Package 'SNPannotator'

May 7, 2026

Type Package

Title Automated Functional Annotation of Genetic Variants and Linked Proxies

Version 1.4.6

Date 2026-01-06

Author Alireza Ani [aut, cre],
Zoha Kamali [aut],
Ahmad Vaez [aut]

Maintainer Alireza Ani <a.ani@umcg.nl>

Depends R (>= 4.0)

Imports data.table (>= 1.15.4), httr (>= 1.4.7), jsonlite (>= 1.8.8), xml2 (>= 1.3.6), openxlsx (>= 4.2.5.2), progress (>= 1.2.3), ggplot2 (>= 3.5.1), kableExtra (>= 1.4.0), methods (>= 4.2.0), rmarkdown (>= 2.26), ini (>= 0.3.1), igraph (>= 2.0.3), png, ggraph (>= 2.2.1), logger (>= 0.4.0), readr (>= 2.1.5)

Description To automated functional annotation of genetic variants and linked proxies. Linked SNPs in moderate to high linkage disequilibrium (e.g. r2>0.50) with the corresponding index SNPs will be selected for further analysis.

URL https://cran.r-project.org/package=SNPannotator

Encoding UTF-8

RoxygenNote 7.3.2

Suggests knitr

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2026-01-08 13:10:02 UTC

Contents

`annotate_shiny` *Run the annotation pipeline on a list of variants from shiny app*

Description

This function should not be used outside shiny app A list of variants and parameters are received and their information is checked on various API servers.

Usage

```
annotate_shiny(config.list)
```

Arguments

`config.list`	List. A list of variants andconfiguration parameters.

Value

a data table with all variant information is returned.

demo_annotation *Demo run of the annotation pipeline*

Description

This function is a demo of the annotation algorithm.

Usage

```
demo_annotation()
```

Value

A data table containing the variant information for testing is returned. Report files are also saved in the current working directory.

EnsemblDatabases *List population from human database (1000 Genomes project)*

Description

This function list the name, description and size of the available populations in 1000 Genomes project database. This database will be used for returning variables in high LD with the target SNP.

Usage

```
EnsemblDatabases(build = 38)
```

Arguments

build	Genome build. Either 37 or 38. default: 38

Value

A data table is returned which includes the name, description and size of the available populations in 1000 Genomes project database.

`EnsemblReleases` *data release available on this REST server.*

Description

Shows the data releases available on this REST server. May return more than one release (infrequent non-standard Ensembl configuration).

Usage

```
EnsemblReleases(build = 38)
```

Arguments

`build`	Genome build. Either 37 or 38. default: 38

Value

a message is displayed to the user

`findGenomicPos` *Query GTEx portal for Variant's genomic position based on rsID Retrieves variant information from the GTEx portal using either an rsID or a variant ID formatted as* `CHR_POS_REF_ALT`. *If an rsID is provided, the function returns the corresponding genomic positions in both GRCh37 and GRCh38 builds. When searching for an rsID based on genomic position, the position parameter should be specified according to the GRCh38 reference genome.*

Description

Query GTEx portal for Variant's genomic position based on rsID Retrieves variant information from the GTEx portal using either an rsID or a variant ID formatted as `CHR_POS_REF_ALT`. If an rsID is provided, the function returns the corresponding genomic positions in both GRCh37 and GRCh38 builds. When searching for an rsID based on genomic position, the position parameter should be specified according to the GRCh38 reference genome.

Usage

```
findGenomicPos(id, type = "rsid", file_path = NULL)
```

Arguments

`id`	Character string representing the rsID (e.g., `"rs12345"`) or the variant ID in the format `"CHR_POS_REF_ALT"` (e.g., `"1_1234567_A_T"`), depending on `type`.
`type`	Character string specifying the type of query. Must be either `"rsid"` or `"varid"`.
`file_path`	character, path to a file for saving results as Excell spreadsheet.

Value

A `data.table` containing variant information including:

- `rsid`: variant id in rsID format
- `chromosome`: chromosome number
- `position_b37`: genomic position
- `position_b38`: genomic position
- `ref`: reference allele
- `alt`: alternate allele

`findPairwiseLD` *Computes and returns LD values between the given variants.*

Description

This function returns a data frame of LD values between the given variants in a selected population.

Usage

```
findPairwiseLD(
  rsList,
  file = NULL,
  pairwise = FALSE,
  build = 38,
  db = "1000GENOMES:phase_3:EUR",
  r2 = 0.1
)
```

Arguments

`rsList`	A vector of rs numbers.
`file`	Path to the Excel file for saving search results.
`pairwise`	If TRUE, compute pairwise LD between all elements of a list. If FALSE, computes the LD between first and other elements of the list. default: FALSE
`build`	Genome build. Either 37 or 38. default: 38
`db`	The population database for calculating LD scores. This can be found using `Ensembl.Databases()` function. default: "1000GENOMES:phase_3:EUR"
`r2`	Only return pairs of variants whose r-squared value is equal to or greater than the value provided. default: 0.1.

Value

A data table with variant information.

findProxy *Finds variants in high LD*

Description

This function returns a list of variables that are in high LD with a list of selected variants using data from the Ensembl website.

Usage

```
findProxy(
  rslist,
  file = NULL,
  build = "38",
  db = "1000GENOMES:phase_3:EUR",
  window_size = 500,
  r2 = 0.8
)
```

Arguments

rslist	A vector of rs numbers.
file	Path to the Excel file for saving search results.
build	Genome build. Either 37 or 38. default: 38
db	The population database for calculating LD scores. This can be found using `Ensembl.Databases()` function. default: 1000GENOMES:phase_3:EUR
window_size	Number of base pairs around the variant for checking LD scores (max = 500kb). default: 500
r2	The minimum LD threshold for selecting variants around the target SNP. default: 0.8.

Value

A data table with variant information.

findRSID *Query Ensembl for variant information based on genomic position*

Description

This function retrieves variant information from Ensembl based on the specified genomic position. It takes the chromosome number, start position, and end position as input parameters and searches for variants within this window, using the specified genomic build. If only the start position is provided, the function automatically sets the end position equal to the start position. This is particularly relevant for SNP variants, where the start and end positions are the same. The function returns all variants found within the defined window.

Usage

```
findRSID(
  chromosome,
  start_position,
  end_position = NULL,
  build = "38",
  file_path = NULL
)
```

Arguments

chromosome	Numeric, specifying the chromosome number.
start_position	Numeric, specifying the starting base pair position.
end_position	Numeric, specifying the ending base pair position.
build	Numeric, specifying the genomic build, default value is 38.
file_path	character, path to a file for saving results as Excell spreadsheet.

Value

A `data.table` containing variant information including:

- `id`: variant id in rsID format
- `alleles`: variant alleles
- `seq_region_name`: chromosome number
- `start`: starting base pair
- `end`: ending base pair

`getConfigFile` *Copy a sample configuration file*

Description

This function provides a sample configuration file. The user can modify the parameters as desired

Usage

```
getConfigFile(dir.path)
```

Arguments

dir.path	The existing folder for copying the file.

mergeResultFiles *Merge multiple output files*

Description

This function merges multiple result files into one.

Usage

```
mergeResultFiles(..., fileName)
```

Arguments

...	list of input files to be merged.
fileName	name of the output file.

Value

A data table is returned.

pingEnsembl *Checks if the service is alive*

Description

This function test whether the Ensembl server is accessible or not

Usage

```
pingEnsembl(server)
```

Arguments

server	name of the server. "https://rest.ensembl.org" can be used for GRCh38 and "https://grch37.rest.ensembl.org" for GRCh37.

Value

a message is displayed to the user

`run_annotation` *Run the annotation pipeline on a list of variants*

Description

This function receives the path to the configuration file. A list of variants is received and their information is checked on various API servers.

Usage

```
run_annotation(configurationFilePath, verbose = TRUE)
```

Arguments

`configurationFilePath`	Character. The path to the configuration file.
`verbose`	Logical. Whether to display messages in the console.

Value

A data table containing all variant information is returned based on the user's selected specifications and parameters. Report files in various formats, including text, HTML, Excel, and image, are saved in the output folder.

`run_stringdb_annotation`
Analyze STRING DB Interactions and perform functional enrichment

Description

This function takes a vector of gene symbols, retrieves their interaction partners from STRING DB, and performs functional enrichment analysis.

Usage

```
run_stringdb_annotation(name, gene_list, required_score = 700, limit = 0, ...)
```

Arguments

`name`	A character string specifying a unique identifier for this analysis run.
`gene_list`	A character vector of gene symbols (e.g., HGNC symbols or Ensembl gene IDs).
`required_score`	Threshold of significance to include an interaction, a number between 0 and 1000.
`limit`	Limits the number of interaction partners retrieved per protein, a number between 0 and 100.
`...`	Additional arguments passed to downstream functions for extended customization.

Value

set of report files, including images, text and excel files containing functional enrichment analysis results.

Index