Package: mvMAPIT (via r-universe)

October 29, 2026

Type Package

Title Multivariate Genome Wide Marginal Epistasis Test

Version 2.0.4

URL https://github.com/lcrawlab/mvMAPIT, https://lcrawlab.github.io/mvMAPIT/

Description Epistasis, commonly defined as the interaction between genetic loci, is known to play an important role in the phenotypic variation of complex traits. As a result, many statistical methods have been developed to identify genetic variants that are involved in epistasis, and nearly all of these approaches carry out this task by focusing on analyzing one trait at a time. Previous studies have shown that jointly modeling multiple phenotypes can often dramatically increase statistical power for association mapping. In this package, we present the 'multivariate MArginal ePIstasis Test' ('mvMAPIT') – a multi-outcome generalization of a recently proposed epistatic detection method which seeks to detect marginal epistasis or the combined pairwise interaction effects between a given variant and all other variants. By searching for marginal epistatic effects, one can identify genetic variants that are involved in epistasis without the need to identify the exact partners with which the variants interact – thus, potentially alleviating much of the statistical and computational burden associated with conventional explicit search based methods. Our proposed 'mvMAPIT' builds upon this strategy by taking advantage of correlation structure between traits to improve the identification of variants involved in epistasis. We formulate 'mvMAPIT' as a multivariate linear mixed model and develop a multi-trait variance component estimation algorithm for efficient parameter inference and P-value computation. Together with reasonable model approximations, our proposed approach is scalable to moderately sized genome-wide association studies. Crawford et al. (2017) <doi:10.1371/journal.pgen.1006869>. Stamp et al. (2023)

<doi:10.1093/g3journal/jkad118>. Stamp et al. (2025) <doi:10.1016/j.ajhg.2025.07.004>.

License GPL (>= 3)

Depends R (>= 3.5)

Imports checkmate, CompQuadForm, dplyr, foreach, harmonicmeanp, logging, mvtnorm, Rcpp, stats, tidyr, truncnorm, utils

Suggests GGally, ggplot2, ggrepel, kableExtra, knitr, markdown, RcppAlgos, rmarkdown, testthat

LinkingTo Rcpp, RcppArmadillo, RcppParallel, RcppProgress, RcppSpdlog, testthat

VignetteBuilder knitr

Encoding UTF-8

LazyData true

LazyDataCompression xz

RoxygenNote 7.3.1

NeedsCompilation yes

Author Julian Stamp [cre, aut] (ORCID: <https://orcid.org/0000-0003-3014-6249>), Lorin Crawford [aut] (ORCID: <https://orcid.org/0000-0003-0178-8242>)

Maintainer Julian Stamp <julian.d.stamp@gmail.com>

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

Date/Publication 2026-02-22 06:10:02 UTC

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

RemoteRef HEAD

RemoteSha 0abbb4e0f67c824d3cf42662c6587774818c27e2

Contents

binary_to_liability 3
cauchy_combined 3
fishers_combined 4
harmonic_combined 5
mvmapit 6
mvmapit_data 8
phillips_data 9
simulate_traits 10
simulated_data 11

Index **13**

binary_to_liability *Convert binary traits to liabilities for low prevalence*

Description

This function implements an approximate conversion of binary traits to liabilties as proposed in the LT-MAPIT model (Crawford and Zhou 2018, https://doi.org/10.1101/374983). Note that this is only good for low prevalence. To run LT-MAPIT (MAPIT on case-control traits), convert the binary traits to liabilities using this function and pass the liabilities to mvmapit as trait.

Usage

```
binary_to_liability(case_control_trait, prevalence)
```

Arguments

case_control_trait	Case-control trait encoded as binary trait with 0 as control or 1 as case.
prevalence	Case prevalence between 0 and 1. Proportion of cases in the population.

Value

A trait vector of same length as y with case-control indicators converted to liabilties.

cauchy_combined *Cauchy p combine method on mvmapit return*

Description

This function takes in the p-values tibble that mvmapit returned. It then computes the combined p-values grouped by variant id.

Usage

```
cauchy_combined(pvalues, group_col = "id", p_col = "p")
```

Arguments

pvalues	Tibble with p-values from mvmapit function call. Grouping is based on the column named "id"
group_col	String that denotes column by which to group and combine p-values.
p_col	String that denotes p-value column.

Value

A Tibble with the combined p-values.

Examples

```
set.seed(837)
p <- 200
n <- 100
d <- 2
X <- matrix(
    runif(p * n),
    ncol = p
)
Y <- matrix(
    runif(d * n),
    ncol = d
)
mapit <- mvmapit(
    t(X),
    t(Y),
    test = "normal", cores = 1, logLevel = "INFO"
)
cauchy <- cauchy_combined(mapit$pvalues)
```

---

fishers_combined *Fisher's combine method on mvmapit return*

Description

This function takes in the p-values tibble that mvmapit returned. It then computes the combined p-values grouped by variant id.

Usage

```
fishers_combined(pvalues, group_col = "id", p_col = "p")
```

Arguments

pvalues	Tibble with p-values from mvmapit function call.
group_col	String that denotes column by which to group and combine p-values.
p_col	String that denotes p-value column.

Value

A Tibble with the combined p-values.

Examples

```
set.seed(837)
p <- 200
n <- 100
d <- 2
```

```
X <- matrix(
    runif(p * n),
    ncol = p
)
Y <- matrix(
    runif(d * n),
    ncol = d
)
mapit <- mvmapit(
    t(X),
    t(Y),
    test = "normal", cores = 1, logLevel = "INFO"
)
fisher <- fishers_combined(mapit$pvalues)
```

`harmonic_combined` *Harmonic mean p combine method on mvmapit return*

Description

This function takes in the p-values tibble that mvmapit returned. It then computes the combined p-values grouped by variant id.

Usage

```
harmonic_combined(pvalues, group_col = "id", p_col = "p")
```

Arguments

`pvalues`	Tibble with p-values from mvmapit function call. Grouping is based on the column named "id"
`group_col`	String that denotes column by which to group and combine p-values.
`p_col`	String that denotes p-value column.

Value

A Tibble with the combined p-values.

Examples

```
set.seed(837)
p <- 200
n <- 100
d <- 2
X <- matrix(
    runif(p * n),
    ncol = p
)
Y <- matrix(
```

```
    runif(d * n),
    ncol = d
)
mapit <- mvmapit(
    t(X),
    t(Y),
    test = "normal", cores = 1, logLevel = "INFO"
)
harmonic <- harmonic_combined(mapit$pvalues)
```

`mvmapit` *Multivariate MArginal ePIstasis Test (mvMAPIT)*

Description

This function runs a multivariate version of the MArginal ePIstasis Test (mvMAPIT) under the following model variations:

Usage

```
mvmapit(
  X,
  Y,
  Z = NULL,
  C = NULL,
  threshold = 0.05,
  accuracy = 1e-08,
  test = c("normal", "davies", "hybrid"),
  cores = 1,
  variantIndex = NULL,
  logLevel = "WARN",
  logFile = NULL
)
```

Arguments

`X`	is the p x n genotype matrix where p is the number of variants and n is the number of samples. Must be a matrix and not a data.frame.
`Y`	is the d x n matrix of d quantitative or continuous traits for n samples.
`Z`	is the matrix q x n matrix of covariates. Must be a matrix and not a data.frame.
`C`	is an n x n covariance matrix detailing environmental effects and population structure effects.
`threshold`	is a parameter detailing the value at which to recalibrate the Z test p values. If nothing is defined by the user, the default value will be 0.05 as recommended by the Crawford et al. (2017).

accuracy	is a parameter setting the davies function numerical approximation accuracy. This parameter is not needed for the normal test. Smaller p-values than the accuracy will be zero.
test	is a parameter defining what hypothesis test should be run. Takes on values 'normal', 'davies', and 'hybrid'. The 'hybrid' test runs first the 'normal' test and then the 'davies' test on the significant variants.
cores	is a parameter detailing the number of cores to parallelize over. It is important to note that this value only matters when the user has installed OPENMP on their operating system.
variantIndex	is a vector containing indices of variants to be included in the computation.
logLevel	is a string parameter defining the log level for the logging package.
logFile	is a string parameter defining the name of the log file for the logging output. Default is stdout.

Details

(1) Standard Model: y = m+g+e where m ~ MVN(0,omega^2K), g ~ MVN(0,sigma^2G), e ~ MVN(0,tau^2M). Recall from Crawford et al. (2017) that m is the combined additive effects from all other variants, represents the additive effect of the k-th variant under the polygenic background of all other variants; K is the genetic relatedness matrix computed using genotypes from all variants other than the k-th; g is the summation of all pairwise interaction effects between the k-th variant and all other variants; G represents a relatedness matrix computed based on pairwise interaction terms between the k-th variant and all other variants. Here, we also denote D = diag(x_k) to be an $n \times n$ diagonal matrix with the genotype vector x_k as its diagonal elements. It is important to note that both K and G change with every new marker k that is considered. Lastly; M is a variant specific projection matrix onto both the null space of the intercept and the corresponding genotypic vector x_k.

(2) Standard + Covariate Model: y = Wa+m+g+e where W is a matrix of covariates with effect sizes a.

(3) Standard + Common Environment Model: y = m+g+c+e i where c ~ MVN(0,eta^2C) controls for extra environmental effects and population structure with covariance matrix C.

(4) Standard + Covariate + Common Environment Model: y = Wa+m+g+c+e

This function will consider the following three hypothesis testing strategies which are featured in Crawford et al. (2017): (1) The Normal or Z test (2) Davies Method (3) Hybrid Method (Z test + Davies Method)

Value

A list of P values and PVEs

Examples

```
set.seed(837)
p <- 200
n <- 100
d <- 2
X <- matrix(
```

```
    runif(p * n),
    ncol = p
)
Y <- matrix(
    runif(d * n),
    ncol = d
)
mapit <- mvmapit(
    t(X),
    t(Y),
    test = "normal", cores = 1, logLevel = "INFO"
)
```

`mvmapit_data` *Multivariate MAPIT analysis and exhaustive search analysis.*

Description

This data set contains the return object from the multivariate MAPIT method, the fisher combined p-values, and the result from an exhaustive search using regression on the SNPs that were significant in the mvMAPIT analysis.

Usage

```
mvmapit_data
```

Format

A nested list containing tibble data frames:

mvmapit mvmapit return object; named list containing tibbles ‘pvalues‘, ‘pves‘, and ‘duration‘.

fisher Tibble containing fisher combined p-values of the mvmapit data.

exhaustive_search A dataframe containing the p-values of an exhaustive search together with the analysed interaction pair.

Source

data-raw/mvmapit_on_simulated_data.R

`phillips_data` *Multivariate MAPIT analysis of binding affinities in broadly neutralizing antibodies.*

Description

This data set contains the return object from the multivariate MAPIT method, applied to two binding affinity traits for two broadly neutralizing antibodies. It also contains the regression coefficients on the same data as reported by Phillips et al. (2021).

Usage

```
phillips_data
```

Format

A named list containing tibble data frames:

fisher Tibble containing among other columns the residue id, p-values, antibody species, trait of the Phillips data. Combined p-value with Fisher's method.

harmonic Tibble containing among other columns the residue id, p-values, antibody species, trait of the Phillips data. Combined p-value with harmonic mean p method.

regression Named list containing two tibbles containing regression coefficients as reported by Phillips et al.

Details

The antibody CR9114 was analyzed with influenza H1 and H3. The antibody CR6261 was analyzed with influenza H1 and H9.

In the data, the p-values are computed for the test whether a given residue position has a marginal epistatic effect on the binding affinities.

Phillips et al. (2021) Binding affinity landscapes constrain the evolution of broadly neutralizing anti-influenza antibodies. eLife 10:e71393

Source

vignette/study-phillips-bnabs.Rmd

`simulate_traits` *Simulate phenotye data*

Description

This function simulates trait data from a genotype matrix.

Usage

```
simulate_traits(
  genotype_matrix,
  n_causal = 1000,
  n_trait_specific = 10,
  n_pleiotropic = 10,
  H2 = 0.6,
  d = 2,
  rho = 0.8,
  marginal_correlation = 0.3,
  epistatic_correlation = 0.3,
  group_ratio_trait = 1,
  group_ratio_pleiotropic = 1,
  maf_threshold = 0.01,
  seed = 67132,
  logLevel = "INFO",
  logFile = NULL
)
```

Arguments

`genotype_matrix`	Genotype matrix with samples as rows, and SNPs as columns.
`n_causal`	Number of SNPs that are causal.
`n_trait_specific`	Number of causal SNPs with single trait epistatic effects.
`n_pleiotropic`	Number of SNPs with epistatic effects on all traits.
`H2`	Broad-sense heritability. Can be vector.
`d`	Number of traits.
`rho`	Proportion of heritability explained by additivity.
`marginal_correlation`	Correlation between the additive effects of the trait.
`epistatic_correlation`	Correlation between the epistatic effects of the trait.
`group_ratio_trait`	Ratio of sizes of trait specific groups that interact, e.g. a ratio 1:3 would be value 3.

group_ratio_pleiotropic
: Ratio of sizes of pleiotropic groups that interact, e.g. a ratio 1:3 would be value 3.

maf_threshold
: is a float parameter defining the threshold for the minor allele frequency not included in causal SNPs.

seed
: Random seed for simulation.

logLevel
: is a string parameter defining the log level for the logging package.

logFile
: is a string parameter defining the name of the log file for the logging output.

Details

This function takes a genotype matrix and simulates trait data under the following model: beta_i ~ MN(0, V_i, I), i in { additive, epistatic, residual}

The effect sizes follow a matrix normal distribution with no correlation between the samples but covariance between the effects for different traits

Value

A list object containing the trait data, the genotype data, as well as the causal SNPs and summary statistics.

Examples

```
p <- 200
f <- 10
g <- 4
n <- 100
d <- 3
X <- matrix(
    runif(p * n),
    ncol = p
)
data <- simulate_traits(
    X, n_causal = f, n_trait_specific = g, n_pleiotropic = g, d = d, maf_threshold = 0,
    logLevel = "ERROR"
)
```

---

simulated_data *Genotype and trait data with epistasis.*

Description

A simulated dataset that has epistatic interactions.

Usage

```
simulated_data
```

Format

A named list with simulated data and simulation parameters:

parameters Tibble containing simulation parameters for each trait.

trait Matrix containing simulated data for 2 traits and 500 samples.

genotype Matrix containing simulated genotype with 500 samples and 1000 variables.

additive Tibble containing all variants with additive effects on the traits as well as the effect sizes.

epistatic Tibble containing all variants with epistatic effects on the traits as well as the effect sizes.

interactions Tibble containing all interactions, effect size, and trait they affect.

snps.filtered SNPs that were used in the simulations.

Source

data-raw/simulate_epistasis.R

Index

∗ **datasets**
    mvmapit_data, 8
    phillips_data, 9
    simulated_data, 11

binary_to_liability, 3

cauchy_combined, 3

fishers_combined, 4

harmonic_combined, 5

mvmapit, 6
mvmapit_data, 8

phillips_data, 9

simulate_traits, 10
simulated_data, 11