Package 'mod09nrt'

July 24, 2016

Type Package

Title Extraction of Bands from MODIS Surface Reflectance Product MOD09 NRT

Version 0.14

Author Rishabh Gupta <rishabh.uk@gmail.com>, Nicholas J. Matzke, Dept. of Integrative Biology, U.C. Berkeley

Maintainer Rishabh Gupta <rishabh.uk@gmail.com>

Description Package for processing downloaded MODIS Surface reflectance Product HDF files. Specifically, MOD09 surface reflectance product files, and the associated MOD03 geolocation files (for MODIS-TERRA). The package will be most effective if the user installs MRTSwath (MODIS Reprojection Tool for swath products; <https://lpdaac.usgs.gov/tools/modis_reprojection_tool_swath>, and adds the directory with the MRTSwath executable to the default R PATH by editing ~/.Rprofile.

SystemRequirements MRTSwath

License GPL (>= 2)

LazyData TRUE

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2016-07-24 13:55:50

R topics documented:

adf 2
check_for_matching_geolocation_files_mod09nrt 2
extract_fn_from_path 4
run_swath2grid_mod09nrt 5
slashslash 7
write_MRTSwath_param_file_mod09nrt 8

Index **10**

adf *Convert to data.frame, without factors*

Description

Shortcut for: `as.data.frame(x, row.names=NULL, stringsAsFactors=FALSE)`

Usage

```
adf(x)
```

Arguments

x	matrix or other object transformable to data.frame

Details

This function is useful for dealing with errors due to automatic conversion of some columns to factors. Another solution may be to prepend `options(stringsAsFactors = FALSE)` at the start of one's script, to turn off all default stringsAsFactors silliness.

Value

data.frame

Author(s)

Nicholas J. Matzke <matzke@berkeley.edu>

Examples

```
x = matrix(c(1,2,3,4,5,6), nrow=3, ncol=2)
adf(x)
```

check_for_matching_geolocation_files_mod09nrt *Checks that every MODIS surface reflectance project HDF has a matching MOD03 file*

Description

Each MOD09 surface reflectance product file requires a corresponding MOD03 geolocation file to be successfully processed with the MRTSwath tool.

Usage

```
check_for_matching_geolocation_files_mod09nrt(moddir = getwd(),
  modtxt = "MOD09", geoloctxt = "MOD03", return_geoloc = FALSE,
  return_product = FALSE)
```

Arguments

moddir	the string describing the directory containing the MOD09 and MOD03 files; both must be in the same directory. Default: getwd(), which gives the present working directory.
modtxt	the text string indicating which HDF files are the MODIS surface reflectance product (or hypothetically, other product). Default: MOD09 (MODIS surface reflectance product)
geoloctxt	the text string indicating which HDF files are the MODIS geolocation files (or hypothetically, another set of files). Default: MOD03
return_geoloc	if TRUE, return the list of unmatched geolocation files (e.g. MOD03)
return_product	if TRUE, return the list of unmatched product files (e.g. MOD09)

Details

MRTSwath is the MRT (MODIS Reprojection Tool) for the MODIS

E.g. this surface reflectance file:

MOD09.A2016204.0500.005.2016204062219.NRT.hdf

...goes with this corresponding geolocation file:

MOD03.A2016204.0500.005.2016204060250.NRT.hdf

...which is a large file (~30 MB) containing detailed information on the position, tilt, etc. of the MODIS satellite. MRTSwath tool needs one of each, however.

Value

data.frame of matching files; or a list of non-matching files, if return_geoloc or return_product are TRUE.

Author(s)

Rishabh Gupta <rishabh.uk@gmail.com>

Examples

```
# Check your working directory
moddir = getwd()

# Here are some example MODIS files in mod09nrt/extdata/
# Code excluded from CRAN check because it depends on modiscdata
## Not run:
library(devtools)
library(modiscdata)
```

```
moddir = system.file("extdata/2002raw/", package="modiscdata")

# You need to have some e.g. MOD files in it (from the MODIS-TERRA platform)
list.files(path=moddir, pattern="MOD")
list.files(path=moddir, pattern="MOD")

# Check for matches (for MODIS-TERRA platform)
check_for_matching_geolocation_files_mod09nrt(moddir=moddir, modtxt="MOD09", geoloctxt="MOD03",
return_geoloc=FALSE, return_product=FALSE)

## End(Not run)
```

`extract_fn_from_path` *Get the filename from a path*

Description

The filename is split on slashes, and the last item is taken; this should be just the filename.

Usage

```
extract_fn_from_path(fn_with_path)
```

Arguments

`fn_with_path`	The filename, with partial or full path

Value

`fn` The extracted filename

Author(s)

Nicholas J. Matzke <matzke@berkeley.edu>

See Also

`strsplit`

Examples

```
fn_with_path = "/Library/Frameworks/R.framework/Versions/2.15/Resources/library/modiscloud/extdata
/2002raw/MYD35_L2.A2002185.1910.005.2007206043609.hdf"
extract_fn_from_path(fn_with_path)
```

run_swath2grid_mod09nrt *Run MRTSwath swath2grid tool*

Description

MRTSwath is the "MODIS Reprojection Tool for swath products". See: https://lpdaac.usgs.gov/tools/modis_reprojection_tool_swath).

Usage

```
run_swath2grid_mod09nrt(mrtpath = "swath2grid", prmfn = "tmpMRTparams.prm",
  tifsdir, modfn, geoloc_fn, ul_lon, ul_lat, lr_lon, lr_lat)
```

Arguments

mrtpath	This is the path to the MRTSwath executable swath2grid. If your ~/.Rprofile file has the location of swath2grid in the PATH, then you can just use mrtpath="swath2grid". Otherwise, the user must provide the full path to swath2grid.
prmfn	The name of the parameter/control file which will be the input to MRTSwath's swath2grid function.
tifsdir	The directory to save the output TIF files in
modfn	The filename of the MODIS data
geoloc_fn	The filename of the corresponding geolocation file (annoyingly, this is a much larger file than the data file!)
ul_lon	Upper left (ul) longitude (x-coordinate) for subsetting
ul_lat	Upper left (ul) latitude (y-coordinate) for subsetting
lr_lon	Lower right (lr) longitude (x-coordinate) for subsetting
lr_lat	Lower right (lr) latitude (y-coordinate) for subsetting

Details

If you want this function to use MRTSwath tool successfully, you should add the directory with the MRTSwath executable to the default R PATH by editing ~/.Rprofile.

Value

cmdstr The string giving the system command that ran swath2grid

Author(s)

Nicholas J. Matzke <matzke@berkeley.edu>

See Also

write_MRTSwath_param_file_mod09nrt

http://landweb.nascom.nasa.gov/cgi-bin/QA_WWW/newPage.cgi?fileName=hdf_filename
@cite NASA2001

Examples

```
#########################################################
# Run MRTSwath tool "swath2grid"
#########################################################

# Source MODIS files (both data and geolocation)
# Code excluded from CRAN check because it depends on modiscdata
## Not run:
library(devtools)
library(modiscdata)
moddir = system.file("extdata/2002raw/", package="modiscdata")

# Get the matching data/geolocation file pairs
fns_df = check_for_matching_geolocation_files(moddir, modtxt="MOD09", geoloctxt="MOD03")
fns_df

# Resulting TIF files go in this directory
tifsdir = getwd()


# Box to subset
ul_lat = 13
ul_lon = -87
lr_lat = 8
lr_lon = -82

for (i in 1:nrow(fns_df))
{

prmfn = write_MRTSwath_param_file_mod09nrt(prmfn="tmpMRTparams.prm", tifsdir=tifsdir,
 modfn=fns_df$mod09_fns[i], geoloc_fn=fns_df$mod03_fns[i], ul_lon=ul_lon, ul_lat=ul_lat,
 lr_lon=lr_lon, lr_lat=lr_lat)
print(scan(file=prmfn, what="character", sep="\n"))

run_swath2grid_mod09nrt(mrtpath="swath2grid", prmfn="tmpMRTparams.prm", tifsdir=tifsdir,
 modfn=fns_df$mod309_fns[i], geoloc_fn=fns_df$mod03_fns[i], ul_lon=ul_lon, ul_lat=ul_lat,
 lr_lon=lr_lon, lr_lat=lr_lat)

}

list.files(tifsdir, pattern=".tif", full.names=TRUE)

## End(Not run)
```

slashslash *Remove double slash (slash a slash)*

Description

Shortcut for: `gsub(pattern="//", replacement="/", x=tmpstr)`

Usage

```
slashslash(tmpstr)
```

Arguments

`tmpstr`	a path that you want to remove double slashes from

Details

This function is useful for removing double slashes that can appear in full pathnames due to inconsistencies in trailing slashes in working directories etc.

Value

outstr a string of the fixed path

Author(s)

Nicholas J. Matzke <matzke@berkeley.edu>

See Also

gsub

Examples

```
tmpstr = "/Library/Frameworks/R.framework/Versions/2.15/Resources/library/modiscloud/extdata/
2002raw//MYD03.A2002185.0645.005.2009192031332.hdf"

outstr = slashslash(tmpstr)
outstr
```

`write_MRTSwath_param_file_mod09nrt` *Write a parameter control file for MRTSwath*

Description

MRTSwath is the "MODIS Reprojection Tool for swath products". See: https://lpdaac.usgs.gov/tools/modis_reprojection_tool_swath).

Usage

```
write_MRTSwath_param_file_mod09nrt(prmfn = "tmpMRTparams.prm", tifsdir, modfn,
  geoloc_fn, ul_lon, ul_lat, lr_lon, lr_lat)
```

Arguments

`prmfn`	The name of the parameter/control file which will be the input to MRTSwath's `swath2grid` function.
`tifsdir`	The directory to save the output TIF files in
`modfn`	The filename of the MODIS data
`geoloc_fn`	The filename of the corresponding geolocation file (annoyingly, this is a much larger file than the data file!)
`ul_lon`	Upper left (ul) longitude (x-coordinate) for subsetting
`ul_lat`	Upper left (ul) latitude (y-coordinate) for subsetting
`lr_lon`	Lower right (lr) longitude (x-coordinate) for subsetting
`lr_lat`	Lower right (lr) latitude (y-coordinate) for subsetting

Details

If you want this function to use MRTSwath tool successfully, you should add the directory with the MRTSwath executable to the default R PATH by editing `~/.Rprofile`.

This function hard-codes these options into the parameter file:
* all the bands are extracted
* the output file is a GeoTIFF
* the output projection is Geographic (plain unprojected Latitude/Longitude)
* the resampling is Nearest Neighbor (NN), which of course is the only one which makes sense when the pixels encode bytes that encode bits that encode discrete classification results, 0/1 error flags, etc.

MRTswath can do many other projections and output formats; users can modify this function to run those options.

Value

`prmfn` The name of the temporary parameter file

Author(s)

Rishabh Gupta <rishabh.uk@gmail.com>

See Also

run_swath2grid_mod09nrt

http://landweb.nascom.nasa.gov/cgi-bin/QA_WWW/newPage.cgi?fileName=hdf_filename
@cite NASA2001

Examples

```
# Source MODIS files (both data and geolocation)
# Code excluded from CRAN check because it depends on modiscdata
## Not run:
library(devtools)
library(modiscdata)
moddir = system.file("extdata/2002raw/", package="modiscdata")

# Get the matching data/geolocation file pairs
fns_df = check_for_matching_geolocation_files_mod09nrt(moddir, modtxt="MOD09", geoloctxt="MOD03")
fns_df

# Resulting TIF files go in this directory
tifsdir = getwd()


# Box to subset
ul_lat = 13
ul_lon = -87
lr_lat = 8
lr_lon = -82

for (i in 1:nrow(fns_df))
{

prmfn = write_MRTSwath_param_file_mod09nrt(prmfn="tmpMRTparams.prm", tifsdir=tifsdir,
modfn=fns_df$mod09_fns[i], geoloc_fn=fns_df$mod03_fns[i], ul_lon=ul_lon, ul_lat=ul_lat,
lr_lon=lr_lon, lr_lat=lr_lat)
print(scan(file=prmfn, what="character", sep="\n"))

}

## End(Not run)
```

Index

adf, 2

check_for_matching_geolocation_files_mod09nrt, 2

extract_fn_from_path, 4

gsub, *7*

run_swath2grid_mod09nrt, 5, *9*

slashslash, 7
strsplit, *4*

write_MRTSwath_param_file_mod09nrt, *6*, 8