Package ‘rLandsat’

July 13, 2018

Type Package

Title Landsat Data Complete Download Process

Version 0.1.0

Author Himanshu Sikaria - SocialCops [aut, cre],
Shilpa Arora [rev],
Akash Tandon [rev],
Kriti Kathuria [rev]

Maintainer Himanshu Sikaria<himanshu.sikaria@socialcops.com>

Description
Easily acquire landsat, a remote sensing satellite, data from R. Search for landsat scenes and download the Landsat data using this library which uses APIs built by Development Seed and ESPA.

License GPL-3 | file LICENSE

URL https://github.com/socialcopsdev/rLandsat

BugReports https://github.com/socialcopsdev/rLandsat/issues

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

Depends R (>= 3.1.2)

Suggests testthat

Imports RCurl, svMisc, httr, jsonlite, dplyr, readr, stringr

NeedsCompilation no

R topics documented:

espa_cancel_order *Cancel Landst espa Orders*

Description

this will cancel the order placed earlier,

Usage

```
espa_cancel_order(order_id, host = "https://espa.cr.usgs.gov/api/v1/",
  username = NULL, password = NULL)
```

Arguments

order_id	vector of order ids to be cancelled
host	the api call host. Default set to espa v1 web api
username	default NULL, which fetches the username from the global environment. If defined otherwise, will run the api with the provided details
password	default NULL, which fetches the password from the global environment. If defined otherwise, will run the api with the provided details

Value

vector of order ids which could NOT be cancelled

Examples

```
# ==== NOT RUN ======
# input the credentials, if not defined earlier
espa_creds("your_espaname", "secret_password")

# Cancel orders
espa_cancel_order(order_id = c("your_order_id1", "your_order_id2"))
```

espa_creds *Input and Save esap-api Credentials*

Description

Save the api credentials as global environment to be used in other functions in rLandsat. This is a pre-requisite for running any of the other functions requiring espa-api

Usage

```
espa_creds(username, password)
```

Arguments

`username` espa account's username

`password` espa account's password corresponding to the username

Details

If you do not have an account with espa, please create one here: https://ers.cr.usgs.gov/register

Value

NULL. Just saves the username and password in .Reviron

Examples

```
espa_creds(username = "your_espaname", password = "secret_password")
```

`espa_order` *Places an Order for the Product IDs*

Description

Places an order in espa for the specified product ids and their products. All the products must be available for the product IDs mentioned for a successful order

Usage

```
espa_order(input_ids, product, file_format = "gtiff",
  resampling_method = "cc", order_note = "Order from R",
  projection = "lonlat", standard_parallel_1 = 29.5,
  central_meridian = -96, datum = "nad83", latitude_of_origin = 23,
  standard_parallel_2 = 45.5, false_northing = 0, false_easting = 0,
  host = "https://espa.cr.usgs.gov/api/v1/", username = NULL,
  password = NULL)
```

Arguments

`input_ids` vector of product ids for which order needs to be places

`product` vector of products required for the product ids mentioned. eg. c("sr", "toa", "sr_ndvi")

`file_format` the required output format of the order. Default "gtiff". Generally available are: "hdf-eos2": "HDF-EOS2", "envi": "ENVI", "gtiff": "GeoTiff", "netcdf": "NetCDF"

`resampling_method`
the required resampling method for the order. Default "cc". Generally available are: "cc": "Cubic Convolution", "bil": "Bilinear Interpolation", "nn": "Nearest Neighbor"

`order_note` the note (meta information) for the order

`projection` the projection of the landsat data for which order is placed. Deafult "lonlat". Avaialble are: "aea" and "lonlat"

`standard_parallel_1`
define numeric value if projection is "aea"

`central_meridian`
define numeric value if projection is "aea"

`datum` define numeric value if projection is "aea"

`latitude_of_origin`
define numeric value if projection is "aea"

`standard_parallel_2`
define numeric value if projection is "aea"

`false_northing` define numeric value if projection is "aea"

`false_easting` define numeric value if projection is "aea"

`host` the api call host. Default set to espa v1 web api

`username` default NULL, which fetches the username from the global environment. If defined otherwise, will run the api with the provided details

`password` default NULL, which fetches the password from the global environment. If defined otherwise, will run the api with the provided details

Value

a list

`order_details` a list of order id and order status if the order was successful, else blank list

`response` the API response message

`product_available`
dataframe with product ids and availability

`query` the json body sent in POST api

Examples

```
# input the credentials, if not defined earlier
espa_creds("your_espaname", "secret_password")

# saving the product ids as a vector
product_ids = c("LC08_L1TP_148047_20180202_20180220_01_T1",
               "LC08_L1TP_134040_20180115_20180120_01_T1")

 # saving the required products as a vector
 prod = c("sr", "sr_ndvi")

 # placing the order
 result = espa_order(input_ids = product_ids, product = prod, projection = "lonlat")
 orderid = result$order_details$orderid # storing the order id for future reference
```

espa_products *Check Available Products for given Product-IDs*

Description

Check Available Products for given Product-IDs

Usage

```
espa_products(input_ids, host = "https://espa.cr.usgs.gov/api/v1/",
  username = NULL, password = NULL)
```

Arguments

input_ids	vector of product ids for which available products are needed
host	the api call host. Default set to espa v1 web api
username	default NULL, which fetches the username from the global environment. If defined otherwise, will run the api with the provided details
password	default NULL, which fetches the password from the global environment. If defined otherwise, will run the api with the provided details

Value

a list :

master	dataframe with product ids as one of the columns and a column for each product with 0 (not available) and 1 (available) values.
no_product	a vector of product_ids which are incorrect
sample_message	sample response from the espa-api

Returns NULL if the espa credentials are not incorrect or the api is unresponsive

Examples

```
# input the credentials, if not defined earlier
espa_creds("your_espaname", "secret_password")

# saving the product ids as a vector
product_ids = c("LC08_L1TP_148047_20180202_20180220_01_T1",
               "LC08_L1TP_134040_20180115_20180120_01_T1",
               "invalid_id")

# running function to get the available products
result = espa_products(input_ids = product_ids)
result = result$master # saving the dataframe from the list
```

espa_status *Get Landsat Order Status and Download URL*

Description

Get Landsat Order Status and Download URL

Usage

```
espa_status(order_id = NULL, min_date = NULL, max_date = NULL,
  getSize = FALSE, host = "https://espa.cr.usgs.gov/api/v1/",
  username = NULL, password = NULL)
```

Arguments

`order_id`	vector of order ids for which status and download url is needed
`min_date`	if order_id is NULL, define the starting date from which order ids need to be fetched
`max_date`	if order_id is NULL, define the ending date till which order ids need to be fetched
`getSize`	logical. if the status is completed for the entire order, then should the file size be calculated. (Output size in Bytes)
`host`	the api call host. Default set to espa v1 web api
`username`	default NULL, which fetches the username from the global environment. If defined otherwise, will run the api with the provided details
`password`	default NULL, which fetches the password from the global environment. If defined otherwise, will run the api with the provided details

Details

if order_id, min_date, max_date are NULL, then will run on all the order ids available till date

Value

a list

`order_details`	a dataframe with order status and download links
`wrong_order_id`	vector of order_ids for which the API failed

Examples

```
# input the credentials, if not defined earlier
espa_creds("your_espaname", "secret_password")

# getting all the order's status
result = espa_status()
result = result$order_details # getting the dataframe from the list
```

espa_user *Validate Espa Credentials*

Description

To check espa credentials and if espa-api is responding. Suggest to use espa_creds function to store your credentials before running this function

Usage

```
espa_user(host = "https://espa.cr.usgs.gov/api/v1/", username = NULL,
  password = NULL)
```

Arguments

host	the api call host. Default set to espa v1 web api
username	default NULL, which fetches the username from the global environment. If defined otherwise, will run the api with the provided details
password	default NULL, which fetches the password from the global environment. If defined otherwise, will run the api with the provided details

Value

logical. TRUE if user is active, FALSE if credentials are wrong or API is unresponsive

Examples

```
# inputting the credentials
espa_creds("your_espaname", "secret_password")
# checking if the user is valid
espa_user() # returns FALSE
```

landsat_download *Downlaod Landsat Files from URL*

Description

Downlaod Landsat Files from URL

Usage

```
landsat_download(download_url, entity_id = NULL, folder_wise = FALSE,
  dest_file = NULL)
```

Arguments

`download_url`	vector of urls to be downloaded
`entity_id`	product id correspoding to the urls if the downloads need to be in folder wise for AWS links
`folder_wise`	if the downloads need to be in folder wise for AWS links
`dest_file`	the destination folder where the files are to be downloaded

Details

Caution: use entity_id and folder_wise ONLY in case of downloading the individual bands (like from AWS). Get the downlaod urls from espa functions in this library. View the demo

Value

vector of failed urls

Examples

```
landsat_download("https://edclpdsftp.cr.usgs.gov/orders/espa-order_id1.tar.gz", dest_file = getwd())
```

`landsat_search` *Search for Landsat8 Products IDs*

Description

Search for landsat8 product ids using country name or row/path

Usage

```
landsat_search(min_date = "2017-03-01", max_date = Sys.Date(),
  country = NULL, path_master = NULL, row_master = NULL,
  source = "sat-api")
```

Arguments

`min_date`	the start date of the products. Format should be %Y-%m-%d
`max_date`	the end date of the products. Format should be %Y-%m-%d
`country`	the country for which product ids is required. NULL if search is not on country. List of available countries are available at data(world_rowpath)
`path_master`	vector of path numbers
`row_master`	vector of row numbers corresponding to the path number. Check details
`source`	search source. Default and recommended is sat-api. Available options: "sat-api", "aws".

Details

for path_master and row_master input is in a pair. For example: If we want path/row of : (147,47) , (147,48) then path_master = c(147, 147), row_master = c(47,48)

Value

dataframe with the product ids and the meta information (cloud cover, path/row) along with it. If source is sat-api then raw value download links from all the sources (AWS, Google, ESPA) are also outputted

Examples

```
result = landsat_search(min_date = "2017-03-01", max_date = "2017-03-03", country = "India")
```

`world_rowpath` *World - Country wise row path mapping*

Description

World - Country wise row path mapping

Usage

```
data("world_rowpath")
```

Format

An object of class dataframe

Examples

```
data("world_rowpath")
```

Index