Package: haploR (via r-universe)

September 3, 2024

Type Package

Title Query 'HaploReg', 'RegulomeDB'

Version 4.0.7

Date 2023-10-08

Maintainer Ilya Y. Zhbannikov <`ilya.zhbannikov@duke.edu`>

Description A set of utilities for querying 'HaploReg' <`https://pubs.broadinstitute.org/mammals/haploreg/haploreg.php`>, 'RegulomeDB' <`https://www.regulomedb.org/regulome-search/`> web-based tools. The package connects to 'HaploReg', 'RegulomeDB' searches and downloads results, without opening web pages, directly from R environment. Results are stored in a data frame that can be directly used in various kinds of downstream analyses.

RoxygenNote 7.1.1

Suggests knitr, rmarkdown,openxlsx

VignetteBuilder knitr

Imports httr,XML,tibble,RUnit,plyr,DT,RCurl,RJSONIO,methods

Depends R (>= 3.4.0)

Encoding UTF-8

License GPL-3

NeedsCompilation no

Author Ilya Y. Zhbannikov [aut, cre], Konstantin G. Arbeev [aut], Anatoliy I. Yashin [aut]

Repository CRAN

Date/Publication 2023-10-09 13:50:02 UTC

Contents

`as.num` *Converts vector of strings to numeric vector*

Description

Converts vector of strings to numeric vector

Usage

```
as.num(x, na.strings = "NA")
```

Arguments

`x`	Input vector of strings.
`na.strings`	A string which represents `NA`. Default: "NA"

Value

A numeric vector

Examples

```
library(haploR)
as.num(c("1", "2", "X"), na.strings="X")
```

`getExtendedView` *This function queries HaploReg web-based tool in order to Extended view for SNP of interest*

Description

This function queries HaploReg web-based tool in order to Extended view for SNP of interest

Usage

```
getExtendedView(snp, url = Haploreg.settings[["extended.view.url"]])
```

Arguments

`snp`	A SNP of interest.
`url`	A url to HaploReg. Default: <https://pubs.broadinstitute.org/mammals/haploreg/detail_v4.1.php?query=&

Value

A list of tables t1, t2, ..., etc depending on information contained in HaploReg database.

Examples

```
tables <- getExtendedView(snp="rs10048158")
tables
```

`getStudyList` *This function queries HaploReg web-based tool in order to see a list of GWAS.*

Description

This function queries HaploReg web-based tool in order to see a list of GWAS.

Usage

```
getStudyList(url = Haploreg.settings[["study.url"]])
```

Arguments

`url`	A url to HaploReg. Default: <https://pubs.broadinstitute.org/mammals/haploreg/haploreg.php>

Value

A list of studies. Each study is itself a list of two: `name`, `id`.

Examples

```
studies <- getStudyList()
studies
```

`queryHaploreg` *This function queries HaploReg web-based tool and returns results.*

Description

This function queries HaploReg web-based tool and returns results.

Usage

```
queryHaploreg(
  query = NULL,
  file = NULL,
  study = NULL,
  ldThresh = 0.8,
  ldPop = "EUR",
  epi = "vanilla",
  cons = "siphy",
  genetypes = "gencode",
  url = Haploreg.settings[["base.url"]],
  timeout = 100,
  encoding = "UTF-8",
  querySNP = FALSE,
  fields = NULL,
  verbose = FALSE
)
```

Arguments

`query`	Query (a vector of rsIDs).
`file`	A text file (one refSNP ID per line).
`study`	A particular study. See function `getHaploRegStudyList(...)`. Default: `NULL`.
`ldThresh`	LD threshold, r2 (select NA to only show query variants). Default: 0.8.
`ldPop`	1000G Phase 1 population for LD calculation. Can be: "AFR", "AMR", "ASN". Default: "EUR".
`epi`	Source for epigenomes. Possible values: `vanilla` for ChromHMM (Core 15-state model); `imputed` for ChromHMM (25-state model using 12 imputed marks); `methyl` for H3K4me1/H3K4me3 peaks; `acetyl` for H3K27ac/H3K9ac peaks. Default: `vanilla`.
`cons`	Mammalian conservation algorithm. Possible values: `gerp` for GERP, `siphy` for SiPhy-omega, `both` for both. Default: `siphy`.
`genetypes`	Show position relative to. Possible values: `gencode` for Gencode genes; `refseq` for RefSeq genes; `both` for both. Default: `gencode`.
`url`	HaploReg url address. Default: <https://pubs.broadinstitute.org/mammals/haploreg/haploreg.php>
`timeout`	A `timeout` parameter for `curl`. Default: 100
`encoding`	sets the `encoding` for correct retrieval web-page content. Default: `UTF-8`
`querySNP`	A flag indicating to return query SNPs only. Default: `FALSE`
`fields`	A set of fields to extract. Refer to the package vignette for available fields. Default: `All`.
`verbose`	Verbosing output. Default: FALSE.

Value

A data frame (table) with results similar to HaploReg uses.

Examples

```
library(haploR)
data <- queryHaploreg(c("rs10048158","rs4791078"))
head(data)
```

`queryRegulome` *This function queries RegulomeDB web-based tool and returns results in a data frame.*

Description

This function queries RegulomeDB web-based tool and returns results in a data frame.

Usage

```
queryRegulome(
  query = NULL,
  genomeAssembly = "GRCh37",
  limit = 1000,
  timeout = 100
)
```

Arguments

`query`	Query (a vector of rsIDs or exact one query region in rsid or like "chr1:39492461-39492462").
`genomeAssembly`	Genome assembly built: can be GRCh37 (default) or GRCh38.
`limit`	It controls how many variants will be queried and returned for a large region. It can be a number (1000 by default) or "all". Please note that large number or "all" may get yourself hurt because you could get timeout or may even crash the server.
`timeout`	A `timeout` parameter for `httr::GET`. Default: 100

Value

a data frame (table) OR a list with the following items: - guery_coordinates - features - regulome_score - variants - nearby_snps, - assembly

Examples

```
data <- queryRegulome(c("rs4791078","rs10048158"))
head(data)
```

`regulomeSearch` *This function queries RegulomeDB web-based tool and returns results in a data frame.*

Description

This function queries RegulomeDB web-based tool and returns results in a data frame.

This function queries RegulomeDB web-based tool and returns results in a data frame.

Usage

```
regulomeSearch(
  query = NULL,
  genomeAssembly = NULL,
  limit = 1000,
  timeout = 100
)

regulomeSummary(
  query = NULL,
  limit = 1000,
  genomeAssembly = NULL,
  timeout = 100
)
```

Arguments

`query`	Query (a vector of rsIDs).
`genomeAssembly`	Genome assembly built: can be GRCh37 or GRCh38
`limit`	It controls how many variants will be queried and returned for a large region. It can be a number (1000 by default) or "all". Please note that large number or "all" may get yourself hurt because you could get timeout or may even crash the server.
`timeout`	A `timeout` parameter for `httr::GET`. Default: 100

Value

a list with the following items: - guery_coordinates - features - regulome_score - variants - nearby_snps, - assembly

a data frame (table)

Examples

```
data <- regulomeSearch("rs4791078")
head(data)
data <- regulomeSummary(c("rs4791078","rs10048158"))
head(data)
```

Index