

Package ‘prism’

November 16, 2015

Type Package

Title Access Data from the Oregon State Prism Climate Project

Description Allows users to access the Oregon State Prism climate data. Using the web service API data can easily downloaded in bulk and loaded into R for spatial analysis. Some user friendly visualizations are also provided.

URL <http://github.com/ropensci/prism>

BugReports <http://github.com/ropensci/prism/issues>

Version 0.0.7

Date 2015-11-11

License MIT + file LICENSE

Imports ggplot2, raster, httr

LazyLoad yes

LazyData yes

VignetteBuilder knitr

Suggests testthat, knitr

Author Hart Edmund [aut, cre],
Kendon Bell [aut]

Maintainer Hart Edmund <Edmund.m.hart@gmail.com>

NeedsCompilation no

Repository CRAN

Date/Publication 2015-11-16 08:20:18

R topics documented:

get_prism_annual	2
get_prism_dailys	3
get_prism_monthlys	4
get_prism_normals	5
ls_prism_data	6

mon_to_string	6
path_check	7
prism_check	7
prism_image	8
prism_md	8
prism_slice	9
prism_stack	9
prism_webservice	10
process_zip	11
pr_parse	11

Index 13

get_prism_annual	<i>Download annual daily averages</i>
------------------	---------------------------------------

Description

Download annual daily average data from the prism project at 4km grid cell resolution for precipitation, mean, min and max temperature

Usage

```
get_prism_annual(type, years = NULL, keepZip = TRUE)
```

Arguments

type	The type of data to download, must be "ppt", "tmean", "tmin", "tmax", or "all", which downloads "ppt", "tmin", and "tmax". Note that tmean == mean(tmin, tmax).
years	a valid numeric year, or vector of years, to download data for. If no month is specified, year averages for that year will be downloaded
keepZip	if true, leave the downloaded zip files in your 'prism.path', if FALSE, they will be deleted

Details

Data is available from 1891 until 2014, however you have to download all data for years prior to 1981. Therefore if you enter a vector of years that bounds 1981, you will automatically download all data for all years in the vector. If the "all" parameter is set to TRUE, it will override any months entered and download all data. Data will be downloaded for all months in all the years in the vectors supplied. You must make sure that you have set up a valid download directory. This must be set as options(prism.path = "YOURPATH")

Examples

```
## Not run:  
### Get all the data for January from 1990 to 2000  
# get_prism_annual(type="tmean", year = 1990:2000, keepZip=FALSE)  
  
## End(Not run)
```

get_prism_dailys	<i>Download daily prism data</i>
------------------	----------------------------------

Description

Download daily data from the prism project at 4km grid cell resolution for precipitation, mean, min and max temperature

Usage

```
get_prism_dailys(type, minDate = NULL, maxDate = NULL, dates = NULL,  
  keepZip = TRUE)
```

Arguments

type	The type of data to download, must be "ppt", "tmean", "tmin", "tmax", which downloads "ppt", "tmin", and "tmax". Note that tmean == mean(tmin, tmax).
minDate	a valid iso-8601 (e.g. YYYY-MM-DD) date to start to download data.
maxDate	a valid iso-8601 (e.g. YYYY-MM-DD) date to end downloading data.
dates	a vector of iso-8601 formatted dates to download data for, can also be a single date.
keepZip	if true, leave the downloaded zip files in your 'prism.path', if FALSE, they will be deleted

Details

Dates must be in the proper format or downloading will not work properly, you can either enter a date range via minDate and maxDate, or a vector of dates, but not both. You must make sure that you have set up a valid download directory. This must be set as options(prism.path = "YOUR-PATH") # get_prism_dailys(type="tmean", minDate = "2013-06-01", maxDate = "2013-06-14", keepZip=FALSE)

get_prism_monthlys *Download monthly prism data*

Description

Download monthly data from the prism project at 4km grid cell resolution for precipitation, mean, min, and max temperature

Usage

```
get_prism_monthlys(type, years = NULL, mon = NULL, keepZip = TRUE)
```

Arguments

type	The type of data to download, must be "ppt", "tmean", "tmin", "tmax", which downloads "ppt", "tmin", and "tmax". Note that tmean == mean(tmin, tmax).
years	a valid numeric years, or vector of years, to download data for. If no month is specified, years averages for that years will be downloaded
mon	a valid numeric month, or vector of mon.
keepZip	if true, leave the downloaded zip files in your 'prism.path', if FALSE, they will be deleted

Details

Data is available from 1891 until 2014, however you have to download all data for years prior to 1981. Therefore if you enter a vector of years that bounds 1981, you will automatically download all data for all years in the vector. If the "all" parameter is set to TRUE, it will override any mon entered and download all data. Data will be downloaded for all mon in all the years in the vectors supplied. You must make sure that you have set up a valid download directory. This must be set as options(prism.path = "YOURPATH")

Examples

```
## Not run:  
### Get all the data for January from 1990 to 2000  
# get_prism_monthlys(type="tmean", years = 1990:2000, mon = 1, keepZip=FALSE)  
  
## End(Not run)
```

get_prism_normals *Download data for 30 year normals of climate variables*

Description

Download data from the prism project for 30 year normals at 4km or 800m grid cell resolution for precipitation, mean, min and max temperature

Usage

```
get_prism_normals(type, resolution, mon = NULL, annual = FALSE,  
  keepZip = TRUE)
```

Arguments

type	The type of data to download, must be "ppt", "tmean", "tmin", "tmax", which downloads "ppt", "tmin", and "tmax". Note that tmean == mean(tmin, tmax).
resolution	The spatial resolution of the data, must be either "4km" or "800m".
mon	a numeric value for month, can be a numeric vector of months.
annual	if TRUE download annual data
keepZip	if TRUE, leave the downloaded zip files in your 'prism.path', if FALSE, they will be deleted

Details

You must make sure that you have set up a valid download directory. This must be set as options(prism.path = "YOURPATH")

Examples

```
## Not run:  
### Get 30 year normal values for rainfall  
# get_prism_normals(type="ppt",resolution = "4km",mon = 1, keepZip=FALSE)  
  
## End(Not run)
```

ls_prism_data *List available datasets*

Description

List the available data sets to load that have already been downloaded.

Usage

```
ls_prism_data(absPath = FALSE, name = FALSE)
```

Arguments

absPath TRUE if you want to return the absolute path.
name TRUE if you want file names and titles of data products.

Value

a data frame of downloaded datasets

Examples

```
## Not run:  
### Just get file names, used in many other prism* fxn  
# get_prism_dailys(type="tmean", minDate = "2013-06-01", maxDate = "2013-06-14", keepZip=FALSE)  
# ls_prism_data()  
  
### Get absolute path values for use with other data  
# ls_prism_data(absPath = TRUE)  
  
### See prism files you have with title of data product  
# ls_prism_data(name=TRUE)  
  
## End(Not run)
```

mon_to_string *helper function for handling months*

Description

Handle numeric month to string conversions

Usage

```
mon_to_string(month)
```

Arguments

month a numeric vector of months (month must be > 0 and <= 12)

Value

a character vector (same length as month) with 2 char month strings.

Examples

```
## Not run:  
# mon_to_string(month = c(1, 3, 2))  
# mon_to_string(month = 12)  
  
## End(Not run)
```

path_check *handle existing directory*

Description

create new directory for user if they don't have one to store prism files

Usage

```
path_check()
```

prism_check *Helper function to check if files already exist*

Description

check if files exist

Usage

```
prism_check(prismfiles)
```

Arguments

prismfiles a list of full paths for prism files

Value

a character vector of file names that already exist

prism_image	<i>Quick image plot</i>
-------------	-------------------------

Description

Quickly make an image plot of a data set.

Usage

```
prism_image(prismfile, col = "heat")
```

Arguments

prismfile	the name of a file to be plotted, this is most easily gotten through <code>ls_prism_data()</code>
col	the color pattern to use. The default is heat, the other valid option is "redblue"

Details

This is meant for rapid vizualization, but more detailed plots will require other methods

Examples

```
## Not run:
# get_prism_dailys(type="tmean", minDate = "2013-06-01", maxDate = "2013-06-14", keepZip=FALSE)
# prism_image(ls_prism_data()[1])

## End(Not run)
```

prism_md	<i>Extract select prism metadata</i>
----------	--------------------------------------

Description

used to extract some prism metadata used in other fxns

Usage

```
prism_md(f, returnDate = FALSE)
```

Arguments

f	a simple directory name
returnDate	TRUE or FALSE? If TRUE, an ISO date is returned. By default years will come back with YYYY-01-01 and months as YYYY-MM-01

Value

a character vector of metadata.

prism_slice	<i>Plot a slice of a raster stack</i>
-------------	---------------------------------------

Description

This function will plot a slice of data at a single point location from a list of prism files

Usage

```
prism_slice(location, prismfile)
```

Arguments

location	a vector of a single location in the form of long,lat
prismfile	a vector of output from <code>ls_prism_data()[,1]</code> giving a list of prism files to extract data from and plot

Details

the list of prism files should be from a continuous data set. Otherwise the plot will look erratic and incorrect.

Value

a ggplot2 plot of the requested slice

Examples

```
## Not run:  
### Assumes you have a clean prism directory  
# get_prism_dailys(type="tmean", minDate = "2013-06-01", maxDate = "2013-06-14", keepZip=FALSE)  
# p <- prism_slice(c(-73.2119,44.4758),ls_prism_data())  
# print(p)  
  
## End(Not run)
```

prism_stack	<i>Stack Prism files</i>
-------------	--------------------------

Description

Create a stack of prism files

Usage

```
prism_stack(prismfile)
```

Arguments

prismfile a vector of file names returned by ls_prism_data()

Examples

```
## Not run:
#get_prism_dailys(type="tmean", minDate = "2013-06-01", maxDate = "2013-06-14", keepZip=FALSE)
#mystack <- prism_stack(ls_prism_data()[1:14])

## End(Not run)
```

prism_webservice *Download PRISM via webservice*

Description

This is the workhorse function that will access the web service to download files

Usage

```
prism_webservice(uri, keepZip = FALSE, returnName = FALSE)
```

Arguments

uri a valid PRISM webservice URI

keepZip TRUE or FALSE, keep zip files once they have been unzipped

returnName TRUE or FALSE, if TRUE the name of the file that was downloaded is returned

Examples

```
## Not run:
### Get all the data for January from 1990 to 2000
# get_prism_annual(type="tmean", year = 1990:2000, keepZip=FALSE)

## End(Not run)
```

process_zip	<i>Process pre 1980 files</i>
-------------	-------------------------------

Description

Files that come prior to 1980 come in one huge zip. This will cause them to mimic all post 1980 downloads

Usage

```
process_zip(pfile, name)
```

Arguments

pfile	the name of the file, should include "all", that is unzipped
name	a vector of names of files that you want to save.

Details

This should match all other files post 1980

Examples

```
## Not run:
# process_zip('PRISM_tmean_stable_4kmM2_1980_all_bil', 'PRISM_tmean_stable_4kmM2_198001_bil')
# process_zip('PRISM_tmean_stable_4kmM2_1980_all_bil',
# c('PRISM_tmean_stable_4kmM2_198001_bil', 'PRISM_tmean_stable_4kmM2_198002_bil'))

## End(Not run)
```

pr_parse	<i>name parse</i>
----------	-------------------

Description

parse the directory name into relevant metadata

Usage

```
pr_parse(p, returnDate = FALSE)
```

Arguments

p	a prism file directory
returnDate	TRUE or FALSE? If TRUE, an ISO date is returned. By default years will come back with YYYY-01-01 and months as YYYY-MM-01

Value

a properly parsed string of human readable names

Index

`get_prism_annual`, 2
`get_prism_dailys`, 3
`get_prism_monthlys`, 4
`get_prism_normals`, 5

`ls_prism_data`, 6

`mon_to_string`, 6

`path_check`, 7
`pr_parse`, 11
`prism_check`, 7
`prism_image`, 8
`prism_md`, 8
`prism_slice`, 9
`prism_stack`, 9
`prism_webservice`, 10
`process_zip`, 11