

Package ‘WorldMapR’

September 3, 2025

Type Package
Title Worldwide or Coordinates-Based Heat Maps
Version 1.3.0
Description Easily plot heat maps of the world, based on continuous or categorical data. Country labels can also be added to the map.
License GPL-3
URL <https://github.com/Luigi-Annic/WorldMapR/>
BugReports <https://github.com/Luigi-Annic/WorldMapR/issues>
Encoding UTF-8
Depends R (>= 4.3.0)
Imports ggplot2 (>= 3.4.4), sf (>= 1.0-14), countrycode (>= 1.5.0),
utils (>= 4.3.0), ggfx (>= 1.0.1)
LazyData true
RoxygenNote 7.3.2
Suggests knitr, rmarkdown, testthat (>= 3.0.0),
VignetteBuilder knitr
Config/testthat/edition 3
NeedsCompilation no
Author Luigi Annicchiarico [cre, aut]
Maintainer Luigi Annicchiarico <luigi.annic@gmail.com>
Repository CRAN
Date/Publication 2025-09-03 10:30:02 UTC

Contents

countrycoord_data	2
empty_editable_map	3
geometries_data	3
map_df0	4

regions_ITBIH	4
testdata1	5
testdata1b	5
testdata1c	6
worldplot	6
worldplotCat	8
Index	11

countrycoord_data	<i>countrycoord_data</i>
-------------------	--------------------------

Description

This function generates a data frame with information about the coordinates of the central point for each country of interest. You can choose whether to keep all the countries or only a subset.

Usage

```
countrycoord_data(  
  countries.list = NULL,  
  crs = 4326,  
  UK_as_GB = TRUE,  
  exclude.iso.na = TRUE  
)
```

Arguments

- `countries.list` List of the ISO 3166-1 alpha-2 codes of countries that are to be included. By default it is set to NULL and all countries are included.
- `crs` Coordinate reference system (EPSG). By default the value is 4326, which corresponds to EPSG::4326 (WGS84)
- `UK_as_GB` Which do you prefer between UK and GB as the code for the United Kingdom? If FALSE, GB is returned in the output data.frame. If TRUE (default), UK is returned.
- `exclude.iso.na` if TRUE (default), countries that do not have a ISO 3166 code are excluded from the table.

Value

an object of class `data.frame`

Examples

```
countrycoord_data(countries.list = c("IT", "FR", "SE"), crs = 3035)  
countrycoord_data(countries.list = c("IT", "FR", "SE"), crs = 3035)  
countrycoord_data(countries.list = c("IT", "FR", "SE", "GB"), crs = 3035, UK_as_GB = TRUE)  
countrycoord_data(countries.list = c("IT", "FR", "SE", "GB"), crs = 3035, UK_as_GB = FALSE)
```

empty_editable_map *empty_editable_map*

Description

Plot a world map which can be edited in powerpoint.

Usage

```
empty_editable_map(lon, lat, crs = 4326, sf_use_s2 = FALSE)
```

Arguments

lon	Longitude limits.
lat	Latitude limits.
crs	Coordinate reference system (EPSG). By default the value is 4326, which corresponds to EPSG::4326 (WGS84)
sf_use_s2	sf_use_s2 option, default is FALSE

Value

a map

Examples

```
## Not run: empty_editable_map(lon = c(2600000, 9300000), lat = c(5450000, 1000000), crs = 3035)
export::graph2ppt(file = "editable_map.pptx", width = 7, height = 4)
## End(Not run)
## Not run: empty_editable_map(lon = c(-15, 40), lat = c(20, 65), crs = 4326)
```

geometries_data *geometries_data*

Description

This function is deprecated and not exported. It has been substituted with `worldMapR::countrycoord_data`. This function generates a data frame with information about geometries and centroid coordinates of countries. You can choose whether to keep all the countries or only a subset.

Usage

```
geometries_data(exclude.iso.na = TRUE, countries.list = NULL)
```

Arguments

- `exclude.iso.na` if TRUE (default), countries that do not have a ISO 3166 code are excluded from the table.
- `countries.list` List of the ISO 3166-1 alpha-2 codes of countries that are to be included. By default it is set to NULL and all countries are included.

Value

an object of class `data.frame` and `sf`.

<code>map_df0</code>	<i>Initial dataset with geometries for each country</i>
----------------------	---

Description

This data set contains information about name, `iso_a2` code and geometry for 242 countries.

Usage

```
data(map_df0)
```

Format

An object of class `data.frame`

Examples

```
data(map_df0)
head(map_df0)
```

<code>regions_ITBIH</code>	<i>Some potentially useful regions for more granular maps</i>
----------------------------	---

Description

This data set contains information about name, `iso_a2` code and geometry for Italian and BIH regions.

Usage

```
data(regions_ITBIH)
```

Format

An object of class `data.frame`

Examples

```
data(regions_ITBIH)
head(regions_ITBIH)
```

`testdata1`*Simulated data set 1*

Description

Data from a random simulation with continuous data.

Usage

```
data(testdata1)
```

Format

An object of class `data.frame`

Examples

```
data(testdata1)
head(testdata1)
```

`testdata1b`*Simulated data set 1b*

Description

Data from a random simulation with continuous and categorical data.

Usage

```
data(testdata1b)
```

Format

An object of class `data.frame`

Examples

```
data(testdata1b)
head(testdata1b)
```

testdata1c	<i>Simulated data set 1c</i>
------------	------------------------------

Description

Data from a random simulation with continuous and categorical data. This data set contains information about 237 countries (countries without unique ISO 3166 code are excluded).

Usage

```
data(testdata1c)
```

Format

An object of class `data.frame`

Examples

```
data(testdata1c)
head(testdata1c)
```

worldplot	<i>worldplot</i>
-----------	------------------

Description

Plot a world heat map based on a continuous variable.

Usage

```
worldplot(
  data,
  ColName,
  CountryName,
  CountryNameType = "isoa2",
  rangeVal,
  longitude = c(-180, 180),
  latitude = c(-90, 90),
  crs = 4326,
  title = "",
  legendTitle = as.character(ColName),
  legend.position = "right",
  annote = FALSE,
  div = 1,
  palette_option = "D",
  label.color = "white",
```

```

    label.size = 2,
    na_colour = "grey80",
    transform_limits = TRUE,
    shadows = TRUE,
    UK_as_GB = TRUE
  )

```

Arguments

<code>data</code>	Data set containing the list of nations and the variable that we want to plot.
<code>ColName</code>	Character variable with the name of the variable of interest.
<code>CountryName</code>	Character variable with the name of the country names column.
<code>CountryNameType</code>	Character variable with the coding for <code>CountryName</code> . One of <code>isoa2</code> (default, standing for ISO 3166-1 alpha-2 code), <code>isoa3</code> , or <code>name</code> .
<code>rangeVal</code>	Limit values (minimum and maximum) that are to be defined for the map. If not specified, the minimum and maximum are taken, and a message is displayed.
<code>longitude</code>	Longitude limits. Default is <code>c(-180, 180)</code> (whole world with crs as EPSG::4326).
<code>latitude</code>	Latitude limits. Default is <code>c(-90, 90)</code> (whole world with crs as EPSG::4326).
<code>crs</code>	Coordinate reference system (EPSG). By default the value is 4326, which corresponds to EPSG::4326 (WGS84)
<code>title</code>	Title of the plot. Default is no title.
<code>legendTitle</code>	Title of the legend. Default is the name of the filling variable.
<code>legend.position</code>	Position of the legend. If set to "none", no legend is displayed
<code>annotate</code>	Do you want to plot country labels (ISO 3166-1 alpha-2 code) on the map? Default is set to FALSE.
<code>div</code>	Parameter for modifying the elements dimensions in the map. Usually, it does not need to be modified. Default value is 1.
<code>palette_option</code>	Character string indicating the palette to be used. Available options range between "A" and "H".
<code>label.color</code>	Color of the labels if <code>annotate = TRUE</code> . Default is white
<code>label.size</code>	Size of the labels if <code>annotate = TRUE</code>
<code>na_colour</code>	The colour to be used for countries with missing information. Default is grey80
<code>transform_limits</code>	Only if <code>crs</code> is specified and different from 4326. If TRUE (the default) the program expects to receive values of longitude and latitude as in EPSG 4326, (i.e., within -180, +180 for longitude and within -90, +90 for latitude) and automatically updates to the new crs. Set to FALSE if you want to define longitude and latitude limits based on the new crs
<code>shadows</code>	If TRUE, add shadows to the country labels (only if <code>annotate = TRUE</code>)
<code>UK_as_GB</code>	Argument passed to <code>countrycoord_data</code> if <code>annotate</code> is set to TRUE. Do you want to translate the GB <code>isoa2</code> code to UK? If FALSE, GB is returned in the output data.frame. If TRUE (default), UK is returned.

Value

a map

Examples

```
data(testdata1b)
worldplot(data = testdata1b,
          div = 1,
          ColName = "VNum",
          CountryName = "Cshort",
          CountryNameType = "isoa2",
          rangeVal = c(0,50),
          annotate = FALSE)
```

worldplotCat

worldplotCat

Description

Plot a world heat map based on a categorical variable.

Usage

```
worldplotCat(
  data,
  ColName,
  CountryName,
  CountryNameType = "isoa2",
  longitude = c(-180, 180),
  latitude = c(-90, 90),
  crs = 4326,
  title = "",
  legendTitle = as.character(ColName),
  legend.position = "right",
  Categories = levels(factor(map_df$MapFiller)),
  na.as.category = TRUE,
  label.color = "white",
  label.size = 2,
  annotate = FALSE,
  div = 1,
  palette_option = "D",
  na_colour = "grey80",
  transform_limits = TRUE,
  shadows = TRUE,
  UK_as_GB = TRUE
)
```

Arguments

<code>data</code>	Data set containing the list of nations and the variable that we want to plot.
<code>ColName</code>	Character variable with the name of the variable of interest.
<code>CountryName</code>	Character variable with the name of the country names column.
<code>CountryNameType</code>	Character variable with the coding for CountryName. One of <code>isoa2</code> (default, standing for ISO 3166-1 alpha-2 code), <code>isoa3</code> , or <code>name</code> .
<code>longitude</code>	Longitude limits. Default is <code>c(-180, 180)</code> (whole world with crs as EPSG::4326).
<code>latitude</code>	Latitude limits. Default is <code>c(-90, 90)</code> (whole world with crs as EPSG::4326).
<code>crs</code>	Coordinate reference system (EPSG). By default the value is 4326, which corresponds to EPSG::4326 (WGS84)
<code>title</code>	Title of the plot. Default is no title.
<code>legendTitle</code>	Title of the legend. Default is the name of the filling variable.
<code>legend.position</code>	Position of the legend. If set to "none", no legend is displayed
<code>Categories</code>	categories labels to be plotted in the legend.
<code>na.as.category</code>	Treat NA as a separate category? If 'TRUE, NA will also appear in the legend as one of the categories.
<code>label.color</code>	Color of the labels if <code>annote = TRUE</code> . Default is white
<code>label.size</code>	Size of the labels if <code>annote = TRUE</code>
<code>annote</code>	Do you want to plot country labels (ISO 3166-1 alpha-2 code) on the map? Default is set to FALSE.
<code>div</code>	Parameter for modifying the elements dimensions in the map. Usually, it does not need to be modified. Default value is 1.
<code>palette_option</code>	Character string indicating the palette to be used. Available options range between "A" and "H". You can also enter a string with a colour for each category
<code>na_colour</code>	The colour to be used for countries with missing information. Default is grey80
<code>transform_limits</code>	Only if <code>crs</code> is specified and different from 4326. If TRUE (the default) the program expects to receive values of longitude and latitude as in EPSG 4326, (i.e., within -180, +180 for longitude and within -90, +90 for latitude) and automatically updates to the new crs. Set to FALSE if you want to define longitude and latitude limits based on the new crs
<code>shadows</code>	If TRUE, add shadows to the country labels (only if <code>annote = TRUE</code>)
<code>UK_as_GB</code>	Argument passed to <code>countrycoord_data</code> if <code>annote</code> is set to TRUE. Do you want to translate the GB <code>isoa2</code> code to UK? If FALSE, GB is returned in the output data.frame. If TRUE (default), UK is returned.

Value

a map

Examples

```
data(testdata1b)
worldplotCat(data = testdata1b,
             div = 1,
             ColName = "VCat",
             CountryName = "Cshort",
             CountryNameType = "isoa2",
             annotate = FALSE)
```

Index

* datasets

- map_df0, [4](#)
- regions_ITBIH, [4](#)
- testdata1, [5](#)
- testdata1b, [5](#)
- testdata1c, [6](#)

countrycoord_data, [2](#)

empty_editable_map, [3](#)

geometries_data, [3](#)

map_df0, [4](#)

regions_ITBIH, [4](#)

testdata1, [5](#)

testdata1b, [5](#)

testdata1c, [6](#)

worldplot, [6](#)

worldplotCat, [8](#)