Package 'neurohcp'

July 10, 2026

Type Package

Title Human 'Connectome' Project Interface

Version 0.12.0

Description Downloads and reads data from Human 'Connectome' Project <https://balsa.wustl.edu/> using Amazon Web Services ('AWS') 'S3' buckets.

License GPL-2

LazyData true

LazyLoad true

Depends R (>= 2.10)

Imports digest, httr, utils, xml2 (>= 1.0.1), base64enc

Suggests testthat (>= 3.0.0), dplyr, knitr, rmarkdown, covr

Config/testthat/edition 3

URL https://balsa.wustl.edu/

BugReports https://github.com/muschellij2/neurohcp/issues

VignetteBuilder knitr

Encoding UTF-8

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author John Muschelli [aut, cre],
Adi Gherman [ctb]

Maintainer John Muschelli <muschellij2@gmail.com>

Repository CRAN

Date/Publication 2026-07-10 19:10:08 UTC

Contents

bucketlist . . . 2
download_hcp_dir . . . 3
download_hcp_file . . . 4
get_hcp_file . . . 5
hcp_1200_scanning_info . . . 6
hcp_900_scanning_info . . . 6
hcp_aws_url . . . 6
hcp_ids . . . 8
hcp_list_files . . . 8
hcp_scanning_info . . . 10
make_aws_call . . . 11
parse_list_files . . . 12
set_aws_api_key . . . 13

Index **14**

`bucketlist` *Retrieve S3 Bucket List*

Description

Lists the Buckets available with the Key/Secrete

Usage

```
bucketlist(
  region = "us-east-1",
  access_key = NULL,
  secret_key = NULL,
  verbose = FALSE,
  ...
)
```

Arguments

`region`	Region of S3 Bucket
`access_key`	Amazon S3 Access Key
`secret_key`	Amazon S3 Secret Key
`verbose`	should progress be shown during the request?
`...`	additional arguments passed to the signed S3 service request

Value

List of Buckets

Examples

```
## Not run:
if (have_aws_key()){
bucketlist()
}

## End(Not run)
```

`download_hcp_dir` *Download an entire directory from HCP*

Description

Downloads a directory/folder from HCP database

Usage

```
download_hcp_dir(
  prefix,
  delimiter = "",
  outdir = tempfile(),
  verbose = TRUE,
  ...
)

download_fcp_dir(...)

download_openneuro_dir(...)
```

Arguments

`prefix`	Folder to download
`delimiter`	Delimiter for files
`outdir`	Output directory
`verbose`	Should diagnostic values be printed?
`...`	additional arguments to pass to `hcp_list_files`

Value

List of return from calling `hcp_list_files`, the output directory, and all destination files (not subsetted by those that did in fact download)

Examples

```
## Not run:
if (have_aws_key()) {
   prefix = "HCP/100307/release-notes"
   res = download_hcp_dir(prefix = prefix, verbose = FALSE)
}

## End(Not run)
```

`download_hcp_file` *Download HCP file*

Description

Downloads a file from HCP S3 Bucket

Usage

```
download_hcp_file(
  path_to_file,
  destfile = NULL,
  verbose = TRUE,
  error = TRUE,
  ...
)

download_fcp_file(...)

download_openneuro_file(...)
```

Arguments

`path_to_file`	Path to file on HCP S3 Bucket
`destfile`	Destination filename
`verbose`	should progress be added to downloading?
`error`	Should the function error if the return was bad?
`...`	arguments to pass to `hcp_aws_url`

Value

Output filename that was downloaded

Examples

```
## Not run:
if (have_aws_key()){
path_to_file <- "HCP_900/100206/MNINonLinear/100206.164k_fs_LR.wb.spec"
download_hcp_file(path_to_file = path_to_file)
}

## End(Not run)
```

`get_hcp_file` *Get HCP file*

Description

Wraps a `make_aws_call` to a `GET` statement to get the file

Usage

```
get_hcp_file(path_to_file = "/", ..., verbose = TRUE, verb = "GET")

head_hcp_file(...)

get_fcp_file(...)

head_fcp_file(...)

get_openneuro_file(...)

head_openneuro_file(...)
```

Arguments

`path_to_file`	Path to file on HCP S3 Bucket
`...`	arguments to pass to `make_aws_call`
`verbose`	Should the URL be printed?
`verb`	passed to `VERB`

Value

Result of `GET`

hcp_1200_scanning_info *Scanning Information for HCP 1200 Data*

Description

A `data.frame` with all the available scanning information for the HCP 1200 data.

Usage

```
hcp_1200_scanning_info
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 69615 rows and 18 columns.

hcp_900_scanning_info *Scanning Information for HCP 900 Data*

Description

A `data.frame` with all the available scanning information for the HCP 900 data.

Usage

```
hcp_900_scanning_info
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 34406 rows and 18 columns.

hcp_aws_url *Construct AWS S3 String*

Description

Constructs url string for AWS S3 Bucket to pass in for HCP downloading

Usage

```
hcp_aws_url(
  path_to_file = "",
  bucket = "hcp-openaccess",
  region = "us-east-1",
  access_key = NULL,
  secret_key = NULL,
  lifetime_minutes = 20,
  query = NULL,
  verb = "GET",
  sign = TRUE
)
```

Arguments

path_to_file	Path to file on HCP S3 Bucket
bucket	Bucket to download from
region	Region of S3 Bucket
access_key	Amazon S3 Access Key
secret_key	Amazon S3 Secret Key
lifetime_minutes	Time that connection can be opened
query	additional query to add to url
verb	httr VERB to be used
sign	Should the url be signed?

Value

Character of the url to be passed to `httr` 'VERB's

Examples

```
## Not run:
if (have_aws_key()){
path_to_file <- "HCP_900/100206/MNINonLinear/100206.164k_fs_LR.wb.spec"
hcp_aws_url(path_to_file)
}

## End(Not run)
```

hcp_ids *Get list of HCP Ids*

Description

Returns list of IDs of folders in the HCP database

Usage

```
hcp_ids(group = c("HCP", "HCP_900", "HCP_1200", "HCP_Retest"))
```

Arguments

group	Group of IDs to return

Value

Character vector

Examples

```
## Not run:
if (have_aws_key()) {
head(hcp_ids("HCP"))
head(hcp_ids("HCP_900"))
}

## End(Not run)
```

hcp_list_files *List files from HCP bucket*

Description

Lists a set of files from an HCP bucket for a specific sub-folder

Usage

```
hcp_list_files(
  prefix = "",
  delimiter = NULL,
  query = NULL,
  marker = NULL,
  ...,
  sign = TRUE
)
```

```
fcp_list_files(prefix = "", delimiter = NULL, query = NULL, marker = NULL, ...)

openneuro_list_files(
  prefix = "",
  delimiter = NULL,
  query = NULL,
  marker = NULL,
  ...
)

hcp_list_dirs(prefix = "HCP/", ...)

fcp_list_dirs(prefix = "data/Projects/", ...)

openneuro_list_dirs(prefix = NULL, ...)
```

Arguments

prefix	directory folder to list files. If "", then it will be the root path
delimiter	Delimiter to list files. For example ".mat", for ".mat"
query	Additional query arguments
marker	the marker to start the listing, needed for pagination of results
...	additional arguments passed to get_hcp_file
sign	Should the url be signed with an API key?

Value

List with the result of the GET command, the parsed result, and the content from the result.

Examples

```
## Not run:
if (have_aws_key()){
x = hcp_list_files(prefix = "HCP/100307/unprocessed/3T/Diffusion",
   delimiter="bval")
stopifnot(x$parsed_result$ListBucketResult$Name[[1]] == "hcp-openaccess")
t1_niis = hcp_list_files(prefix ="HCP/100307/T1w",
delimiter = ".nii.gz")
all_dirs = hcp_list_dirs("HCP/")
}

## End(Not run)

## Not run:
if (have_aws_key()){
  res = hcp_list_dirs("HCP/")
  projects = unlist(parse_list_files(res)$prefixes)
  projects = unname(projects)
```

```
  projects = unname(projects)
  head(projects)
  head(basename(projects))
  stopifnot("100307" %in% basename(projects))
}

## End(Not run)
## Not run:
if (have_aws_key()){
res = fcp_list_dirs()
projects = unlist(parse_list_files(res)$prefixes)
projects = unname(projects)
head(projects)
head(basename(projects))
stopifnot("ABIDE" %in% basename(projects))
}

## End(Not run)
## Not run:
res = openneuro_list_dirs()
projects = unlist(parse_list_files(res)$prefixes)
projects = unname(projects)
head(projects)
if (length(projects) > 0) {
head(basename(projects))
stopifnot("ds000002" %in% basename(projects))
}

## End(Not run)
```

`hcp_scanning_info` *Scanning Information for HCP Data*

Description

A `data.frame` with all the available scanning information for the HCP data.

Usage

```
hcp_scanning_info
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 34406 rows and 18 columns.

make_aws_call *Construct AWS S3 Call*

Description

Constructs GET information string for AWS S3 Bucket

Usage

```
make_aws_call(
  path_to_file = "/",
  bucket = "hcp-openaccess",
  region = "us-east-1",
  access_key = NULL,
  secret_key = NULL,
  lifetime_minutes = 5,
  query = NULL,
  verb = "GET",
  sign = TRUE
)
```

Arguments

path_to_file	Path to file on HCP S3 Bucket
bucket	Bucket to download from
region	Region of S3 Bucket
access_key	Amazon S3 Access Key
secret_key	Amazon S3 Secret Key
lifetime_minutes	Time that connection can be opened
query	additional query to add to verb command
verb	httr VERB to be used
sign	Should the url be signed?

Value

Character of the url to be passed to httr 'VERB's

Examples

```
## Not run:
if (have_aws_key()){
path_to_file <- paste0(
   "HCP_900/100206/MNINonLinear/",
   "100206.164k_fs_LR.wb.spec")
```

```
result = make_aws_call(path_to_file)
# don't want this stuff printed because it has keys
result$headers$secret_key = NULL
result$headers$access_key = NULL
result$query$AWSAccessKeyId = NULL
result$query$Signature = NULL
result
}

## End(Not run)
```

`parse_list_files` *Parse listed files from HCP bucket*

Description

This parses the result from `hcp_list_files` and organizes the files into `data.frames`

Usage

```
parse_list_files(ret)
```

Arguments

`ret`	object with element `parsed_result`, usually from `hcp_list_files`

Value

List of 2 data.frames, the contents and the `commonprefixes` elements from the list

Examples

```
## Not run:
if (have_aws_key()){
   ret = hcp_list_files(prefix = "HCP/100307/unprocessed/3T/Diffusion")
   parsed = parse_list_files(ret)
   stopifnot(!is.null(parsed$contents))
 }

## End(Not run)
```

set_aws_api_key *Set Amazon AWS Key*

Description

Sets and returns the AWS keys. This will error if not all are specified.

Usage

```
set_aws_api_key(
  access_key = NULL,
  secret_key = NULL,
  default_region = "us-east-1",
  error = TRUE
)

have_aws_key()
```

Arguments

access_key	Amazon access key. If NULL then looks at the AWS_ACCESS_KEY_ID system variable.
secret_key	Amazon secret key. If NULL then looks at the AWS_SECRET_ACCESS_KEY system variable.
default_region	Amazon default region. If NULL then looks at the AWS_DEFAULT_REGION system variable.
error	Should this function error if things are not specified?

Value

List of access_key, secret_key, and default_region

Index

∗ **datasets**
    hcp_1200_scanning_info, 6
    hcp_900_scanning_info, 6
    hcp_scanning_info, 10

bucketlist, 2

download_fcp_dir *(*download_hcp_dir*)*, 3
download_fcp_file *(*download_hcp_file*)*, 4
download_hcp_dir, 3
download_hcp_file, 4
download_openneuro_dir
        *(*download_hcp_dir*)*, 3
download_openneuro_file
        *(*download_hcp_file*)*, 4

fcp_list_dirs *(*hcp_list_files*)*, 8
fcp_list_files *(*hcp_list_files*)*, 8

GET, *5*
get_fcp_file *(*get_hcp_file*)*, 5
get_hcp_file, 5, *9*
get_openneuro_file *(*get_hcp_file*)*, 5

have_aws_key *(*set_aws_api_key*)*, 13
hcp_1200_scanning_info, 6
hcp_900_scanning_info, 6
hcp_aws_url, *4*, 6
hcp_ids, 8
hcp_list_dirs *(*hcp_list_files*)*, 8
hcp_list_files, *3*, 8, *12*
hcp_scanning_info, 10
head_fcp_file *(*get_hcp_file*)*, 5
head_hcp_file *(*get_hcp_file*)*, 5
head_openneuro_file *(*get_hcp_file*)*, 5

make_aws_call, *5*, 11

openneuro_list_dirs *(*hcp_list_files*)*, 8
openneuro_list_files *(*hcp_list_files*)*, 8

parse_list_files, 12

set_aws_api_key, 13

VERB, *5*