Package 'echodata'

September 17, 2021

Type Package

Title echoverse module: Fine-mapping results data

Version 0.99.0

Description echoverse module: Fine-mapping results data.

URL https://github.com/RajLabMSSM/echodata

BugReports https://github.com/RajLabMSSM/echodata/issues

Encoding UTF-8

LazyData true

Depends R (>= 3.6.0)

SystemRequirements Python (>= 3.7.0)

biocViews

Imports magrittr,
　　dplyr,
　　data.table,
　　utils,
　　R.utils,
　　stats,
　　methods,
　　tidyr,
　　parallel,
　　httr,
　　stringr

Suggests markdown,
　　rmarkdown,
　　remotes,
　　knitr,
　　BiocStyle,
　　covr,
　　testthat (>= 3.0.0)

RoxygenNote 7.1.2

VignetteBuilder knitr

License GPL-3

Config/testthat/edition 3

R topics documented:

---

example_fullSS   *Store* fullSS_dat

Description

Save downsampled genome-wide summary statistics from different GWAS to disk.

Usage

```
example_fullSS(
  save_dir = tempdir(),
  dataset = c("Nalls2019", "Kunkle2019"),
  nThread = 1,
  verbose = TRUE
)
```

Arguments

save_dir	Path to save the example summary statistics to. It is usually best to provide the absolute path rather than the relative path.
dataset	Name of the example GWAS dataset to use.
nThread	Number of threads to use.
verbose	Print messages.

Details

- "Nalls2019"Parkinson's Disease GWAS from Nalls et al. 2019 (excluding 23andMe data).

- "Kunkle2019"Alzheimer's Disease GWAS from Kunkle et al. 2019.

Value

Path to saved summary statistics file.

Examples

```
fullSS_path <- example_fullSS()
```

---

github_list_files	*List files in GitHub repo*

Description

Search for files within a public GitHub repository and return their paths.

Usage

```
github_list_files(
  creator = "RajLabMSSM",
  repo = "Fine_Mapping_Shiny",
  branch = c("main", "master"),
  query = NULL,
  return_download_api = TRUE,
  verbose = TRUE
)
```

Arguments

creator	Repo creator name (can be organization or individual GitHub account name).
repo	GitHub repository name.
branch	Which branch to search.
query	Query substring.
return_download_api	
	Return the link to download each file (instead of its path).
verbose	Print messages.

Value

A list of paths.

Kunkle2019	*Example GWAS summary statistics*

Description

Downsampled GWAS summary statistics from Kunkle et al. 2019 Alzheimer's Disease GWAS.

Usage

```
data("Kunkle2019")
```

Format

path string

Source

```
 # meta <-MungeSumstats::find_sumstats(ids = "ieu_b_2") # Kunkle2019 <-MungeSumstats::import_sumstat
= meta$id[1]) #path <-file.path("/Volumes/bms20/projects/neurogenomics-lab/live",#
"GWAS_sumstats/OpenGWAS/ieu-b-2.tsv.gz") path <-"~/Desktop/ieu-b-2.tsv.gz" Kunkle2019
<-data.table::fread(path) Kunkle2019 <-subset(Kunkle2019,P<5e-8) usethis::use_data(Kunkle2019,ove
= TRUE)
```

---

locus_dir   *Example results path for* Nall2019 *BST1 locus*

Description

Example results path for Nall2019 BST1 locus

Usage

```
data("locus_dir")
```

Format

path string

Source

```
 locus_dir <-"results/GWAS/Nalls23andMe_2019/BST1" usethis::use_data(locus_dir,overwrite
= TRUE)
```

Nalls2019   *Example GWAS summary statistics*

Description

Downsampled GWAS summary statistics from Nalls et al. 2019 Parkinson's Disease GWAS (without 23andMe data).

Usage

```
data("Nalls2019")
```

Format

path string

Source

```
 path <-echolocatoR::example_fullSS() Nalls2019 <-data.table::fread(path) usethis::use_data(Nalls2
= TRUE)
```

---

portal_metadata	*Import metadata*

Description

Import metadata for all fine-mapping results files stored on the echolocatoR Fine-mapping Portal.

Usage

```
portal_metadata(verbose = TRUE)
```

Arguments

verbose   Print messages.

Examples

```
meta <- portal_metadata()
```

---

portal_query	*Search and download fine-mapping files*

Description

Search the echolocatoR Fine-mapping Portal for fine-mapping results, LD, and locus plots.

Usage

```
portal_query(
  dataset_types = NULL,
  datasets = NULL,
  phenotypes = NULL,
  loci = NULL,
  LD_panels = c("UKB", "1KGphase1", "1KGphase3"),
  file_types = c("multi_finemap", "LD", "plot"),
  results_dir = tempdir(),
  overwrite = FALSE,
  nThread = 1,
  verbose = TRUE
)
```

Arguments

dataset_types   Dataset types to search for (e.g. "GWAS", "QTL").

datasets   Dataset names to search for (e.g. "Ripke_2014", "Wray_2018").

phenotypes   Phenotypes to search for (e.g. "Schizophrenia", "Major Depressive Disorder").

loci   Locus names to search for (e.g. "BST1", "CHRNB1", "LRRK2").

LD_panels   LD panels to search for, or results generated using different LD panels (e.g. "UKB", "1KGphase1", "1KGphase3").

file_types   File types to search for:

- "multi_finemap":Fine-mapping results merged with GWAS/QTL summary statistics.
- "LD":Linkage Disequilibrium with the lead SNP in each locus.
- "plot" : Locus plots of fine-mapped results.

results_dir   Where to save the matching queries.

overwrite   Whether to overwrite previously saved queries with the same names.

nThread   Number of threads to parallelise downloads across.

verbose   Print messages.

Value

List of local paths where the requested files were downloaded to.

Examples

```
## Not run:
local_finemap <- portal_query(
    dataset_types = "GWAS",
    phenotypes = c("schizophrenia", "parkinson"),
    file_types = "multi_finemap",
    loci = c("BST1", "CHRNB1", "LRRK2"),
    LD_panels = "1KGphase3"
)

## End(Not run)
```

Index