured.

Usage

```r
leaf_stars_col(
  data,  
  col_var, 
  pal = NULL, 
  pal_na = "#7F7F7F", 
  pal_rev = FALSE, 
  alpha_fill = 1, 
  basemap = "light", 
  col_breaks_n = 4, 
  col_cuts = NULL, 
  col_intervals_left = TRUE, 
  col_labels = NULL, 
  col_legend_none = FALSE, 
  col_method = NULL, 
  col_na_rm = FALSE, 
  col_title = NULL, 
  group_id = NULL, 
  legend_id = NULL, 
  map_id = "map"
)
```

Arguments

- `data`  
  A stars object. Required input.
- `col_var`  
  Unquoted attribute to colour the features by. Required input.
- `pal`  
  Character vector of hex codes.
- `pal_na`  
  The hex code or name of the NA colour to be used.
- `pal_rev`  
  Reverses the palette. Defaults to FALSE.
alpha_fill  The opacity of the fill. Defaults to 1.
basemap  The underlying basemap. Either "light", "dark", "satellite", "street", or "ocean". Defaults to "light". Only applicable where shiny equals FALSE.
col_breaks_n  For a numeric colour variable, the desired number of intervals on the colour scale.
col_cuts  A vector of cuts to colour a numeric variable. If "bin" is selected, the first number in the vector should be either -Inf or 0, and the final number Inf. If "quantile" is selected, the first number in the vector should be 0 and the final number should be 1. Defaults to quartiles.
col_intervals_left  For a numeric colour variable, TRUE or FALSE of whether bins or quantiles are to be cut left-closed. Defaults to TRUE.
col_labels  A function or named vector to modify the colour scale labels. Defaults to stringr::str_to_sentence if categorical, and scales::label_comma if numeric. Use function(x) x to keep labels untransformed.
col_legend_none  TRUE or FALSE of whether to remove the legend.
col_method  The method of colouring features, either "bin", "quantile", "continuous", or "category." If numeric, defaults to "bin".
col_na_rm  TRUE or FALSE of whether to visualise col_var NA values. Defaults to FALSE.
col_title  A title string that will be wrapped into the legend.
group_id  The id name for the stars group.
legend_id  The id name for the layerId of the legend.
map_id  The map id for the leaflet map. Defaults to "map".

Value
A leaflet object.

Examples
## Not run:
library(simplevis)

leaf_stars_col(example_stars,  
col_var = nitrate,  
col_na_rm = TRUE)

## End(Not run)
Index

* datasets
  example_borders, 3
  example_point, 3
  example_polygon, 4
  example_stars, 4

add_tooltip, 2

example_borders, 3
example_point, 3
example_polygon, 4
example_stars, 4

leaf_basemap, 4
leaf_clear, 5
leaf_sf, 6
leaf_sf_col, 7
leaf_stars, 10
leaf_stars_col, 11