Package: gpmapr (via r-universe)

November 7, 2026

Title Query the 'OpenGWAS' Genotype-Phenotype Map

Version 1.0.0

Description Client for the 'OpenGWAS' genotype-phenotype map ('GPMap') API <https://gpmap.opengwas.io>, which brings together colocalisation, rare variant and molecular QTL results across thousands of traits. Functions retrieve traits, genes, variants and genomic regions with their colocalising studies, associations, linkage disequilibrium and pleiotropy summaries, search the map by name or variant, and upload GWAS summary statistics for analysis against it.

License MIT + file LICENSE

URL https://github.com/MRCIEU/gpmapr, https://gpmap.opengwas.io

BugReports https://github.com/MRCIEU/gpmapr/issues

Encoding UTF-8

RoxygenNote 7.3.2

Depends R (>= 4.1.0)

Imports dplyr, httr, jsonlite, readr, stats, utils

Suggests ggplot2, ggrepel, knitr, RColorBrewer, reshape2, rmarkdown, stringr, testthat (>= 3.1.7), tidyr

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation no

Author Andrew Elmore [aut, cre] (ORCID: <https://orcid.org/0000-0003-0292-447X>), Gibran Hemani [aut] (ORCID: <https://orcid.org/0000-0003-0920-1055>)

Maintainer Andrew Elmore <andrew.elmore@bristol.ac.uk>

Repository https://cran.r-universe.dev

Date/Publication 2026-11-07 16:56:55 UTC

RemoteUrl https://github.com/cran/gpmapr

RemoteRef HEAD

RemoteSha 3f6b7f6d13cf999896050b5063e7c1ac75c366e7

Contents

all_genes . . . 2
all_traits . . . 3
associations . . . 4
compare_upload . . . 5
delete_gwas . . . 6
gene . . . 7
genes . . . 11
get_all_gene_pleiotropies . . . 15
get_all_variant_pleiotropies . . . 16
get_gwas . . . 16
gpmap_available . . . 17
health_api . . . 18
ld_matrix . . . 18
ld_proxies . . . 19
region . . . 20
search_gpmap . . . 24
trait . . . 26
trait_duplicates . . . 30
traits . . . 31
upload_gwas . . . 35
variant . . . 37
variants . . . 41

Index **47**

`all_genes` *All genes*

Description

Get all genes from the API

Usage

```
all_genes()
```

Value

A dataframe containing all genes with the following columns:

- id: the id of the gene
- gene: the name of the gene
- description: the description of the gene
- gene_biotype: the gene biotype
- chr: the chromosome of the gene

- start: the start position of the gene
- stop: the end position of the gene
- strand: the strand of the gene
- source: the source of the gene
- distinct_trait_categories: the number of trait categories that the gene is associated with via coloc groups
- distinct_protein_coding_genes: the number of genes that the gene is associated with via coloc groups
- num_study_extractions: the number of study extractions for this gene
- num_coloc_groups: the number of coloc groups for this gene
- num_coloc_studies: the number of studies that have coloc results for this gene
- num_rare_groups: the number of rare groups for this gene

Examples

```
genes <- all_genes()
head(genes[, c("id", "gene", "chr")])
```

`all_traits` *All traits*

Description

Get all traits from the API

Usage

```
all_traits()
```

Value

A dataframe containing all traits with the following columns:

- id: the id of the trait
- data_type: the data type of the trait
- pipeline_id: the string id the GPMap pipeline uses for the trait, e.g. its OpenGWAS dataset id
- trait_name: the name of the trait
- trait_category: the trait category of the trait
- variant_type: the type of variant
- sample_size: the sample size of the trait
- variable_type: the type of variable the trait is (continuous, categorical)
- ancestry: the ancestry of the trait

- heritability: the LDSC heritability score of the trait
- heritability_se: the standard error of the LDSC heritability score of the trait
- num_study_extractions: the number of study extractions for this trait
- num_coloc_groups: the number of coloc groups for this trait
- num_coloc_studies: the number of studies that have coloc results for this trait
- num_rare_results: the number of rare results for this trait

Examples

```
traits <- all_traits()
head(traits[, c("id", "trait_name")])
```

`associations` *Get Associations by SNP ID and Study ID*

Description

Get associations from the API by SNP id and study id

Usage

```
associations(variant_ids, study_ids)
```

Arguments

`variant_ids`	A vector of numeric values specifying the SNP IDs
`study_ids`	A vector of numeric values specifying the Study IDs

Value

A dataframe containing the associations

associations_dataframe

The associations dataframe contains information about which studies have association results. It has the following columns:

- variant_id: the id of the SNP associated with this association
- study_id: the id of the study associated with this association
- beta: the beta value of the association
- se: the standard error of the association
- p: the p-value of the association
- eaf: the estimated allele frequency of the association
- imputed: whether the association is imputed

Associations of variants a study did not report are placeholders in the GPMap (beta 0, se 1, p 1), and are returned with `beta`, `se`, `p` and `eaf` set to `NA` (and `imputed` TRUE).

Which associations are stored

The GPMap does not store every study's association at every variant. It stores:

- at the variant of every coloc group and rare result: the association of every study with a study extraction in that LD block, whether or not the study is in the group. A coloc group's variant can be looked up across all of those studies.
- at a study extraction's own lead variant (its `display_snp`): that study's association only.

So other studies have no association at a study extraction's lead variant, unless that variant is also the variant of a coloc group or rare result somewhere in the map. To compare studies at a locus, use the variant of their coloc group, not the study extractions' lead variants. Asking `associations()` for a variant and study with nothing stored returns no row for that pair.

`trait(include_full_associations = TRUE)` returns the stored associations at the trait's coloc group, rare result and study extraction variants, for the trait's own studies and the studies that colocalise with it. Within those, the rules above apply: at its coloc groups' variants most of those studies have an association; at the lead variant of one of its study extractions that is in none of its coloc groups, usually only the trait's own study does.

Examples

```
associations(variant_ids = 80717, study_ids = c(926, 927))
```

`compare_upload` *Compare a GWAS upload with a trait*

Description

Line up a GWAS upload with another trait (or another uploaded GWAS it was compared with) at the places they colocalise.

For each coloc group containing both the upload and the trait, it gives the effect of each at the group's variant. Both effects are read from the upload's own result, where they are on the same alleles, so their signs can be compared directly.

Usage

```
compare_upload(upload, trait_id)
```

Arguments

`upload`	A GWAS upload result from `get_gwas()` or `trait()` with a GUID, fetched with `include_associations = TRUE`
`trait_id`	The id of a trait in the GPMap, or the GUID of another GWAS upload that `upload` was compared with (`compare_with_upload_guids` in `upload_gwas()`)

Value

A list with two dataframes:

- coloc_groups: one row per coloc group containing both the upload and the trait, with `coloc_group_id`, `ld_block_id`, `ld_block`, `variant_id`, `display_snp`, `h4_connectedness`, the upload's lead study extraction in the group (`upload_study_extraction_id`) and its `upload_beta`, `upload_se` and `upload_p`, and the same for the trait (`trait_name`, `trait_study_extraction_id`, `trait_beta`, `trait_se`, `trait_p`). The lead is the study extraction with the smallest p-value.
- coloc_pairs: the coloc pairs between one of the upload's study extractions and one of the trait's, with `ld_block_id`, `upload_study_extraction_id`, `upload_display_snp`, `trait_name`, `trait_study_extraction_id`, `trait_display_snp`, `h3`, `h4` and `in_shared_coloc_group` (whether the pair's LD block has a coloc group in `coloc_groups`). Pairs have no effects, as the two study extractions are at different variants.

Both are empty when the upload and the trait don't colocalise.

Examples

```
upload <- get_gwas("7a289615-c1b4-91f3-3d97-887f60de9155", include_associations = TRUE)
comparison <- compare_upload(upload, 2527)
head(comparison$coloc_groups)
```

---

delete_gwas *Delete a GWAS upload*

Description

Permanently delete a GWAS upload and all of its results and files. The email must match the one used for the upload, and uploads that are still processing cannot be deleted.

Usage

```
delete_gwas(gwas_id, email)
```

Arguments

`gwas_id`	The GUID of the GWAS upload
`email`	The email address used for the upload

Value

Invisibly, a list containing the API response message

Examples

```
## Not run:
# Permanently deletes an upload; needs the email address used to upload it.
delete_gwas("00000000-0000-0000-0000-000000000000", email = "you@example.com")

## End(Not run)
```

---

gene *Gene*

Description

A collection of studies that are associated with a particular gene.

Usage

```
gene(
  gene_id,
  include_associations = FALSE,
  include_coloc_pairs = FALSE,
  include_trans = TRUE,
  h4_threshold = 0.8
)
```

Arguments

gene_id	A numeric value specifying the gene id
include_associations	A logical value specifying whether to add each study's association at its variant (beta, se, p, eaf, imputed) to the coloc_groups, rare_results and study_extractions, defaults to FALSE
include_coloc_pairs	A logical value specifying whether to include coloc pairs, defaults to FALSE
include_trans	A logical value specifying whether to include trans genetic effects, defaults to TRUE
h4_threshold	A numeric value specifying the h4 threshold for coloc pairs, defaults to 0.8

Details

The dataframes returned by this function are as follows:

Value

A list which contains the following elements:

- gene: A list containing metadata about the gene, including region, and neighboring genes.
- coloc_groups: a dataframe containing information about which studies have coloc results for this gene. See below for details.
- study_extractions: a list of dataframes containing the study extractions for this trait. See below for details.
- rare_results: (optional) a list of dataframes containing the rare results for this trait
- coloc_pairs: (optional) a dataframe containing all pairwise coloc results for this trait.
- variants: a dataframe containing the variants for each associated coloc group or rare group.

See below for details.

coloc_groups_dataframe

The coloc_groups dataframe contains information about which studies have coloc results. It has the following columns:

- coloc_group_id: the unique id for this group of colocalised results
- study_id: the id of the study
- study_extraction_id: the id of the study extraction
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- h4_connectedness: the proportion of the pairs of studies in the group that colocalise (posterior probability of H4, one shared causal variant, of at least 0.8)
- h3_connectedness: the proportion of the pairs of studies in the group with a posterior probability of H3 (two distinct causal variants) of at least 0.8
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- display_snp: the display SNP name
- rsid: the rsID of the SNP
- gene: the gene whose expression, splicing or protein level the study measures (for molecular QTLs)
- gene_id: the id of the gene
- situated_gene: the gene the measured site is in, for methylation studies (which have no gene)
- situated_gene_id: the id of the situated gene
- trait_id: the id of the trait
- trait_name: the name of the trait

- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- cell_type: the cell type of the trait
- source_id: the id of the source the study comes from
- source_name: the name of the source the study comes from
- source_url: a link to the study at its source
- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the association of the study at the SNP
- source, gwas_upload_id: (GWAS upload results only) whether the row is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Identify a study extraction by `source` and `study_extraction_id` together.

study_extractions_dataframe

The study_extractions dataframe contains information about which studies have coloc results. It has the following columns:

- id: the unique id for this study extraction
- study_id: the id of the study associated with this study extraction
- variant_id: the id of the SNP
- snp: the SNP name
- ld_block_id: the id of the LD block
- unique_study_id: the unique id for this study
- study: the study name
- file: the file name
- svg_file: the SVG file name
- file_with_lbfs: the file name with lbfs
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait

- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the study's association at the study extraction's own lead variant
- source, gwas_upload_id: (GWAS upload results only) whether the study extraction is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Upload and existing study extractions have separate ids, so identify a study extraction by `source` and `id` together.

rare_results_dataframe

The rare_results dataframe contains information about which studies have coloc results. It has the following columns:

- rare_result_group_id: the unique id for this rare result group
- study_id: the id of the study associated with this rare result
- study_extraction_id: the id of the study extraction associated with this rare result
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- display_snp: the display SNP name
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- ld_block: the LD block of the SNP

coloc_pairs_dataframe

The coloc_pairs dataframe contains information about which studies have coloc pairs. It has the following columns:

- study_extraction_a_id: the id of the study extraction associated with this coloc pair
- study_extraction_b_id: the id of the study extraction associated with this coloc pair
- source_a, source_b: (GWAS upload results only) whether each study extraction is from an uploaded GWAS ("upload") or already in the GPMap ("existing")
- trait_name_a, data_type_a, tissue_a, gene_a: the trait name, data type, tissue and gene of study extraction a (and the same with `_b` for study extraction b), joined from `study_extractions`. Not added to the coloc pairs of `trait()`, whose endpoint doesn't return the study extractions its pairs refer to.

- ld_block_id: the id of the LD block
- h3: the h3 value for this coloc pair
- h4: the h4 value for this coloc pair
- spurious: whether this coloc pair is spurious

variants_dataframe

The variants dataframe contains variant information that is pulled from the Variant Effect Predictor (VEP) database. It has the following columns, along side many more columns from VEP:

- id: the id of the SNP
- gene_id: the id of the gene as predicted by VEP
- gene: the gene name as predicted by VEP

Examples

```
wnt7b <- gene("WNT7B")
head(wnt7b$coloc_groups)
```

`genes` *Genes*

Description

Get specific genes from the API. The API returns collapsed/combined data for all requested genes.

Usage

```
genes(
  gene_ids,
  include_associations = FALSE,
  include_coloc_pairs = FALSE,
  include_trans = TRUE,
  h4_threshold = 0.8
)
```

Arguments

`gene_ids`	A vector of gene ids (1 or more)
`include_associations`	A logical value specifying whether to add each study's association at its variant (beta, se, p, eaf, imputed) to the coloc_groups, rare_results and study_extractions, defaults to FALSE
`include_coloc_pairs`	A logical value specifying whether to include coloc pairs, defaults to FALSE
`include_trans`	A logical value specifying whether to include trans genetic effects, defaults to TRUE
`h4_threshold`	A numeric value specifying the h4 threshold for coloc pairs, defaults to 0.8

Details

The dataframes returned by this function are as follows:

Value

A list which contains the following elements:

- genes: gene metadata for the requested genes
- coloc_groups: a dataframe containing information about which studies have coloc results for all genes
- study_extractions: a dataframe containing the study extractions for all genes
- rare_results: a dataframe containing the rare results for all genes

coloc_groups_dataframe

The coloc_groups dataframe contains information about which studies have coloc results. It has the following columns:

- coloc_group_id: the unique id for this group of colocalised results
- study_id: the id of the study
- study_extraction_id: the id of the study extraction
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- h4_connectedness: the proportion of the pairs of studies in the group that colocalise (posterior probability of H4, one shared causal variant, of at least 0.8)
- h3_connectedness: the proportion of the pairs of studies in the group with a posterior probability of H3 (two distinct causal variants) of at least 0.8
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- display_snp: the display SNP name
- rsid: the rsID of the SNP
- gene: the gene whose expression, splicing or protein level the study measures (for molecular QTLs)
- gene_id: the id of the gene
- situated_gene: the gene the measured site is in, for methylation studies (which have no gene)
- situated_gene_id: the id of the situated gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait

- data_type: the data type of the trait
- tissue: the tissue of the trait
- cell_type: the cell type of the trait
- source_id: the id of the source the study comes from
- source_name: the name of the source the study comes from
- source_url: a link to the study at its source
- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the association of the study at the SNP
- source, gwas_upload_id: (GWAS upload results only) whether the row is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Identify a study extraction by `source` and `study_extraction_id` together.

study_extractions_dataframe

The study_extractions dataframe contains information about which studies have coloc results. It has the following columns:

- id: the unique id for this study extraction
- study_id: the id of the study associated with this study extraction
- variant_id: the id of the SNP
- snp: the SNP name
- ld_block_id: the id of the LD block
- unique_study_id: the unique id for this study
- study: the study name
- file: the file name
- svg_file: the SVG file name
- file_with_lbfs: the file name with lbfs
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait

- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the study's association at the study extraction's own lead variant
- source, gwas_upload_id: (GWAS upload results only) whether the study extraction is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Upload and existing study extractions have separate ids, so identify a study extraction by `source` and `id` together.

rare_results_dataframe

The rare_results dataframe contains information about which studies have coloc results. It has the following columns:

- rare_result_group_id: the unique id for this rare result group
- study_id: the id of the study associated with this rare result
- study_extraction_id: the id of the study extraction associated with this rare result
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- display_snp: the display SNP name
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- ld_block: the LD block of the SNP

coloc_pairs_dataframe

The coloc_pairs dataframe contains information about which studies have coloc pairs. It has the following columns:

- study_extraction_a_id: the id of the study extraction associated with this coloc pair
- study_extraction_b_id: the id of the study extraction associated with this coloc pair
- source_a, source_b: (GWAS upload results only) whether each study extraction is from an uploaded GWAS ("upload") or already in the GPMap ("existing")
- trait_name_a, data_type_a, tissue_a, gene_a: the trait name, data type, tissue and gene of study extraction a (and the same with `_b` for study extraction b), joined from `study_extractions`. Not added to the coloc pairs of `trait()`, whose endpoint doesn't return the study extractions its pairs refer to.

- ld_block_id: the id of the LD block
- h3: the h3 value for this coloc pair
- h4: the h4 value for this coloc pair
- spurious: whether this coloc pair is spurious

Examples

```
result <- genes(c("WNT7B", "WNT7A"))
result$genes
```

`get_all_gene_pleiotropies` *Get All Gene Pleiotropies*

Description

Get gene pleiotropy from the API by gene id

Usage

```
get_all_gene_pleiotropies()
```

Value

A list containing the gene pleiotropy

- gene_id: the id of the gene
- gene: the name of the gene
- distinct_trait_categories: the number of trait categories that the gene is associated with via coloc groups
- distinct_protein_coding_genes: the number of genes that the gene is associated with via coloc groups

Examples

```
head(get_all_gene_pleiotropies())
```

get_all_variant_pleiotropies *Get All SNP Pleiotropies*

Description

Get all SNP pleiotropies from the API

Usage

```
get_all_variant_pleiotropies()
```

Value

A list containing the SNP pleiotropies

- variant_id: the id of the SNP
- display_snp: the name of the SNP
- distinct_trait_categories: the number of trait categories that the SNP is associated with via coloc groups
- distinct_protein_coding_genes: the number of genes that the SNP is associated with via coloc groups

Examples

```
head(get_all_variant_pleiotropies())
```

get_gwas *Get a GWAS from the API*

Description

Get a GWAS from the API

Usage

```
get_gwas(gwas_id, include_associations = FALSE, include_summary_stats = FALSE)
```

Arguments

gwas_id	The ID of the GWAS
include_associations	Whether to add each study's association at its variant (beta, se, p, eaf, imputed) to the coloc_groups and study_extractions
include_summary_stats	Whether to include summary statistics

Value

A list containing the GWAS information:

- trait: the upload's details, e.g. its `id`, `guid`, `name` and `status`
- study_extractions: the upload's study extractions and the existing study extractions its coloc groups and pairs refer to, in one dataframe. See below.
- coloc_groups: the colocalisation groups the upload's study extractions are in, one row per study extraction in each group. See below.
- coloc_pairs: one row per colocalising pair of study extractions. Each side (`a`, `b`) is `study_extraction_a_id` with `source_a`, and `trait_name_a`, `data_type_a`, `tissue_a` and `gene_a` describe it (and the same for `b`).

Upload and existing rows

The rows of `study_extractions` and `coloc_groups` (and each side of `coloc_pairs`) come either from an uploaded GWAS or from a study already in the GPMap, which are kept in separate databases with their own ids:

- `source` is `"upload"` or `"existing"`. A study extraction is identified by `source` together with its id (`id` in `study_extractions`, `study_extraction_id` in `coloc_groups`), as an upload and an existing study extraction can have the same id. Join on both.
- `gwas_upload_id` is the id of the upload a row comes from, and `NA` for existing rows. Rows from another upload this one was compared with (`compare_with_upload_guids`) have that upload's id.
- The upload's own rows are those with `source == "upload" & gwas_upload_id == result$trait$id`.
- Upload rows are named after their upload (`trait_name`), have `data_type` "Phenotype", and have no `trait_id` or `study_id`; their `study` is the upload's GUID.

To line up the upload's effects with another trait's, see `compare_upload()`.

Associations of variants a study did not report are placeholders in the GPMap, and are returned with `beta`, `se`, `p` and `eaf` set to `NA` (and `imputed` TRUE).

Examples

```
gwas <- get_gwas("7a289615-c1b4-91f3-3d97-887f60de9155")
gwas$trait$name
```

`gpmap_available` *Is the GPMap API available?*

Description

Checks whether the GPMap API can be reached and is a version this version of gpmapr supports, without raising an error. Useful before running code that needs the API, for example in examples, vignettes or tests.

Usage

```
gpmap_available()
```

Value

TRUE if the API's health endpoint responds and its version is supported, otherwise FALSE.

Examples

```
gpmap_available()
```

`health_api` *Get API Health*

Description

Get the health status of the API

Usage

```
health_api()
```

Value

A list containing the health status

Examples

```
health_api()$status
```

`ld_matrix` *LD Matrix*

Description

Get the LD between the given variants from the API. The GPMap stores LD only between lead variants and their proxies, not between every pair of variants, so this returns the pairs among `variant_ids` where one is a lead variant and the other its proxy.

Usage

```
ld_matrix(variant_ids = c())
```

Arguments

variant_ids	A character string specifying the Variant ID. Variant IDs can be SNP IDs or variant IDs.

Value

A list containing lds, a dataframe of the stored LD pairs among variant_ids, with no rows when there are none

ld_dataframe

The ld dataframe contains information about the LD matrix. It has the following columns:

- lead_variant_id: the id of the lead SNP
- proxy_variant_id: the id of the variant SNP
- ld_block_id: the id of the LD block
- r: the r value between the lead and variant SNPs
- rsq: the r-squared between the lead and variant SNPs, rounded to 2 decimal places

Examples

```
ld_matrix(c(5553693, 5553688))
```

---

ld_proxies *LD Proxies*

Description

Get LD proxies from the API by Variant ID

Usage

```
ld_proxies(variant_ids = c())
```

Arguments

variant_ids	A character string specifying the Variant ID. Variant IDs can be SNP IDs or variant IDs.

Value

A list containing the LD proxies

ld_dataframe

The ld dataframe contains information about the LD matrix. It has the following columns:

- lead_variant_id: the id of the lead SNP
- proxy_variant_id: the id of the variant SNP
- ld_block_id: the id of the LD block
- r: the r value between the lead and variant SNPs
- rsq: the r-squared between the lead and variant SNPs, rounded to 2 decimal places

Examples

```
proxies <- ld_proxies(c(5553693, 5553694))
head(proxies$lds)
```

---

region *Region*

Description

A collection of studies that are associated with a particular region.

Usage

```
region(
  region_id,
  include_associations = FALSE,
  include_coloc_pairs = FALSE,
  h4_threshold = 0.8
)
```

Arguments

region_id	A numeric value specifying the region id
include_associations	A logical value specifying whether to add each study's association at its variant (beta, se, p, eaf, imputed) to the coloc_groups, rare_results and study_extractions, defaults to FALSE
include_coloc_pairs	A logical value specifying whether to include coloc pairs, defaults to FALSE
h4_threshold	A numeric value specifying the h4 threshold for coloc pairs, defaults to 0.8

Details

The dataframes returned by this function are as follows:

Value

A list which contains the following elements:

- gene: A list containing metadata about the gene, including region, and neighboring genes.
- coloc_groups: a dataframe containing information about which studies have coloc results for this gene. See below for details.
- study_extractions: a list of dataframes containing the study extractions for this trait. See below for details.
- rare_results: (optional) a list of dataframes containing the rare results for this trait
- coloc_pairs: (optional) a dataframe containing all pairwise coloc results for this trait.
- variants: a dataframe containing the variants for each associated coloc group or rare group.

See below for details.

coloc_groups_dataframe

The coloc_groups dataframe contains information about which studies have coloc results. It has the following columns:

- coloc_group_id: the unique id for this group of colocalised results
- study_id: the id of the study
- study_extraction_id: the id of the study extraction
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- h4_connectedness: the proportion of the pairs of studies in the group that colocalise (posterior probability of H4, one shared causal variant, of at least 0.8)
- h3_connectedness: the proportion of the pairs of studies in the group with a posterior probability of H3 (two distinct causal variants) of at least 0.8
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- display_snp: the display SNP name
- rsid: the rsID of the SNP
- gene: the gene whose expression, splicing or protein level the study measures (for molecular QTLs)
- gene_id: the id of the gene
- situated_gene: the gene the measured site is in, for methylation studies (which have no gene)
- situated_gene_id: the id of the situated gene
- trait_id: the id of the trait
- trait_name: the name of the trait

- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- cell_type: the cell type of the trait
- source_id: the id of the source the study comes from
- source_name: the name of the source the study comes from
- source_url: a link to the study at its source
- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the association of the study at the SNP
- source, gwas_upload_id: (GWAS upload results only) whether the row is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Identify a study extraction by `source` and `study_extraction_id` together.

genes_in_region_dataframe

The genes_in_region dataframe contains information about which genes are in a region. It has the following columns:

- id: the id of the gene
- ensembl_id: the ensembl id of the gene
- gene: the name of the gene
- description: the description of the gene
- gene_biotype: the gene biotype
- chr: the chromosome of the gene
- start: the start position of the gene
- stop: the stop position of the gene
- strand: the strand of the gene
- source: the source of the gene

study_extractions_dataframe

The study_extractions dataframe contains information about which studies have coloc results. It has the following columns:

- id: the unique id for this study extraction
- study_id: the id of the study associated with this study extraction
- variant_id: the id of the SNP
- snp: the SNP name
- ld_block_id: the id of the LD block
- unique_study_id: the unique id for this study
- study: the study name
- file: the file name

- svg_file: the SVG file name
- file_with_lbfs: the file name with lbfs
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the study's association at the study extraction's own lead variant
- source, gwas_upload_id: (GWAS upload results only) whether the study extraction is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Upload and existing study extractions have separate ids, so identify a study extraction by `source` and `id` together.

rare_results_dataframe

The rare_results dataframe contains information about which studies have coloc results. It has the following columns:

- rare_result_group_id: the unique id for this rare result group
- study_id: the id of the study associated with this rare result
- study_extraction_id: the id of the study extraction associated with this rare result
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- display_snp: the display SNP name
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- ld_block: the LD block of the SNP

coloc_pairs_dataframe

The coloc_pairs dataframe contains information about which studies have coloc pairs. It has the following columns:

- study_extraction_a_id: the id of the study extraction associated with this coloc pair
- study_extraction_b_id: the id of the study extraction associated with this coloc pair
- source_a, source_b: (GWAS upload results only) whether each study extraction is from an uploaded GWAS ("upload") or already in the GPMap ("existing")
- trait_name_a, data_type_a, tissue_a, gene_a: the trait name, data type, tissue and gene of study extraction a (and the same with `_b` for study extraction b), joined from `study_extractions`. Not added to the coloc pairs of `trait()`, whose endpoint doesn't return the study extractions its pairs refer to.
- ld_block_id: the id of the LD block
- h3: the h3 value for this coloc pair
- h4: the h4 value for this coloc pair
- spurious: whether this coloc pair is spurious

variants_dataframe

The variants dataframe contains variant information that is pulled from the Variant Effect Predictor (VEP) database. It has the following columns, along side many more columns from VEP:

- id: the id of the SNP
- gene_id: the id of the gene as predicted by VEP
- gene: the gene name as predicted by VEP

Examples

```
result <- region(459)
head(result$coloc_groups)
```

---

search_gpmap *Search the Genotype-Phenotype Map*

Description

Search the GP Map for Traits, Genes or Variants

Usage

```
search_gpmap(search_text, rsquared_threshold = 0.8)
```

Arguments

search_text	A character string specifying the search text
rsquared_threshold	A numeric value specifying the rsquared threshold for proxy variants, defaults to 0.8

Details

After calling search, you can use call the subsequent data as described in the call column of the search results.

Value

A dataframe containing the search results, ordered by the number of colocalisation groups plus rare results (most first). For variant searches the searched variant is always first. Columns include:

- type: the type of the search result: "original_variant", "proxy_variant", "trait", "gene"
- name: the name of the search result
- type_id: the type_id of the search result. This is the internal id in which the data can be accessed.
- rsq: (variant searches only) the r-squared between a proxy variant and the searched variant; NA for the searched variant itself
- num_study_extractions: the number of study extractions (NA for variants)
- num_coloc_groups: the number of colocalisation groups
- num_coloc_studies: the number of colocalising studies
- num_rare_results: the number of rare results
- call: the call to get the search result: "variant(type_id)", "trait(type_id)", "gene(type_id)"

Trait and gene results also carry the descriptive columns of the search options (e.g. aliases, sample_size, ancestry).

Examples

```
search_gpmap("haemoglobin")
search_gpmap("rs9939609")
```

`trait` *Trait*

Description

A collection of studies that are associated with a particular phenotype. A trait will include a common study and occasionally a rare study. When trait_id is a GUID (from GWAS upload), fetches the upload result instead, as `get_gwas()` does: see its documentation for how upload results mark the upload's own rows.

Usage

```
trait(
  trait_id,
  include_associations = FALSE,
  include_full_associations = FALSE,
  include_coloc_pairs = FALSE,
  h4_threshold = 0.8
)
```

Arguments

`trait_id`	A numeric value or GUID (from GWAS upload) specifying the trait id
`include_associations`	A logical value specifying whether to add each study's association at its variant (beta, se, p, eaf, imputed) to the coloc_groups, rare_results and study_extractions, defaults to FALSE
`include_full_associations`	A logical value specifying whether to include the stored associations of the trait's studies and the studies that colocalise with it, at the trait's coloc group, rare result and study extraction variants (see "Which associations are stored"), defaults to FALSE
`include_coloc_pairs`	A logical value specifying whether to include coloc pairs, defaults to FALSE
`h4_threshold`	A numeric value specifying the h4 threshold for coloc pairs, defaults to 0.8

Details

The dataframes returned by this function are as follows:

Value

A list which contains the following elements:

- trait: A list containing metadata about the trait, including its `pipeline_id`, `source_url`, and the common and rare studies associated with it

- coloc_groups: a dataframe containing information about which studies have coloc results for this trait. See below for details.
- study_extractions: a list of dataframes containing the study extractions for this trait. See below for details.
- rare_results: (optional) a list of dataframes containing the rare results for this trait
- coloc_pairs: (optional) a dataframe containing all pairwise coloc results for this trait.
- full_associations: (optional) a dataframe of associations (`variant_id`, `study_id`, `beta`, `se`, `p`, ...): see "Which associations are stored" below for which variants and studies it covers.

See below for details.

coloc_groups_dataframe

The coloc_groups dataframe contains information about which studies have coloc results. It has the following columns:

- coloc_group_id: the unique id for this group of colocalised results
- study_id: the id of the study
- study_extraction_id: the id of the study extraction
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- h4_connectedness: the proportion of the pairs of studies in the group that colocalise (posterior probability of H4, one shared causal variant, of at least 0.8)
- h3_connectedness: the proportion of the pairs of studies in the group with a posterior probability of H3 (two distinct causal variants) of at least 0.8
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- display_snp: the display SNP name
- rsid: the rsID of the SNP
- gene: the gene whose expression, splicing or protein level the study measures (for molecular QTLs)
- gene_id: the id of the gene
- situated_gene: the gene the measured site is in, for methylation studies (which have no gene)
- situated_gene_id: the id of the situated gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait

- tissue: the tissue of the trait
- cell_type: the cell type of the trait
- source_id: the id of the source the study comes from
- source_name: the name of the source the study comes from
- source_url: a link to the study at its source
- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the association of the study at the SNP
- source, gwas_upload_id: (GWAS upload results only) whether the row is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Identify a study extraction by `source` and `study_extraction_id` together.

study_extractions_dataframe

The study_extractions dataframe contains information about which studies have coloc results. It has the following columns:

- id: the unique id for this study extraction
- study_id: the id of the study associated with this study extraction
- variant_id: the id of the SNP
- snp: the SNP name
- ld_block_id: the id of the LD block
- unique_study_id: the unique id for this study
- study: the study name
- file: the file name
- svg_file: the SVG file name
- file_with_lbfs: the file name with lbfs
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the study's association at the study extraction's own lead variant
- source, gwas_upload_id: (GWAS upload results only) whether the study extraction is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Upload and existing study extractions have separate ids, so identify a study extraction by `source` and `id` together.

rare_results_dataframe

The rare_results dataframe contains information about which studies have coloc results. It has the following columns:

- rare_result_group_id: the unique id for this rare result group
- study_id: the id of the study associated with this rare result
- study_extraction_id: the id of the study extraction associated with this rare result
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- display_snp: the display SNP name
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- ld_block: the LD block of the SNP

coloc_pairs_dataframe

The coloc_pairs dataframe contains information about which studies have coloc pairs. It has the following columns:

- study_extraction_a_id: the id of the study extraction associated with this coloc pair
- study_extraction_b_id: the id of the study extraction associated with this coloc pair
- source_a, source_b: (GWAS upload results only) whether each study extraction is from an uploaded GWAS ("upload") or already in the GPMap ("existing")
- trait_name_a, data_type_a, tissue_a, gene_a: the trait name, data type, tissue and gene of study extraction a (and the same with `_b` for study extraction b), joined from `study_extractions`. Not added to the coloc pairs of `trait()`, whose endpoint doesn't return the study extractions its pairs refer to.
- ld_block_id: the id of the LD block
- h3: the h3 value for this coloc pair
- h4: the h4 value for this coloc pair
- spurious: whether this coloc pair is spurious

variants_dataframe

The variants dataframe contains variant information that is pulled from the Variant Effect Predictor (VEP) database. It has the following columns, along side many more columns from VEP:

- id: the id of the SNP
- gene_id: the id of the gene as predicted by VEP
- gene: the gene name as predicted by VEP

Which associations are stored

The GPMap does not store every study's association at every variant. It stores:

- at the variant of every coloc group and rare result: the association of every study with a study extraction in that LD block, whether or not the study is in the group. A coloc group's variant can be looked up across all of those studies.
- at a study extraction's own lead variant (its `display_snp`): that study's association only.

So other studies have no association at a study extraction's lead variant, unless that variant is also the variant of a coloc group or rare result somewhere in the map. To compare studies at a locus, use the variant of their coloc group, not the study extractions' lead variants. Asking `associations()` for a variant and study with nothing stored returns no row for that pair.

`trait(include_full_associations = TRUE)` returns the stored associations at the trait's coloc group, rare result and study extraction variants, for the trait's own studies and the studies that colocalise with it. Within those, the rules above apply: at its coloc groups' variants most of those studies have an association; at the lead variant of one of its study extractions that is in none of its coloc groups, usually only the trait's own study does.

Examples

```
result <- trait(5020)
head(result$coloc_groups)
```

`trait_duplicates` *Trait duplicates*

Description

Get all traits that have been marked as a duplicate of another trait

Usage

```
trait_duplicates()
```

Value

A dataframe containing the duplicated traits with the following columns:

- trait_id: the id of the duplicated trait
- trait_name: the name of the duplicated trait
- parent_trait_id: the id of the trait it duplicates
- parent_trait_name: the name of the trait it duplicates

Examples

```
head(trait_duplicates())
```

---

traits *Traits*

Description

Get specific traits from the API. The API returns collapsed/combined data for all requested traits. When a trait ID is a GUID (from GWAS upload), fetches the upload result instead.

Usage

```
traits(
  trait_ids,
  include_associations = FALSE,
  include_coloc_pairs = FALSE,
  h4_threshold = 0.8
)
```

Arguments

trait_ids	A vector of trait ids (numeric) or GUIDs (from GWAS upload)
include_associations	A logical value specifying whether to add each study's association at its variant (beta, se, p, eaf, imputed) to the coloc_groups, rare_results and study_extractions, defaults to FALSE
include_coloc_pairs	A logical value specifying whether to include coloc pairs, defaults to FALSE. Coloc pairs are fetched from a separate endpoint per trait.
h4_threshold	A numeric value specifying the h4 threshold for coloc pairs, defaults to 0.8

Details

The dataframes returned by this function are as follows:

Value

A list which contains the following elements:

- traits: trait metadata for the requested traits
- coloc_groups: a dataframe containing information about which studies have coloc results for all traits. See below for details.
- study_extractions: a dataframe containing the study extractions for all traits. See below for details.
- rare_results: a dataframe containing the rare results for all traits
- coloc_pairs: (optional) a dataframe containing all pairwise coloc results for all traits.

coloc_groups_dataframe

The coloc_groups dataframe contains information about which studies have coloc results. It has the following columns:

- coloc_group_id: the unique id for this group of colocalised results
- study_id: the id of the study
- study_extraction_id: the id of the study extraction
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- h4_connectedness: the proportion of the pairs of studies in the group that colocalise (posterior probability of H4, one shared causal variant, of at least 0.8)
- h3_connectedness: the proportion of the pairs of studies in the group with a posterior probability of H3 (two distinct causal variants) of at least 0.8
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- display_snp: the display SNP name
- rsid: the rsID of the SNP
- gene: the gene whose expression, splicing or protein level the study measures (for molecular QTLs)
- gene_id: the id of the gene
- situated_gene: the gene the measured site is in, for methylation studies (which have no gene)
- situated_gene_id: the id of the situated gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait

- tissue: the tissue of the trait
- cell_type: the cell type of the trait
- source_id: the id of the source the study comes from
- source_name: the name of the source the study comes from
- source_url: a link to the study at its source
- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the association of the study at the SNP
- source, gwas_upload_id: (GWAS upload results only) whether the row is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Identify a study extraction by `source` and `study_extraction_id` together.

study_extractions_dataframe

The study_extractions dataframe contains information about which studies have coloc results. It has the following columns:

- id: the unique id for this study extraction
- study_id: the id of the study associated with this study extraction
- variant_id: the id of the SNP
- snp: the SNP name
- ld_block_id: the id of the LD block
- unique_study_id: the unique id for this study
- study: the study name
- file: the file name
- svg_file: the SVG file name
- file_with_lbfs: the file name with lbfs
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the study's association at the study extraction's own lead variant
- source, gwas_upload_id: (GWAS upload results only) whether the study extraction is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Upload and existing study extractions have separate ids, so identify a study extraction by `source` and `id` together.

rare_results_dataframe

The rare_results dataframe contains information about which studies have coloc results. It has the following columns:

- rare_result_group_id: the unique id for this rare result group
- study_id: the id of the study associated with this rare result
- study_extraction_id: the id of the study extraction associated with this rare result
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- display_snp: the display SNP name
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- ld_block: the LD block of the SNP

coloc_pairs_dataframe

The coloc_pairs dataframe contains information about which studies have coloc pairs. It has the following columns:

- study_extraction_a_id: the id of the study extraction associated with this coloc pair
- study_extraction_b_id: the id of the study extraction associated with this coloc pair
- source_a, source_b: (GWAS upload results only) whether each study extraction is from an uploaded GWAS ("upload") or already in the GPMap ("existing")
- trait_name_a, data_type_a, tissue_a, gene_a: the trait name, data type, tissue and gene of study extraction a (and the same with `_b` for study extraction b), joined from `study_extractions`. Not added to the coloc pairs of `trait()`, whose endpoint doesn't return the study extractions its pairs refer to.
- ld_block_id: the id of the LD block
- h3: the h3 value for this coloc pair
- h4: the h4 value for this coloc pair
- spurious: whether this coloc pair is spurious

Examples

```
result <- traits(c(4405, 4872))
result$traits
```

upload_gwas *Upload a GWAS to the API*

Description

Upload a GWAS to the API. Column names are specified with the `*_col` arguments, e.g. `chr_col = "chr"`. `chr_col`, `bp_col`, `ea_col`, `oa_col` and `p_col` are required, along with either `beta_col` and `se_col`, or `or_col`, `or_lb_col` and `or_ub_col`.

Usage

```
upload_gwas(
  file,
  name,
  p_value_threshold = 5e-08,
  column_names = NULL,
  email = NA,
  variable_type = "continuous",
  is_published = FALSE,
  doi = NA,
  should_be_added = FALSE,
  ancestry = "EUR",
  sample_size = NA,
  reference_build = "GRCh38",
  compare_with_upload_guids = NA,
  chr_col = NULL,
  bp_col = NULL,
  ea_col = NULL,
  oa_col = NULL,
  p_col = NULL,
  eaf_col = NULL,
  beta_col = NULL,
  se_col = NULL,
  or_col = NULL,
  or_lb_col = NULL,
  or_ub_col = NULL,
  snp_col = NULL,
  rsid_col = NULL,
  category = NULL
)
```

Arguments

`file`	The path to the GWAS file, maximum size is 1GB
`name`	The name of the GWAS
`p_value_threshold`	The p-value threshold for the GWAS

column_names	Deprecated, use the *_col arguments instead. A named list of column names in the format of: list(CHR = "chr", BP = "pos"...). Accepted names are SNP, RSID, CHR, BP, EA, OA, P, EAF, BETA, SE, OR, OR_LB (or LB) and OR_UB (or UB).
email	The email of the user
variable_type	The type of variable the GWAS is of. Only "continuous" and "categorical" are accepted.
is_published	Whether the GWAS is published
doi	The DOI of the GWAS
should_be_added	Whether the GWAS should be added to the API
ancestry	The ancestry of the GWAS. Currently only "EUR" is accepted.
sample_size	The sample size of the GWAS
reference_build	The reference build of the GWAS. Only "GRCh37" and "GRCh38" are accepted.
compare_with_upload_guids	A vector of GUIDs of uploads to compare with
chr_col	The name of the chromosome column (required)
bp_col	The name of the base pair position column (required)
ea_col	The name of the effect allele column (required)
oa_col	The name of the other allele column (required)
p_col	The name of the p-value column (required)
eaf_col	The name of the effect allele frequency column (optional)
beta_col	The name of the beta column (required with se_col, unless using odds ratios)
se_col	The name of the standard error column (required with beta_col, unless using odds ratios)
or_col	The name of the odds ratio column (required with or_lb_col and or_ub_col, unless using beta)
or_lb_col	The name of the odds ratio lower confidence bound column
or_ub_col	The name of the odds ratio upper confidence bound column
snp_col	The name of the SNP identifier column (optional)
rsid_col	The name of the rsID column (optional)
category	Deprecated, use variable_type instead.

Value

A list containing the GWAS information, including the guid used to fetch results

Examples

```
## Not run:
# Uploads summary statistics to the GPMap server; needs an email address.
result <- upload_gwas(
  "my_gwas.tsv", name = "My GWAS", email = "you@example.com", sample_size = 10000,
  chr_col = "chr", bp_col = "pos", ea_col = "ea", oa_col = "oa",
  p_col = "p", beta_col = "beta", se_col = "se"
)

## End(Not run)
```

---

variant *Variant*

Description

A collection of studies that are associated with a particular variant.

Usage

```
variant(
  variant_id,
  include_coloc_pairs = FALSE,
  h4_threshold = 0.8,
  include_summary_stats = FALSE
)
```

Arguments

variant_id	A character string specifying the SNP ID
include_coloc_pairs	A logical value specifying whether to include coloc pairs
h4_threshold	A numeric value specifying the cutoff for included coloc pairs, defaults to 0.8. Only used if include_coloc_pairs is TRUE.
include_summary_stats	A logical value specifying whether to include summary stats

Details

The dataframes returned by this function are as follows:

Value

A list which contains the following elements:

- variant: named list containing the variant information
- coloc_groups: a dataframe containing information about which studies have coloc results for this variant

- rare_results: a list of dataframes containing the rare variants
- study_extractions: a list of dataframes containing the study extractions
- coloc_groups, rare_results and study_extractions include each study's association at its variant (beta, se, p, eaf, imputed)
- summary_stats (optional): a list of dataframes containing the summary stats for each study, where the name of each element is the study_id. Column names are uppercase (e.g. SNP, BP, BETA, SE, LBF_1).
- coloc_pairs (optional): a dataframe containing information about which studies have coloc pairs for this variant where the study_extraction_a_id and study_extraction_b_id are the study_extraction_ids of the two studies. h4_threshold is the cutoff for included coloc pairs, defaults to 0.8

coloc_groups_dataframe

The coloc_groups dataframe contains information about which studies have coloc results. It has the following columns:

- coloc_group_id: the unique id for this group of colocalised results
- study_id: the id of the study
- study_extraction_id: the id of the study extraction
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- h4_connectedness: the proportion of the pairs of studies in the group that colocalise (posterior probability of H4, one shared causal variant, of at least 0.8)
- h3_connectedness: the proportion of the pairs of studies in the group with a posterior probability of H3 (two distinct causal variants) of at least 0.8
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- display_snp: the display SNP name
- rsid: the rsID of the SNP
- gene: the gene whose expression, splicing or protein level the study measures (for molecular QTLs)
- gene_id: the id of the gene
- situated_gene: the gene the measured site is in, for methylation studies (which have no gene)
- situated_gene_id: the id of the situated gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait

- tissue: the tissue of the trait
- cell_type: the cell type of the trait
- source_id: the id of the source the study comes from
- source_name: the name of the source the study comes from
- source_url: a link to the study at its source
- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the association of the study at the SNP
- source, gwas_upload_id: (GWAS upload results only) whether the row is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Identify a study extraction by `source` and `study_extraction_id` together.

rare_results_dataframe

The rare_results dataframe contains information about which studies have coloc results. It has the following columns:

- rare_result_group_id: the unique id for this rare result group
- study_id: the id of the study associated with this rare result
- study_extraction_id: the id of the study extraction associated with this rare result
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- display_snp: the display SNP name
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- ld_block: the LD block of the SNP

study_extractions_dataframe

The study_extractions dataframe contains information about which studies have coloc results. It has the following columns:

- id: the unique id for this study extraction
- study_id: the id of the study associated with this study extraction
- variant_id: the id of the SNP

- snp: the SNP name
- ld_block_id: the id of the LD block
- unique_study_id: the unique id for this study
- study: the study name
- file: the file name
- svg_file: the SVG file name
- file_with_lbfs: the file name with lbfs
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the study's association at the study extraction's own lead variant
- source, gwas_upload_id: (GWAS upload results only) whether the study extraction is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Upload and existing study extractions have separate ids, so identify a study extraction by `source` and `id` together.

summary_statistics_dataframe

The summary_statistics dataframe contains information about which studies have summary statistics. From the API, column names are typically uppercase (SNP, CHR, BP, EA, OA, EAF, Z, BETA, SE, P, LBF_1, etc.). It has the following columns (names may be upper or lower case depending on source):

- SNP / variant_id: the id of the SNP
- CHR / chr: the chromosome of the SNP
- BP / bp: the base pair position of the SNP
- EA / ea: the effect allele
- OA / oa: the other allele
- EAF / eaf: the estimated allele frequency
- Z / z: the z-score
- BETA / beta: the beta value

- SE / se: the standard error
- P / p: the p-value
- imputed: whether the summary statistics are imputed
- LBF_* / lbf_*: all different finemapped log-bayes factors for each credible set. Each credible set is numbered from 1 to 10. If finemapped failed or only returned 1 credible set, the LBF_1 column is just converted directly from the z-score.

coloc_pairs_dataframe

The coloc_pairs dataframe contains information about which studies have coloc pairs. It has the following columns:

- study_extraction_a_id: the id of the study extraction associated with this coloc pair
- study_extraction_b_id: the id of the study extraction associated with this coloc pair
- source_a, source_b: (GWAS upload results only) whether each study extraction is from an uploaded GWAS ("upload") or already in the GPMap ("existing")
- trait_name_a, data_type_a, tissue_a, gene_a: the trait name, data type, tissue and gene of study extraction a (and the same with _b for study extraction b), joined from `study_extractions`. Not added to the coloc pairs of `trait()`, whose endpoint doesn't return the study extractions its pairs refer to.
- ld_block_id: the id of the LD block
- h3: the h3 value for this coloc pair
- h4: the h4 value for this coloc pair
- spurious: whether this coloc pair is spurious

Examples

```
result <- variant(5553693)
head(result$coloc_groups)
```

`variants` *Variants*

Description

Get specific variants from the API. The API accepts variant identifiers (variant_ids, rsids, or strings) and returns collapsed/combined data. The API distinguishes between identifier types automatically. Max 10 variants when expand=TRUE.

Usage

```
variants(
  variants,
  expand = FALSE,
  include_associations = FALSE,
  include_coloc_pairs = FALSE,
  h4_threshold = 0.8
)
```

Arguments

variants	A vector of variant identifiers (variant_ids, rsids, or strings)
expand	Logical. FALSE (default) returns minimal data. TRUE returns full VariantResponse (max 10)
include_associations	Logical. Whether to add each study's association at its variant (beta, se, p, eaf, imputed) to the coloc_groups, rare_results and study_extractions. Only when expand=TRUE
include_coloc_pairs	Logical. Whether to include coloc pairs. Only when expand=TRUE
h4_threshold	Numeric. H4 threshold for coloc pairs, defaults to 0.8

Details

The dataframes returned by this function are as follows:

Value

A list which contains the following elements:

- variants: a dataframe containing the variants for all requested variants
- coloc_groups: (if expanded) a dataframe containing the coloc groups for all variants
- study_extractions: (if expanded) a dataframe containing the study extractions for all variants
- rare_results: (if expanded) a dataframe containing the rare results for all variants

coloc_groups_dataframe

The coloc_groups dataframe contains information about which studies have coloc results. It has the following columns:

- coloc_group_id: the unique id for this group of colocalised results
- study_id: the id of the study
- study_extraction_id: the id of the study extraction
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block

- h4_connectedness: the proportion of the pairs of studies in the group that colocalise (posterior probability of H4, one shared causal variant, of at least 0.8)
- h3_connectedness: the proportion of the pairs of studies in the group with a posterior probability of H3 (two distinct causal variants) of at least 0.8
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- display_snp: the display SNP name
- rsid: the rsID of the SNP
- gene: the gene whose expression, splicing or protein level the study measures (for molecular QTLs)
- gene_id: the id of the gene
- situated_gene: the gene the measured site is in, for methylation studies (which have no `gene`)
- situated_gene_id: the id of the situated gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- cell_type: the cell type of the trait
- source_id: the id of the source the study comes from
- source_name: the name of the source the study comes from
- source_url: a link to the study at its source
- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the association of the study at the SNP
- source, gwas_upload_id: (GWAS upload results only) whether the row is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Identify a study extraction by `source` and `study_extraction_id` together.

study_extractions_dataframe

The study_extractions dataframe contains information about which studies have coloc results. It has the following columns:

- id: the unique id for this study extraction
- study_id: the id of the study associated with this study extraction
- variant_id: the id of the SNP
- snp: the SNP name
- ld_block_id: the id of the LD block

- unique_study_id: the unique id for this study
- study: the study name
- file: the file name
- svg_file: the SVG file name
- file_with_lbfs: the file name with lbfs
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- cis_trans: the cis/trans status of the SNP
- ld_block: the LD block of the SNP
- gene: the gene associated with the SNP
- gene_id: the id of the gene
- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- beta, se, p, eaf, imputed: (with `include_associations = TRUE`) the study's association at the study extraction's own lead variant
- source, gwas_upload_id: (GWAS upload results only) whether the study extraction is from an uploaded GWAS ("upload", with the upload's id in `gwas_upload_id`) or already in the GPMap ("existing"). Upload and existing study extractions have separate ids, so identify a study extraction by `source` and `id` together.

rare_results_dataframe

The rare_results dataframe contains information about which studies have coloc results. It has the following columns:

- rare_result_group_id: the unique id for this rare result group
- study_id: the id of the study associated with this rare result
- study_extraction_id: the id of the study extraction associated with this rare result
- variant_id: the id of the SNP
- ld_block_id: the id of the LD block
- chr: the chromosome of the SNP
- bp: the base pair position of the SNP
- min_p: the minimum p-value related to the study_extraction_id
- display_snp: the display SNP name
- gene: the gene associated with the SNP
- gene_id: the id of the gene

- trait_id: the id of the trait
- trait_name: the name of the trait
- trait_category: the category of the trait
- data_type: the data type of the trait
- tissue: the tissue of the trait
- ld_block: the LD block of the SNP

summary_statistics_dataframe

The summary_statistics dataframe contains information about which studies have summary statistics. From the API, column names are typically uppercase (SNP, CHR, BP, EA, OA, EAF, Z, BETA, SE, P, LBF_1, etc.). It has the following columns (names may be upper or lower case depending on source):

- SNP / variant_id: the id of the SNP
- CHR / chr: the chromosome of the SNP
- BP / bp: the base pair position of the SNP
- EA / ea: the effect allele
- OA / oa: the other allele
- EAF / eaf: the estimated allele frequency
- Z / z: the z-score
- BETA / beta: the beta value
- SE / se: the standard error
- P / p: the p-value
- imputed: whether the summary statistics are imputed
- LBF_* / lbf_*: all different finemapped log-bayes factors for each credible set. Each credible set is numbered from 1 to 10. If finemapped failed or only returned 1 credible set, the LBF_1 column is just converted directly from the z-score.

coloc_pairs_dataframe

The coloc_pairs dataframe contains information about which studies have coloc pairs. It has the following columns:

- study_extraction_a_id: the id of the study extraction associated with this coloc pair
- study_extraction_b_id: the id of the study extraction associated with this coloc pair
- source_a, source_b: (GWAS upload results only) whether each study extraction is from an uploaded GWAS ("upload") or already in the GPMap ("existing")
- trait_name_a, data_type_a, tissue_a, gene_a: the trait name, data type, tissue and gene of study extraction a (and the same with _b for study extraction b), joined from `study_extractions`. Not added to the coloc pairs of `trait()`, whose endpoint doesn't return the study extractions its pairs refer to.
- ld_block_id: the id of the LD block
- h3: the h3 value for this coloc pair
- h4: the h4 value for this coloc pair
- spurious: whether this coloc pair is spurious

Examples

```
result <- variants(c(5553693, 5553694))
result$variants
```

Index

all_genes, 2
all_traits, 3
associations, 4
associations(), *5*, *30*

compare_upload, 5
compare_upload(), *17*

delete_gwas, 6

gene, 7
genes, 11
get_all_gene_pleiotropies, 15
get_all_variant_pleiotropies, 16
get_gwas, 16
get_gwas(), *5*, *26*
gpmap_available, 17

health_api, 18

ld_matrix, 18
ld_proxies, 19

region, 20

search_gpmap, 24

trait, 26
trait(), *5*
trait_duplicates, 30
traits, 31

upload_gwas, 35
upload_gwas(), *5*

variant, 37
variants, 41