

Package ‘cmsafvis’

October 13, 2020

Title Visualize CM SAF NetCDF Data

Version 1.0.1

Description The Satellite Application Facility on Climate Monitoring (CM SAF) is a ground segment of the European Organization for the Exploitation of Meteorological Satellites (EUMETSAT) and one of EUMETSAT's Satellite Application Facilities. The CM SAF contributes to the sustainable monitoring of the climate system by providing essential climate variables related to the energy and water cycle of the atmosphere (<<http://www.cmsaf.eu>>). It is a joint cooperation of eight National Meteorological and Hydrological Services. The 'cmsafvis' R-package provides a collection of R-operators for the analysis and visualization of CM SAF NetCDF data. CM SAF climate data records are provided for free via (<<https://wui.cmsaf.eu>>). Detailed information and test data are provided on the CM SAF webpage (<http://www.cmsaf.eu/R_toolbox>).

License GPL (>= 3)

Depends R (>= 3.6)

Imports animation (>= 2.6), assertthat (>= 0.2.1), cmsafops (>= 1.0.0), colorspace (>= 1.4), countrycode (>= 1.1), fields (>= 10.3), mapproj (>= 1.2.7), maps (>= 3.3.0), maptools (>= 0.9), methods (>= 3.6), ncdf4 (>= 1.17), png (>= 0.1), progress (>= 1.2.2), raster (>= 3.0), rworldxtra (>= 1.01), sp (>= 1.4), yaml (>= 2.2)

NeedsCompilation no

Repository CRAN

LazyData true

RoxygenNote 7.1.1

Encoding UTF-8

Language en-US

Suggests plotKML (>= 0.6), rgdal (>= 1.4), rnaturalearth (>= 0.1), spelling (>= 2.1), testthat (>= 2.3)

Author Steffen Kothe [aut, cre]

Maintainer Steffen Kothe <Steffen.Kothe@dwd.de>

Date/Publication 2020-10-13 08:50:05 UTC

R topics documented:

absolute_map	2
anomaly_map	5
climatology_map	8
cmsafvis	12
fieldmean_and_anomaly_map	13
fieldmean_plot	16
monitor_climate	20
quicklook	23
recalculateImageDimensions	25
render_hist_plot	26
render_instat_plot	27
render_plot	28
render_plot_1d	32
render_preview_plot	34
render_region_plot	35

Index 39

absolute_map	<i>A 'cmsaf' extension for creating absolute valued plots.</i>
--------------	--

Description

This plotting routine generates graphical output for the given variable within the given time range and area. Dependent on the output format a PNG or MP4 is created.

Usage

```
absolute_map(
  config = NULL,
  variable = NULL,
  accumulate = FALSE,
  infile = NULL,
  temp_dir = tempdir(),
  out_dir = getwd(),
  start_date = NULL,
  end_date = NULL,
  country_code = "S_A",
  lon_min = NULL,
  lon_max = NULL,
  lat_min = NULL,
  lat_max = NULL,
  outfile_name = NULL,
  output_format = "animation",
  animation_pace = 0.07,
  freeze_animation = FALSE,
```

```

    min_value = NULL,
    max_value = NULL,
    nbreaks = NULL,
    language = "eng",
    keep_files = TRUE,
    states = FALSE,
    attach = FALSE,
    infile_attach = "auto",
    verbose = TRUE
)

```

Arguments

config	Path to YAML config file (character). The config file does not have to specify all arguments. Each addressed argument has to be formatted according to the example config file: (#TODO: LINK EXAMPLE CONFIG FILE!).
variable	Name of variable in infile (NULL or character). If NULL then the first variable from the infile is taken.
accumulate	Whether the input file should be accumulated (logical).
infile	Path to NetCDF file (NULL or character). If NULL then it needs to be specified in the config file.
temp_dir	Path to temporary working directory (character).
out_dir	Path to output directory (character).
start_date	Start date in format of 'YYYY-MM-DD' (NULL or character). If NULL then the first date of the infile is used.
end_date	End date in format of 'YYYY-MM-DD' (NULL or character). If NULL then the last date of the infile is used.
country_code	Either a country code in iso3c format or from the following: 'AFR' for Africa, 'EUR' for Europe, 'TOT' for the total disc, or 'S_A' for an arbitrary region selection (character). If a country is passed the data from within this country is extracted, else a rectangular box is visualized. Directly provided latitude and longitude ranges will be ignored in case of 'AFR', 'EUR' or 'TOT'.
lon_min	Longitude of lower left corner (NULL or numeric). If NULL then the smallest longitude of the infile is used.
lon_max	Longitude of upper right left corner (NULL or numeric). If NULL then the largest longitude of the infile is used.
lat_min	Latitude of lower left corner (NULL or numeric). If NULL then the smallest latitude of the infile is used.
lat_max	Latitude of upper right corner (NULL or numeric). If NULL then the largest latitude of the infile is used.
outfile_name	Filename of the PNG or MP4 outfile (NULL or character). If NULL then a name is computed from the current configuration. Please match the file ending according to the output_format.
output_format	Specification of output format (either 'graphic' for PNG or 'animation' for MP4).

animation_pace	Pace of the animation in seconds (positive numeric). This only has an effect if <code>output_format == 'animation'</code> .
freeze_animation	If TRUE then the animation will freeze at the last frame (logical).
min_value	Lower values than this are ignored (NULL or numeric). If NULL, no values are ignored.
max_value	Larger values than this are ignored (NULL or numeric). If NULL, no values are ignored.
nbreaks	Number of color breaks (NULL or positive integer). A value will be computed if NULL is passed.
language	Language used for title, legend, etc. in plots (either 'eng' for English or 'deu' for German).
keep_files	A flag indicating whether all files created in the process of obtaining the output file should be kept (logical). If false, all intermediate results are deleted, otherwise all are kept. Keeping these files could improve performance in further function calls.
states	Whether to crop/plot administration level of states (logical).
attach	Whether to temporarily merge the infile to an already existing one. (logical).
infile_attach	File to attach the infile to. When 'auto', a suitable file will be searched in <code>out_dir</code> . If attach is false, this will be ignored(character).
verbose	Whether to display progress messages (logical).

Details

You can pass a YAML config file and/or specify the arguments directly. Argument prioritization is done in the following way: Direct argument > config argument > default argument. Thus, if you pass a existing config file but also want to modify a specific argument you can do that easily.

Examples

```
## Create an example NetCDF file with a similar structure as used by CM
## SAF. The file is created with the ncdf4 package. Alternatively
## example data can be freely downloaded here: <https://wui.cmsaf.eu/>

library(ncdf4)

## create some (non-realistic) example data

lon <- seq(5, 15, 0.5)
lat <- seq(45, 55, 0.5)
time <- seq(as.Date("2010-01-01"), as.Date("2010-01-01"), "days")
origin <- as.Date("1983-01-01 00:00:00")
time <- as.numeric(difftime(time, origin, units = "hour"))
data <- array(250:350, dim = c(21, 21, 1))

## create example NetCDF
infile <- tempfile("input", fileext = ".nc")
```

```

x <- ncdim_def(name = "lon", units = "degrees_east", vals = lon)
y <- ncdim_def(name = "lat", units = "degrees_north", vals = lat)
t <- ncdim_def(name = "time", units = "hours since 1983-01-01 00:00:00",
  vals = time, unlim = TRUE)
var1 <- ncvar_def("SDU", "W m-2", list(x, y, t), -1, prec = "short")
vars <- list(var1)
ncnew <- nc_create(infile, vars)
ncvar_put(ncnew, var1, data)
ncatt_put(ncnew, "lon", "standard_name", "longitude", prec = "text")
ncatt_put(ncnew, "lat", "standard_name", "latitude", prec = "text")
nc_close(ncnew)

## this will save 'output.png' in temp directory
cmsafvis::absolute_map(
  infile = infile,
  out_dir = tempdir(),
  outfile_name = "output",
  output_format = "graphic",
  keep_files = FALSE,
  verbose = FALSE
)

```

anomaly_map

A 'cmsaf' extension for creating an anomaly map.

Description

This plotting routine generates a graph showing the anomaly of a given variable within the given time range and area. The intended application is for daily accumulated data, such as sunshine duration. Dependent on the output format a PNG or MP4 is created.

Usage

```

anomaly_map(
  config = NULL,
  variable = NULL,
  accumulate = FALSE,
  infile = NULL,
  temp_dir = tempdir(),
  out_dir = getwd(),
  climate_dir = NULL,
  climate_year_start = 1983,
  climate_year_end = 2018,
  start_date = NULL,
  end_date = NULL,
  country_code = "S_A",
  lon_min = NULL,
  lon_max = NULL,

```

```

lat_min = NULL,
lat_max = NULL,
outfile_name = NULL,
output_format = "animation",
animation_pace = 0.07,
freeze_animation = FALSE,
min_value = NULL,
max_value = NULL,
nbreaks = NULL,
language = "eng",
keep_files = TRUE,
states = FALSE,
attach = FALSE,
infile_attach = "auto",
verbose = TRUE
)

```

Arguments

config	Path to YAML config file (character). The config file does not have to specify all arguments. Each addressed argument has to be formatted according to the example config file: (#TODO: LINK EXAMPLE CONFIG FILE!).
variable	Name of variable in infile (NULL or character). If NULL then the first variable from the infile is taken.
accumulate	Whether the input file should be accumulated (logical).
infile	Path to NetCDF file (NULL or character). If NULL then it needs to be specified in the config file.
temp_dir	Path to temporary working directory (character).
out_dir	Path to output directory (character).
climate_dir	Path to directory in which climatology is computed or contained (NULL or character). If NULL then the temp_dir directory is taken.
climate_year_start	Start year of climatology (integer).
climate_year_end	End year of climatology (integer).
start_date	Start date in format of 'YYYY-MM-DD' (NULL or character). If NULL then the first date of the infile is used.
end_date	End date in format of 'YYYY-MM-DD' (NULL or character). If NULL then the last date of the infile is used.
country_code	Either a country code in iso3c format or from the following: 'AFR' for Africa, 'EUR' for Europe, 'TOT' for the total disc, or 'S_A' for an arbitrary region selection (character). If a country is passed the data from within this country is extracted, else a rectangular box is visualized. Directly provided latitude and longitude ranges will be ignored in case of 'AFR', 'EUR' or 'TOT'.
lon_min	Longitude of lower left corner (NULL or numeric). If NULL then the smallest longitude of the infile is used.

lon_max	Longitude of upper right left corner (NULL or numeric). If NULL then the largest longitude of the infile is used.
lat_min	Latitude of lower left corner (NULL or numeric). If NULL then the smallest latitude of the infile is used.
lat_max	Latitude of upper right corner (NULL or numeric). If NULL then the largest latitude of the infile is used.
outfile_name	Filename of the PNG or MP4 outfile (NULL or character). If NULL then a name is computed from the current configuration. Please match the file ending according to the output_format.
output_format	Specification of output format (either 'graphic' for PNG or 'animation' for MP4).
animation_pace	Pace of the animation in seconds (positive numeric). This only has an effect if output_format == 'animation'.
freeze_animation	If TRUE then the animation will freeze at the last frame (logical).
min_value	Lower values than this are ignored (NULL or numeric). If NULL, no values are ignored.
max_value	Larger values than this are ignored (NULL or numeric). If NULL, no values are ignored.
nbreaks	Number of color breaks (NULL or positive integer). A value will be computed if NULL is passed.
language	Language used for title, legend, etc. in plots (either 'eng' for English or 'deu' for German).
keep_files	A flag indicating whether all files created in the process of obtaining the output file should be kept (logical). If false, all intermediate results are deleted, otherwise all are kept. Keeping these files could improve performance in further function calls.
states	Whether to crop/plot administration level of states (logical).
attach	Whether to temporarily merge the infile to an already existing one. (logical).
infile_attach	File to attach the infile to. When 'auto', a suitable file will be searched in out_dir. If attach is false, this will be ignored(character).
verbose	Whether to display progress messages (logical).

Details

You can pass a YAML config file and / or specify the arguments directly. Argument prioritization is done in the following way: Direct argument > config argument > default argument. Thus, if you pass a existing config file but also want to modify a specific argument you can do that easily.

Examples

```
## Create an example NetCDF file with a similar structure as used by CM
## SAF. The file is created with the ncdf4 package. Alternatively
## example data can be freely downloaded here: <https://wui.cmsaf.eu/>

library(ncdf4)
```

```

## create some (non-realistic) example data

lon <- seq(5, 15, 0.5)
lat <- seq(45, 55, 0.5)
time <- seq(as.Date("2010-01-01"), as.Date("2011-12-31"), "days")
origin <- as.Date("1983-01-01 00:00:00")
time <- as.numeric(difftime(time, origin, units = "hour"))
data <- array(250:350, dim = c(21, 21, 2 * 365))

## create example NetCDF
infile <- tempfile("input", fileext = ".nc")

x <- ncdim_def(name = "lon", units = "degrees_east", vals = lon)
y <- ncdim_def(name = "lat", units = "degrees_north", vals = lat)
t <- ncdim_def(name = "time", units = "hours since 1983-01-01 00:00:00",
  vals = time, unlim = TRUE)
var1 <- ncvar_def("SDU", "W m-2", list(x, y, t), -1, prec = "short")
vars <- list(var1)
ncnew <- nc_create(infile, vars)
ncvar_put(ncnew, var1, data)
ncatt_put(ncnew, "lon", "standard_name", "longitude", prec = "text")
ncatt_put(ncnew, "lat", "standard_name", "latitude", prec = "text")
nc_close(ncnew)

## this will save 'output.png' in temp directory
cmsafvis::anomaly_map(
  accumulate = TRUE,
  infile = infile,
  out_dir = tempdir(),
  climate_dir = tempdir(),
  climate_year_start = 2010,
  climate_year_end = 2011,
  end_date = "2010-01-05",
  outfile_name = "output",
  output_format = "graphic",
  keep_files = FALSE,
  verbose = FALSE
)

```

climatology_map

A 'cmsaf' extension for creating a climatology map.

Description

This plotting routine generates the climatological mean for the given variable within the given time range and area. The intended application is for daily accumulated data, such as sunshine duration. Dependent on the output format a PNG or MP4 is created.

Usage

```

climatology_map(
  config = NULL,
  variable = NULL,
  accumulate = FALSE,
  infile = NULL,
  temp_dir = tempdir(),
  out_dir = getwd(),
  climate_dir = NULL,
  climate_year_start = 1983,
  climate_year_end = 2018,
  start_date = NULL,
  end_date = NULL,
  country_code = "S_A",
  lon_min = NULL,
  lon_max = NULL,
  lat_min = NULL,
  lat_max = NULL,
  outfile_name = NULL,
  output_format = "animation",
  animation_pace = 0.07,
  freeze_animation = FALSE,
  min_value = NULL,
  max_value = NULL,
  nbreaks = NULL,
  language = "eng",
  keep_files = TRUE,
  states = FALSE,
  attach = FALSE,
  infile_attach = "auto",
  verbose = TRUE
)

```

Arguments

config	Path to YAML config file (character). The config file does not have to specify all arguments. Each addressed argument has to be formatted according to the example config file: (#TODO: LINK EXAMPLE CONFIG FILE!).
variable	Name of variable in infile (NULL or character). If NULL then the first variable from the infile is taken.
accumulate	Whether the input file should be accumulated (logical).
infile	Path to NetCDF file (NULL or character). If NULL then it needs to be specified in the config file.
temp_dir	Path to temporary working directory (character).
out_dir	Path to output directory (character).
climate_dir	Path to directory in which climatology is computed or contained (NULL or character). If NULL then the temp_dir directory is taken.

climate_year_start	Start year of climatology (integer).
climate_year_end	End year of climatology (integer).
start_date	Start date in format of 'YYYY-MM-DD' (NULL or character). If NULL then the first date of the infile is used.
end_date	End date in format of 'YYYY-MM-DD' (NULL or character). If NULL then the last date of the infile is used.
country_code	Either a country code in iso3c format or from the following: 'AFR' for Africa, 'EUR' for Europe, 'TOT' for the total disc, or 'S_A' for an arbitrary region selection (character). If a country is passed the data from within this country is extracted, else a rectangular box is visualized. Directly provided latitude and longitude ranges will be ignored in case of 'AFR', 'EUR' or 'TOT'.
lon_min	Longitude of lower left corner (NULL or numeric). If NULL then the smallest longitude of the infile is used.
lon_max	Longitude of upper right left corner (NULL or numeric). If NULL then the largest longitude of the infile is used.
lat_min	Latitude of lower left corner (NULL or numeric). If NULL then the smallest latitude of the infile is used.
lat_max	Latitude of upper right corner (NULL or numeric). If NULL then the largest latitude of the infile is used.
outfile_name	Filename of the PNG or MP4 outfile (NULL or character). If NULL then a name is computed from the current configuration. Please match the file ending according to the output_format.
output_format	Specification of output format (either 'graphic' for PNG or 'animation' for MP4).
animation_pace	Pace of the animation in seconds (positive numeric). This only has an effect if output_format == 'animation'.
freeze_animation	If TRUE then the animation will freeze at the last frame (logical).
min_value	Lower values than this are ignored (NULL or numeric). If NULL, no values are ignored.
max_value	Larger values than this are ignored (NULL or numeric). If NULL, no values are ignored.
nbreaks	Number of color breaks (NULL or positive integer). A value will be computed if NULL is passed.
language	Language used for title, legend, etc. in plots (either 'eng' for English or 'deu' for German).
keep_files	A flag indicating whether all files created in the process of obtaining the output file should be kept (logical). If false, all intermediate results are deleted, otherwise all are kept. Keeping these files could improve performance in further function calls.
states	Whether to crop/plot administration level of states (logical).
attach	Whether to temporarily merge the infile to an already existing one. (logical).

`infile_attach` File to attach the infile to. When 'auto', a suitable file will be searched in `out_dir`. If attach is false, this will be ignored(character).

`verbose` Whether to display progress messages (logical).

Details

You can pass a YAML config file and/or specify the arguments directly. Argument prioritization is done in the following way: Direct argument > config argument > default argument. Thus, if you pass a existing config file but also want to modify a specific argument you can do that easily.

Examples

```
## Create an example NetCDF file with a similar structure as used by CM
## SAF. The file is created with the ncdf4 package. Alternatively
## example data can be freely downloaded here: <https://wui.cmsaf.eu/>
```

```
library(ncdf4)
```

```
## create some (non-realistic) example data
```

```
lon <- seq(5, 15, 0.5)
lat <- seq(45, 55, 0.5)
time <- seq(as.Date("2010-01-01"), as.Date("2011-12-31"), "days")
origin <- as.Date("1983-01-01 00:00:00")
time <- as.numeric(difftime(time, origin, units = "hour"))
data <- array(250:350, dim = c(21, 21, 2 * 365))
```

```
## create example NetCDF
```

```
infile <- tempfile("input", fileext = ".nc")
```

```
x <- ncdim_def(name = "lon", units = "degrees_east", vals = lon)
y <- ncdim_def(name = "lat", units = "degrees_north", vals = lat)
t <- ncdim_def(name = "time", units = "hours since 1983-01-01 00:00:00",
  vals = time, unlim = TRUE)
var1 <- ncvar_def("SDU", "W m-2", list(x, y, t), -1, prec = "short")
vars <- list(var1)
ncnew <- nc_create(infile, vars)
ncvar_put(ncnew, var1, data)
ncatt_put(ncnew, "lon", "standard_name", "longitude", prec = "text")
ncatt_put(ncnew, "lat", "standard_name", "latitude", prec = "text")
nc_close(ncnew)
```

```
## this will save 'output.png' in temp directory
```

```
cmsafvis::climatology_map(
  accumulate = TRUE,
  infile = infile,
  out_dir = tempdir(),
  climate_dir = tempdir(),
  climate_year_start = 2010,
  climate_year_end = 2011,
  end_date = "2010-01-05",
  outfile_name = "output",
```

```
output_format = "graphic",
keep_files = FALSE,
verbose = FALSE
)
```

cmsafvis

cmsafvis: A 'cmsaf' package extension for visualization of CM SAF NetCDF data.

Description

The 'cmsafvis' plotting routines are designed to analyze climate files by generating graphics and videos. The functions were designed and tested for daily valued CM SAF NetCDF data. As interface to NetCDF data the [ncdf4 package](#) is used.

Absolute

[absolute_map](#)

Anomaly

[anomaly_map](#)

Climatology

[climatology_map](#)

Fieldmean

[fieldmean_plot](#)

Fieldmean and anomaly plots

[fieldmean_and_anomaly_map](#)

Author(s)

Maintainer: Steffen Kothe <Steffen.Kothe@dwd.de>

Contact: CM SAF Team <contact.cmsaf@dwd.de>

References

http://www.cmsaf.eu/R_toolbox

Kothe, S.; Hollmann, R.; Pfeifroth, U.; Träger-Chatterjee, C.; Trentmann, J. The CM SAF R Toolbox—A Tool for the Easy Usage of Satellite-Based Climate Data in NetCDF Format. ISPRS Int. J. Geo-Inf. 2019, 8, 109. <https://doi.org/10.3390/ijgi8030109>

fieldmean_and_anomaly_map

A 'cmsaf' extension for creating both, a spatial mean and an anomaly map.

Description

This plotting routine generates a graph showing the evolution of the spatial mean of a given variable and the corresponding anomaly map within the given time range and area. The intended application is for daily accumulated data, such as sunshine duration. Dependent on the output format a PNG or MP4 is created.

Usage

```
fieldmean_and_anomaly_map(  
  config = NULL,  
  variable = NULL,  
  accumulate = FALSE,  
  infile = NULL,  
  temp_dir = tempdir(),  
  out_dir = getwd(),  
  climate_dir = NULL,  
  climate_year_start,  
  climate_year_end,  
  show_extreme_climate_years = NULL,  
  climatology_until_eoy = FALSE,  
  start_date = NULL,  
  end_date = NULL,  
  country_code = "S_A",  
  lon_min = NULL,  
  lon_max = NULL,  
  lat_min = NULL,  
  lat_max = NULL,  
  outfile_name = NULL,  
  output_format = "animation",  
  animation_pace = 0.07,  
  freeze_animation = FALSE,  
  min_value = NULL,  
  max_value = NULL,  
  nbreaks = NULL,  
  language = "eng",  
  keep_files = TRUE,  
  states = FALSE,  
  attach = FALSE,  
  infile_attach = "auto",  
  verbose = TRUE  
)
```

Arguments

<code>config</code>	Path to YAML config file (character). The config file does not have to specify all arguments. Each addressed argument has to be formatted according to the example config file: (#TODO: LINK EXAMPLE CONFIG FILE!).
<code>variable</code>	Name of variable in infile (NULL or character). If NULL then the first variable from the infile is taken.
<code>accumulate</code>	Whether the input file should be accumulated (logical).
<code>infile</code>	Path to NetCDF file (NULL or character). If NULL then it needs to be specified in the config file.
<code>temp_dir</code>	Path to temporary working directory (character).
<code>out_dir</code>	Path to output directory (character).
<code>climate_dir</code>	Path to directory in which climatology is computed or contained (NULL or character). If NULL then the <code>temp_dir</code> directory is taken.
<code>climate_year_start</code>	Start year of climatology (integer).
<code>climate_year_end</code>	End year of climatology (integer).
<code>show_extreme_climate_years</code>	Whether the minimum and maximum of the climate years should be titled in the fieldmean plot (NULL or logical). This is usually only of interest when plotting accumulated data. If the default NULL is chosen, then it will be set to the value of <code>accumulate</code> .
<code>climatology_until_eoy</code>	Plot the climatology and fieldmeans until the end of year (logical). Only affects fieldmean plots analyzed from January 1st.
<code>start_date</code>	Start date in format of 'YYYY-MM-DD' (NULL or character). If NULL then the first date of the infile is used.
<code>end_date</code>	End date in format of 'YYYY-MM-DD' (NULL or character). If NULL then the last date of the infile is used.
<code>country_code</code>	Either a country code in iso3c format or from the following: 'AFR' for Africa, 'EUR' for Europe, 'TOT' for the total disc, or 'S_A' for an arbitrary region selection (character). If a country is passed the data from within this country is extracted, else a rectangular box is visualized. Directly provided latitude and longitude ranges will be ignored in case of 'AFR', 'EUR' or 'TOT'.
<code>lon_min</code>	Longitude of lower left corner (NULL or numeric). If NULL then the smallest longitude of the infile is used.
<code>lon_max</code>	Longitude of upper right left corner (NULL or numeric). If NULL then the largest longitude of the infile is used.
<code>lat_min</code>	Latitude of lower left corner (NULL or numeric). If NULL then the smallest latitude of the infile is used.
<code>lat_max</code>	Latitude of upper right corner (NULL or numeric). If NULL then the largest latitude of the infile is used.

outfile_name	Filename of the PNG or MP4 outfile (NULL or character). If NULL then a name is computed from the current configuration. Please match the file ending according to the output_format.
output_format	Specification of output format (either 'graphic' for PNG or 'animation' for MP4).
animation_pace	Pace of the animation in seconds (positive numeric). This only has an effect if output_format == 'animation'.
freeze_animation	If TRUE then the animation will freeze at the last frame (logical).
min_value	Lower values than this are ignored (NULL or numeric). If NULL, no values are ignored.
max_value	Larger values than this are ignored (NULL or numeric). If NULL, no values are ignored.
nbreaks	Number of color breaks (NULL or positive integer). A value will be computed if NULL is passed.
language	Language used for title, legend, etc. in plots (either 'eng' for English or 'deu' for German).
keep_files	A flag indicating whether all files created in the process of obtaining the output file should be kept (logical). If false, all intermediate results are deleted, otherwise all are kept. Keeping these files could improve performance in further function calls.
states	Whether to crop/plot administration level of states (logical).
attach	Whether to temporarily merge the infile to an already existing one. (logical).
infile_attach	File to attach the infile to. When 'auto', a suitable file will be searched in out_dir. If attach is false, this will be ignored(character).
verbose	Whether to display progress messages (logical).

Details

You can pass a YAML config file and/or specify the arguments directly. Argument prioritization is done in the following way: Direct argument > config argument > default argument. Thus, if you pass a existing config file but also want to modify a specific argument you can do that easily.

Examples

```
## Create an example NetCDF file with a similar structure as used by CM
## SAF. The file is created with the ncdf4 package. Alternatively
## example data can be freely downloaded here: <https://wui.cmsaf.eu/>

library(ncdf4)

## create some (non-realistic) example data

lon <- seq(5, 15, 0.5)
lat <- seq(45, 55, 0.5)
time <- seq(as.Date("2010-01-01"), as.Date("2011-12-31"), "days")
origin <- as.Date("1983-01-01 00:00:00")
```

```

time <- as.numeric(difftime(time, origin, units = "hour"))
data <- array(250:350, dim = c(21, 21, 2 * 365))

## create example NetCDF
infile <- tempfile("input", fileext = ".nc")

x <- ncdim_def(name = "lon", units = "degrees_east", vals = lon)
y <- ncdim_def(name = "lat", units = "degrees_north", vals = lat)
t <- ncdim_def(name = "time", units = "hours since 1983-01-01 00:00:00",
  vals = time, unlim = TRUE)
var1 <- ncvar_def("SDU", "W m-2", list(x, y, t), -1, prec = "short")
vars <- list(var1)
ncnew <- nc_create(infile, vars)
ncvar_put(ncnew, var1, data)
ncatt_put(ncnew, "lon", "standard_name", "longitude", prec = "text")
ncatt_put(ncnew, "lat", "standard_name", "latitude", prec = "text")
nc_close(ncnew)

## this will save 'output.png' in temp directory
cmsafvis::fieldmean_and_anomaly_map(
  accumulate = TRUE,
  infile = infile,
  out_dir = tempdir(),
  climate_dir = tempdir(),
  climate_year_start = 2010,
  climate_year_end = 2011,
  end_date = "2010-01-05",
  outfile_name = "output",
  output_format = "graphic",
  keep_files = FALSE,
  verbose = FALSE
)

```

fieldmean_plot

A 'cmsaf' extension for creating spatial mean plots.

Description

This plotting routine generates a graph showing the evolution of the spatial mean of a given variable within the given time range and area. The intended application is for daily accumulated data, such as sunshine duration. Dependent on the output format a PNG or MP4 is created.

Usage

```

fieldmean_plot(
  config = NULL,
  variable = NULL,
  accumulate = FALSE,
  infile = NULL,
  temp_dir = tempdir(),

```



```

out_dir = getwd(),
climate_dir = NULL,
climate_year_start = 1983,
climate_year_end = 2018,
show_extreme_climate_years = NULL,
climatology_until_eoy = FALSE,
start_date = NULL,
end_date = NULL,
country_code = "S_A",
lon_min = NULL,
lon_max = NULL,
lat_min = NULL,
lat_max = NULL,
outfile_name = NULL,
output_format = "animation",
animation_pace = 0.07,
freeze_animation = FALSE,
language = "eng",
keep_files = TRUE,
states = FALSE,
attach = FALSE,
infile_attach = "auto",
verbose = TRUE
)

```

Arguments

config	Path to YAML config file (character). The config file does not have to specify all arguments. Each addressed argument has to be formatted according to the example config file: (#TODO: LINK EXAMPLE CONFIG FILE!).
variable	Name of variable in infile (NULL or character). If NULL then the first variable from the infile is taken.
accumulate	Whether the input file should be accumulated (logical).
infile	Path to NetCDF file (NULL or character). If NULL then it needs to be specified in the config file.
temp_dir	Path to temporary working directory (character).
out_dir	Path to output directory (character).
climate_dir	Path to directory in which climatology is computed or contained (NULL or character). If NULL then the temp_dir directory is taken.
climate_year_start	Start year of climatology (integer).
climate_year_end	End year of climatology (integer).
show_extreme_climate_years	Whether the minimum and maximum of the climate years should be titled in the fieldmean plot (NULL or logical). This is usually only of interest when plotting

	accumulated data. If the default NULL is chosen, then it will be set to the value of accumulate.
climatology_until_eoy	Plot the climatology and fieldmeans until the end of year (logical). Only affects fieldmean plots analyzed from January 1st.
start_date	Start date in format of 'YYYY-MM-DD' (NULL or character). If NULL then the first date of the infile is used.
end_date	End date in format of 'YYYY-MM-DD' (NULL or character). If NULL then the last date of the infile is used.
country_code	Either a country code in iso3c format or from the following: 'AFR' for Africa, 'EUR' for Europe, 'TOT' for the total disc, or 'S_A' for an arbitrary region selection (character). If a country is passed the data from within this country is extracted, else a rectangular box is visualized. Directly provided latitude and longitude ranges will be ignored in case of 'AFR', 'EUR' or 'TOT'.
lon_min	Longitude of lower left corner (NULL or numeric). If NULL then the smallest longitude of the infile is used.
lon_max	Longitude of upper right left corner (NULL or numeric). If NULL then the largest longitude of the infile is used.
lat_min	Latitude of lower left corner (NULL or numeric). If NULL then the smallest latitude of the infile is used.
lat_max	Latitude of upper right corner (NULL or numeric). If NULL then the largest latitude of the infile is used.
outfile_name	Filename of the PNG or MP4 outfile (NULL or character). If NULL then a name is computed from the current configuration. Please match the file ending according to the output_format.
output_format	Specification of output format (either 'graphic' for PNG or 'animation' for MP4).
animation_pace	Pace of the animation in seconds (positive numeric). This only has an effect if output_format == 'animation'.
freeze_animation	If TRUE then the animation will freeze at the last frame (logical).
language	Language used for title, legend, etc. in plots (either 'eng' for English or 'deu' for German).
keep_files	A flag indicating whether all files created in the process of obtaining the output file should be kept (logical). If false, all intermediate results are deleted, otherwise all are kept. Keeping these files could improve performance in further function calls.
states	Whether to crop/plot administration level of states (logical).
attach	Whether to temporarily merge the infile to an already existing one. (logical).
infile_attach	File to attach the infile to. When 'auto', a suitable file will be searched in out_dir. If attach is false, this will be ignored(character).
verbose	Whether to display progress messages (logical).

Details

You can pass a YAML config file and/or specify the arguments directly. Argument prioritization is done in the following way: Direct argument > config argument > default argument. Thus, if you pass a existing config file but also want to modify a specific argument you can do that easily.

Examples

```
## Create an example NetCDF file with a similar structure as used by CM
## SAF. The file is created with the ncdf4 package. Alternatively
## example data can be freely downloaded here: <https://wui.cmsaf.eu/>

library(ncdf4)

## create some (non-realistic) example data

lon <- seq(5, 15, 0.5)
lat <- seq(45, 55, 0.5)
time <- seq(as.Date("2010-01-01"), as.Date("2011-12-31"), "days")
origin <- as.Date("1983-01-01 00:00:00")
time <- as.numeric(difftime(time, origin, units = "hour"))
data <- array(250:350, dim = c(21, 21, 2 * 365))

## create example NetCDF
infile <- tempfile("input", fileext = ".nc")

x <- ncdim_def(name = "lon", units = "degrees_east", vals = lon)
y <- ncdim_def(name = "lat", units = "degrees_north", vals = lat)
t <- ncdim_def(name = "time", units = "hours since 1983-01-01 00:00:00",
  vals = time, unlim = TRUE)
var1 <- ncvar_def("SDU", "W m-2", list(x, y, t), -1, prec = "short")
vars <- list(var1)
ncnew <- nc_create(infile, vars)
ncvar_put(ncnew, var1, data)
ncatt_put(ncnew, "lon", "standard_name", "longitude", prec = "text")
ncatt_put(ncnew, "lat", "standard_name", "latitude", prec = "text")
nc_close(ncnew)

## this will save 'output.png' in temp directory
cmsafvis::fieldmean_plot(
  accumulate = TRUE,
  infile = infile,
  out_dir = tempdir(),
  climate_dir = tempdir(),
  climate_year_start = 2010,
  climate_year_end = 2011,
  end_date = "2010-01-05",
  outfile_name = "output",
  output_format = "graphic",
  keep_files = FALSE,
  verbose = FALSE
)
```

monitor_climate *A 'cmsaf' extension for creating various climate plots.*

Description

This plotting routine generates graphical output of the evolution of the given variable within the given time range and area. The intended application is for daily accumulated data, such as sunshine duration. Dependent on the output format a PNG or MP4 is created.

Usage

```
monitor_climate(  
  plot_type = "absolute_map",  
  config = NULL,  
  variable = NULL,  
  accumulate = FALSE,  
  infile = NULL,  
  temp_dir = tempdir(),  
  out_dir = getwd(),  
  climate_dir = NULL,  
  climate_year_start = 1983,  
  climate_year_end = 2018,  
  show_extreme_climate_years = NULL,  
  climatology_until_eoy = FALSE,  
  start_date = NULL,  
  end_date = NULL,  
  country_code = "S_A",  
  lon_min = NULL,  
  lon_max = NULL,  
  lat_min = NULL,  
  lat_max = NULL,  
  outfile_name = NULL,  
  output_format = "animation",  
  animation_pace = 0.07,  
  freeze_animation = FALSE,  
  min_value = NULL,  
  max_value = NULL,  
  nbreaks = NULL,  
  language = "eng",  
  keep_files = TRUE,  
  states = FALSE,  
  attach = FALSE,  
  infile_attach = "auto",  
  verbose = TRUE  
)
```

Arguments

plot_type	Specifies the type of the plot ('absolute_map', 'anomaly_map', 'climatology_map', 'fieldmean_plot', or 'fieldmean_and_anomaly_map').
config	Path to YAML config file (character). The config file does not have to specify all arguments. Each addressed argument has to be formatted according to the example config file: (#TODO: LINK EXAMPLE CONFIG FILE!).
variable	Name of variable in infile (NULL or character). If NULL then the first variable from the infile is taken.
accumulate	Whether the input file should be accumulated (logical).
infile	Path to NetCDF file (NULL or character). If NULL then it needs to be specified in the config file.
temp_dir	Path to temporary working directory (character).
out_dir	Path to output directory (character).
climate_dir	Path to directory in which climatology is computed or contained (NULL or character). If NULL then the temp_dir directory is taken.
climate_year_start	Start year of climatology (integer).
climate_year_end	End year of climatology (integer).
show_extreme_climate_years	Whether the minimum and maximum of the climate years should be titled in the fieldmean plot (NULL or logical). This is usually only of interest when plotting accumulated data. If the default NULL is chosen, then it will be set to the value of accumulate.
climatology_until_eoy	Plot the climatology and fieldmeans until the end of year (logical). Only affects fieldmean plots analyzed from January 1st.
start_date	Start date in format of 'YYYY-MM-DD' (NULL or character). If NULL then the first date of the infile is used.
end_date	End date in format of 'YYYY-MM-DD' (NULL or character). If NULL then the last date of the infile is used.
country_code	Either a country code in iso3c format or from the following: 'AFR' for Africa, 'EUR' for Europe, 'TOT' for the total disc, or 'S_A' for an arbitrary region selection (character). If a country is passed the data from within this country is extracted, else a rectangular box is visualized. Directly provided latitude and longitude ranges will be ignored in case of 'AFR', 'EUR' or 'TOT'.
lon_min	Longitude of lower left corner (NULL or numeric). If NULL then the smallest longitude of the infile is used.
lon_max	Longitude of upper right left corner (NULL or numeric). If NULL then the largest longitude of the infile is used.
lat_min	Latitude of lower left corner (NULL or numeric). If NULL then the smallest latitude of the infile is used.
lat_max	Latitude of upper right corner (NULL or numeric). If NULL then the largest latitude of the infile is used.

outfile_name	Filename of the PNG or MP4 outfile (NULL or character). If NULL then a name is computed from the current configuration. Please match the file ending according to the output_format.
output_format	Specification of output format (either 'graphic' for PNG or 'animation' for MP4).
animation_pace	Pace of the animation in seconds (positive numeric). This only has an effect if output_format == 'animation'.
freeze_animation	If TRUE then the animation will freeze at the last frame (logical).
min_value	Lower values than this are ignored (NULL or numeric). If NULL, no values are ignored.
max_value	Larger values than this are ignored (NULL or numeric). If NULL, no values are ignored.
nbreaks	Number of color breaks (NULL or positive integer). A value will be computed if NULL is passed.
language	Language used for title, legend, etc. in plots (either 'eng' for English or 'deu' for German).
keep_files	A flag indicating whether all files created in the process of obtaining the output file should be kept (logical). If false, all intermediate results are deleted, otherwise all are kept. Keeping these files could improve performance in further function calls.
states	Whether to crop/plot administration level of states (logical).
attach	Whether to temporarily merge the infile to an already existing one. (logical).
infile_attach	File to attach the infile to. When 'auto', a suitable file will be searched in out_dir. If attach is false, this will be ignored(character).
verbose	Whether to display progress messages (logical).

Details

You can pass a YAML config file and/or specify the arguments directly. Argument prioritization is done in the following way: Direct argument > config argument > default argument. Thus, if you pass a existing config file but also want to modify a specific argument you can do that easily.

Examples

```
## Create an example NetCDF file with a similar structure as used by CM
## SAF. The file is created with the ncdf4 package. Alternatively
## example data can be freely downloaded here: <https://wui.cmsaf.eu/>

library(ncdf4)

## create some (non-realistic) example data

lon <- seq(5, 15, 0.5)
lat <- seq(45, 55, 0.5)
time <- seq(as.Date("2010-01-01"), as.Date("2011-12-31"), "days")
origin <- as.Date("1983-01-01 00:00:00")
```

```

time <- as.numeric(difftime(time, origin, units = "hour"))
data <- array(250:350, dim = c(21, 21, 2 * 365))

## create example NetCDF
infile <- tempfile("input", fileext = ".nc")

x <- ncdim_def(name = "lon", units = "degrees_east", vals = lon)
y <- ncdim_def(name = "lat", units = "degrees_north", vals = lat)
t <- ncdim_def(name = "time", units = "hours since 1983-01-01 00:00:00",
  vals = time, unlim = TRUE)
var1 <- ncvar_def("SDU", "W m-2", list(x, y, t), -1, prec = "short")
vars <- list(var1)
ncnew <- nc_create(infile, vars)
ncvar_put(ncnew, var1, data)
ncatt_put(ncnew, "lon", "standard_name", "longitude", prec = "text")
ncatt_put(ncnew, "lat", "standard_name", "latitude", prec = "text")
nc_close(ncnew)

## this will save 'output.png' in temp directory
cmsafvis::monitor_climate(
  plot_type = "anomaly_map",
  accumulate = TRUE,
  infile = infile,
  out_dir = tempdir(),
  climate_dir = tempdir(),
  climate_year_start = 2010,
  climate_year_end = 2011,
  end_date = "2010-01-05",
  outfile_name = "output",
  output_format = "graphic",
  keep_files = FALSE,
  verbose = FALSE
)

```

quicklook

Create a quicklook of NetCDF data

Description

The function creates a plot of the variables in NetCDF file(s) specified in the config file. Only NetCDF files that conform to the **CMSAF naming convention** are supported.

Usage

```

quicklook(
  config,
  filelist,
  outpath = getwd(),
  jpeg_quality = 75,
  dpi = 72,

```

```

    iwidth = 720,
    logo = TRUE,
    copyright = TRUE,
    bluemarble = FALSE
  )

```

Arguments

config	filename of configuration file. This may include the directory (character).
filelist	list of NetCDF file to create plots from (character).
outpath	directory in which to save the output files. (character).
jpeg_quality	jpeg quality for the image in percent, see grDevices::jpeg()
dpi	resolution of the image in dots per inch, see grDevices::jpeg()
iwidth	width of the resulting image in pixels, see grDevices::jpeg()
logo	logical; should the cmsaf logo be added to the plot?
copyright	logical; should the copyright text be added to the plot?
bluemarble	logical; should the data be plotted onto a NASA bluemarble (only available for MSG/Seviri based data)? Due to data size this option is not available for the cmsafvis package on CRAN. Please have a look at our website https://www.cmsaf.eu/R_toolbox

Value

A jpeg file with the same name as the original NetCDF file.

Examples

```

## Create an example NetCDF file with a similar structure as used by CM
## SAF. The file is created with the ncdf4 package. Alternatively
## example data can be freely downloaded here: <https://wui.cmsaf.eu/>

library(ncdf4)

## create some (non-realistic) example data

lon <- seq(5, 15, 0.5)
lat <- seq(45, 55, 0.5)
time <- c(as.Date("2000-01-01"))
origin <- as.Date("1983-01-01 00:00:00")
time <- as.numeric(difftime(time, origin, units = "hour"))
data <- array(1:441, dim = c(21, 21, 1))

## create example NetCDF
infile1 <- file.path(tempdir(), "SISmm200001010000003231000101MA.nc")

x <- ncdim_def(name = "lon", units = "degrees_east", vals = lon)
y <- ncdim_def(name = "lat", units = "degrees_north", vals = lat)
t1 <- ncdim_def(name = "time", units = "hours since 1983-01-01 00:00:00",
  vals = time[1], unlim = TRUE)
var1 <- ncvar_def("SIS", "W m-2", list(x, y, t1), -1, prec = "short")

```



```

vars1 <- list(var1)
ncnew1 <- nc_create(infile1, vars1)
ncvar_put(ncnew1, var1, data[,1])
ncatt_put(ncnew1, "lon", "standard_name", "longitude", prec = "text")
ncatt_put(ncnew1, "lat", "standard_name", "latitude", prec = "text")
nc_close(ncnew1)

config_path <- system.file(
  "extdata",
  "quicklook_config.yml",
  package = "cmsafvis",
  mustWork = TRUE
)
## Create quicklooks to tempdir
quicklook(config_path, list.files(tempdir(), "SIS", full.names = TRUE), outpath = tempdir())

```

```
recalculateImageDimensions
```

Get rectangular image dimensions

Description

Given regional bounds image width and height are computed in order to display a region without distortion.

Usage

```

recalculateImageDimensions(
  visualizeVariables,
  lon_bounds,
  lat_bounds,
  image_def,
  ihsf
)

```

Arguments

visualizeVariables	A dataframe containing \$lon and \$lat values which will be bounded by lon/lat_bounds (data.frame).
lon_bounds	Array containing two values for longitude min and max (numeric).
lat_bounds	Array containing two values for latitude min and max (numeric).
image_def	Minimal image default size for width and height (numeric).
ihsf	Image height rescaling factor (numeric).

Examples

```

visualizeVariables <- list(
  plot_dim = 2,
  date.time = c(format(Sys.time(), format = "%Y-%m-%d %H:%M:%S")),
  data = array(1:4, dim = c(2,2)),
  min_lon = 6,
  max_lon = 9,
  min_lat = 49,
  max_lat = 52,
  lon = c(7, 8),
  lat = c(50, 51),
  vn = "SIS",
  varname = "Surface Downwelling Shortwave Radiation",
  copyrightText = "Example Copyright"
)

ihf <- 0.1
image_def <- 800
lon_bounds <- c(6, 9)
lat_bounds <- c(49, 52)

recalculateImageDimensions(visualizeVariables, lon_bounds, lat_bounds, image_def, ihf)

```

render_hist_plot *Creating a simple histogram.*

Description

This routine was implemented for creating histograms in the CM SAF R Toolbox.

Usage

```

render_hist_plot(
  dastat,
  shortDescription,
  grid_col,
  bordercolor,
  linesize,
  xlab
)

```

Arguments

dastat	Statistics given to hist plot (numeric).
shortDescription	A title will be generated using "Histogram of" + description (character).
grid_col	Color used for the grid.

bordercolor	Color used for borders.
linesize	Line width to be used (positive numeric).
xlab	Label for x axis (character).

Examples

```
dastat <- c(1,1,2)
shortDescription <- "Short description"
bordercolor <- "gray20"
grid_col <- "cornsilk2"
linesize <- 1.5
xlab <- "x label"

render_hist_plot(dastat, shortDescription, grid_col, bordercolor, linesize, xlab)
```

render_instat_plot *Creating a simple instat plot.*

Description

This function creates a simple r-instat plot.

Usage

```
render_instat_plot(
  co.data,
  shortDescription,
  grid_col,
  bordercolor,
  linesize,
  ylab
)
```

Arguments

co.data	Statistics given to hist plot (data.frame).
shortDescription	A title will be generated using "Comparison of" + description (character).
grid_col	Color used for the grid.
bordercolor	Color used for borders.
linesize	Line width to be used (positive numeric).
ylab	Label for y axis (character).

Examples

```
co.data <- list(
  lon_station = c(1,2,3),
  lat_station = c(10,20,30),
  data_station = c(1,2,3),
  data_sat = c(4,3,2)
)

render_instat_plot(
  co.data = co.data,
  shortDescription = "thing 1 vs thing 2",
  grid_col = "cornsilk2",
  bordercolor = "gray20",
  linesize = 1.2,
  ylab = "y label"
)
```

render_plot

Plotting routine designed for the CM SAF R Toolbox.

Description

This function renders a 2D image usually called by the CM SAF R Toolbox.

Usage

```
render_plot(
  plot_rinstat,
  outfile = NULL,
  fileExtension = ".png",
  visualizeVariables,
  visualizeDataTimestep,
  nc_path_visualize,
  visualizeDataMax,
  lon_bounds,
  lat_bounds,
  lon_loc_vec,
  lat_loc_vec,
  name_loc_vec,
  timestep,
  num_tick,
  num_rmin,
  num_rmax,
  num_brk,
  co.data,
  proj,
  xort,
```

```

    yort,
    rort,
    slider1,
    slider2,
    imagewidth,
    imageheight,
    location,
    int,
    text1,
    text2,
    text3,
    textsize,
    bordercolor,
    linesize,
    na.color,
    PAL,
    palettes,
    reverse,
    plot_grid,
    grid_col
)

```

Arguments

plot_rinstat	Whether to create an R-Instat plot (logical).
outfile	Name of the outfile (NULL or character). Should match the fileExtension. If NULL is passed a file is created in the R session temporary directory.
fileExtension	The file extension of the image (character). Has to be one of the following: 'png', 'jpg', 'tif', 'kml', 'pdf'.
visualizeVariables	A data frame containing all meta data for the plotting process (data.frame).
visualizeDataTimestep	The data to be plotted.
nc_path_visualize	The nc file path of which the plot is generated for.
visualizeDataMax	Maximal data for computing breaks.
lon_bounds	Array containing two values for longitude min and max (numeric).
lat_bounds	Array containing two values for latitude min and max (numeric).
lon_loc_vec	All longitude entries for points at (lat_loc_vec, lon_loc_vec) to be specified on the map (numeric).
lat_loc_vec	All latitude entries for points at (lat_loc_vec, lon_loc_vec) to be specified on the map (numeric).
name_loc_vec	Names for the points at (lat_loc_vec, lon_loc_vec) to be specified on the map (numeric).
timestep	The current timestep chosen.

num_tick	Number of ticks (numeric).
num_rmin	Color scale range minimum (numeric).
num_rmax	Color scale range maximum (numeric).
num_brk	Number of breaks (numeric).
co.data	Data to be plotted in R-Instat mode (data.frame).
proj	The chosen projection (either 'rect' for rectangular or 'ortho' for orthographic).
xort	Centering the globe at longitude xort (numeric). Only in orthographic mode.
yort	Centering the globe at latitude yort (numeric). Only in orthographic mode.
rort	Rotation of the globe (numeric). Only in orthographic mode.
slider1	Controlling the horizontal plot position as vector of two values min and max (numeric).
slider2	Controlling the vertical plot position as vector of two values min and max (numeric).
imagewidth	Width of the image (numeric).
imageheight	Height of the image (numeric).
location	Whether points specified by (lat_loc_vec, lon_loc_vec, name_loc_vec) should be added to the map (logical).
int	Whether interior country borders should be added (logical).
text1	Title text (character).
text2	Text to be passed to graphics::mtext (character).
text3	Text to be added to the legend (character).
textsize	Textsize to be used (cex).
bordercolor	Color used for borders.
linesize	Line width to be used (positive numeric).
na.color	The color to be used for NA values.
PAL	Color palette.
palettes	Color palettes to be used.
reverse	Whether to revert the color palette (logical).
plot_grid	Whether to plot a grid using color_grid_col (logical).
grid_col	Color used for the grid.

Examples

```
visualizeVariables <- list(
  plot_dim = 2,
  date.time = c(format(Sys.time(), format = "%Y-%m-%d %H:%M:%S")),
  data = array(1:4, dim = c(2,2)),
  min_lon = 6,
  max_lon = 9,
  min_lat = 49,
  max_lat = 52,
```

```
lon = c(7, 8),
lat = c(50, 51),
vn = "VAR",
varname = "Example var name",
copyrightText = "Example Copyright",
unit = "example unit"
)

palettes <- data.frame(
  type = "qual",
  h1 = 0,
  h2 = 360,
  c1 = 35,
  c2 = NA,
  l1 = 85,
  l2 = NA,
  p1 = NA,
  p2 = NA,
  cmax = NA,
  fixup = 1
)

rownames(palettes) = "Pastel 1"

render_plot(
  plot_rinstat = FALSE,
  visualizeVariables = visualizeVariables,
  visualizeDataTimestep = array(1:4, dim = c(2,2)),
  nc_path_visualize = NULL,
  visualizeDataMax = 4,
  lon_bounds = c(6, 9),
  lat_bounds = c(49, 52),
  timestep = c(format(Sys.time(), format = "%Y-%m-%d")),
  num_tick = 4,
  num_rmin = 1,
  num_rmax = 4,
  num_brk = 2,
  co.data = NULL,
  proj = "rect",
  xort = 0,
  yort = 0,
  rort = 0,
  slider1 = c(6, 9),
  slider2 = c(49, 52),
  imagewidth = 800,
  imageheight = 800,
  location = FALSE,
  int = FALSE,
  text1 = "Text 1",
  text2 = "Text 2",
  text3 = "Text 3",
  textsize = 1.2,
  bordercolor = "gray20",
```

```
linesize = 1.5,  
na.color = "gray80",  
PAL = "Pastel 1",  
palettes = palettes,  
reverse = FALSE,  
plot_grid = TRUE,  
grid_col = "cornsilk2"  
)
```

render_plot_1d

Plotting routine designed for the CM SAF R Toolbox.

Description

This function renders a 1D plot of data at one single lon / lat point.

Usage

```
render_plot_1d(  
  outfile = NULL,  
  fileExtension = ".png",  
  visualizeVariables,  
  ticknumber,  
  dateformat,  
  analyze_timeseries,  
  addTrend,  
  sliderx,  
  slidery,  
  checkGroup_type,  
  imagewidth,  
  imageheight,  
  text1_1d,  
  text2_1d,  
  textsize,  
  linesize,  
  col  
)
```

Arguments

outfile	Name of the outfile (NULL or character). Should match the fileExtension. If NULL is passed a file is created in the R session temporary directory.
fileExtension	The file extension of the image (character). Has to be one of the following: 'png', 'jpg', 'tif', 'kml', 'pdf'.
visualizeVariables	A data frame containing all meta data for the plotting process (data.frame).

ticknumber	Number of ticks (numeric).
dateformat	Date format for constructing a date label.
analyze_timeseries	Whether or not to analyze the timeseries of the given point (logical).
addTrend	Whether to add a trend line (logical).
sliderx	Limiting the time series with a two valued vector for min and max (numeric).
slidery	Limiting the y axis with a two valued vector for min and max (numeric).
checkGroup_type	An integer between 1 and 5 indicating group type (numeric). 1 for Line, 2 for Points, 3 for Line and Points, 4 for steps, 5 for histogram.
imagewidth	Width of the image (numeric).
imageheight	Height of the image (numeric).
text1_1d	Title text (character).
text2_1d	Text to be passed to graphics::mtext (character).
textsize	Textsize to be used (cex).
linesize	Line width to be used (positive numeric).
col	A color chosen via colourpicker::colourInput.

Examples

```
visualizeVariables <- list(
  plot_dim = 1,
  date.time = seq.Date(as.Date("2020-01-01"),as.Date("2020-01-31"), by = 1),
  data = array(1:31, dim = c(1,1,31)),
  x_range = 31,
  y_label = "Y LABEL",
  lon = 7,
  lat = 50,
  vn = "VAR",
  varname = "Example var name",
  copyrightText = "Example Copyright",
  unit = "example unit",
  fitted = array(1:31, dim = c(1,1,31))
)
```

```
render_plot_1d(
  visualizeVariables = visualizeVariables,
  ticknumber = 6,
  dateformat = "2",
  analyze_timeseries = FALSE,
  addTrend = FALSE,
  sliderx = c(1, 31),
  slidery = c(1, 31),
  checkGroup_type = "1",
  imagewidth = 800,
  imageheight = 533,
  text1_1d = "TEXT 1",
```

```
text2_1d = "TEXT 2",  
textsize = 1.2,  
linesize = 1.5,  
col = "royalblue4"  
)
```

render_preview_plot *Creates a preview plot of a selected area*

Description

This function creates a simple preview plot via `maps::map` of a given region.

Usage

```
render_preview_plot(spatial_lon_range, spatial_lat_range, lonRange, latRange)
```

Arguments

`spatial_lon_range` Array containing two values for longitude min and max (numeric).

`spatial_lat_range` Array containing two values for latitude min and max (numeric).

`lonRange` Array containing two values for longitude min and max (numeric).

`latRange` Array containing two values for latitude min and max (numeric).

Examples

```
render_preview_plot(  
  spatial_lon_range = c(-20, 55),  
  spatial_lat_range = c(-40, 40),  
  lonRange = c(-5, 5),  
  latRange = c(0, 15)  
)
```

render_region_plot	<i>Function to create a plot of a selected country.</i>
--------------------	---

Description

This function generates a plot of within a certain region (e.g. a country).

Usage

```
render_region_plot(  
    infile,  
    outfile = NULL,  
    fileExtension = ".png",  
    visualizeVariables,  
    visualizeDataMax,  
    lon_bounds,  
    lat_bounds,  
    lon_loc_vec,  
    lat_loc_vec,  
    name_loc_vec,  
    division,  
    selectedRegion,  
    region_data,  
    timestep,  
    num_tick,  
    num_rmin,  
    num_rmax,  
    location,  
    text1,  
    text2,  
    text3,  
    PAL,  
    palettes,  
    num_brk,  
    reverse,  
    textsize,  
    bordercolor,  
    plot_grid,  
    grid_col,  
    image_def,  
    ihsf  
)
```

Arguments

infile	The nc file to be visualized.
--------	-------------------------------

outfile	Name of the outfile (NULL or character). Should match the fileExtension. If NULL is passed a file is created in the R session temporary directory.
fileExtension	The file extension of the image (character). Has to be one of the following: 'png', 'jpg', 'tif', 'kml', 'pdf'.
visualizeVariables	A data frame containing all meta data for the plotting process (data.frame).
visualizeDataMax	Maximal data for computing breaks.
lon_bounds	Array containing two values for longitude min and max (numeric).
lat_bounds	Array containing two values for latitude min and max (numeric).
lon_loc_vec	All longitude entries for points at (lat_loc_vec, lon_loc_vec) to be specified on the map (numeric).
lat_loc_vec	All latitude entries for points at (lat_loc_vec, lon_loc_vec) to be specified on the map (numeric).
name_loc_vec	Names for the points at (lat_loc_vec, lon_loc_vec) to be specified on the map (numeric).
division	Division to contain region (either 'COUNTRY' or something coherent to region_data).
selectedRegion	The region to be cropped according to division. If division == "COUNTRY", the country's 3-character ISO code should be used. Otherwise it has to fit to the passed region data.
region_data	If Division is not 'COUNTRY' then region_data has to contain spatial data of the given division.
timestep	Timestep to visualize at (character).
num_tick	Number of ticks (numeric).
num_rmin	Color scale range minimum (numeric).
num_rmax	Color scale range maximum (numeric).
location	Whether points specified by (lat_loc_vec, lon_loc_vec, name_loc_vec) should be added to the map (logical).
text1	Title text (character).
text2	Text to be passed to graphics::mtext (character).
text3	Text to be added to the legend (character).
PAL	Color palette.
palettes	Color palettes to be used.
num_brk	Number of breaks (numeric).
reverse	Whether to revert the color palette (logical).
textsize	Textsize to be used (cex).
bordercolor	Color used for borders.
plot_grid	Whether to plot a grid using color grid_col (logical).
grid_col	Color used for the grid.
image_def	Default size (positive numeric).
ihsf	Image height scaling factor (positive numeric).

Examples

```

library(ncdf4)

## create some (non-realistic) example data
lon <- seq(-7.5, 7.5, 0.5)
lat <- seq(0, 15, 0.5)
time <- seq(as.Date("2000-01-01"), as.Date("2000-12-31"), "month")
origin <- as.Date("1983-01-01 00:00:00")
time <- as.numeric(difftime(time, origin, units = "hour"))
data <- array(1:31*31*12, dim = c(31, 31, 12))

x <- ncdim_def(name = "lon", units = "degrees_east", vals = lon)
y <- ncdim_def(name = "lat", units = "degrees_north", vals = lat)
t <- ncdim_def(name = "time", units = "hours since 1983-01-01 00:00:00",
              vals = time, unlim = TRUE)
var1 <- ncvar_def("VAR", "EX UNIT", list(x, y, t), -1, prec = "short")
vars <- list(var1)
ncfile <- tempfile(fileext = ".nc")
ncnew <- nc_create(ncfile, vars)
ncvar_put(ncnew, var1, data)
ncatt_put(ncnew, "lon", "standard_name", "longitude", prec = "text")
ncatt_put(ncnew, "lat", "standard_name", "latitude", prec = "text")
nc_close(ncnew)

visualizeVariables <- list(
  plot_dim = 2,
  date.time = seq(as.Date("2000-01-01"), as.Date("2010-12-31"), "month"),
  data = data,
  min_lon = -7.5,
  max_lon = 7.5,
  min_lat = 0,
  max_lat = 15,
  lon = lon,
  lat = lat,
  vn = "VAR",
  varname = "Example var name",
  copyrightText = "Example Copyright",
  unit = "example unit"
)

palettes <- data.frame(
  type = "qual",
  h1 = 0,
  h2 = 360,
  c1 = 35,
  c2 = NA,
  l1 = 85,
  l2 = NA,
  p1 = NA,
  p2 = NA,
  cmax = NA,
  fixup = 1
)

```

```
)  
  
rownames(palettes) = "Pastel 1"  
  
render_region_plot(  
  infile = ncfile,  
  visualizeVariables = visualizeVariables,  
  visualizeDataMax = 31*31,  
  lon_bounds = c(-7.5, 7.5),  
  lat_bounds = c(0, 15),  
  division = "COUNTRY",  
  selectedRegion = "GHA",  
  timestep = as.Date("2000-01-01"),  
  num_tick = 10,  
  num_rmin = 0,  
  num_rmax = 31*31*12,  
  num_brk = 10,  
  location = FALSE,  
  text1 = "Text 1",  
  text2 = "Text 2",  
  text3 = "Text 3",  
  textsize = 1.2,  
  bordercolor = "gray20",  
  PAL = "Pastel 1",  
  palettes = palettes,  
  reverse = FALSE,  
  plot_grid = TRUE,  
  grid_col = "cornsilk2",  
  image_def = 800,  
  ihsf = 0.1  
)
```

Index

- * **datagen**
 - cmsafvis, [12](#)
- * **manip**
 - cmsafvis, [12](#)
- * **package**
 - cmsafvis, [12](#)
- * **spatial**
 - cmsafvis, [12](#)
- * **ts**
 - cmsafvis, [12](#)
- * **univar**
 - cmsafvis, [12](#)

[absolute_map](#), [2](#), [12](#)
[anomaly_map](#), [5](#), [12](#)

[climatology_map](#), [8](#), [12](#)
[cmsafvis](#), [12](#)

[fieldmean_and_anomaly_map](#), [12](#), [13](#)
[fieldmean_plot](#), [12](#), [16](#)

[grDevices::jpeg\(\)](#), [24](#)

[monitor_climate](#), [20](#)

[ncdf4](#) package, [12](#)

[quicklook](#), [23](#)

[recalculateImageDimensions](#), [25](#)
[render_hist_plot](#), [26](#)
[render_instat_plot](#), [27](#)
[render_plot](#), [28](#)
[render_plot_1d](#), [32](#)
[render_preview_plot](#), [34](#)
[render_region_plot](#), [35](#)