

Package ‘RSwissMaps’

June 9, 2019

Title Plot and Save Customised Swiss Maps

Version 0.1.0.1

Description Allows to link data to Swiss administrative divisions (municipalities, districts, cantons) and to plot and save customised maps thereof. Furthermore, the package allows to generate tailored templates for data collection. The used geodata is publicly available on the Swiss Federal Statistical Office website <<https://www.bfs.admin.ch/bfs/de/home/dienstleistungen/geostat/geodaten-bundesstatistik.html>>.

Depends R (>= 2.10)

License GPL (>= 2)

Encoding UTF-8

LazyData true

Date 2017-10-01

RoxygenNote 6.0.1

BugReports <https://github.com/zumbov2/RSwissMaps/issues>

Imports ggplot2, dplyr, downloader

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author David Zumbach [aut, cre]

Maintainer David Zumbach <david.zumbach@gfzb.ch>

Repository CRAN

Date/Publication 2019-06-09 18:45:35 UTC

R topics documented:

can.plot	2
can.template	4
dis.plot	4
dis.template	6

map.load	7
mapCH2016	8
mun.plot	9
mun.template	11

Index	12
--------------	-----------

can.plot	<i>Plotting canton-level maps of Switzerland</i>
----------	--

Description

can.plot offers the possibility to visualise Swiss canton-level data on a map. It uses the geodata that is made publicly available by the [Swiss Federal Statistical Office](#). Geodata is available for the period between 2001 to 2017. The argument cantons makes it easy to plot customised maps. The function [can.template](#) provides templates for data collection.

Usage

```
can.plot(bfs_id, data, year, add_geodata_path = ".", endofyear = FALSE,
         cantons = NULL, lakes = "all", boundaries = "c",
         boundaries_size = 0.2, boundaries_color = "white", extrema = "global",
         continuous = TRUE, color_continuous = c("#fee5d9", "#a50f15"),
         color_discrete = c("seq", "1"), color_na = "gray90", title = NULL,
         subtitle = NULL, caption = NULL, legend_title = NULL,
         legend_position = "bottom", save = FALSE, filename = "can_plot.png",
         dpi = 300, width = NA, height = NA, units = "cm")
```

Arguments

bfs_id	a numeric vector with canton identification numbers as used by the Swiss Federal Statistical Office corresponding to data. For templates, see can.template .
data	a vector of data at canton-level.
year	numeric. Indicates what map version to plot. Available with package: 2016. For additional geodata, see: map.load.
add_geodata_path	a character string with the path to the folder where additional geodata is saved.
endofyear	if TRUE, administrative boundaries as by December 31 of year are plotted. Available: year > 2010.
cantons	input either numeric (canton identification numbers) or character (canton abbreviations) vector to plot district of selected cantons.
lakes	input either numeric (lake identification numbers) or character (lakes names) vector to plot selected lakes. If "none", no lakes are plotted. If "all", all available lakes are plotted.
boundaries	character vector naming the boundaries to map. <ul style="list-style-type: none"> "c": canton boundaries

	<ul style="list-style-type: none"> • "n": national boundaries
boundaries_size	numeric vector that specifies the size of the corresponding element in boundaries.
boundaries_color	character vector that specifies the color of the corresponding element in boundaries.
extrema	if local, data of municipalities that are not selected by cantons is not considered in the color scaling process.
continuous	set to FALSE, if data is non-continuous.
color_continuous	character vector with 2 elements to specify the plot color if continuous = TRUE. color_continuous = c(low, high) resulting in a two color gradient from color low to color high.
color_discrete	character vector with 2 elements to specify the plot color if continuous = FALSE. color_discrete = c(type, palette) resulting in a color scheme from ColorBrewer that are particularly well suited to display discrete values on a map. <ul style="list-style-type: none"> • type: one of seq (sequential), div (diverging) or qual (qualitative) • palette: index into the list of palettes of appropriate type; character needed!
color_na	color of missing values.
title, subtitle, caption, legend_title	text for titles or caption below the plot.
legend_position	the position of the legend ("none", "left", "right", "bottom", "top")
save	if TRUE, the plot will be saved to disk.
filename	file name to create on disk incl. image file format (".jpeg", ".png", ".tiff", etc.).
dpi	plot resolution. Applies only to raster output types.
width, height, units	plot size in units ("in", "cm", or "mm"). If not supplied, uses the size of current graphics device.

Examples

```
# Generating sample data:
dt <- can.template(2016)
for(i in 1:nrow(dt)){dt$values[i] <- sample(c(300:700), 1)/1000}

# Plotting sample data:
can.plot(dt$bfs_nr, dt$values, 2016)
```

can.template	<i>Generating templates for canton-level data of Switzerland</i>
--------------	--

Description

can.template offers the possibility to generate Swiss canton-level data templates in the form of data frames. The package comes with data for 2016. Additional data can be downloaded with map.load. The templates contain the canton identification numbers as used by the Swiss Federal Statistical Office and the canton abbreviations.

Usage

```
can.template(year, add_geodata_path = ".", endofyear = FALSE,
             cantons = NULL)
```

Arguments

year	numeric. Available with package: 2016. For additional years, see: map.load.
add_geodata_path	a character string with the path to the folder where additional geodata is saved.
endofyear	if TRUE, cantons as by December 31 of year are used for the template.
cantons	input either numeric (canton identification numbers) or character (canton abbreviations) vector to create tailor-made templates.

Examples

```
# Generating template:
can.template(2016)
```

dis.plot	<i>Plotting district-level maps of Switzerland</i>
----------	--

Description

dis.plot offers the possibility to visualise Swiss district-level data on a map. It uses the geodata that is made publicly available by the [Swiss Federal Statistical Office](#). Geodata is available for the period between 2001 to 2017. The arguments cantons, and districts make it easy to plot customised maps. The function [dis.template](#) provides templates for data collection.

Usage

```
dis.plot(bfs_id, data, year, add_geodata_path = ".", endofyear = FALSE,
        cantons = NULL, districts = NULL, lakes = "all", boundaries = "d",
        boundaries_size = 0.2, boundaries_color = "white", extrema = "global",
        continuous = TRUE, color_continuous = c("#fee5d9", "#a50f15"),
        color_discrete = c("seq", "1"), color_na = "gray90", title = NULL,
        subtitle = NULL, caption = NULL, legend_title = NULL,
        legend_position = "bottom", save = FALSE, filename = "dis_plot.png",
        dpi = 300, width = NA, height = NA, units = "cm")
```

Arguments

<code>bfs_id</code>	a numeric vector with district identification numbers as used by the Swiss Federal Statistical Office corresponding to data. For templates, see dis.template .
<code>data</code>	a vector of data at district-level.
<code>year</code>	numeric. Indicates what map version to plot. Available with package: 2016. For additional geodata, see: <code>map.load</code> .
<code>add_geodata_path</code>	a character string with the path to the folder where additional geodata is saved.
<code>endofyear</code>	if TRUE, administrative boundaries as by December 31 of year are plotted. Available: <code>year > 2010</code> .
<code>cantons</code>	input either numeric (canton identification numbers) or character (canton abbreviations) vector to plot district of selected cantons.
<code>districts</code>	input either numeric (district identification numbers) or character (district names) vector to plot selected districts.
<code>lakes</code>	input either numeric (lake identification numbers) or character (lakes names) vector to plot selected lakes. If "none", no lakes are plotted. If "all", all available lakes are plotted.
<code>boundaries</code>	character vector naming the boundaries to map. <ul style="list-style-type: none"> • "d": district boundaries • "c": canton boundaries • "n": national boundaries
<code>boundaries_size</code>	numeric vector that specifies the size of the corresponding element in <code>boundaries</code> .
<code>boundaries_color</code>	character vector that specifies the color of the corresponding element in <code>boundaries</code> .
<code>extrema</code>	if local, data of municipalities that are not selected by cantons or districts is not considered in the color scaling process.
<code>continuous</code>	set to FALSE, if data is non-continuous.
<code>color_continuous</code>	character vector with 2 elements to specify the plot color if <code>continuous = TRUE</code> . <code>color_continuous = c(low, high)</code> resulting in a two color gradient from color low to color high.

`color_discrete` character vector with 2 elements to specify the plot color if `continuous = FALSE`.
`color_discrete = c(type, palette)` resulting in a color scheme from **ColorBrewer** that are particularly well suited to display discrete values on a map.

- `type`: one of `seq` (sequential), `div` (diverging) or `qual` (qualitative)
- `palette`: index into the list of palettes of appropriate type; character needed!

`color_na` color of missing values.

`title`, `subtitle`, `caption`, `legend_title`
text for titles or caption below the plot.

`legend_position`
the position of the legend ("`none`", "`left`", "`right`", "`bottom`", "`top`")

`save`
if `TRUE`, the plot will be saved to disk.

`filename`
file name to create on disk incl. image file format ("`.jpeg`", "`.png`", "`.tiff`", etc.).

`dpi`
plot resolution. Applies only to raster output types.

`width`, `height`, `units`
plot size in units ("`in`", "`cm`", or "`mm`"). If not supplied, uses the size of current graphics device.

Examples

```
# Generating sample data:
dt <- dis.template(2016)
for(i in 1:nrow(dt)){dt$values[i] <- sample(c(300:700), 1)/1000}

# Plotting sample data:
dis.plot(dt$bfs_nr, dt$values, 2016)

# Plotting sample data for the canton of Aargau:
dis.plot(dt$bfs_nr, dt$values, 2016, cantons = c("AG"),
lakes = c("Hallwilersee"))
```

dis.template

Generating templates for district-level data of Switzerland

Description

`dis.template` offers the possibility to generate Swiss district-level data templates in the form of data frames. The package comes with data for 2016. Additional data can be downloaded with `map.load`. The templates contain the district identification numbers as used by the Swiss Federal Statistical Office and the district names.

Usage

```
dis.template(year, add_geodata_path = ".", endofyear = FALSE,
cantons = NULL, districts = NULL)
```

Arguments

year	numeric. Available with package: 2016. For additional years, see: map.load.
add_geodata_path	a character string with the path to the folder where additional geodata is saved.
endofyear	if TRUE, districts as by December 31 of year are used for the template.
cantons	input either numeric (canton identification numbers) or character (canton abbreviations) vector to create canton-specific templates.
districts	input either numeric (district identification numbers) or character (district names) vector to create tailor-made templates.

Examples

```
# Generating template:
dis.template(2016)

# Generating template for the districts of the canton of Aargau:
dis.template(2016, cantons = c("AG", "ZH"))
```

map.load	<i>Download additional geodata of Swiss municipalities, districts, cantons and lakes</i>
----------	--

Description

map.load offers the possibility to download geodata of Swiss subdivisions and lakes for the years 2001-2017 from the official [RSwissMaps repository](#) on GitHub.

Usage

```
map.load(year, add_geodata_path = ".")
```

Arguments

year	numeric. Available years: 2001-2017.
add_geodata_path	a character string with the path to the folder where downloaded geodata is saved.

Examples

```
## Not run:
#Download Swiss geodata for 2017
map.load(2017)
## End(Not run)
```

`mapCH2016`*Geodata of Swiss municipalities, districts, cantons and lakes (2016)*

Description

A ggplot2-compatible data frame containing geodata of Swiss municipalities, districts, cantons and lakes for 2016. The data is made publicly available by the Swiss Federal Statistical Office. The administrative divisions are scaled at 1:2,000,000.

Usage`mapCH2016`**Format**

A data frame with 162,148 rows and 10 variables:

year year of geodata

endofyear if 0, the data dates from year-1-1, else from year-31-12

long longitude of point

lat latitude of point

group if two adjacent points are in the same group, then they get connected

type type of object point belongs to: municipality, district, canton, country, lake)

id official identification number of object

name name of object

dis district object belongs to

can canton object belongs to

Author(s)

David Zumbach <david.zumbach@gfzb.ch>

Source

Swiss Federal Statistical Office, GEOSTAT (last download 2017-09-07)

References

<https://www.bfs.admin.ch/bfs/de/home/dienstleistungen/geostat/geodaten-bundesstatistik/administrative-grenzen/generalisierte-gemeindegrenzen.html>

mun.plot

*Plotting municipality-level maps of Switzerland***Description**

mun.plot offers the possibility to visualise Swiss municipality-level data on a map. It uses the geodata that is made publicly available by the [Swiss Federal Statistical Office](#). The package comes with geodata for 2016. Additional geodata can be downloaded with map.load. The arguments cantons, districts and municipalities make it easy to plot customised maps. The function [mun.template](#) provides templates for data collection.

Usage

```
mun.plot(bfs_id, data, year, add_geodata_path = ".", endofyear = FALSE,
         cantons = NULL, districts = NULL, municipalities = NULL,
         lakes = "all", boundaries = "m", boundaries_size = 0.2,
         boundaries_color = "white", extrema = "global", continuous = TRUE,
         color_continuous = c("#fee5d9", "#a50f15"), color_discrete = c("seq",
         "1"), color_na = "gray90", title = NULL, subtitle = NULL,
         caption = NULL, legend_title = NULL, legend_position = "bottom",
         save = FALSE, filename = "mun_plot.png", dpi = 300, width = NA,
         heigth = NA, units = "cm")
```

Arguments

bfs_id	a numeric vector with municipality identification numbers as used by the Swiss Federal Statistical Office corresponding to data. For templates, see mun.template .
data	a vector of data at municipality-level.
year	numeric. Indicates what map version to plot. Available with package: 2016. For additional geodata, see: map.load.
add_geodata_path	a character string with the path to the folder where additional geodata is saved.
endofyear	if TRUE, administrative boundaries as by December 31 of year are plotted.
cantons	input either numeric (canton identification numbers) or character (canton abbreviations) vector to plot municipalities of selected cantons.
districts	input either numeric (district identification numbers) or character (district names) vector to plot municipalities of selected districts.
municipalities	input either numeric (municipality identification numbers) or character (municipality names) vector to plot selected municipalities.
lakes	input either numeric (lake identification numbers) or character (lakes names) vector to plot selected lakes. If "none", no lakes are plotted. If "all", all available lakes are plotted.
boundaries	character vector naming the boundaries to map. <ul style="list-style-type: none"> • "m": municipality boundaries (default)

	<ul style="list-style-type: none"> • "d": district boundaries • "c": canton boundaries • "n": national boundaries
boundaries_size	numeric vector that specifies the size of the corresponding element in boundaries.
boundaries_color	character vector that specifies the color of the corresponding element in boundaries.
extrema	if local, data of municipalities that are not selected by cantons, districts, or municipalities is not considered in the color scaling process.
continuous	set to FALSE, if data is non-continuous.
color_continuous	character vector with 2 elements to specify the plot color if continuous = TRUE. color_continuous = c(low, high) resulting in a two color gradient from color low to color high.
color_discrete	character vector with 2 elements to specify the plot color if continuous = FALSE. color_discrete = c(type, palette) resulting in a color scheme from ColorBrewer that are particularly well suited to display discrete values on a map. <ul style="list-style-type: none"> • type: one of seq (sequential), div (diverging) or qual (qualitative) • palette: index into the list of palettes of appropriate type; character needed!
color_na	color of missing values.
title, subtitle, caption, legend_title	text for titles or caption below the plot.
legend_position	the position of the legend ("none", "left", "right", "bottom", "top")
save	if TRUE, the plot will be saved to disk.
filename	file name to create on disk incl. image file format (".jpeg", ".png", ".tiff", etc.).
dpi	plot resolution. Applies only to raster output types.
width, height, units	plot size in units ("in", "cm", or "mm"). If not supplied, uses the size of current graphics device.

Examples

```
# Generating sample data:
dt <- mun.template(2016)
for(i in 1:nrow(dt)){dt$values[i] <- sample(c(300:700), 1)/1000}

# Plotting sample data:
mun.plot(dt$bfs_nr, dt$values, 2016)

# Plotting sample data for the canton of Aargau:
mun.plot(dt$bfs_nr, dt$values, 2016, cantons = c("AG"),
lakes = c("Hallwilersee"))
```

`mun.template`*Generating templates for municipality-level data of Switzerland*

Description

`mun.template` offers the possibility to generate Swiss municipality-level data templates in the form of data frames. The package comes with data for 2016. Additional data can be downloaded with `map.load`. The templates contain the municipality identification numbers as used by the Swiss Federal Statistical Office and the municipality names.

Usage

```
mun.template(year, add_geodata_path = ".", endofyear = FALSE,
             cantons = NULL, districts = NULL, municipalities = NULL)
```

Arguments

<code>year</code>	numeric. Available with package: 2016. For additional years, see: <code>map.load</code> .
<code>add_geodata_path</code>	a character string with the path to the folder where additional geodata is saved.
<code>endofyear</code>	if TRUE, municipalities as by December 31 of <code>year</code> are used for the template.
<code>cantons</code>	input either numeric (canton identification numbers) or character (canton abbreviations) vector to create canton-specific templates.
<code>districts</code>	input either numeric (district identification numbers) or character (district names) vector to create district-specific templates.
<code>municipalities</code>	input either numeric (municipality identification numbers) or character (municipality names) vector to create tailor-made templates.

Examples

```
# Generating template:
mun.template(2016)

# Generating template for the municipalities of the canton of Aargau:
mun.template(2016, cantons = c("AG", "ZH"))
```

Index

*Topic **datasets**

mapCH2016, 8

can.plot, 2

can.template, 2, 4

dis.plot, 4

dis.template, 4, 5, 6

map.load, 7

mapCH2016, 8

mun.plot, 9

mun.template, 9, 11