Package 'evolMap'

July 1, 2025

Type Package

Version 1.3.14

Date 2025-06-27

Title Dynamic and Interactive Maps

Description Dynamic and Interactive Maps with R, powered by 'leaflet' <https://leafletjs.com>. 'evolMap' generates a web page with interactive and dynamic maps to which you can add geometric entities (points, lines or colored geographic areas), and/or markers with optional links between them. The dynamic ability of these maps allows their components to evolve over a continuous period of time or by periods.

License BSD_2_clause + file LICENSE

Depends R (>= 3.5.0)

Imports jsonlite, curl, sf

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Maintainer Modesto Escobar <modesto@usal.es>

Encoding UTF-8

RoxygenNote 7.1.2

Author Modesto Escobar [cre, aut, cph] (ORCID: <https://orcid.org/0000-0003-2072-6071>), Carlos Prieto [aut] (ORCID: <https://orcid.org/0000-0001-8178-9768>), David Barrios [aut]

Repository CRAN

Contents

---

add_description   *Add description of the map.*

Description

Add a description of the evolMap object to be showned on screen.

Usage

```
add_description(map, content = "", width = NULL)
```

Arguments

map	an object of class evolMap.
content	a character string indicating a description text for the graph.
width	a percentage indicating the width for the description panel (25% of the window by default).

Value

Object of class evolMap.

Author(s)

Modesto Escobar <modesto@usal.es>, Department of Sociology and Communication, University of Salamanca, Carlos Prieto <cprietos@usal.es>, and David Barrios <metal@usal.es>, Bioinformatics Service, University of Salamanca.

Examples

```
# create map
map <- create_map()
map <- add_description(map, "Lorem ipsum dolor sit amet, consectetur adipiscing elit.")


# plot map
plot(map)
```

---

add_entities   *Add entities.*

Description

Add entities to the interactive map.

Usage

```
add_entities(map, entities, attributes = NULL, name = NULL,
  label = NULL, color = NULL, text = NULL,
  info = NULL, infoFrame = c("right","left"), rightFrameWidth = NULL,
  start = NULL, end = NULL, period = NULL, opacity = 0.2)
```

Arguments

map	an object of class evolMap.
entities	a spatial object of geometries.
attributes	a data frame with information to show for each geometry. Its columns names can be passed as parameters to the arguments.
name	name of the column with names in the entities or attributes data frame.
label	name of the column with labels in the entities or attributes data frame.
color	name of the column with color variable in the entities or attributes data frame.
text	name of the column in the entities or attributes data frame with texts in the entities or attributes data frame. This information will be shown as a popup.
info	name of the column with information to display in a panel in the data matrix.
infoFrame	Panel (right or left) where the information is to be displayed. The left panel is only available if a description is provided with add_description.
rightFrameWidth	
	a percentage indicating the width for the right information panel.
start	name of the column with the start time of a period in the entities or attributes data frame.
end	name of the column with the end time of a period in the entities or attributes data frame.
period	name of the column with the period name in the entities or attributes data frame.
opacity	Entity opacity expressed as a numeric vector between 0 and 1. Default: 0.2.

Value

Object of class `evolMap`.

Author(s)

Modesto Escobar <modesto@usal.es>, Department of Sociology and Communication, University of Salamanca, Carlos Prieto <cprietos@usal.es>, and David Barrios <metal@usal.es>, Bioinformatics Service, University of Salamanca.

Examples

```
data(World)
map <- create_map()
map <- add_entities(map, World, color="pop")

# plot map
plot(map)
```

---

add_links   *Add links.*

Description

Add links to the interactive map.

Usage

```
add_links(map, links, color = NULL, start = NULL, end = NULL, period = NULL,
  curve = TRUE, arrows = FALSE)
```

Arguments

map	an object of class `evolMap`.
links	a link data frame with two first columns as source and target marker names (specified with the name argument in [add_markers](#)) and any other column with link attributes. Column names can be passed as parameters to the arguments.
color	name of the column with color variable in the link data frame.
start	name of the column with the start time of a period in the link data frame.
end	name of the column with the end time of a period in the link data frame.
period	name of the column with the period name in the link data frame.
curve	Links will be displayed as a curve.
arrows	Links will show an arrow at the end.

Value

Object of class `evolMap`.

Author(s)

Modesto Escobar <modesto@usal.es>, Department of Sociology and Communication, University of Salamanca, Carlos Prieto <cprietos@usal.es>, and David Barrios <metal@usal.es>, Bioinformatics Service, University of Salamanca.

Examples

```
# create data
markers <- data.frame(
  lat=c(39,47,53,40,53),
  lon=c(-5, 2,-8,-8,-1),
  name=c("m1","m2","m3","m4","m5")
)

links <- data.frame(
  source=c("m4","m1","m2","m5"),
  target=c("m1","m2","m5","m3"),
  num=1:4
)

# create map
map <- create_map()
map <- add_markers(map, markers, name="name", color="name", markerCluster=FALSE)
map <- add_links(map, links, color="num")


# plot map
plot(map)
```

---

add_markers   *Add markers.*

Description

Add markers to the interactive map.

Usage

```
add_markers(map, data, latitude = NULL, longitude = NULL, name = NULL,
  label = NULL, image = NULL, size = NULL, color = NULL, shape = NULL,
  text = NULL, info = NULL, infoFrame = c("right","left"),
  rightFrameWidth = NULL, start = NULL, end = NULL, period = NULL,
  markerCluster = FALSE, roundedIcons = TRUE, jitteredPoints = 0,
  coords = FALSE)
```

Arguments

map	an object of class evolMap.
data	a marker data frame with the locations and information to show in each marker. Its column names can be passed as parameters to the arguments of the function.
latitude	name of the column with the latitude coordinates for each marker (first column by default).
longitude	name of the column with the longitude coordinates for each marker (second column by default).
name	name of the column with names in the marker data frame.
label	name of the column with labels in the marker data frame.
image	name of the column with the path to marker image files in the marker data frame.
size	name of the column with size in the nodes data frame.
color	name of the column with color variable in the marker data frame.
shape	name of the column with shape variable in the marker data frame.
text	name of the column with texts in the marker data frame. This information will be shown as a popup.
info	name of the column with information to display in a panel in the marker data frame.
infoFrame	Panel (right or left) where the information is to be displayed. The left panel is only available if a description is provided with add_description.
rightFrameWidth	a percentage indicating the width for the right information panel.
start	name of the column with the start time of a period in the marker data frame.
end	name of the column with the end time of a period in the marker data frame.
period	name of the column with the period name in the marker data frame.
markerCluster	enable joining of nearby markers when zooming out.
roundedIcons	display markers with rounded borders.
jitteredPoints	add an amount of noise to markers to avoid overlapping.
coords	should coordinates (latitude and longitude) must be shown in the table and filters? (FALSE by default)

Value

Object of class evolMap.

Author(s)

Modesto Escobar <modesto@usal.es>, Department of Sociology and Communication, University of Salamanca, Carlos Prieto <cprietos@usal.es>, and David Barrios <metal@usal.es>, Bioinformatics Service, University of Salamanca.

Examples

```
# load data
data(sociologists)
data(locations)

# load pictures
sociologists$picture <- system.file("extdata", sociologists$picture,
  package="evolMap")

# join locations
sociologists$latitude <- locations[,1]
sociologists$longitude <- locations[,2]

# create map
map <- create_map()
map <- add_markers(map, sociologists,
  latitude = "latitude", longitude = "longitude",
  label = "label", image = "picture",
  start = "birth", end = "death")


# plot map
plot(map)
```

---

add_periods   *Add Periods.*

Description

Add periods to the interactive map.

Usage

```
add_periods(map, periods, name = NULL, start = NULL, end = NULL,
  latitude = NULL, longitude = NULL, zoom = NULL,
  description = NULL, popup = FALSE, duration = NULL, periodrep = TRUE)
```

Arguments

map	an object of class evolMap.
periods	a period data frame defining periods with the following columns: name, start and end.
name	name of the column with names in the period data frame (first column by default).
start	name of the column with the start time of a period in the period data frame (second column by default).

end	name of the column with the end time of a period in the period data frame (third column by default).
latitude	name of the column with the latitude coordinates to center zoom on each period.
longitude	name of the column with the longitude coordinates for center zoom in each period.
zoom	name of the column with the zoom size to display each period.
description	name of the column with the description of each period.
popup	should the description be displayed as a popup?.
duration	name of the column with the period duration in seconds.
periodrep	If false, time will run by year, and the period will only be shown in the header. Otherwise, time will run by period.

Value

Object of class evolMap.

Author(s)

Modesto Escobar <modesto@usal.es>, Department of Sociology and Communication, University of Salamanca, Carlos Prieto <cprietos@usal.es>, and David Barrios <metal@usal.es>, Bioinformatics Service, University of Salamanca.

Examples

```
data(sociologists)
data(locations)

# load pictures
sociologists$picture <- system.file("extdata", sociologists$picture,
  package="evolMap")

# join locations
sociologists$latitude <- locations[,1]
sociologists$longitude <- locations[,2]

# create map
map <- create_map()
map <- add_markers(map, sociologists,
  latitude = "latitude", longitude = "longitude",
  label = "label", image = "picture",
  start = "birth", end = "death",
  period = "generation", markerCluster = FALSE)

periods <- data.frame(
name = c("1775-1799","1800-1824","1825-1850","1851-1874"),
start = c(1775,1800,1825,1851),
end = c(1799,1824,1850,1874)
)
map <- add_periods(map, periods)
```

```
# plot map
plot(map)
```

---

add_tutorial   *Adds a tutorial for the map.*

Description

`add_tutorial` adds a tutorial for a map.

Usage

```
add_tutorial(map, image = NULL, description = NULL)
```

Arguments

map	object of class `evolMap`.
image	character vector indicating the image path, header for the tutorial.
description	a character string indicating a desription text to insert in the tutorial.

Value

Object of class `evolMap`.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca.

Examples

```
# create map
map <- create_map()
map <- add_tutorial(map)


# plot map
plot(map)
```

---

create_map   *Interactive map.*

Description

create_map produces the structure of an interactive map with 'Leaflet'.

Usage

```
create_map(center = NULL, zoom = NULL, zoomStep = NULL,
  provider = "OpenStreetMap", main = NULL, note = NULL, mode = 1,
  defaultColor = "#2f7bee", controls = 1:4, language = c("en","es","ca"))
```

Arguments

center	a numeric two size vector length giving latitude and longitude to set the initial view. (default: 0,0)
zoom	a number greater than or equal to 0 to stablish the startet zoom. (default: 3)
zoomStep	a number greater than 0 to determine the zoom step. (default: 0.25)
provider	character string with the map provider to represent as background, OpenStreetMap by default. See [list_providers](#) for available map providers.
main	upper title of the graph.
note	text to appear at the bottom of the map.
mode	a number indicating the graphic style. (default: 1)
defaultColor	a character vector giving a valid html color for marker representation.
controls	a numeric vector indicating which controls will be shown. 1 = tools (filters in mode=2), 2 = buttons, 3 = legends, 4 = search box. NULL hides all controls, negative values deny each control and 0 denies all.
language	a character string indicating the language of the graph (en=english (default); es=spanish; ca=catalan).

Value

Object of class evolMap.

Author(s)

Modesto Escobar <modesto@usal.es>, Department of Sociology and Communication, University of Salamanca, Carlos Prieto <cprietos@usal.es>, and David Barrios <metal@usal.es>, Bioinformatics Service, University of Salamanca.

Examples

```
# create map
map <- create_map()


# plot map
plot(map)
```

---

get_location   *Get location coordinates.*

Description

get_location returns the latitude and longitude of each input place name.

Usage

```
get_location(x)
```

Arguments

x   a vector string with place names ("city, country") whose coordinates will be
                    downloaded from OpenStreetMap.

Value

Matrix with the latitude and longitude coordinates of each input place.

Author(s)

Modesto Escobar <modesto@usal.es>, Department of Sociology and Communication, University
of Salamanca, Carlos Prieto <cprietos@usal.es>, and David Barrios <metal@usal.es>, Bioin-
formatics Service, University of Salamanca.

Examples

```
get_location(c("Salamanca, Spain","New York, United States"))
```

---

list_providers   *Get location coordinates.*

Description

List of all providers with their map variations.

Usage

```
list_providers()
```

Value

This function returns a list of available map providers for the `create_map` function.

Author(s)

Modesto Escobar <modesto@usal.es>, Department of Sociology and Communication, University of Salamanca, Carlos Prieto <cprietos@usal.es>, and David Barrios <metal@usal.es>, Bioinformatics Service, University of Salamanca.

Examples

```
list_providers()
```

---

locations   *Data: Birthplaces locations of classical sociologists.*

Description

Matrix with birthplace locations of certain sociologists.

Usage

```
data("locations")
```

Format

A matrix with birthplace locations of 16 sociologists:

lat : latitude.

lon : longitude.

Source

Own elaboration from manuals of sociology and OpenStreetMap.

References

See [sociologists](#).

Examples

```
data(locations)
```

---

sociologists	*Data: Classical sociologists.*

Description

Data frame with names, birth and death year data, birth country and school of thought.

Usage

```
data("sociologists")
```

Format

A data frame of 16 sociologists with 11 variables to study time coincidences:

name : first and last name of the sociologist.

birth : birth year.

death : death year.

birth_place : birth city.

birth_country : birth country.

death_place : death city.

death_country : death country.

label : combination of name, birth and death dates.

generation : generation (every 25 years) of the sociologist.

school : school of thought.

picture : name and path of the file where their picture is.

Source

Own elaboration from manuals of sociology.

Examples

```
data(sociologists)
head(sociologists, 10)
tail(sociologists, 10)
```

Description

The object loaded is an sf object containing world map data from Natural Earth with a few variables from World Bank.

Usage

```
data("World")
```

Format

Formal class 'sf' [package "sf"]; the data contains a data frame with 177 obs. of 11 variables:

iso_a2 : character vector of ISO 2 character country codes

name_long : character vector of country names

continent : character vector of continent names

region_un : character vector of region names

subregion : character vector of subregion names

type : character vector of type names

area_km2 : integer vector of area values

pop : integer vector of population in 2014

lifeExp : integer vector of life expectancy at birth in 2014

gdpPercap : integer vector of per-capita GDP in 2014

geom : sfc_MULTIPOLYGON

The object is in geographical coordinates using the WGS84 datum.

Source

https://www.naturalearthdata.com/ https://data.worldbank.org/

Examples

```
if (requireNamespace("sf", quietly = TRUE)) {
  library(sf)
  data(World)
  plot(World)
}
```

Index