Package 'rsnps'

September 20, 2018

Title Get 'SNP' ('Single-Nucleotide' 'Polymorphism') Data on the Web

Description A programmatic interface to various 'SNP' 'datasets' on the web: 'OpenSNP' (<https://opensnp.org>), and 'NBCIs' 'dbSNP' database (<https://www.ncbi.nlm.nih.gov/projects/SNP>). Functions are included for searching for 'NCBI'. For 'OpenSNP', functions are included for getting 'SNPs', and data for 'genotypes', 'phenotypes', annotations, and bulk downloads of data by user.

Version 0.3.0

License MIT + file LICENSE

URL https://github.com/ropensci/rsnps

BugReports https://github.com/ropensci/rsnps/issues

VignetteBuilder knitr

Encoding UTF-8

Imports plyr, stringr, crul (>= 0.5.2), data.table, XML, xml2, jsonlite

Suggests testthat, knitr

RoxygenNote 6.1.0

NeedsCompilation no

Author Scott Chamberlain [aut, cre] (<https://orcid.org/0000-0003-1444-9135>), Kevin Ushey [aut], Hao Zhu [aut]

Maintainer Scott Chamberlain <myrmecocystus@gmail.com>

Repository CRAN

Date/Publication 2018-09-20 04:50:03 UTC

R topics documented:

`rsnps-package` *Get SNP (Single-Nucleotide Polymorphism) Data on the Web*

Description

This package gives you access to data from OpenSNP (https://opensnp.org) via their API (https://opensnp.org/faq#api).

NCBI Authenication

This applies the the functions `ncbi_snp_query()`, `ncbi_snp_query2()`, and `ncbi_snp_summary()`:

You can optionally use an API key, if you do it will allow higher rate limits (more requests per time period)

If you don't have an NCBI API key, get one at https://www.ncbi.nlm.nih.gov/account/

Create your key from your account. After generating your key set an environment variable as `ENTREZ_KEY` in .Renviron.

```
ENTREZ_KEY='youractualkeynotthisstring'
```

You can optionally pass in your API key to the key parameter in NCBI functions in this package. However, it's much better from a security perspective to set an environment variable.

Author(s)

Scott Chamberlain <myrmecocystus@gmail.com>

Kevin Ushey <kevinushey@gmail.com>

Hao Zhu <haozhu233@gmail.com>

`allgensnp` *Get openSNP genotype data for all users at a particular snp.*

Description

Get openSNP genotype data for all users at a particular snp.

Usage

```
allgensnp(snp = NA, ...)
```

Arguments

`snp`	(character) A SNP name
`...`	Curl options passed on to crul::HttpClient

Value

data.frame of genotypes for all users at a certain SNP

See Also

Other opensnp-fxns: `allphenotypes`, `annotations`, `download_users`, `fetch_genotypes`, `genotypes`, `phenotypes_byid`, `phenotypes`, `users`

Examples

```
## Not run:
x <- allgensnp(snp = 'rs7412')
head(x)

## End(Not run)
```

`allphenotypes` *Get all openSNP phenotypes, their variations, and how many users have data available for a given phenotype.*

Description

Either return data.frame with all results, or output a list, then call the charicteristic by id (paramater = "id") or name (paramater = "characteristic").

Usage

```
allphenotypes(df = FALSE, ...)
```

Arguments

df	Return a data.frame of all data. The column known_variations can take multiple values, so the other columns id, characteristic, and number_of_users are replicated in the data.frame. Default: FALSE
...	Curl options passed on to crul::HttpClient

Value

data.frame of results, or list if df=FALSE

See Also

Other opensnp-fxns: allgensnp, annotations, download_users, fetch_genotypes, genotypes, phenotypes_byid, phenotypes, users

Examples

```
## Not run:
# Get all data
allphenotypes(df = TRUE)

# Output a list, then call the characterisitc of interest by 'id' or
# 'characteristic'
datalist <- allphenotypes()
names(datalist) # get list of all characteristics you can call
datalist[["ADHD"]] # get data.frame for 'ADHD'
datalist[c("mouth size","SAT Writing")] # get data.frame for 'ADHD'

## End(Not run)
```

---

annotations *Get all openSNP phenotypes, their variations, and how many users have data available for a given phenotype.*

Description

Either return data.frame with all results, or output a list, then call the charicteristic by id (paramater = "id") or name (paramater = "characteristic").

Usage

```
annotations(snp = NA, output = c("all", "plos", "mendeley", "snpedia",
  "metadata"), ...)
```

Arguments

snp	SNP name.
output	Name the source or sources you want annotations from (options are: 'plos', 'mendeley', 'snpedia', 'metadata'). 'metadata' gives the metadata for the response.
...	Curl options passed on to crul::HttpClient

Value

data.frame of results

See Also

Other opensnp-fxns: allgensnp, allphenotypes, download_users, fetch_genotypes, genotypes, phenotypes_byid, phenotypes, users

Examples

```
## Not run:
# Get all data
## get just the metadata
annotations(snp = 'rs7903146', output = 'metadata')

## just from plos
annotations(snp = 'rs7903146', output = 'plos')

## just from snpedia
annotations(snp = 'rs7903146', output = 'snpedia')

## get all annotations
annotations(snp = 'rs7903146', output = 'all')

## End(Not run)
```

---

download_users *Download openSNP user files.*

Description

Download openSNP user files.

Usage

```
download_users(name = NULL, id = NULL, dir = "~/", ...)
```

Arguments

name	User name
id	User id
dir	Directory to save file to
...	Curl options passed on to crul::HttpClient

Value

File downloaded to directory you specify (or default), nothing returned in R.

See Also

Other opensnp-fxns: allgensnp, allphenotypes, annotations, fetch_genotypes, genotypes, phenotypes_byid, phenotypes, users

Examples

```
## Not run:
# Download a single user file, by id
download_users(id = 14)

# Download a single user file, by user name
download_users(name = 'kevinmcc')

# Download many user files
lapply(c(14,22), function(x) download_users(id=x))
read_users(id=14, nrows=5)

## End(Not run)
```

---

fetch_genotypes *Download openSNP genotype data for a user*

Description

Download openSNP genotype data for a user

Usage

```
fetch_genotypes(url, rows = 100, filepath = NULL, quiet = TRUE, ...)
```

Arguments

`url`	(character) URL for the download. See example below of function use.
`rows`	(integer) Number of rows to read in. Useful for getting a glimpse of the data. Negative and other invalid values are ignored, giving back all data. Default: 100
`filepath`	(character) If none is given the file is saved to a temporary file, which will be lost after your session is closed. Save to a file if you want to access it later.
`quiet`	(logical) Should download progress be suppressed. Default: `TRUE`
`...`	Further args passed on to `download.file()`

Details

Beware, not setting the rows parameter means that you download the entire file, which can be large (e.g., 15MB), and so take a while to download depending on your connection speed. Therefore, rows is set to 10 by default to sort of protect the user.

Internally, we use `download.file()` to download each file, then `read.table()` to read the file to a data.frame.

Value

data.frame for a single user, with four columns:

- rsid (character)
- chromsome (integer)
- position (integer)
- genotype (character)

See Also

Other opensnp-fxns: `allgensnp`, `allphenotypes`, `annotations`, `download_users`, `genotypes`, `phenotypes_byid`, `phenotypes`, `users`

Examples

```
## Not run:
# get a data.frame of the users data
data <- users(df = TRUE)
head( data[[1]] ) # users with links to genome data
mydata <- fetch_genotypes(url = data[[1]][1,"genotypes.download_url"],
  file="~/myfile.txt")

# see some data right away
mydata

# Or read in data later separately
read.table("~/myfile.txt", nrows=10)

## End(Not run)
```

genotypes *Get openSNP genotype data for one or multiple users.*

Description

Get openSNP genotype data for one or multiple users.

Usage

```
genotypes(snp = NA, userid = NA, df = FALSE, ...)
```

Arguments

snp	SNP name.
userid	ID of openSNP user.
df	Return data.frame (TRUE) or not (FALSE). Default: FALSE
...	Curl options passed on to crul::HttpClient]

Value

List (or data.frame) of genotypes for specified user(s) at a certain SNP.

See Also

Other opensnp-fxns: allgensnp, allphenotypes, annotations, download_users, fetch_genotypes, phenotypes_byid, phenotypes, users

Examples

```
## Not run:
genotypes(snp='rs9939609', userid=1)
genotypes('rs9939609', userid='1,6,8', df=TRUE)
genotypes('rs9939609', userid='1-2', df=FALSE)

## End(Not run)
```

`ncbi_snp_query` *Query NCBI's dbSNP for information on a set of SNPs*

Description

This function queries NCBI's dbSNP for information related to the latest dbSNP build and latest reference genome for information on the vector of SNPs submitted.

Usage

```
ncbi_snp_query(SNPs, key = NULL, ...)
```

Arguments

`SNPs`	(character) A vector of SNPs (rs numbers).
`key`	(character) NCBI Entrez API key. optional. See "NCBI Authenication" in rsnps-package
`...`	Curl options passed on to crul::HttpClient

Details

This function currently pulling data for Assembly 38 - in particular note that if you think the BP position is wrong, that you may be hoping for the BP position for a different Assembly. With ENTREZ we cannot specify which assembly to pull data from, so it's stuck with 38.

Note that you are limited in the number of SNPs you pass in to one request because URLs can only be so long. Around 600 is likely the max you can pass in, though may be somewhat more. Break up your vector of SNP codes into pieces of 600 or less and do repeated requests to get all data.

Value

A dataframe with columns:

- Query: The rs ID that was queried.
- Chromosome: The chromosome that the marker lies on.
- Marker: The name of the marker. If the rs ID queried has been merged, the up-to-date name of the marker is returned here, and a warning is issued.
- Class: The marker's 'class'. See `http://www.ncbi.nlm.nih.gov/projects/SNP/snp_legend.cgi?legend=snpClass` for more details.
- Gene: If the marker lies within a gene (either within the exon or introns of a gene), the name of that gene is returned here; otherwise, `NA`. Note that the gene may not be returned if the marker lies too far upstream or downstream of the particular gene of interest.
- Alleles: The alleles associated with the SNP if it is a SNV; otherwise, if it is an INDEL, microsatellite, or other kind of polymorphism the relevant information will be available here.
- Major: The major allele of the SNP, on the forward strand, given it is an SNV; otherwise, `NA`.
- Minor: The minor allele of the SNP, on the forward strand, given it is an SNV; otherwise, `NA`.

- MAF: The minor allele frequency of the SNP, given it is an SNV. This is drawn from the current global reference population used by NCBI.
- BP: The chromosomal position, in base pairs, of the marker, as aligned with the current genome used by dbSNP. we add 1 to the base pair position in the BP column in the output data.frame to agree with what the dbSNP website has.

References

https://www.ncbi.nlm.nih.gov/projects/SNP/

See Also

ncbi_snp_query2()

Examples

```
## Not run:
## an example with both merged SNPs, non-SNV SNPs, regular SNPs,
## SNPs not found, microsatellite
SNPs <- c("rs332", "rs420358", "rs1837253", "rs1209415715", "rs111068718")
ncbi_snp_query(SNPs)
# ncbi_snp_query("123456") ##invalid: must prefix with 'rs'
ncbi_snp_query("rs420358")
ncbi_snp_query("rs332") # warning that its merged into another, try that
ncbi_snp_query("rs121909001")
ncbi_snp_query("rs1837253")
ncbi_snp_query("rs1209415715")
ncbi_snp_query("rs111068718")
ncbi_snp_query(SNPs='rs9970807')

# Curl debugging
ncbi_snp_query("rs121909001")
ncbi_snp_query("rs121909001", verbose = TRUE)

## End(Not run)
```

`ncbi_snp_query2` *Query NCBI's dbSNP for information on a set of SNPs*

Description

Query NCBI's dbSNP for information on a set of SNPs

Usage

```
ncbi_snp_query2(SNPs, key = NULL, ...)
```

Arguments

SNPs	A vector of SNPs (rs numbers).
key	(character) NCBI Entrez API key. optional. See "NCBI Authenication" in rsnps-package
...	Curl options passed on to crul::HttpClient

See Also

ncbi_snp_query()

Examples

```
## Not run:
x <- c("rs332", "rs420358", "rs1837253", "rs1209415715", "rs111068718")
ncbi_snp_query2(x)
# ncbi_snp_query2("123456") ## invalid: must prefix with 'rs'
ncbi_snp_query2("rs420358")
ncbi_snp_query2("rs332") # warning, merged into new one
ncbi_snp_query2("rs121909001")
ncbi_snp_query2("rs1837253")
ncbi_snp_query2("rs1209415715") # no data available
ncbi_snp_query2("rs111068718") # chromosomal information may be unmapped

## End(Not run)
```

---

ncbi_snp_summary *Query NCBI's dbSNP for summary information on a set of SNPs*

Description

Query NCBI's dbSNP for summary information on a set of SNPs

Usage

```
ncbi_snp_summary(x, key = NULL, ...)
```

Arguments

x	A vector of SNPs (with or without 'rs' prefix)
key	(character) NCBI Entrez API key. optional. See "NCBI Authenication" in rsnps-package
...	Curl options passed on to crul::HttpClient

See Also

ncbi_snp_query2()

Examples

```
## Not run:
# use with 'rs' or without it
ncbi_snp_summary("rs420358")
ncbi_snp_summary("420358")

# you can pass > 1
x <- c("rs332", "rs420358", "rs1837253", "rs1209415715", "rs111068718")
ncbi_snp_summary(x)

ncbi_snp_summary("rs420358")
ncbi_snp_summary("rs332") # warning, merged into new one
ncbi_snp_summary("rs121909001")
ncbi_snp_summary("rs1837253")
ncbi_snp_summary("rs1209415715") # no data available
ncbi_snp_summary("rs111068718") # chromosomal information may be unmapped

## End(Not run)
```

`phenotypes` *Get openSNP phenotype data for one or multiple users.*

Description

Get openSNP phenotype data for one or multiple users.

Usage

```
phenotypes(userid = NA, df = FALSE, ...)
```

Arguments

`userid`	ID of openSNP user.
`df`	Return data.frame (`TRUE`) or not (`FALSE`). Default: `FALSE`
`...`	Curl options passed on to crul::HttpClient

Value

List of phenotypes for specified user(s).

See Also

Other opensnp-fxns: `allgensnp`, `allphenotypes`, `annotations`, `download_users`, `fetch_genotypes`, `genotypes`, `phenotypes_byid`, `users`

Examples

```
## Not run:
phenotypes(userid=1)
phenotypes(userid='1,6,8', df=TRUE)
phenotypes(userid='1-8', df=TRUE)

# coerce to data.frame
library(plyr)
df <- ldply(phenotypes(userid='1-8', df=TRUE))
head(df); tail(df)

# pass on curl options
phenotypes(1, verbose = TRUE)

## End(Not run)
```

`phenotypes_byid` *Get all openSNP known variations and all users sharing that phenotype for one phenotype(-ID).*

Description

Get all openSNP known variations and all users sharing that phenotype for one phenotype(-ID).

Usage

```
phenotypes_byid(phenotypeid = NA, return_ = c("description",
  "knownvars", "users"), ...)
```

Arguments

`phenotypeid`	ID of openSNP phenotype.
`return_`	Return data.frame (`TRUE`) or not (`FALSE`). Default: `FALSE`
`...`	Curl options passed on to crul::HttpClient

Value

List of description of phenotype, list of known variants, or data.frame of variants for each user with that phenotype.

See Also

Other opensnp-fxns: `allgensnp`, `allphenotypes`, `annotations`, `download_users`, `fetch_genotypes`, `genotypes`, `phenotypes`, `users`

Examples

```
## Not run:
phenotypes_byid(phenotypeid=12, return_ = 'desc')
phenotypes_byid(phenotypeid=12, return_ = 'knownvars')
phenotypes_byid(phenotypeid=12, return_ = 'users')

# pass on curl options
phenotypes_byid(phenotypeid=12, return_ = 'desc', verbose = TRUE)

## End(Not run)
```

`read_users` *Read in openSNP user files from local storage.*

Description

Beware, these tables can be large. Check your RAM before executing. Or possibly read in a subset of the data. This function reads in the whole kitten kaboodle.

Usage

```
read_users(name = NULL, id = NULL, path = NULL, ...)
```

Arguments

`name`	User name
`id`	User id
`path`	Path to file to read from.
`...`	Parameters passed on to `read.table()`

Details

If you specify a name or id, this function reads environment variables written in the function download_users, and then searches against those variables for the path to the file saved. Alternatively, you can supply the path.

Value

A data.frame.

Examples

```
## Not run:
# dat <- read_users(name = "kevinmcc")
# head(dat)
# dat <- read_users(id = 285)

## End(Not run)
```

rsnps-defunct *Defunct functions in rsnps*

Description

- `LDSearch()`: Function name changed to ld_search
- `ld_search()`: The Broad Institute took the service down, see https://www.broadinstitute.org/snap/snap
- `NCBI_snp_query()`: Function name changed to ncbi_snp_query
- `NCBI_snp_query2()`: Function name changed to ncbi_snp_query2

users *Get openSNP users.*

Description

Get openSNP users.

Usage

```
users(df = FALSE, ...)
```

Arguments

`df`	Return data.frame (`TRUE`) or not (`FALSE`). Default: `FALSE`
`...`	Curl options passed on to crul::HttpClient

Value

List of openSNP users, their ID numbers, and XX if available.

See Also

Other opensnp-fxns: `allgensnp`, `allphenotypes`, `annotations`, `download_users`, `fetch_genotypes`, `genotypes`, `phenotypes_byid`, `phenotypes`

Examples

```
## Not run:
# just the list
data <- users(df = FALSE)
data

# get a data.frame of the users data
data <- users(df = TRUE)
data[[1]] # users with links to genome data
data[[2]] # users without links to genome data

## End(Not run)
```

Index