Package 'STAARpipeline'

October 31, 2023

Type Package

Title STAARpipeline for Analyzing Whole-Genome/Whole-Exome Sequencing Data

Version 0.9.7

Date 2023-10-31

Author Xihao Li [aut, cre], Zilin Li [aut, cre], Sheila M. Gaynor [aut], Han Chen [aut]

Maintainer Xihao Li <xihaoli@unc.edu>, Zilin Li <lizl@nenu.edu.cn>

Description An R package for performing STAARpipeline in analyzing whole-genome/whole-exome sequencing data.

License GPL-3

Copyright See COPYRIGHTS for details.

Imports Rcpp, STAAR, MultiSTAAR, SCANG, dplyr, SeqArray, SeqVarTools, GenomicFeatures, TxDb.Hsapiens.UCSC.hg38.knownGene, GMMAT, GENESIS, Matrix, methods

Encoding UTF-8

LazyData true

Depends R (>= 3.2.0)

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 7.2.3

Suggests knitr, rmarkdown

VignetteBuilder knitr

R topics documented:

`Dynamic_Window_SCANG` *Genetic region analysis of dynamic windows using SCANG-STAAR procedure*

Description

The `Dynamic_Window_SCANG` function takes in chromosome, starting location, ending location, the object of opened annotated GDS file, and the object from fitting the null model to analyze the association between a quantitative/dichotomous phenotype and variants in a genetic region by using SCANG-STAAR procedure. For each dynamic window, the scan statistic of SCANG-STAAR-O is the set-based p-value of an omnibus test that aggregated p-values across different types of multiple annotation-weighted variant-set tests SKAT(1,1), SKAT(1,25), Burden(1,1) and Burden(1,25) using ACAT method; the scan statistic of SCANG-STAAR-S is the set-based p-value of STAAR-S, which is an omnibus test that aggregated p-values across multiple annotation-weighted variant-set tests SKAT(1,1) and SKAT(1,25) using ACAT method; the scan statistic of SCANG-STAAR-B is the set-based p-value of STAAR-B, which is an omnibus test that aggregated p-values across multiple annotation-weighted variant-set tests Burden(1,1) and Burden(1,25) using ACAT method.

Usage

```
Dynamic_Window_SCANG(
  chr,
  start_loc,
  end_loc,
  genofile,
  obj_nullmodel,
  Lmin = 40,
  Lmax = 300,
  steplength = 10,
  rare_maf_cutoff = 0.01,
  p_filter = 1e-08,
  f = 0,
  alpha = 0.1,
  QC_label = "annotation/filter",
  variant_type = c("SNV", "Indel", "variant"),
  geno_missing_imputation = c("mean", "minor"),
  Annotation_dir = "annotation/info/FunctionalAnnotation",
  Annotation_name_catalog,
  Use_annotation_weights = c(TRUE, FALSE),
  Annotation_name = NULL,
  silent = FALSE
)
```

Arguments

`chr`	chromosome.
`start_loc`	starting location (position) of the genetic region to be analyzed using SCANG-STAAR procedure.
`end_loc`	ending location (position) of the genetic region to be analyzed using SCANG-STAAR procedure.
`genofile`	an object of opened annotated GDS (aGDS) file.
`obj_nullmodel`	an object from fitting the null model, which is the output from `fit_nullmodel` function and transformed using the `staar2scang_nullmodel` function.
`Lmin`	minimum number of variants in searching windows (default = 40).
`Lmax`	maximum number of variants in searching windows (default = 300).
`steplength`	difference of number of variants in searching windows, that is, the number of variants in searching windows are Lmin, Lmin+steplength, Lmin+steplength,..., Lmax (default = 10).
`rare_maf_cutoff`	a cutoff of maximum minor allele frequency in defining rare variants (default = 0.01).
`p_filter`	a filtering threshold of screening method for SKAT in SCANG-STAAR. SKAT p-values are calculated for regions whose p-value is possibly smaller than the filtering threshold (default = 1e-8).
`f`	an overlap fraction, which controls for the overlapping proportion of of detected regions. For example, when f=0, the detected regions are non-overlapped with each other, and when f=1, we keep every susceptive region as detected regions (default = 0).
`alpha`	family-wise/genome-wide significance level (default = 0.1).
`QC_label`	channel name of the QC label in the GDS/aGDS file (default = "annotation/filter").
`variant_type`	type of variant included in the analysis. Choices include "SNV", "Indel", or "variant" (default = "SNV").
`geno_missing_imputation`	method of handling missing genotypes. Either "mean" or "minor" (default = "mean").
`Annotation_dir`	channel name of the annotations in the aGDS file (default = "annotation/info/FunctionalAnnotation").
`Annotation_name_catalog`	a data frame containing the name and the corresponding channel name in the aGDS file.
`Use_annotation_weights`	use annotations as weights or not (default = TRUE).
`Annotation_name`	a vector of annotation names used in SCANG-STAAR (default = NULL).
`silent`	logical: should the report of error messages be suppressed (default = FALSE).

Value

The function returns a list with the following members:

`SCANG_O_res`: A matrix that summarizes the significant region detected by SCANG-STAAR-O, including the negative log transformation of SCANG-STAAR-O p-value ("-logp"), chromosome

("chr"), start position ("start_pos"), end position ("end_pos"), family-wise/genome-wide error rate (GWER) and the number of variants ("SNV_num").

`SCANG_O_top1`: A vector of length 4 which summarizes the top 1 region detected by SCANG-STAAR-O. including the negative log transformation of SCANG-STAAR-O p-value ("-logp"), chromosome ("chr"), start position ("start_pos"), end position ("end_pos"), family-wise/genome-wide error rate (GWER) and the number of variants ("SNV_num").

`SCANG_O_emthr`: A vector of Monte Carlo simulation sample for generating the empirical threshold. The 1-alpha quantile of this vector is the empirical threshold.

`SCANG_S_res, SCANG_S_top1, SCANG_S_emthr`: Analysis results using SCANG-STAAR-S. Details see SCANG-STAAR-O.

`SCANG_B_res, SCANG_B_top1, SCANG_B_emthr`: Analysis results using SCANG-STAAR-B. Details see SCANG-STAAR-O.

References

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

Li, Z., Li, X., et al. (2019). Dynamic scan procedure for detecting rare-variant association regions in whole-genome sequencing studies. *The American Journal of Human Genetics*, *104*(5), 802-814. (pub)

Li, X., Li, Z., et al. (2020). Dynamic incorporation of multiple in silico functional annotations empowers rare variant association analysis of large whole-genome sequencing studies at scale. *Nature Genetics*, *52*(9), 969-983. (pub)

Liu, Y., et al. (2019). Acat: A fast and powerful p value combination method for rare-variant analysis in sequencing studies. *The American Journal of Human Genetics*, *104*(3), 410-421. (pub)

`fit_nullmodel` *Fitting generalized linear mixed model with known relationship matrices under the null hypothesis.*

Description

The `fit_nullmodel` function is a wrapper of the `glmmkin` function from the `GMMAT` package that fits a regression model under the null hypothesis, which provides the preliminary step for subsequent variant-set tests in whole-genome sequencing data analysis. See `glmmkin` for more details.

Usage

```
fit_nullmodel(
  fixed,
  data = parent.frame(),
  kins,
  use_sparse = NULL,
  kins_cutoff = 0.022,
  id,
  random.slope = NULL,
  groups = NULL,
  family = binomial(link = "logit"),
  method = "REML",
```

```
  method.optim = "AI",
  maxiter = 500,
  tol = 1e-05,
  taumin = 1e-05,
  taumax = 1e+05,
  tauregion = 10,
  verbose = FALSE,
  ...
)
```

Arguments

fixed	an object of class formula (or one that can be coerced to that class): a symbolic description of the fixed effects model to be fitted. For multiple phenotype analysis, formula recognized by lm, such as cbind(y1,y2,y3) ~ x1 + x2, can be used in fixed as fixed effects.
data	a data frame or list (or object coercible by as.data.frame to a data frame) containing the variables in the model.
kins	a known positive semi-definite relationship matrix (e.g. kinship matrix in genetic association studies) or a list of known positive semi-definite relationship matrices. The rownames and colnames of these matrices must at least include all samples as specified in the id column of the data frame data. If kins is NULL, fit_nullmodel will switch to the generalized linear model with no random effects.
use_sparse	a logical switch of whether the provided dense kins matrix should be transformed to a sparse matrix (default = NULL).
kins_cutoff	the cutoff value for clustering samples to make the output matrix sparse block-diagonal (default = 0.022).
id	a column in the data frame data, indicating the id of samples. When there are duplicates in id, the data is assumed to be longitudinal with repeated measures.
random.slope	an optional column indicating the random slope for time effect used in a mixed effects model for longitudinal data. It must be included in the names of data. There must be duplicates in id and method.optim must be "AI" (default = NULL).
groups	an optional categorical variable indicating the groups used in a heteroscedastic linear mixed model (allowing residual variances in different groups to be different). This variable must be included in the names of data, and family must be "gaussian" and method.optim must be "AI" (default = NULL).
family	a description of the error distribution and link function to be used in the model. This can be a character string naming a family function, a family function or the result of a call to a family function. (See family for details of family functions).
method	method of fitting the generalized linear mixed model. Either "REML" or "ML" (default = "REML").
method.optim	optimization method of fitting the generalized linear mixed model. Either "AI", "Brent" or "Nelder-Mead" (default = "AI").
maxiter	a positive integer specifying the maximum number of iterations when fitting the generalized linear mixed model (default = 500).
tol	a positive number specifying tolerance, the difference threshold for parameter estimates below which iterations should be stopped (default = 1e-5).

taumin	the lower bound of search space for the variance component parameter τ (default = 1e-5), used when method.optim = "Brent". See Details.
taumax	the upper bound of search space for the variance component parameter τ (default = 1e5), used when method.optim = "Brent". See Details.
tauregion	the number of search intervals for the REML or ML estimate of the variance component parameter τ (default = 10), used when method.optim = "Brent". See Details.
verbose	a logical switch for printing detailed information (parameter estimates in each iteration) for testing and debugging purpose (default = FALSE).
...	additional arguments that could be passed to glm.

Value

The function returns an object of the model fit from glmmkin (obj_nullmodel) and whether the kins matrix is sparse when fitting the null model. See glmmkin for more details.

References

Chen, H., et al. (2016). Control for population structure and relatedness for binary traits in genetic association studies via logistic mixed models. *The American Journal of Human Genetics*, *98*(4), 653-666. (pub)

Chen, H., et al. (2019). Efficient variant set mixed model association tests for continuous and binary traits in large-scale whole-genome sequencing studies. *The American Journal of Human Genetics*, *104*(2), 260-274. (pub)

Chen, H. (2021). GMMAT: Generalized linear Mixed Model Association Tests Version 1.3.2. (web)

---

genesis2staar_nullmodel

Transforming the null model object fitted using GENESIS to the null model object to be used for STAAR

Description

The genesis2staar_nullmodel function takes in the object from fitting the null model using the GENESIS package and transforms it to the object from fitting the null model to be used for STAAR procedure.

Usage

```
genesis2staar_nullmodel(obj_nullmodel_genesis)
```

Arguments

obj_nullmodel_genesis	an object from fitting the null model, which is the output from fitNullModel function in the GENESIS package.

Value

an object from fitting the null model for related samples to be used for STAAR procedure, which is the output from fit_nullmodel function.

References

Gogarten, S.M., Sofer, T., Chen, H., et al. (2019). Genetic association testing using the GENESIS R/Bioconductor package. *Bioinformatics*, *35*(24), 5346-5348. (pub)

Li, X., Li, Z., et al. (2020). Dynamic incorporation of multiple in silico functional annotations empowers rare variant association analysis of large whole-genome sequencing studies at scale. *Nature Genetics*, *52*(9), 969-983. (pub)

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

`Gene_Centric_Coding` *Gene-centric analysis of coding functional categories using STAAR procedure*

Description

The `Gene_Centric_Coding` function takes in chromosome, gene name, functional category, the object of opened annotated GDS file, and the object from fitting the null model to analyze the association between a quantitative/dichotomous phenotype and coding functional categories of a gene by using STAAR procedure. For each coding functional category, the STAAR-O p-value is a p-value from an omnibus test that aggregated SKAT(1,25), SKAT(1,1), Burden(1,25), Burden(1,1), ACAT-V(1,25), and ACAT-V(1,1) together with p-values of each test weighted by each annotation using Cauchy method. For multiple phenotype analysis (`obj_nullmodel$n.pheno > 1`), the results correspond to multi-trait association p-values (e.g. MultiSTAAR-O) by leveraging the correlation structure between multiple phenotypes.

Usage

```
Gene_Centric_Coding(
  chr,
  gene_name,
 category = c("all_categories", "plof", "plof_ds", "missense", "disruptive_missense",
    "synonymous"),
  genofile,
  obj_nullmodel,
  rare_maf_cutoff = 0.01,
  rv_num_cutoff = 2,
  QC_label = "annotation/filter",
  variant_type = c("SNV", "Indel", "variant"),
  geno_missing_imputation = c("mean", "minor"),
  Annotation_dir = "annotation/info/FunctionalAnnotation",
  Annotation_name_catalog,
  Use_annotation_weights = c(TRUE, FALSE),
  Annotation_name = NULL,
  silent = FALSE
)
```

Arguments

`chr`	chromosome.
`gene_name`	name of the gene to be analyzed using STAAR procedure.

`category` the coding functional category to be analyzed using STAAR procedure. Choices include `all_categories`, `plof`, `plof_ds`, `missense`, `disruptive_missense`, `synonymous` (default = `all_categories`).

`genofile` an object of opened annotated GDS (aGDS) file.

`obj_nullmodel` an object from fitting the null model, which is either the output from `fit_nullmodel` function, or the output from `fitNullModel` function in the `GENESIS` package and transformed using the `genesis2staar_nullmodel` function.

`rare_maf_cutoff` the cutoff of maximum minor allele frequency in defining rare variants (default = 0.01).

`rv_num_cutoff` the cutoff of minimum number of variants of analyzing a given variant-set (default = 2).

`QC_label` channel name of the QC label in the GDS/aGDS file (default = "annotation/filter").

`variant_type` type of variant included in the analysis. Choices include "SNV", "Indel", or "variant" (default = "SNV").

`geno_missing_imputation` method of handling missing genotypes. Either "mean" or "minor" (default = "mean").

`Annotation_dir` channel name of the annotations in the aGDS file (default = "annotation/info/FunctionalAnnotation").

`Annotation_name_catalog` a data frame containing the name and the corresponding channel name in the aGDS file.

`Use_annotation_weights` use annotations as weights or not (default = TRUE).

`Annotation_name` a vector of annotation names used in STAAR (default = NULL).

`silent` logical: should the report of error messages be suppressed (default = FALSE).

Value

a list of data frames containing the STAAR p-values (including STAAR-O) corresponding to the coding functional category of the given gene.

References

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

Li, X., Li, Z., et al. (2020). Dynamic incorporation of multiple in silico functional annotations empowers rare variant association analysis of large whole-genome sequencing studies at scale. *Nature Genetics*, *52*(9), 969-983. (pub)

`Gene_Centric_Coding_cond` *Gene-centric conditional analysis of coding functional categories using STAAR procedure*

Description

The `Gene_Centric_Coding_cond` function takes in chromosome, gene name, functional category, the object of opened annotated GDS file, the object from fitting the null model, and the set of known variants to be adjusted for in conditional analysis to analyze the conditional association between a quantitative/dichotomous phenotype and coding functional categories of a gene by using STAAR procedure. For each coding functional category, the conditional STAAR-O p-value is a p-value from an omnibus test that aggregated conditional SKAT(1,25), SKAT(1,1), Burden(1,25), Burden(1,1), ACAT-V(1,25), and ACAT-V(1,1) together with conditional p-values of each test weighted by each annotation using Cauchy method. For multiple phenotype analysis (`obj_nullmodel$n.pheno > 1`), the results correspond to multi-trait conditional p-values (e.g. conditional MultiSTAAR-O) by leveraging the correlation structure between multiple phenotypes.

Usage

```
Gene_Centric_Coding_cond(
  chr,
  gene_name,
 category = c("plof", "plof_ds", "missense", "disruptive_missense", "synonymous"),
  genofile,
  obj_nullmodel,
  known_loci = NULL,
  rare_maf_cutoff = 0.01,
  rv_num_cutoff = 2,
  method_cond = c("optimal", "naive"),
  QC_label = "annotation/filter",
  variant_type = c("SNV", "Indel", "variant"),
  geno_missing_imputation = c("mean", "minor"),
  Annotation_dir = "annotation/info/FunctionalAnnotation",
  Annotation_name_catalog,
  Use_annotation_weights = c(TRUE, FALSE),
  Annotation_name = NULL
)
```

Arguments

`chr`	chromosome.
`gene_name`	name of the gene to be analyzed using STAAR procedure.
`category`	the coding functional category to be analyzed using STAAR procedure. Choices include `plof`, `plof_ds`, `missense`, `disruptive_missense`, `synonymous` (default = `plof`).
`genofile`	an object of opened annotated GDS (aGDS) file.
`obj_nullmodel`	an object from fitting the null model, which is either the output from `fit_nullmodel` function, or the output from `fitNullModel` function in the `GENESIS` package and transformed using the `genesis2staar_nullmodel` function.

`known_loci` the data frame of variants to be adjusted for in conditional analysis and should contain 4 columns in the following order: chromosome (CHR), position (POS), reference allele (REF), and alternative allele (ALT) (default = NULL).

`rare_maf_cutoff` the cutoff of maximum minor allele frequency in defining rare variants (default = 0.01).

`rv_num_cutoff` the cutoff of minimum number of variants of analyzing a given variant-set (default = 2).

`method_cond` a character value indicating the method for conditional analysis. `optimal` refers to regressing residuals from the null model on `known_loci` as well as all covariates used in fitting the null model (fully adjusted) and taking the residuals; `naive` refers to regressing residuals from the null model on `known_loci` and taking the residuals (default = `optimal`).

`QC_label` channel name of the QC label in the GDS/aGDS file (default = "annotation/filter").

`variant_type` type of variant included in the analysis. Choices include "SNV", "Indel", or "variant" (default = "SNV").

`geno_missing_imputation` method of handling missing genotypes. Either "mean" or "minor" (default = "mean").

`Annotation_dir` channel name of the annotations in the aGDS file (default = "annotation/info/FunctionalAnnotation").

`Annotation_name_catalog` a data frame containing the name and the corresponding channel name in the aGDS file.

`Use_annotation_weights` use annotations as weights or not (default = TRUE).

`Annotation_name` a vector of annotation names used in STAAR (default = NULL).

Value

a data frame containing the conditional STAAR p-values (including STAAR-O) corresponding to each coding functional category of the given gene.

References

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

Li, X., Li, Z., et al. (2020). Dynamic incorporation of multiple in silico functional annotations empowers rare variant association analysis of large whole-genome sequencing studies at scale. *Nature Genetics*, *52*(9), 969-983. (pub)

Sofer, T., et al. (2019). A fully adjusted two-stage procedure for rank-normalization in genetic association studies. *Genetic Epidemiology*, *43*(3), 263-275. (pub)

`Gene_Centric_Noncoding` *Gene-centric analysis of noncoding functional categories using STAAR procedure*

Description

The `Gene_Centric_Noncoding` function takes in chromosome, gene name, functional category, the object of opened annotated GDS file, and the object from fitting the null model to analyze the association between a quantitative/dichotomous phenotype and noncoding functional categories of a gene by using STAAR procedure. For each noncoding functional category, the STAAR-O p-value is a p-value from an omnibus test that aggregated SKAT(1,25), SKAT(1,1), Burden(1,25), Burden(1,1), ACAT-V(1,25), and ACAT-V(1,1) together with p-values of each test weighted by each annotation using Cauchy method. For multiple phenotype analysis (`obj_nullmodel$n.pheno > 1`), the results correspond to multi-trait association p-values (e.g. MultiSTAAR-O) by leveraging the correlation structure between multiple phenotypes.

Usage

```
Gene_Centric_Noncoding(
  chr,
  gene_name,
 category = c("all_categories", "downstream", "upstream", "UTR", "promoter_CAGE",
    "promoter_DHS", "enhancer_CAGE", "enhancer_DHS"),
  genofile,
  obj_nullmodel,
  rare_maf_cutoff = 0.01,
  rv_num_cutoff = 2,
  QC_label = "annotation/filter",
  variant_type = c("SNV", "Indel", "variant"),
  geno_missing_imputation = c("mean", "minor"),
  Annotation_dir = "annotation/info/FunctionalAnnotation",
  Annotation_name_catalog,
  Use_annotation_weights = c(TRUE, FALSE),
  Annotation_name = NULL,
  silent = FALSE
)
```

Arguments

`chr`	chromosome.
`gene_name`	name of the gene to be analyzed using STAAR procedure.
`category`	the noncoding functional category to be analyzed using STAAR procedure. Choices include `all_categories`, `downstream`, `upstream`, `UTR`, `promoter_CAGE`, `promoter_DHS`, `enhancer_CAGE`, `enhancer_DHS` (default = `all_categories`).
`genofile`	an object of opened annotated GDS (aGDS) file.
`obj_nullmodel`	an object from fitting the null model, which is either the output from `fit_nullmodel` function, or the output from `fitNullModel` function in the `GENESIS` package and transformed using the `genesis2staar_nullmodel` function.

`rare_maf_cutoff`
: the cutoff of maximum minor allele frequency in defining rare variants (default = 0.01).

`rv_num_cutoff`
: the cutoff of minimum number of variants of analyzing a given variant-set (default = 2).

`QC_label`
: channel name of the QC label in the GDS/aGDS file (default = "annotation/filter").

`variant_type`
: type of variant included in the analysis. Choices include "SNV", "Indel", or "variant" (default = "SNV").

`geno_missing_imputation`
: method of handling missing genotypes. Either "mean" or "minor" (default = "mean").

`Annotation_dir`
: channel name of the annotations in the aGDS file (default = "annotation/info/FunctionalAnnotation").

`Annotation_name_catalog`
: a data frame containing the name and the corresponding channel name in the aGDS file.

`Use_annotation_weights`
: use annotations as weights or not (default = TRUE).

`Annotation_name`
: a vector of annotation names used in STAAR (default = NULL).

`silent`
: logical: should the report of error messages be suppressed (default = FALSE).

Value

a list of data frames containing the STAAR p-values (including STAAR-O) corresponding to each noncoding functional category of the given gene.

References

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

Li, X., Li, Z., et al. (2020). Dynamic incorporation of multiple in silico functional annotations empowers rare variant association analysis of large whole-genome sequencing studies at scale. *Nature Genetics*, *52*(9), 969-983. (pub)

`Gene_Centric_Noncoding_cond`
: *Gene-centric conditional analysis of noncoding functional categories using STAAR procedure*

Description

The `Gene_Centric_Noncoding_cond` function takes in chromosome, gene name, functional category, the object of opened annotated GDS file, the object from fitting the null model, and the set of known variants to be adjusted for in conditional analysis to analyze the conditional association between a quantitative/dichotomous phenotype and noncoding functional categories of a gene by using STAAR procedure. For each noncoding functional category, the conditional STAAR-O p-value is a p-value from an omnibus test that aggregated conditional SKAT(1,25), SKAT(1,1), Burden(1,25), Burden(1,1), ACAT-V(1,25), and ACAT-V(1,1) together with conditional p-values

of each test weighted by each annotation using Cauchy method. For multiple phenotype analysis (`obj_nullmodel$n.pheno > 1`), the results correspond to multi-trait conditional p-values (e.g. conditional MultiSTAAR-O) by leveraging the correlation structure between multiple phenotypes.

Usage

```
Gene_Centric_Noncoding_cond(
  chr,
  gene_name,
  category = c("downstream", "upstream", "UTR", "promoter_CAGE", "promoter_DHS",
    "enhancer_CAGE", "enhancer_DHS"),
  genofile,
  obj_nullmodel,
  known_loci = NULL,
  rare_maf_cutoff = 0.01,
  rv_num_cutoff = 2,
  method_cond = c("optimal", "naive"),
  QC_label = "annotation/filter",
  variant_type = c("SNV", "Indel", "variant"),
  geno_missing_imputation = c("mean", "minor"),
  Annotation_dir = "annotation/info/FunctionalAnnotation",
  Annotation_name_catalog,
  Use_annotation_weights = c(TRUE, FALSE),
  Annotation_name = NULL
)
```

Arguments

`chr`	chromosome.
`gene_name`	name of the gene to be analyzed using STAAR procedure.
`category`	the noncoding functional category to be analyzed using STAAR procedure. Choices include `downstream`, `upstream`, `UTR`, `promoter_CAGE`, `promoter_DHS`, `enhancer_CAGE`, `enhancer_DHS` (default = `downstream`).
`genofile`	an object of opened annotated GDS (aGDS) file.
`obj_nullmodel`	an object from fitting the null model, which is either the output from `fit_nullmodel` function, or the output from `fitNullModel` function in the `GENESIS` package and transformed using the `genesis2staar_nullmodel` function.
`known_loci`	the data frame of variants to be adjusted for in conditional analysis and should contain 4 columns in the following order: chromosome (CHR), position (POS), reference allele (REF), and alternative allele (ALT) (default = NULL).
`rare_maf_cutoff`	the cutoff of maximum minor allele frequency in defining rare variants (default = 0.01).
`rv_num_cutoff`	the cutoff of minimum number of variants of analyzing a given variant-set (default = 2).
`method_cond`	a character value indicating the method for conditional analysis. `optimal` refers to regressing residuals from the null model on `known_loci` as well as all covariates used in fitting the null model (fully adjusted) and taking the residuals; `naive` refers to regressing residuals from the null model on `known_loci` and taking the residuals (default = `optimal`).

`QC_label` channel name of the QC label in the GDS/aGDS file (default = "annotation/filter").

`variant_type` type of variant included in the analysis. Choices include "SNV", "Indel", or "variant" (default = "SNV").

`geno_missing_imputation` method of handling missing genotypes. Either "mean" or "minor" (default = "mean").

`Annotation_dir` channel name of the annotations in the aGDS file (default = "annotation/info/FunctionalAnnotation").

`Annotation_name_catalog` a data frame containing the name and the corresponding channel name in the aGDS file.

`Use_annotation_weights` use annotations as weights or not (default = TRUE).

`Annotation_name` a vector of annotation names used in STAAR (default = NULL).

Value

a data frame containing the conditional STAAR p-values (including STAAR-O) corresponding to the noncoding functional category of the given gene.

References

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

Li, X., Li, Z., et al. (2020). Dynamic incorporation of multiple in silico functional annotations empowers rare variant association analysis of large whole-genome sequencing studies at scale. *Nature Genetics*, *52*(9), 969-983. (pub)

Sofer, T., et al. (2019). A fully adjusted two-stage procedure for rank-normalization in genetic association studies. *Genetic Epidemiology*, *43*(3), 263-275. (pub)

`Individual_Analysis` *Individual-variant analysis using score test*

Description

The `Individual_Analysis` function takes in chromosome, starting location, ending location, the object of opened annotated GDS file, and the object from fitting the null model to analyze the association between a quantitative/dichotomous phenotype and each individual variant in a genetic region by using score test. For multiple phenotype analysis (`obj_nullmodel$n.pheno > 1`), the results correspond to multi-trait score test p-values by leveraging the correlation structure between multiple phenotypes.

Usage

```
Individual_Analysis(
  chr,
  start_loc,
  end_loc,
```

```
  genofile,
  obj_nullmodel,
  mac_cutoff = 20,
  subset_variants_num = 5000,
  QC_label = "annotation/filter",
  variant_type = c("variant", "SNV", "Indel"),
  geno_missing_imputation = c("mean", "minor")
)
```

Arguments

chr	chromosome.
start_loc	starting location (position) of the genetic region for each individual variant to be analyzed using score test.
end_loc	ending location (position) of the genetic region for each individual variant to be analyzed using score test.
genofile	an object of opened annotated GDS (aGDS) file.
obj_nullmodel	an object from fitting the null model, which is either the output from fit_nullmodel function, or the output from fitNullModel function in the GENESIS package and transformed using the genesis2staar_nullmodel function.
mac_cutoff	the cutoff of minimum minor allele count in defining individual variants (default = 20).
subset_variants_num	the number of variants to run per subset for each time (default = 5e3).
QC_label	channel name of the QC label in the GDS/aGDS file (default = "annotation/filter").
variant_type	type of variant included in the analysis. Choices include "variant", "SNV", or "Indel" (default = "variant").
geno_missing_imputation	method of handling missing genotypes. Either "mean" or "minor" (default = "mean").

Value

a data frame containing the score test p-value and the estimated effect size of the minor allele for each individual variant in the given genetic region. The first 4 columns correspond to chromosome (CHR), position (POS), reference allele (REF), and alternative allele (ALT).

References

Chen, H., et al. (2016). Control for population structure and relatedness for binary traits in genetic association studies via logistic mixed models. *The American Journal of Human Genetics*, *98*(4), 653-666. (pub)

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

`Individual_Analysis_cond` *Individual-variant conditional analysis using score test*

Description

The `Individual_Analysis_cond` function takes in the data frame of individual variants, the object of opened annotated GDS file, the object from fitting the null model, and the set of known variants to be adjusted for in conditional analysis to analyze the conditional association between a quantitative/dichotomous phenotype and each (significant) individual variant by using score test. For multiple phenotype analysis (`obj_nullmodel$n.pheno > 1`), the results correspond to multi-trait conditional score test p-values by leveraging the correlation structure between multiple phenotypes.

Usage

```
Individual_Analysis_cond(
  chr,
  individual_results,
  genofile,
  obj_nullmodel,
  known_loci = NULL,
  method_cond = c("optimal", "naive"),
  QC_label = "annotation/filter",
  variant_type = c("variant", "SNV", "Indel"),
  geno_missing_imputation = c("mean", "minor"),
  geno_position_ascending = TRUE
)
```

Arguments

`chr`	chromosome.
`individual_results`	the data frame of (significant) individual variants for conditional analysis using score test. The first 4 columns should correspond to chromosome (CHR), position (POS), reference allele (REF), and alternative allele (ALT).
`genofile`	an object of opened annotated GDS (aGDS) file.
`obj_nullmodel`	an object from fitting the null model, which is either the output from `fit_nullmodel` function, or the output from `fitNullModel` function in the `GENESIS` package and transformed using the `genesis2staar_nullmodel` function.
`known_loci`	the data frame of variants to be adjusted for in conditional analysis and should contain 4 columns in the following order: chromosome (CHR), position (POS), reference allele (REF), and alternative allele (ALT) (default = NULL).
`method_cond`	a character value indicating the method for conditional analysis. `optimal` refers to regressing residuals from the null model on `known_loci` as well as all covariates used in fitting the null model (fully adjusted) and taking the residuals; `naive` refers to regressing residuals from the null model on `known_loci` and taking the residuals (default = `optimal`).
`QC_label`	channel name of the QC label in the GDS/aGDS file (default = "annotation/filter").

`variant_type` type of variant included in the analysis. Choices include "variant", "SNV", or "Indel" (default = "variant").

`geno_missing_imputation`
method of handling missing genotypes. Either "mean" or "minor" (default = "mean").

`geno_position_ascending`
logical: are the variant positions in ascending order in the GDS/aGDS file (default = TRUE).

Value

a data frame containing the conditional score test p-value and the estimated effect size of the minor allele for each (significant) individual variant in `individual_results`.

References

Chen, H., et al. (2016). Control for population structure and relatedness for binary traits in genetic association studies via logistic mixed models. *The American Journal of Human Genetics*, *98*(4), 653-666. (pub)

Sofer, T., et al. (2019). A fully adjusted two-stage procedure for rank-normalization in genetic association studies. *Genetic Epidemiology*, *43*(3), 263-275. (pub)

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

`LD_pruning` *Linkage disequilibrium (LD) pruning procedure*

Description

The `LD_pruning` function takes in chromosome, the object of opened annotated GDS file, the object from fitting the null model, and a given list of variants to perform LD pruning among these variants in sequential conditional analysis by using score test. For multiple phenotype analysis (`obj_nullmodel$n.pheno > 1`), the results correspond to multi-trait sequential conditional analysis by leveraging the correlation structure between multiple phenotypes.

Usage

```
LD_pruning(
  chr,
  genofile,
  obj_nullmodel,
  variants_list,
  maf_cutoff = 0.01,
  cond_p_thresh = 1e-04,
  method_cond = c("optimal", "naive"),
  QC_label = "annotation/filter",
  variant_type = c("variant", "SNV", "Indel"),
  geno_missing_imputation = c("mean", "minor"),
  geno_position_ascending = TRUE
)
```

Arguments

chr	chromosome.
genofile	an object of opened annotated GDS (aGDS) file.
obj_nullmodel	an object from fitting the null model, which is either the output from fit_nullmodel function, or the output from fitNullModel function in the GENESIS package and transformed using the genesis2staar_nullmodel function.
variants_list	the data frame of variants to be LD-pruned in sequential conditional analysis and should contain 4 columns in the following order: chromosome (CHR), position (POS), reference allele (REF), and alternative allele (ALT).
maf_cutoff	the cutoff of minimum minor allele frequency in defining individual variants to be LD-pruned (default = 0.01).
cond_p_thresh	the cutoff of maximum conditional p-value allowed for variants to be kept in the LD-pruned list of variants (default = 1e-04).
method_cond	a character value indicating the method for conditional analysis. optimal refers to regressing residuals from the null model on known_loci as well as all covariates used in fitting the null model (fully adjusted) and taking the residuals; naive refers to regressing residuals from the null model on known_loci and taking the residuals (default = optimal).
QC_label	channel name of the QC label in the GDS/aGDS file (default = "annotation/filter").
variant_type	type of variant included in the analysis. Choices include "variant", "SNV", or "Indel" (default = "variant").
geno_missing_imputation	method of handling missing genotypes. Either "mean" or "minor" (default = "mean").
geno_position_ascending	logical: are the variant positions in ascending order in the GDS/aGDS file (default = TRUE).

Value

a data frame containing the list of LD-pruned variants in the given chromosome.

References

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

---

ncRNA *Gene-centric analysis of long noncoding RNA (ncRNA) category using STAAR procedure*

Description

The ncRNA function takes in chromosome, gene name, the object of opened annotated GDS file, and the object from fitting the null model to analyze the association between a quantitative/dichotomous phenotype and the exonic and splicing category of an ncRNA gene by using STAAR procedure. For each ncRNA category, the STAAR-O p-value is a p-value from an omnibus test that aggregated SKAT(1,25), SKAT(1,1), Burden(1,25), Burden(1,1), ACAT-V(1,25), and ACAT-V(1,1) together

with p-values of each test weighted by each annotation using Cauchy method. For multiple phenotype analysis (`obj_nullmodel$n.pheno > 1`), the results correspond to multi-trait association p-values (e.g. MultiSTAAR-O) by leveraging the correlation structure between multiple phenotypes.

Usage

```
ncRNA(
  chr,
  gene_name,
  genofile,
  obj_nullmodel,
  rare_maf_cutoff = 0.01,
  rv_num_cutoff = 2,
  QC_label = "annotation/filter",
  variant_type = c("SNV", "Indel", "variant"),
  geno_missing_imputation = c("mean", "minor"),
  Annotation_dir = "annotation/info/FunctionalAnnotation",
  Annotation_name_catalog,
  Use_annotation_weights = c(TRUE, FALSE),
  Annotation_name = NULL,
  silent = FALSE
)
```

Arguments

`chr`	chromosome.
`gene_name`	name of the ncRNA gene to be analyzed using STAAR procedure.
`genofile`	an object of opened annotated GDS (aGDS) file.
`obj_nullmodel`	an object from fitting the null model, which is either the output from `fit_nullmodel` function, or the output from `fitNullModel` function in the `GENESIS` package and transformed using the `genesis2staar_nullmodel` function.
`rare_maf_cutoff`	the cutoff of maximum minor allele frequency in defining rare variants (default = 0.01).
`rv_num_cutoff`	the cutoff of minimum number of variants of analyzing a given variant-set (default = 2).
`QC_label`	channel name of the QC label in the GDS/aGDS file (default = "annotation/filter").
`variant_type`	type of variant included in the analysis. Choices include "SNV", "Indel", or "variant" (default = "SNV").
`geno_missing_imputation`	method of handling missing genotypes. Either "mean" or "minor" (default = "mean").
`Annotation_dir`	channel name of the annotations in the aGDS file (default = "annotation/info/FunctionalAnnotation").
`Annotation_name_catalog`	a data frame containing the name and the corresponding channel name in the aGDS file.
`Use_annotation_weights`	use annotations as weights or not (default = TRUE).

`Annotation_name`
: a vector of annotation names used in STAAR (default = NULL).

`silent`
: logical: should the report of error messages be suppressed (default = FALSE).

Value

a data frame containing the STAAR p-values (including STAAR-O) corresponding to the exonic and splicing category of the given ncRNA gene.

References

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

Li, X., Li, Z., et al. (2020). Dynamic incorporation of multiple in silico functional annotations empowers rare variant association analysis of large whole-genome sequencing studies at scale. *Nature Genetics*, *52*(9), 969-983. (pub)

---

ncRNA_cond — *Gene-centric conditional analysis of long noncoding RNA (ncRNA) category using STAAR procedure*

Description

The `ncRNA_cond` function takes in chromosome, gene name, the object of opened annotated GDS file, the object from fitting the null model, and the set of known variants to be adjusted for in conditional analysis to analyze the conditional association between a quantitative/dichotomous phenotype and the exonic and splicing category of an ncRNA gene by using STAAR procedure. For each ncRNA category, the conditional STAAR-O p-value is a p-value from an omnibus test that aggregated conditional SKAT(1,25), SKAT(1,1), Burden(1,25), Burden(1,1), ACAT-V(1,25), and ACAT-V(1,1) together with conditional p-values of each test weighted by each annotation using Cauchy method. For multiple phenotype analysis (`obj_nullmodel$n.pheno > 1`), the results correspond to multi-trait conditional p-values (e.g. conditional MultiSTAAR-O) by leveraging the correlation structure between multiple phenotypes.

Usage

```
ncRNA_cond(
  chr,
  gene_name,
  genofile,
  obj_nullmodel,
  known_loci = NULL,
  rare_maf_cutoff = 0.01,
  rv_num_cutoff = 2,
  method_cond = c("optimal", "naive"),
  QC_label = "annotation/filter",
  variant_type = c("SNV", "Indel", "variant"),
  geno_missing_imputation = c("mean", "minor"),
  Annotation_dir = "annotation/info/FunctionalAnnotation",
  Annotation_name_catalog,
  Use_annotation_weights = c(TRUE, FALSE),
  Annotation_name = NULL
)
```

Arguments

chr	chromosome.
gene_name	name of the ncRNA gene to be analyzed using STAAR procedure.
genofile	an object of opened annotated GDS (aGDS) file.
obj_nullmodel	an object from fitting the null model, which is either the output from fit_nullmodel function, or the output from fitNullModel function in the GENESIS package and transformed using the genesis2staar_nullmodel function.
known_loci	the data frame of variants to be adjusted for in conditional analysis and should contain 4 columns in the following order: chromosome (chr), position (pos), reference allele (ref), and alternative allele (alt) (default = NULL).
rare_maf_cutoff	the cutoff of maximum minor allele frequency in defining rare variants (default = 0.01).
rv_num_cutoff	the cutoff of minimum number of variants of analyzing a given variant-set (default = 2).
method_cond	a character value indicating the method for conditional analysis. optimal refers to regressing residuals from the null model on known_loci as well as all covariates used in fitting the null model (fully adjusted) and taking the residuals; naive refers to regressing residuals from the null model on known_loci and taking the residuals (default = optimal).
QC_label	channel name of the QC label in the GDS/aGDS file (default = "annotation/filter").
variant_type	type of variant included in the analysis. Choices include "SNV", "Indel", or "variant" (default = "SNV").
geno_missing_imputation	method of handling missing genotypes. Either "mean" or "minor" (default = "mean").
Annotation_dir	channel name of the annotations in the aGDS file (default = "annotation/info/FunctionalAnnotation").
Annotation_name_catalog	a data frame containing the name and the corresponding channel name in the aGDS file.
Use_annotation_weights	use annotations as weights or not (default = TRUE).
Annotation_name	a vector of annotation names used in STAAR (default = NULL).

Value

a data frame containing the conditional STAAR p-values (including STAAR-O) corresponding to the exonic and splicing category of the given ncRNA gene.

References

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

Li, X., Li, Z., et al. (2020). Dynamic incorporation of multiple in silico functional annotations empowers rare variant association analysis of large whole-genome sequencing studies at scale. *Nature Genetics*, *52*(9), 969-983. (pub)

Sofer, T., et al. (2019). A fully adjusted two-stage procedure for rank-normalization in genetic association studies. *Genetic Epidemiology*, *43*(3), 263-275. (pub)

`Sliding_Window` *Genetic region analysis of sliding windows using STAAR procedure*

Description

The `Sliding_Window` function takes in chromosome, starting location, ending location, sliding window length, the object of opened annotated GDS file, and the object from fitting the null model to analyze the association between a quantitative/dichotomous phenotype and variants in a genetic region by using STAAR procedure. For each sliding window, the STAAR-O p-value is a p-value from an omnibus test that aggregated SKAT(1,25), SKAT(1,1), Burden(1,25), Burden(1,1), ACAT-V(1,25), and ACAT-V(1,1) together with p-values of each test weighted by each annotation using Cauchy method. For multiple phenotype analysis (`obj_nullmodel$n.pheno > 1`), the results correspond to multi-trait association p-values (e.g. MultiSTAAR-O) by leveraging the correlation structure between multiple phenotypes.

Usage

```
Sliding_Window(
  chr,
  start_loc,
  end_loc,
  sliding_window_length = 2000,
  type = c("single", "multiple"),
  genofile,
  obj_nullmodel,
  rare_maf_cutoff = 0.01,
  rv_num_cutoff = 2,
  QC_label = "annotation/filter",
  variant_type = c("SNV", "Indel", "variant"),
  geno_missing_imputation = c("mean", "minor"),
  Annotation_dir = "annotation/info/FunctionalAnnotation",
  Annotation_name_catalog,
  Use_annotation_weights = c(TRUE, FALSE),
  Annotation_name = NULL,
  silent = FALSE
)
```

Arguments

`chr`	chromosome.
`start_loc`	starting location (position) of the genetic region to be analyzed using STAAR procedure.
`end_loc`	ending location (position) of the genetic region to be analyzed using STAAR procedure.
`sliding_window_length`	the (fixed) length of the sliding window to be analyzed using STAAR procedure.
`type`	the type of sliding window to be analyzed using STAAR procedure. Choices include `single, multiple` (default = `single`).
`genofile`	an object of opened annotated GDS (aGDS) file.

`obj_nullmodel` an object from fitting the null model, which is either the output from `fit_nullmodel` function, or the output from `fitNullModel` function in the `GENESIS` package and transformed using the `genesis2staar_nullmodel` function.

`rare_maf_cutoff`
the cutoff of maximum minor allele frequency in defining rare variants (default = 0.01).

`rv_num_cutoff` the cutoff of minimum number of variants of analyzing a given variant-set (default = 2).

`QC_label` channel name of the QC label in the GDS/aGDS file (default = "annotation/filter").

`variant_type` type of variant included in the analysis. Choices include "SNV", "Indel", or "variant" (default = "SNV").

`geno_missing_imputation`
method of handling missing genotypes. Either "mean" or "minor" (default = "mean").

`Annotation_dir` channel name of the annotations in the aGDS file
(default = "annotation/info/FunctionalAnnotation").

`Annotation_name_catalog`
a data frame containing the name and the corresponding channel name in the aGDS file.

`Use_annotation_weights`
use annotations as weights or not (default = TRUE).

`Annotation_name`
a vector of annotation names used in STAAR (default = NULL).

`silent` logical: should the report of error messages be suppressed (default = FALSE).

Value

a data frame containing the STAAR p-values (including STAAR-O) corresponding to each sliding window in the given genetic region.

References

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

Li, X., Li, Z., et al. (2020). Dynamic incorporation of multiple in silico functional annotations empowers rare variant association analysis of large whole-genome sequencing studies at scale. *Nature Genetics*, *52*(9), 969-983. (pub)

`Sliding_Window_cond` *Genetic region conditional analysis of sliding windows using STAAR procedure*

Description

The `Sliding_Window_cond` function takes in chromosome, starting location, ending location, the object of opened annotated GDS file, the object from fitting the null model, and the set of known variants to be adjusted for in conditional analysis to analyze the conditional association between a quantitative/dichotomous phenotype and variants in a genetic region by using STAAR procedure. For each sliding window, the conditional STAAR-O p-value is a p-value from an omnibus test that aggregated conditional SKAT(1,25), SKAT(1,1), Burden(1,25), Burden(1,1), ACAT-V(1,25), and ACAT-V(1,1) together with conditional p-values of each test weighted by each annotation using Cauchy method. For multiple phenotype analysis (`obj_nullmodel$n.pheno > 1`), the results correspond to multi-trait conditional p-values (e.g. conditional MultiSTAAR-O) by leveraging the correlation structure between multiple phenotypes.

Usage

```
Sliding_Window_cond(
  chr,
  start_loc,
  end_loc,
  genofile,
  obj_nullmodel,
  known_loci = NULL,
  rare_maf_cutoff = 0.01,
  rv_num_cutoff = 2,
  method_cond = c("optimal", "naive"),
  QC_label = "annotation/filter",
  variant_type = c("SNV", "Indel", "variant"),
  geno_missing_imputation = c("mean", "minor"),
  Annotation_dir = "annotation/info/FunctionalAnnotation",
  Annotation_name_catalog,
  Use_annotation_weights = c(TRUE, FALSE),
  Annotation_name = NULL
)
```

Arguments

`chr`	chromosome.
`start_loc`	starting location (position) of the sliding window to be analyzed using STAAR procedure.
`end_loc`	ending location (position) of the sliding window to be analyzed using STAAR procedure.
`genofile`	an object of opened annotated GDS (aGDS) file.
`obj_nullmodel`	an object from fitting the null model, which is either the output from `fit_nullmodel` function, or the output from `fitNullModel` function in the `GENESIS` package and transformed using the `genesis2staar_nullmodel` function.
`known_loci`	the data frame of variants to be adjusted for in conditional analysis and should contain 4 columns in the following order: chromosome (CHR), position (POS), reference allele (REF), and alternative allele (ALT) (default = NULL).
`rare_maf_cutoff`	the cutoff of maximum minor allele frequency in defining rare variants (default = 0.01).

`rv_num_cutoff` the cutoff of minimum number of variants of analyzing a given variant-set (default = 2).

`method_cond` a character value indicating the method for conditional analysis. `optimal` refers to regressing residuals from the null model on `known_loci` as well as all covariates used in fitting the null model (fully adjusted) and taking the residuals; `naive` refers to regressing residuals from the null model on `known_loci` and taking the residuals (default = `optimal`).

`QC_label` channel name of the QC label in the GDS/aGDS file (default = "annotation/filter").

`variant_type` type of variant included in the analysis. Choices include "SNV", "Indel", or "variant" (default = "SNV").

`geno_missing_imputation`
method of handling missing genotypes. Either "mean" or "minor" (default = "mean").

`Annotation_dir` channel name of the annotations in the aGDS file (default = "annotation/info/FunctionalAnnotation").

`Annotation_name_catalog`
a data frame containing the name and the corresponding channel name in the aGDS file.

`Use_annotation_weights`
use annotations as weights or not (default = TRUE).

`Annotation_name`
a vector of annotation names used in STAAR (default = NULL).

Value

a data frame containing the conditional STAAR p-values (including STAAR-O) corresponding to the sliding window in the given genetic region.

References

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

Li, X., Li, Z., et al. (2020). Dynamic incorporation of multiple in silico functional annotations empowers rare variant association analysis of large whole-genome sequencing studies at scale. *Nature Genetics*, *52*(9), 969-983. (pub)

Sofer, T., et al. (2019). A fully adjusted two-stage procedure for rank-normalization in genetic association studies. *Genetic Epidemiology*, *43*(3), 263-275. (pub)

`staar2scang_nullmodel` *Transforming the null model object fitted using STAAR to the null model object to be used for SCANG-STAAR*

Description

The `staar2scang_nullmodel` function takes in the object from fitting the null model and transforms it to the object from fitting the null model to be used for SCANG-STAAR procedure.

Usage

```
staar2scang_nullmodel(obj_nullmodel)
```

Arguments

obj_nullmodel	an object from fitting the null model, which is either the output from fit_nullmodel function, or the output from fitNullModel function in the GENESIS package and transformed using the genesis2staar_nullmodel function.

Value

an object from fitting the null model for related samples to be used for SCANG-STAAR procedure, which is the output from fit_null_glmmkin_SCANG function for related samples in the SCANG package.

References

Li, Z., Li, X., et al. (2022). A framework for detecting noncoding rare-variant associations of large-scale whole-genome sequencing studies. *Nature Methods*, *19*(12), 1599-1611. (pub)

Li, X., Li, Z., et al. (2020). Dynamic incorporation of multiple in silico functional annotations empowers rare variant association analysis of large whole-genome sequencing studies at scale. *Nature Genetics*, *52*(9), 969-983. (pub)

Li, Z., Li, X., et al. (2019). Dynamic scan procedure for detecting rare-variant association regions in whole-genome sequencing studies. *The American Journal of Human Genetics*, *104*(5), 802-814. (pub)

Index