

Package: nneo (via r-universe)

May 26, 2026

Type Package

Title 'NEON' 'API' Client

Description 'NEON' 'API' (<<http://data.neonscience.org/data-api>>) client. Includes methods for interacting with all 'API' routes.

Version 0.1.1.9313

License MIT + file LICENSE

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

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

LazyData true

Roxygen list(markdown = TRUE)

Imports crul (>= 0.1.0), jsonlite (>= 1.1), tibble (>= 1.2), data.table, fauxpas, stringr

Suggests roxygen2 (>= 6.0.1), testthat

RoxygenNote 6.1.0

Config/pak/sysreqs libicu-dev libssl-dev

Repository <https://pecanproject.r-universe.dev>

Date/Publication 2022-05-10 13:56:54 UTC

RemoteUrl <https://github.com/ropensci-archive/nneo>

RemoteRef HEAD

RemoteSha 2dbd0ceff3e3189db1d449668e502857d5960b29

Contents

nneo-package	2
nneo_data	3
nneo_locations	4
nneo_products	5
nneo_sites	6
nneo_wrangle	6

Index	9
--------------	----------

nneo-package

nneo - R client for the NEON API

Description

NEON API docs: <http://data.neonscience.org/data-api>

nneo API

- [nneo_products\(\)](#)
- [nneo_product\(\)](#)
- [nneo_sites\(\)](#)
- [nneo_site\(\)](#)
- [nneo_locations\(\)](#)
- [nneo_location\(\)](#)
- [nneo_data\(\)](#)
- [nneo_file\(\)](#)
- [nneo_wrangle\(\)](#)

Curl options

Curl options follow those in the **curl** package. Find the curl options by looking at `curl::curl_options()`. Pass them on into any **nneo** function call.

User-Agent

A user agent string is sent in every request just to let the NEON servers know that the request is coming from R and from this package. No personal info is shared, just the package name and version for **nneo** and the underlying HTTP packages (**crul**, **curl**)

Author(s)

Scott Chamberlain <myrmecocystus@gmail.com>

Josh Roberti <jaroberti87@gmail.com>

nneo_data

Data

Description

Data

Usage

```
nneo_data(product_code, site_code, year_month, package = NULL, ...)
```

```
nneo_file(product_code, site_code, year_month, filename, ...)
```

Arguments

product_code	(character) a product code. required.
site_code	(character) a site code. required.
year_month	(character) YYYY-MM month to check for files. required.
package	(character) Package type to return, basic or expanded. optional.
...	named curl options passed on to HttpClient , e.g., <code>list(verbose = TRUE)</code> or <code>list(timeout_ms = 1000)</code>
filename	(character) a file name. optional.

Details

nneo_data gets files available for a given product/site/month combination.

nneo_file gets a file, and returns a data.frame

Value

nneo_data returns a list, while nneo_file returns a tibble/data.frame

Examples

```
## Not run:
nneo_data(product_code = "DP1.00098.001", site_code = "HEAL",
  year_month = "2016-05")

nneo_data(product_code = "DP1.00098.001", site_code = "HEAL",
  year_month = "2016-05")

## with a package
nneo_data(product_code = "DP1.00098.001", site_code = "HEAL",
  year_month = "2016-05", package = "basic")
nneo_data(product_code = "DP1.00098.001", site_code = "HEAL",
  year_month = "2016-05", package = "expanded")
```

```
## with a file name
fname <- "NEON.D19.HEAL.DP1.00098.001.003.000.030.RH_30min.2016-05.expanded.20171026T085604Z.csv"
nneo_file(product_code = "DP1.00098.001", site_code = "HEAL",
  year_month = "2016-05", filename = fname)

## curl options
nneo_data(product_code = "DP1.00098.001", site_code = "HEAL",
  year_month = "2016-05", verbose = TRUE)

## End(Not run)
```

nneo_locations	<i>Locations</i>
----------------	------------------

Description

Locations

Usage

```
nneo_locations(...)

nneo_location(x, ...)
```

Arguments

...	named curl options passed on to HttpClient , e.g., <code>list(verbose = TRUE)</code> or <code>list(timeout_ms = 1000)</code>
x	(character) a location code name

Value

a tibble (data.frame)

Examples

```
## Not run:
## list products
nneo_locations()

## get a product
res <- nneo_location("HARV")
res
res$locationProperties
res$locationChildren

## End(Not run)
```

nneo_products	<i>Products</i>
---------------	-----------------

Description

Products

Usage

```
nneo_products(...)
```

```
nneo_product(x, ...)
```

Arguments

... named curl options passed on to [HttpClient](#), e.g., `list(verbose = TRUE)` or `list(timeout_ms = 1000)`

x (character) a product code

Value

nneo_products returns a tibble (data.frame), and nneo_product returns a list

Examples

```
## Not run:
## list products
nneo_products()

## get a product
res <- nneo_product("DP3.30018.001")
res$productDescription

### get many products
ids <- c("DP3.30018.001", "DP4.00001.001", "DP3.30025.001")
lapply(ids, nneo_product)

## curl options
nneo_product("DP3.30018.001", verbose = TRUE)
nneo_product("DP3.30018.001", verbose = TRUE, useragent = "hello world")

## End(Not run)
```

nneo_sites

Sites

Description

Sites

Usage

```
nneo_sites(...)
```

```
nneo_site(x, ...)
```

Arguments

... named curl options passed on to [HttpClient](#), e.g., `list(verbose = TRUE)` or `list(timeout_ms = 1000)`

x (character) a site code

Value

a tibble (data.frame)

Examples

```
## Not run:
## list sites
nneo_sites()

## get a site
res <- nneo_site("JORN")
res$dataProducts
res$siteDescription

## End(Not run)
```

nneo_wrangle

Download and wrangle NEON's sensor-based data products

Description

Retrieve related, sensor-based datasets at a given NEON site for a custom time period, merged per measurement level and/or variable

Usage

```
nneo_wrangle(site_code = "CPER", time_start = "2017-06-20",
             time_end = NULL, data_var = "temperature", time_agr = 30,
             package = "basic", ...)
```

Arguments

site_code	(character) a site code. Required.
time_start	(character) YYYY-MM-DD start day to check for files. Required if time_end=NULL
time_end	(character) YYYY-MM-DD end day to check for files. Required if time_start=NULL
data_var	(character) NEON data product(s) to be downloaded. The user may enter a specific, controlled, NEON data product or they may enter a generic term if wanting multiple, similar, NEON data products. For instance, defining data_var as "Photosynthetically Active Radiation (Quantum Line)" would satisfy the former, and defining data_var as "Active Radiation" or just "Radiation" would satisfy the latter. Required
time_agr	(numeric) Aggregation period, given in minutes, of NEON data product(s) to be downloaded (e.g. 1 corresponds to a 1-minute data product). Required.
package	(character) Package type to return, basic or expanded. 'Expanded' datasets are only available for the smallest time_agr of each NEON data product. NOTE: 'expanded' datasets are much larger than 'basic' and will take considerably more time to download. Required
...	named curl options passed on to HttpClient , e.g., list(verbose = TRUE) or list(timeout_ms = 1000)

Value

Returns a tibble of relevant data products from all spatial locations at a NEON site for the custom time period. Data products are displayed via 'productName.spatialLocation', e.g., 'difRad-Mean.000.060' is mean, diffuse, shortwave radiation as measured on the 6th measurement level of a NEON tower (000.060), while 'linePARMean.005.000' is mean, photosynthetically Active Radiation (PAR) as measured at the 5th soil plot (005.000) of a NEON site. More information regarding spatial location and identifiers can be found at the references below.

Author(s)

Josh Roberti <jaroberti87@gmail.com>
 Dave Durden
 Robert Lee

References

[NEON Data Portal](#)
[NEON Data Availability](#)

See Also

Currently none

Examples

```
## Not run:
#download 30-minute, radiation data from NEON's Bartlett site for Summer 2016
nneo_wrangle(site_code="BART", time_start="2016-06-20",
             time_end="2016-09-21", data_var="radiation")
#download 30-minute, temperature data from NEON's Sterling (STER) site for 2017-03-04
nneo_wrangle(site_code="STER", time_start="2017-03-04",
             data_var="temperature", time_agr=30)

## End(Not run)
```

Index

- * **(NEON)**,
 - nneo_wrangle, 6
- * **Ecological**
 - nneo_wrangle, 6
- * **Ecology**,
 - nneo_wrangle, 6
- * **NEON**,
 - nneo_wrangle, 6
- * **National**
 - nneo_wrangle, 6
- * **Network**
 - nneo_wrangle, 6
- * **Observatory**
 - nneo_wrangle, 6
- * **atmosphere**
 - nneo_wrangle, 6
- * **climate**,
 - nneo_wrangle, 6
- * **data**,
 - nneo_wrangle, 6
- * **data**
 - nneo_wrangle, 6
- * **environmental**
 - nneo_wrangle, 6
- * **package**
 - nneo-package, 2
- * **products**,
 - nneo_wrangle, 6

curl::curl_options(), 2

HttpClient, 3–7

nneo (nneo-package), 2

nneo-package, 2

nneo_data, 3

nneo_data(), 2

nneo_file (nneo_data), 3

nneo_file(), 2

nneo_location (nneo_locations), 4

nneo_location(), 2

nneo_locations, 4

nneo_locations(), 2

nneo_product (nneo_products), 5

nneo_product(), 2

nneo_products, 5

nneo_products(), 2

nneo_site (nneo_sites), 6

nneo_site(), 2

nneo_sites, 6

nneo_sites(), 2

nneo_wrangle, 6

nneo_wrangle(), 2