Package: RtD3 (via r-universe)

July 17, 2024

Title Rt Visualization in D3

Version 0.0.1

Description Create interactive visualisations of Rt estimates using 'D3.js' (Gibbs et al. (2020) <doi:10.5281/zenodo.4011842>). Developed primarily targeting Rt estimates generated by the 'EpiNow2' package, 'RtD3' aims to make simple, beautiful visualisations that help researchers explore their results and share them with others.

License MIT + file LICENSE

URL https:/epiforecasts.io/RtD3, https://github.com/epiforecasts/RtD3

BugReports https://github.com/epiforecasts/RtD3/issues

Imports jsonlite, htmlwidgets, geojsonsf, data.table, purrr

Suggests testthat, spelling, knitr, covr, dplyr, sf, rgeos, rnaturalearth

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

Language en-US

Repository https://epiforecasts.r-universe.dev

RemoteUrl https://github.com/epiforecasts/RtD3

RemoteRef HEAD

RemoteSha a0dfdac9e7c4aa2c01400ee521d24f1ac8876b85

Contents

---

default_data_ref   *default_data_ref*

Description

Create a default reference for a input datasets. Also useful as a template when defining your own.

Usage

```
default_data_ref()
```

---

default_map_legend_ref

default_map_legend_ref

Description

Create a default reference for a map legend. Also useful as a template when defining your own.

Usage

```
default_map_legend_ref()
```

---

default_ts_colors   *default_ts_colors*

Description

Define default colors for time series plots

Can be overridden with a list of the same format

Usage

```
default_ts_colors()
```

---

default_ts_color_ref   *default_ts_color_ref*

Description

Create a default reference for time series colors. Also useful as a template when defining your own.

Usage

```
default_ts_color_ref()
```

---

getSubregionalUrls   *Get Subregional Estimate Urls*

Description

Get Subregional Estimate Urls

Usage

```
getSubregionalUrls(path, areas)
```

Arguments

path	A character string container the overall path to subnational estimates
areas	A character vector listing the subregional estimates (assuming that listed in the geoData with capitalisation and without capitalisation in the path).

Value

A named list of subnational urls.

Examples

```
getSubregionalUrls(path = "https://epiforecasts.io/covid/posts/national/",
                   areas = c('Afghanistan', 'Brazil', 'Colombia', 'United States'))
```

---

joinRtData   *Join RtData*

Description

Joins two nested lists in the format required by summaryWidget. This may be useful for merging estimates from disparate data sources or linking national level estimates with subnational estimates

Usage

```
joinRtData(rtData, rtData2)
```

Arguments

rtData   A nested list as required by summaryWidget

rtData2   A nested list as required by summaryWidget

Value

A nested list as required by summaryWidget

Examples

```
base_url <- "https://raw.githubusercontent.com/epiforecasts/covid-rt-estimates/master/"
subnational <-   national <- list("Cases" = readInEpiNow2(
path = paste0(base_url, "subnational/italy/cases/summary"),
region_var = "region"))


national <- list("Cases" = readInEpiNow2(
path = paste0(base_url, "national/cases/summary"),
region_var = "country"),
regions = "Italy")

out <- list()
out$Cases <- joinRtData(subnational$Cases, national$Cases)
```

---

legend_qualitative   *legend_qualitative*

Description

Create a qualitative legend.

Usage

```
legend_qualitative(variable_name, legend_values)
```

Arguments

variable_name   string, name of the variable of this legend.

legend_values   list, reference for legend colors in the format: 'value':'color', ...

---

legend_sequential   *legend_sequential*

Description

Create a qualitative legend. 'legend_scale" accepts scales listed in d3-scale.

Usage

```
legend_sequential(
  variable_name,
  legend_scale = "scaleLinear",
  color_low = "white",
  color_high = "green",
  color_no_data = "lightgrey"
)
```

Arguments

variable_name   string, name of the variable of this legend.

legend_scale   string, type of legend scale, must be the name of a d3 scale. Default: "scaleLin-ear".

color_low   string, color for the lowest value of the legend.

color_high   string, color for the highest value of the legend.

color_no_data   string, color for no data entries.

---

readInEpiNow2   *Read in Results from EpiNow2*

Description

Reads in results from `EpiNow2` and converts them into the `RtD3` format. Supports either input via a list object or from a file path/url.

Usage

```
readInEpiNow2(input_list, path, region_var = "region", regions)
```

Arguments

`input_list`	A list of results as returned by `EpiNow2::regional_summary`
`path`	A character string indicating the path (either file or URL) to the summary results
`region_var`	A character string that identifies the region name used.
`regions`	A character string indicating the regions of interest to returns. Defaults to all regions.

Value

A named list in the format required by `summaryWidget` along with a summary table.

Examples

```
# Read in each summary folder

base_path <- "https://raw.githubusercontent.com/epiforecasts/covid-rt-estimates/"
rtData <- readInEpiNow2(
  path = paste0(base_path, "master/national/cases/summary"),
  region_var = "country")

rtData



france <- readInEpiNow2(
  path = paste0(base_path, "master/national/cases/summary"),
  region_var = "country",
  regions = "France")

france
```

summaryWidget   *summaryWidget*

Description

Create an Rt visualisation using D3. Need convenience functions to define defaults

Usage

```
summaryWidget(
  geoData = NULL,
  rtData = NULL,
  data_ref = NULL,
  subregional_ref = NULL,
  ts_color_ref = NULL,
  ts_bar_color = "lightgrey",
  projection = "geoEquirectangular",
  map_legend_ref = NULL,
  credible_threshold = 10,
  width = NULL,
  activeArea = "United Kingdom",
  downloadUrl = NULL,
  dryRun = FALSE
)
```

Arguments

geoData	sf object, map data
rtData	data.frame, rt estimates in the format 'Source':'rtData':x, 'casesInfectionData':x, 'casesReportData':x, 'obsCasesData':x, ...
data_ref	list, reference for input data column names. Specify the column holding geometry to be symbolized 'rtData':'geometry_name':'region', ...
subregional_ref	
	list, reference to subnational estimates in the format 'country_name':'url', ...
ts_color_ref	list, reference for colors for time series plots.
ts_bar_color	string, color of observed cases bars in time series plots.
projection	string, map projection, must be named in d3-geo-projection.
map_legend_ref	list, reference for map legend variables
credible_threshold	
	integer, Threshold for credible intervals, maximum observed cases * this value will be removed.
width	integer, Width of widget in pixels.
activeArea	string, Area to symbolize first.
downloadUrl	string, URL to download data.

dryRun   Logical, defaults to FALSE. Should the function be tested without the widget
                        being created. Useful for checking the integrity of input data.

---

summaryWidget-shiny   *Shiny bindings for summaryWidget*

Description

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

Usage

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

rendersummaryWidget(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 RtD3

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.

Index