

Package ‘smapr’

October 14, 2022

Type Package

Title Acquisition and Processing of NASA Soil Moisture Active-Passive (SMAP) Data

Version 0.2.1

Depends R (>= 3.2.5)

biocViews

Imports httr (>= 1.1.0), rappdirs (>= 0.3.1), raster (>= 2.5), rhdf5 (>= 2.14), rvest, xml2

Maintainer Maxwell Joseph <maxwell1.b.joseph@colorado.edu>

Description Facilitates programmatic access to NASA Soil Moisture Active Passive (SMAP) data with R. It includes functions to search for, acquire, and extract SMAP data.

License GPL-3

LazyData TRUE

RoxygenNote 6.1.1

Suggests knitr, rgdal, rmarkdown, roxygen2, sp, testthat, utils, covr

VignetteBuilder knitr

URL <https://github.com/ropensci/smapr>

BugReports <https://github.com/ropensci/smapr/issues>

Encoding UTF-8

NeedsCompilation no

Author Maxwell Joseph [aut, cre],
Matthew Oakley [aut],
Zachary Schira [aut]

Repository CRAN

Date/Publication 2019-04-22 04:40:02 UTC

R topics documented:

smapr-package	2
download_smap	2
extract_smap	3
find_smap	4
list_smap	5
set_smap_credentials	6

Index	7
--------------	----------

smapr-package	<i>smapr: A package for acquisition and processing of NASA SMAP data.</i>
---------------	---

Description

The smapr package provides a means to discover, acquire, and process NASA Soil Moisture Active Passive (SMAP) data.

Author(s)

Max Joseph <maxwell.b.joseph@colorado.edu>

download_smap	<i>Download SMAP data</i>
---------------	---------------------------

Description

This function downloads SMAP data in HDF5 format.

Usage

```
download_smap(files, directory = NULL, overwrite = TRUE,
              verbose = TRUE)
```

Arguments

files	A data.frame produced by find_smap() that specifies data files to download.
directory	A local directory path in which to save data, specified as a character string. If left as NULL, data are stored in a user's cache directory.
overwrite	TRUE or FALSE: should existing data files be overwritten?
verbose	TRUE or FALSE: should messages be printed to indicate that files are being downloaded?

Details

This function requires a username and password from NASA's Earthdata portal. If you have an Earthdata username and password, pass them in using the `set_smap_credentials()` function.

If you do not yet have a username and password, register for one here: <https://urs.earthdata.nasa.gov/>

Value

Returns a `data.frame` that appends a column called `local_dir` to the input data frame, which consists of a character vector specifying the local directory containing the downloaded files.

Examples

```
## Not run:
files <- find_smap(id = "SPL4SMGP", dates = "2015-03-31", version = 4)
# files[1, ] refers to the first available data file
downloads <- download_smap(files[1, ])

## End(Not run)
```

extract_smap	<i>Extracts contents of SMAP data</i>
--------------	---------------------------------------

Description

Extracts datasets from SMAP data files.

Usage

```
extract_smap(data, name, in_memory = FALSE)
```

Arguments

<code>data</code>	A data frame produced by <code>download_smap()</code> that specifies input files from which to extract data.
<code>name</code>	The path in the HDF5 file pointing to data to extract.
<code>in_memory</code>	Logical. Should the result be stored in memory? If not, then raster objects are stored on disk in the cache directory. By default the result is stored on disk.

Details

The arguments `group` and `dataset` must refer specifically the group and name within group for the input file, such as can be obtained with `list_smap()`. This function will extract that particular dataset, returning a Raster object.

Value

Returns a RasterStack object.

Examples

```
## Not run:
files <- find_smap(id = "SPL4SMGP", dates = "2015-03-31", version = 4)
downloads <- download_smap(files[1, ])
sm_raster <- extract_smap(downloads, name = '/Geophysical_Data/sm_surface')

## End(Not run)
```

find_smap

Find SMAP data

Description

This function searches for SMAP data on a specific date, returning a data.frame describing available data.

Usage

```
find_smap(id, dates, version)
```

Arguments

id	A character string that refers to a specific SMAP dataset, e.g., "SPL4SMGP" for SMAP L4 Global 3-hourly 9 km Surface and Rootzone Soil Moisture Geophysical Data. See "Details" for a list of supported data types and their associated id codes.
dates	An object of class Date or a character string formatted as To search for one specific date, this can be a Date object of length one. To search over a time interval, it can be a multi-element object of class Date such as produced by seq.Date.
version	Which data version would you like to search for? Version information for each data product can be found at https://nsidc.org/data/smap/data_versions

Details

There are many SMAP data products that can be accessed with this function. Currently, smapr supports level 3 and level 4 data products, each of which has an associated Data Set ID which is specified by the id argument, described at <https://nsidc.org/data/smap/smap-data.html> and summarized below:

SPL2SMAP_S SMAP/Sentinel-1 Radiometer/Radar Soil Moisture

SPL3FTA Radar Northern Hemisphere Daily Freeze/Thaw State

SPL3SMA Radar Global Daily Soil Moisture

SPL3SMP Radiometer Global Soil Moisture

SPL3SMAP Radar/Radiometer Global Soil Moisture

SPL4SMAU Surface/Rootzone Soil Moisture Analysis Update
SPL4SMGP Surface/Rootzone Soil Moisture Geophysical Data
SPL4SMLM Surface/Rootzone Soil Moisture Land Model Constants
SPL4CMDL Carbon Net Ecosystem Exchange

This function requires a username and password from NASA's Earthdata portal. If you have an Earthdata username and password, pass them in using the `set_smap_credentials()` function.

If you do not yet have a username and password, register for one here: <https://urs.earthdata.nasa.gov/>

Value

A data.frame with the names of the data files, the remote directory, and the date.

Examples

```
## Not run:
# looking for data on one day:
find_smap(id = "SPL4SMGP", dates = "2015-03-31", version = 4)

# searching across a date range
start_date <- as.Date("2015-03-31")
end_date <- as.Date("2015-04-02")
date_sequence <- seq(start_date, end_date, by = 1)
find_smap(id = "SPL4SMGP", dates = date_sequence, version = 4)

## End(Not run)
```

list_smap	<i>Lists the contents of SMAP data files</i>
-----------	--

Description

This function returns a list of the contents of SMAP data files.

Usage

```
list_smap(files, all = FALSE)
```

Arguments

`files` A data.frame produced by `download_smap()` that specifies input data files.
`all` If TRUE a longer, more detailed list of information on each entry is provided.

Value

Returns a list of data.frame objects that list the contents of each data file in files.

Examples

```
## Not run:
files <- find_smmap(id = "SPL4SMGP", dates = "2015-03-31", version = 4)
files <- download_smmap(files[1, ])
list_smmap(files)
list_smmap(files, all = TRUE)

## End(Not run)
```

set_smmap_credentials *Set credentials for NASA's Earthdata portal*

Description

To use smmap, users need to provide NASA Earthdata portal credentials. This function allows users to interactively set these credentials via the user's Earthdata username and password.

Usage

```
set_smmap_credentials(username, password, save = TRUE,
  overwrite = FALSE)
```

Arguments

username	A character string of your Earthdata portal username
password	A character string of your Earthdata portal password
save	Logical: whether to save your credentials to your .Renviron file (e.g., ~/.Renviron). Previous Earthdata credentials will not be overwritten unless <code>overwrite = TRUE</code> .
overwrite	Logical: whether to overwrite previous Earthdata credentials in your .Renviron file (only applies when <code>save = TRUE</code>)

Details

If you do not yet have a username and password, register for one here: <https://urs.earthdata.nasa.gov/>

A warning: do not commit your username and password to a public repository! This function is meant to be used interactively, and not embedded within a script that you would share.

Value

A data.frame with the names of the data files, the remote directory, and the date.

Examples

```
## Not run:
set_smmap_credentials('myusername', 'mypassword')

## End(Not run)
```

Index

* **package**

smapr-package, [2](#)

download_smap, [2](#)

extract_smap, [3](#)

find_smap, [4](#)

list_smap, [5](#)

set_smap_credentials, [6](#)

set_smap_credentials(), [3](#), [5](#)

smapr-package, [2](#)