

Package: flowmapblue (via r-universe)

September 7, 2024

Title Flow Map Rendering

Version 0.0.2

Description Create interactive flow maps using 'FlowmapBlue' 'TypeScript' library <<https://github.com/FlowmapBlue/FlowmapBlue>>, which is a free tool for representing aggregated numbers of movements between geographic locations as flow maps. It is used to visualize urban mobility, commuting behavior, bus, subway and air travels, bicycle sharing, human and bird migration, refugee flows, freight transportation, trade, supply chains, scientific collaboration, epidemiological and historical data and many other topics. The package allows to either create standalone flow maps in form of 'htmlwidgets' and save them in 'HTML' files, or integrate flow maps into 'Shiny' applications.

License MIT + file LICENSE

URL <https://github.com/FlowmapBlue/flowmapblue.R>,
<https://flowmapblue.github.io/flowmapblue.R/>

BugReports <https://github.com/FlowmapBlue/flowmapblue.R/issues>

Depends R (>= 2.10)

Imports htmlwidgets

Suggests quarto

VignetteBuilder quarto

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Repository <https://flowmapblue.r-universe.dev>

RemoteUrl <https://github.com/flowmapblue/flowmapblue.r>

RemoteRef HEAD

RemoteSha e2cae3cee616f159d84e0e9c890cbe52bef04b21

Contents

ch_flows	2
ch_locations	2
flowmapblue	3
flowmapblue-shiny	6

Index	7
--------------	----------

ch_flows	<i>Swiss Flows Dataset</i>
----------	----------------------------

Description

A dataset containing flow data between various locations in Switzerland. This data represents the flow counts between origin and destination locations, identified by their unique codes.

Usage

ch_flows

Format

ch_flows:

A data frame with 676 rows and 3 columns:

origin A character vector representing the origin location identifier (must match the id in the ch_locations dataset).

dest A character vector representing the destination location identifier (must match the id in the ch_locations dataset).

count An integer vector representing the flow count between the origin and destination locations.

ch_locations	<i>Swiss Locations Dataset</i>
--------------	--------------------------------

Description

A dataset containing geographic information about 26 locations in Switzerland. This data includes unique identifiers, names, and geographic coordinates (latitude and longitude) for each location.

Usage

ch_locations

Format

`ch_locations`:

A data frame with 26 rows and 4 columns:

id A character vector representing the unique identifier for each Swiss location (e.g., "JU", "LU").

name A character vector representing the name of each location (e.g., "Jura", "Luzern").

lat A numeric vector representing the latitude of each location in WGS84 (EPSG: 4326) coordinate reference system.

lon A numeric vector representing the longitude of each location in WGS84 (EPSG: 4326) coordinate reference system.

flowmapblue

Create an interactive flow map

Description

Creates an interactive flow map visualizing flows between various locations and outputs it as an HTML widget. This function utilizes the [FlowmapBlue](#) library to create maps with customizable options such as clustering, animation, and dark mode. The widget can be rendered in R Markdown, Shiny, or viewed in a browser. It can also be saved to html file with [htmlwidgets::saveWidget\(\)](#). See examples for more details.

Usage

```
flowmapblue(  
  locations,  
  flows,  
  mapboxAccessToken = NULL,  
  clustering = TRUE,  
  animation = FALSE,  
  darkMode = FALSE  
)
```

Arguments

locations A `data.frame` containing the location data. The `data.frame` should have the following columns:

id A character vector representing the unique identifier for each location (e.g., "JU", "LU").

name (Optional) A character vector representing the name of each location (e.g., "Jura", "Luzern").

lat A numeric vector representing the latitude of each location in WGS84 (EPSG: 4326) coordinate reference system.

lon A numeric vector representing the longitude of each location in WGS84 (EPSG: 4326) coordinate reference system.

flows	<p>A data.frame containing the flow data between locations. The data.frame should have the following columns:</p> <p>origin A character vector representing the origin location identifier (must match the id in locations).</p> <p>dest A character vector representing the destination location identifier (must match the id in locations).</p> <p>count An integer vector representing the flow count between the origin and destination locations.</p> <p>time (Optional) A vector of POSIXct or Date objects representing the date or date and time of the flow.</p>
mapboxAccessToken	A character string representing the Mapbox access token. This is required to render the map using Mapbox tiles. You can obtain a free token at https://account.mapbox.com/ .
clustering	A logical value indicating whether to enable clustering of locations on the map. Defaults to TRUE.
animation	A logical value indicating whether to enable animation of flows on the map. Defaults to FALSE.
darkMode	A logical value indicating whether to enable dark mode for the map. Defaults to FALSE.

Value

An HTML widget of class flowmapblue and htmlwidget that can be rendered in R Markdown, Shiny, or viewed in a browser. It can also be saved to html file with `htmlwidgets::saveWidget()`. See examples for more details.

Examples

```
## Not run:
# example 1, normal flows
# set your Mapbox access token
Sys.setenv(MAPBOX_API_TOKEN = "YOUR_MAPBOX_ACCESS_TOKEN")

# load locations and flows for Switzerland
locations <- data(ch_locations)
flows <- data(ch_flows)

flowmap <- flowmapblue(
  locations,
  flows,
  mapboxAccessToken = Sys.getenv('MAPBOX_API_TOKEN'),
  clustering = TRUE,
  darkMode = TRUE,
  animation = FALSE
)

# view the map
flowmap
```

```
# or save it as an HTML file
htmlwidgets::saveWidget(flowmap, file = "flowmap.html")

# example 2, flows with date in time column
# set your Mapbox access token
Sys.setenv(MAPBOX_API_TOKEN = "YOUR_MAPBOX_ACCESS_TOKEN")

# load locations and flows for Switzerland
locations <- data(ch_locations)
flows <- data(ch_flows)

# generate fake datetime
flows$time <- seq(from = as.POSIXct("2020-01-01"),
  to = as.POSIXct("2020-01-05"), length.out = nrow(flows))

flowmap <- flowmapblue(
  locations,
  flows,
  mapboxAccessToken = Sys.getenv('MAPBOX_API_TOKEN'),
  clustering = TRUE,
  darkMode = TRUE,
  animation = FALSE
)

# view the map
flowmap

# example 3, flows with date in time column
# set your Mapbox access token
Sys.setenv(MAPBOX_API_TOKEN = "YOUR_MAPBOX_ACCESS_TOKEN")

# load locations and flows for Switzerland
locations <- data(ch_locations)
flows <- data(ch_flows)
# generate fake dates
flows$time <- seq(from = as.Date("2020-01-01"),
  to = as.Date("2020-06-01"), length.out = nrow(flows))

flowmap <- flowmapblue(
  locations,
  flows,
  mapboxAccessToken = Sys.getenv('MAPBOX_API_TOKEN'),
  clustering = TRUE,
  darkMode = TRUE,
  animation = FALSE
)

# view the map
flowmap

## End(Not run)
```

flowmapblue-shiny *Shiny bindings for flowmapblue*

Description

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

Usage

```
flowmapblueOutput(outputId, width = "100%", height = "400px")  
renderFlowmapblue(expr, env = parent.frame(), quoted = FALSE)
```

Arguments

outputId	output variable to read from.
width, height	Must be a valid CSS unit (like '100%', '400px', 'auto') or a number, which will be coerced to a string and have 'px' appended.
expr	An expression that generates a flowmapblue widget.
env	The environment in which to evaluate expr.
quoted	Is expr a quoted expression (with quote())? This is useful if you want to save an expression in a variable.

Value

`flowmapblueOutput` Returns a `shiny.tag.list` object that can be included in a Shiny UI to display the flowmapblue widget.

`renderFlowmapblue` Returns a `shiny.render.function` that is used to generate the flowmapblue widget on the server side in a Shiny application.

See Also

[shinyWidgetOutput](#), [shinyRenderWidget](#)

Index

* datasets

ch_flows, [2](#)

ch_locations, [2](#)

ch_flows, [2](#)

ch_locations, [2](#)

flowmapblue, [3](#)

flowmapblue-shiny, [6](#)

flowmapblueOutput (flowmapblue-shiny), [6](#)

htmlwidgets:saveWidget(), [3](#), [4](#)

renderFlowmapblue (flowmapblue-shiny), [6](#)

shinyRenderWidget, [6](#)

shinyWidgetOutput, [6](#)