

# Package ‘STAAR’

May 7, 2020

**Type** Package

**Title** STAAR Procedure for Dynamic Incorporation of Multiple Functional Annotations in Whole Genome Sequencing Studies

**Version** 0.9.5

**Date** 2020-05-07

**Author** Xihao Li [aut, cre], Zilin Li [aut, cre], Han Chen [aut]

**Maintainer** Xihao Li <xihao.li@hhs.harvard.edu>, Zilin Li <li@hsph.harvard.edu>

**Description** An R package for performing STAAR procedure in whole genome sequencing studies.

**License** GPL-3

**Copyright** See COPYRIGHTS for details.

**Imports** Rcpp, GMMAT, GENESIS, Matrix, methods

**Encoding** UTF-8

**LazyData** true

**Depends** R (>= 3.2.0)

**LinkingTo** Rcpp, RcppArmadillo

**RoxygenNote** 7.1.0

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

## R topics documented:

|  |           |
|--|-----------|
| CCT . . . . .                          | 2         |
| fit_null_glm . . . . .                 | 2         |
| fit_null_glmmkin . . . . .             | 3         |
| Indiv_Score_Test_Region . . . . .      | 5         |
| Indiv_Score_Test_Region_cond . . . . . | 6         |
| STAAR . . . . .                        | 7         |
| STAAR_cond . . . . .                   | 9         |
| <b>Index</b>                           | <b>11</b> |

---

|     |   |
|-----|---|
| CCT | <i>An analytical p-value combination method using the Cauchy distribution</i> |
|-----|---|

---

### Description

The CCT function takes in a numeric vector of p-values, a numeric vector of non-negative weights, and return the aggregated p-value using Cauchy method.

### Usage

```
CCT(pvals, weights = NULL)
```

### Arguments

|         |  |
|---------|--|
| pvals   | a numeric vector of p-values, where each of the element is between 0 to 1, to be combined. |
| weights | a numeric vector of non-negative weights. If NULL, the equal weights are assumed.          |

### Value

the aggregated p-value combining p-values from the vector pvals.

### References

Liu, Y., & Xie, J. (2019). Cauchy combination test: a powerful test with analytic p-value calculation under arbitrary dependency structures. *Journal of American Statistical Association* 1-18. ([pub](#))

### Examples

```
pvalues <- c(2e-02, 4e-04, 0.2, 0.1, 0.8)
CCT(pvals=pvalues)
```

---

|              |  |
|--------------|--|
| fit_null_glm | <i>Fit generalized linear model under the null hypothesis for unrelated samples.</i> |
|--------------|--|

---

### Description

The fit\_null\_glm function is a wrapper of the [glm](#) function from the [stats](#) package that fits a regression model under the null hypothesis for unrelated samples, which provides the preliminary step for subsequent variant-set tests in whole genome sequencing data analysis. See [glm](#) for more details.

### Usage

```
fit_null_glm(fixed, data, family = binomial(link = "logit"), ...)
```

**Arguments**

|        |   |
|--------|---|
| fixed  | an object of class <a href="#">formula</a> (or one that can be coerced to that class): a symbolic description of the fixed effects model to be fitted.  |
| data   | a data frame or list (or object coercible by <a href="#">as.data.frame</a> to a data frame) containing the variables in the model.  |
| 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 <a href="#">family</a> for details of family functions). Can be either "gaussian" for continuous phenotype or "binomial" for binary phenotype. |
| ...    | additional arguments that could be passed to <a href="#">glm</a> .  |

**Value**

The function returns an object of the model fit from [glm](#) (`obj_nullmodel`), with an additional element indicating the samples are unrelated (`obj_nullmodel$relatedness = FALSE`). See [glm](#) for more details.

---

|                               |   |
|-------------------------------|---|
| <code>fit_null_glmmkin</code> | <i>Fitting generalized linear mixed model with known relationship matrices under the null hypothesis for related samples.</i> |
|-------------------------------|---|

---

**Description**

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

**Usage**

```
fit_null_glmmkin(
  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**

|                           |   |
|---------------------------|---|
| <code>fixed</code>        | an object of class <a href="#">formula</a> (or one that can be coerced to that class): a symbolic description of the fixed effects model to be fitted.  |
| <code>data</code>         | a data frame or list (or object coercible by <a href="#">as.data.frame</a> to a data frame) containing the variables in the model.  |
| <code>kins</code>         | 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 <code>id</code> column of the data frame <code>data</code> .                         |
| <code>use_sparse</code>   | a logical switch of whether the provided dense <code>kins</code> matrix should be transformed to a sparse matrix (default = <code>NULL</code> ).  |
| <code>kins_cutoff</code>  | the cutoff value for clustering samples to make the output matrix sparse block-diagonal (default = 0.022).  |
| <code>id</code>           | a column in the data frame <code>data</code> , indicating the id of samples. When there are duplicates in <code>id</code> , the data is assumed to be longitudinal with repeated measures.  |
| <code>random.slope</code> | 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 <code>data</code> . There must be duplicates in <code>id</code> and <code>method.optim</code> must be "AI" (default = <code>NULL</code> ).  |
| <code>groups</code>       | 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 <code>data</code> , and <code>family</code> must be "gaussian" and <code>method.optim</code> must be "AI" (default = <code>NULL</code> ). |
| <code>family</code>       | 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 <a href="#">family</a> for details of family functions).   |
| <code>method</code>       | method of fitting the generalized linear mixed model. Either "REML" or "ML" (default = "REML").   |
| <code>method.optim</code> | optimization method of fitting the generalized linear mixed model. Either "AI", "Brent" or "Nelder-Mead" (default = "AI").  |
| <code>maxiter</code>      | a positive integer specifying the maximum number of iterations when fitting the generalized linear mixed model (default = 500).   |
| <code>tol</code>          | a positive number specifying tolerance, the difference threshold for parameter estimates below which iterations should be stopped (default = 1e-5).   |
| <code>taumin</code>       | the lower bound of search space for the variance component parameter $\tau$ (default = 1e-5), used when <code>method.optim</code> = "Brent". See Details.   |
| <code>taumax</code>       | the upper bound of search space for the variance component parameter $\tau$ (default = 1e5), used when <code>method.optim</code> = "Brent". See Details.  |
| <code>tauregion</code>    | the number of search intervals for the REML or ML estimate of the variance component parameter $\tau$ (default = 10), used when <code>method.optim</code> = "Brent". See Details.   |
| <code>verbose</code>      | a logical switch for printing detailed information (parameter estimates in each iteration) for testing and debugging purpose (default = <code>FALSE</code> ).   |
| <code>...</code>          | additional arguments that could be passed to <a href="#">glm</a> .  |

**Value**

The function returns an object of the model fit from `glmmkin` (`obj_nullmodel`), with additional elements indicating the samples are related (`obj_nullmodel$relatedness = TRUE`), 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. *American Journal of Humann 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. *American Journal of Humann Genetics* 104(2), 260-274. ([pub](#))

Chen, H. (2019). GMMAT: Generalized Linear Mixed Model Association Tests. ([web](#))

---

Indiv\_Score\_Test\_Region

*Score test for individual variants in a given variant-set*

---

**Description**

The `Indiv_Score_Test_Region` function takes in `genotype`, the object from fitting the null model to analyze the association between a quantitative/dichotomous phenotype and all individual variants in a given variant-set by using score test.

**Usage**

```
Indiv_Score_Test_Region(
  genotype,
  obj_nullmodel,
  rare_maf_cutoff = 0.01,
  rv_num_cutoff = 2
)
```

**Arguments**

- |                              |   |
|------------------------------|---|
| <code>genotype</code>        | an <code>n</code> * <code>p</code> genotype matrix (dosage matrix) of the target sequence, where <code>n</code> is the sample size and <code>p</code> is the number of genetic variants.  |
| <code>obj_nullmodel</code>   | an object from fitting the null model, which is the output from either <code>fit_null_glm</code> function for unrelated samples or <code>fit_null_glmmkin</code> function for related samples. Note that <code>fit_null_glmmkin</code> is a wrapper of <code>glmmkin</code> function from the <code>GMMAT</code> package. |
| <code>rare_maf_cutoff</code> | the cutoff of maximum minor allele frequency in defining rare variants. (Default is 0.01).  |
| <code>rv_num_cutoff</code>   | the cutoff of minimum number of variants of analyzing a given variant-set. (Default is 2).  |

**Value**

a data frame with  $p$  rows corresponding to the  $p$  genetic variants in the given variant-set and three columns: Score (the score test statistic), SE (the standard error associated with the score test statistic), and pvalue (the score test p-value). If a variant in the given variant-set has minor allele frequency = 0 or greater than `rare_maf_cutoff`, the corresponding row will be NA. If a variant in the given variant-set has standard error equal to 0, the p-value will be set as 1.

---

Indiv\_Score\_Test\_Region\_cond

*Conditional score test for individual variants in a given variant-set*


---

**Description**

The `Indiv_Score_Test_Region_cond` function takes in `genotype`, the genotype of variant(s) to be adjusted for in conditional analysis, the object from fitting the null model to analyze the conditional association between a quantitative/dichotomous phenotype and all individual variants in a given variant-set by using score test.

**Usage**

```
Indiv_Score_Test_Region_cond(
  genotype,
  genotype_adj,
  obj_nullmodel,
  rare_maf_cutoff = 0.01,
  rv_num_cutoff = 2
)
```

**Arguments**

- |                              |  |
|------------------------------|--|
| <code>genotype</code>        | an $n \times p$ genotype matrix (dosage matrix) of the target sequence, where $n$ is the sample size and $p$ is the number of genetic variants.  |
| <code>genotype_adj</code>    | an $n \times p_{\text{adj}}$ genotype matrix (dosage matrix) of the target sequence, where $n$ is the sample size and $p_{\text{adj}}$ is the number of genetic variants adjusted for conditional analysis (or a vector of a single variant with length $n$ if $p_{\text{adj}}$ is 1).                                 |
| <code>obj_nullmodel</code>   | an object from fitting the null model, which is the output from either <code>fit_null_glm</code> function for unrelated samples or <code>fit_null_glmkin</code> function for related samples. Note that <code>fit_null_glmkin</code> is a wrapper of <code>glmkin</code> function from the <code>GMMAT</code> package. |
| <code>rare_maf_cutoff</code> | the cutoff of maximum minor allele frequency in defining rare variants. (Default is 0.01).   |
| <code>rv_num_cutoff</code>   | the cutoff of minimum number of variants of analyzing a given variant-set. (Default is 2).   |

**Value**

a data frame with  $p$  rows corresponding to the  $p$  genetic variants in the given variant-set and three columns: `Score_cond` (the conditional score test statistic adjusting for variants in `genotype_adj`), `SE_cond` (the standard error associated with the conditional score test statistic), and `pvalue_cond` (the conditional score test p-value). If a variant in the given variant-set has minor allele frequency = 0 or greater than `rare_maf_cutoff`, the corresponding row will be NA. If a variant in the given variant-set has standard error equal to 0, the p-value will be set as 1.

STAAR

*STAAR procedure using omnibus test***Description**

The STAAR function takes in `genotype`, the object from fitting the null model, and functional annotation data to analyze the association between a quantitative/dichotomous phenotype and a variant-set by using STAAR procedure. For each variant-set, 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.

**Usage**

```
STAAR(
  genotype,
  obj_nullmodel,
  annotation_phred = NULL,
  rare_maf_cutoff = 0.01,
  rv_num_cutoff = 2
)
```

**Arguments**

- |                               |  |
|-------------------------------|--|
| <code>genotype</code>         | an $n \times p$ genotype matrix (dosage matrix) of the target sequence, where $n$ is the sample size and $p$ is the number of genetic variants.  |
| <code>obj_nullmodel</code>    | an object from fitting the null model, which is the output from either <code>fit_null_glm</code> function for unrelated samples or <code>fit_null_glmmkin</code> function for related samples. Note that <code>fit_null_glmmkin</code> is a wrapper of the <code>glmmkin</code> function from the <code>GMMAT</code> package.  |
| <code>annotation_phred</code> | a data frame or matrix of functional annotation data of dimension $p \times q$ (or a vector of a single annotation score with length $p$ ). Continuous scores should be given in PHRED score scale, where the PHRED score of $j$ -th variant is defined to be $-10 \times \log_{10}(\text{rank}(-\text{score}_j)/\text{total})$ across the genome. (Binary) categorical scores should be taking values 0 or 1, where 1 is functional and 0 is non-functional. If not provided, STAAR will perform the SKAT(1,25), SKAT(1,1), Burden(1,25), Burden(1,1), ACAT-V(1,25), ACAT-V(1,1) and ACAT-O tests (default = NULL). |
| <code>rare_maf_cutoff</code>  | the cutoff of maximum minor allele frequency in defining rare variants. (Default is 0.01).   |

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

## Value

a list with the following members:

`num_variant`: the number of variants with minor allele frequency  $> 0$  and less than `rare_maf_cutoff` in the given variant-set that are used for performing the variant-set using STAAR.

`RV_label`: the boolean vector indicating whether each variant in the given variant-set has minor allele frequency  $> 0$  and less than `rare_maf_cutoff`.

`results_STAAR_O`: the STAAR-O p-value 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.

`results_ACAT_O`: the ACAT-O p-value that aggregated SKAT(1,25), SKAT(1,1), Burden(1,25), Burden(1,1), ACAT-V(1,25), and ACAT-V(1,1) using Cauchy method.

`results_STAAR_S_1_25`: a vector of STAAR-S(1,25) p-values, including SKAT(1,25) p-value weighted by MAF, the SKAT(1,25) p-values weighted by each annotation, and a STAAR-S(1,25) p-value by aggregating these p-values using Cauchy method.

`results_STAAR_S_1_1`: a vector of STAAR-S(1,1) p-values, including SKAT(1,1) p-value weighted by MAF, the SKAT(1,1) p-values weighted by each annotation, and a STAAR-S(1,1) p-value by aggregating these p-values using Cauchy method.

`results_STAAR_B_1_25`: a vector of STAAR-B(1,25) p-values, including Burden(1,25) p-value weighted by MAF, the Burden(1,25) p-values weighted by each annotation, and a STAAR-B(1,25) p-value by aggregating these p-values using Cauchy method.

`results_STAAR_B_1_1`: a vector of STAAR-B(1,1) p-values, including Burden(1,1) p-value weighted by MAF, the Burden(1,1) p-values weighted by each annotation, and a STAAR-B(1,1) p-value by aggregating these p-values using Cauchy method.

`results_STAAR_A_1_25`: a vector of STAAR-A(1,25) p-values, including ACAT-V(1,25) p-value weighted by MAF, the ACAT-V(1,25) p-values weighted by each annotation, and a STAAR-A(1,25) p-value by aggregating these p-values using Cauchy method.

`results_STAAR_A_1_1`: a vector of STAAR-A(1,1) p-values, including ACAT-V(1,1) p-value weighted by MAF, the ACAT-V(1,1) p-values weighted by each annotation, and a STAAR-A(1,1) p-value by aggregating these p-values using Cauchy method.

## References

Li, X., Li, Z. et al. (2019+). Dynamic incorporation of multiple in-silico functional annotations empowers rare variant association analysis of large whole genome sequencing studies at scale. *Submitted*.

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



## Description

The STAAR\_cond function takes in genotype, the genotype of variant(s) to be adjusted for in conditional analysis, the object from fitting the null model, and functional annotation data to analyze the conditional association between a quantitative/dichotomous phenotype and a variant-set by using STAAR procedure, adjusting for a given list of variants. For each variant-set, 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.

## Usage

```
STAAR_cond(
  genotype,
  genotype_adj,
  obj_nullmodel,
  annotation_phred = NULL,
  rare_maf_cutoff = 0.01,
  rv_num_cutoff = 2
)
```

## Arguments

- |                  |  |
|------------------|--|
| genotype         | an n*p genotype matrix (dosage matrix) of the target sequence, where n is the sample size and p is the number of genetic variants.   |
| genotype_adj     | an n*p_adj genotype matrix (dosage matrix) of the target sequence, where n is the sample size and p_adj is the number of genetic variants adjusted for conditional analysis (or a vector of a single variant with length n if p_adj is 1).   |
| obj_nullmodel    | an object from fitting the null model, which is the output from either <a href="#">fit_null_glm</a> function for unrelated samples or <a href="#">fit_null_glmmkin</a> function for related samples. Note that <a href="#">fit_null_glmmkin</a> is a wrapper of the <a href="#">glmmkin</a> function from the <a href="#">GMMAT</a> package.   |
| annotation_phred | a data frame or matrix of functional annotation data of dimension p*q (or a vector of a single annotation score with length p). Continuous scores should be given in PHRED score scale, where the PHRED score of j-th variant is defined to be $-10 \cdot \log_{10}(\text{rank}(-\text{score}_j)/\text{total})$ across the genome. (Binary) categorical scores should be taking values 0 or 1, where 1 is functional and 0 is non-functional. If not provided, STAAR will perform the SKAT(1,25), SKAT(1,1), Burden(1,25), Burden(1,1), ACAT-V(1,25), ACAT-V(1,1) and ACAT-O tests (default = NULL). |
| rare_maf_cutoff  | the cutoff of maximum minor allele frequency in defining rare variants. (Default is 0.01).   |
| rv_num_cutoff    | the cutoff of minimum number of variants of analyzing a given variant-set. (Default is 2).   |

**Value**

a list with the following members:

`num_variant`: the number of variants with minor allele frequency  $> 0$  and less than `rare_maf_cutoff` in the given variant-set that are used for performing the variant-set using STAAR.

`RV_label`: the boolean vector indicating whether each variant in the given variant-set has minor allele frequency  $> 0$  and less than `rare_maf_cutoff`.

`results_STAAR_O_cond`: the conditional STAAR-O p-value 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.

`results_ACAT_O_cond`: the conditional ACAT-O p-value that aggregated conditional SKAT(1,25), SKAT(1,1), Burden(1,25), Burden(1,1), ACAT-V(1,25), and ACAT-V(1,1) using Cauchy method.

`results_STAAR_S_1_25_cond`: a vector of conditional STAAR-S(1,25) p-values, including conditional SKAT(1,25) p-value weighted by MAF, the conditional SKAT(1,25) p-values weighted by each annotation, and a conditional STAAR-S(1,25) p-value by aggregating these p-values using Cauchy method.

`results_STAAR_S_1_1_cond`: a vector of conditional STAAR-S(1,1) p-values, including conditional SKAT(1,1) p-value weighted by MAF, the conditional SKAT(1,1) p-values weighted by each annotation, and a conditional STAAR-S(1,1) p-value by aggregating these p-values using Cauchy method.

`results_STAAR_B_1_25_cond`: a vector of conditional STAAR-B(1,25) p-values, including conditional Burden(1,25) p-value weighted by MAF, the conditional Burden(1,25) p-values weighted by each annotation, and a conditional STAAR-B(1,25) p-value by aggregating these p-values using Cauchy method.

`results_STAAR_B_1_1_cond`: a vector of conditional STAAR-B(1,1) p-values, including conditional Burden(1,1) p-value weighted by MAF, the conditional Burden(1,1) p-values weighted by each annotation, and a conditional STAAR-B(1,1) p-value by aggregating these p-values using Cauchy method.

`results_STAAR_A_1_25_cond`: a vector of conditional STAAR-A(1,25) p-values, including conditional ACAT-V(1,25) p-value weighted by MAF, the conditional ACAT-V(1,25) p-values weighted by each annotation, and a conditional STAAR-A(1,25) p-value by aggregating these p-values using Cauchy method.

`results_STAAR_A_1_1_cond`: a vector of conditional STAAR-A(1,1) p-values, including conditional ACAT-V(1,1) p-value weighted by MAF, the conditional ACAT-V(1,1) p-values weighted by each annotation, and a conditional STAAR-A(1,1) p-value by aggregating these p-values using Cauchy method.

**References**

Li, X., Li, Z. et al. (2019+). Dynamic incorporation of multiple in-silico functional annotations empowers rare variant association analysis of large whole genome sequencing studies at scale. *Submitted*.

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

# Index

`as.data.frame`, [3](#), [4](#)

`CCT`, [2](#)

`family`, [3](#), [4](#)

`fit_null_glm`, [2](#), [5–7](#), [9](#)

`fit_null_glmmkin`, [3](#), [5–7](#), [9](#)

`formula`, [3](#), [4](#)

`glm`, [2–4](#)

`glmmkin`, [3](#), [5–7](#), [9](#)

`GMMAT`, [3](#), [5–7](#), [9](#)

`Indiv_Score_Test_Region`, [5](#)

`Indiv_Score_Test_Region_cond`, [6](#)

`STAAR`, [7](#)

`STAAR_cond`, [9](#)

`stats`, [2](#)