Package ‘spMaps’

November 27, 2019

Type Package
Title Europe SpatialPolygonsDataFrame Builder
Version 0.4.0
Description Build custom Europe SpatialPolygonsDataFrame, if you don't know what is a SpatialPolygonsDataFrame see SpatialPolygons() in 'sp', by example for mapLayout() in 'antaresViz'. Antares is a powerful software developed by RTE to simulate and study electric power systems (more information about 'Antares' here: <https://antares-simulator.org/>).

URL https://github.com/rte-antares-rpackage/spMaps
BugReports https://github.com/rte-antares-rpackage/spMaps/issues
License GPL (>= 2) | file LICENSE
LazyData TRUE
Encoding UTF-8
Depends R (>= 2.10), sp, raster, rgeos
RoxygenNote 7.0.1
Suggests testthat, covr, antaresViz
NeedsCompilation no
Author Veronique Bachelier [aut, cre],
Jalal-Edine ZAWAM [aut],
Benoit Thieurmel [aut],
RTE [cph]
Maintainer Veronique Bachelier <veronique.bachelier@rte-france.com>
Repository CRAN
Date/Publication 2019-11-27 12:10:02 UTC

R topics documented:

getEuropeReferenceTable .................................................. 2

Index 4
getEuropeReferenceTable

*Get custom Europe map*(SpatialPolygonsDataFrame)

**Description**

This function builds a custom Europe map and returns a SpatialPolygonsDataFrame. The output can be used by example in `mapLayout` with the `map` argument.

**Usage**

getEuropeReferenceTable()

gEuropeCountries(mergeCountry = TRUE)

gEuropeStates()

gSpMaps(countries = "all", states = NULL, mergeCountry = TRUE)

**Arguments**

- **mergeCountry** boolean. Merge country? (UK, Belgium ...). Default to TRUE.
- **countries** character. Vector of wanted countries, without details / states. Must refer to code column of the reference table `getEuropeReferenceTable`. "all" (default) keep all countries.
- **states** character. Vector of wanted countries, with details / states. Must refer to code column of the reference table `getEuropeReferenceTable`. "all" keep all countries. NULL as default.

**Value**

SpatialPolygonsDataFrame

**Examples**

# default map: Europe without states
europe_cty <- getSpMaps()
plot(europe_cty)

# subset on some countries
ref_table <- getEuropeReferenceTable()

italy_spain_fra <- getSpMaps(countries = c("FRA", "ITA", "ESP"))
plot(italy_spain_fra)

## Not run:
italy_spain_fra_states <- getSpMaps(countries = NULL, states = c("FRA", "ITA", "ESP"))
getEuropeReferenceTable

```r
plot(italy_spain_fra_states)

# combine countries and states
combine_map <- getSpMaps(countries = c("ITA", "ESP"), states = "FRA")
plot(combine_map)

# build your custom map: you can use directly data
# to subset the area you really want
europe_states <- getEuropeStates()
europe_countries <- getEuropeCountries()

# for example, have a look to GBR states map
data(europe_states)
gebr_states_districts <- europe_states[
  europe_states$code %in% "GBR" &
  europe_states$type %in% "Administrative County",
]
plot(gbr_states_districts)

# combine with another map: you just have to have the same columns...
# getSpMaps only return "name" and "code" column
custom_states <- rbind(
  getSpMaps(countries = NULL, states = "FRA"),
  gbr_states_districts[, c("name", "code"), drop = FALSE])

plot(custom_states)
```

## End(Not run)
Index

gEuropeCountries
  (getEuropeReferenceTable), 2
getEuropeReferenceTable, 2
gEuropeStates
  (getEuropeReferenceTable), 2
gSpMaps (getEuropeReferenceTable), 2

mapLayout, 2

spMaps (getEuropeReferenceTable), 2