

Package: GEint (via r-universe)

May 26, 2026

Type Package

Title Misspecified Models for Gene-Environment Interaction

Version 1.1

Date 2025-09-18

Description The first major functionality is to compute the bias in regression coefficients of misspecified linear gene-environment interaction models. The most generalized function for this objective is `GE_bias()`. However `GE_bias()` requires specification of many higher order moments of covariates in the model. If users are unsure about how to calculate/estimate these higher order moments, it may be easier to use `GE_bias_normal_squaredmis()`. This function places many more assumptions on the covariates (most notably that they are all jointly generated from a multivariate normal distribution) and is thus able to automatically calculate many of the higher order moments automatically, necessitating only that the user specify some covariances. There are also functions to solve for the bias through simulation and non-linear equation solvers; these can be used to check your work. Second major functionality is to implement the Bootstrap Inference with Correct Sandwich (BICS) testing procedure, which we have found to provide better finite-sample performance than other inference procedures for testing GxE interaction. More details on these functions are available in Sun, Carroll, Christiani, and Lin (2018) <[doi:10.1111/biom.12813](https://doi.org/10.1111/biom.12813)>.

Imports mvtnorm, bindata, nleqslv, pracma, speedglm, rje, geepack, stats

License GPL-3

RoxygenNote 6.1.1

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

NeedsCompilation no

Author Ryan Sun [aut, cre], Richard Barfield [ctb]

Maintainer Ryan Sun <ryansun.work@gmail.com>

Config/pak/sysreqs libicu-dev

Repository <https://ryansunwork.r-universe.dev>

Date/Publication 2025-10-01 08:42:12 UTC

RemoteUrl <https://github.com/cran/GEint>

RemoteRef HEAD

RemoteSha be47d47382ef49f4610e8a4983571da02cef6a07

Contents

GE_bias	2
GE_bias_normal_squaredmis	3
GE_BICS	4
GE_nleqslv	5
GE_scoreeq_sim	6
Index	8

GE_bias

GE_bias.R

Description

A function to calculate the bias in testing for GxE interaction.

Usage

```
GE_bias(beta_list, cov_list, cov_mat_list, mu_list, HOM_list)
```

Arguments

- | | |
|--------------|--|
| beta_list | A list of the effect sizes in the true model. Use the order beta_0, beta_G, beta_E, beta_I, beta_Z, beta_M. If G or Z or M is a vector, then beta_G/beta_Z/beta_M should be vectors. If Z and/or M/W do not exist in your model, then set beta_Z and/or beta_M = 0. |
| cov_list | A list of expectations (which happen to be covariances if all covariates are centered at 0) in the order specified by GE_enumerate_inputs(). If Z and/or M/W do not exist in your model, then treat them as constants 0. For example, if Z doesn't exist and W includes 2 covariates, then set cov(EZ) = 0 and cov(ZW) = (0,0). If describing expectations relating two vectors, i.e. Z includes two covariates and W includes three covariates, sort by the first term and then the second. Thus in the example, the first three terms of cov(ZW) are cov(Z_1,W_1),cov(Z_1,W_2), cov(Z_1,W_3), and the last three terms are cov(Z_3,W_1), cov(Z_3,W_2), cov(Z_3,W_3). |
| cov_mat_list | A list of matrices of expectations as specified by GE_enumerate_inputs(). |

mu_list A list of means as specified by GE_enumerate_inputs().
 HOM_list A list of higher order moments as specified by GE_enumerate_inputs().

Value

A list of the fitted coefficients alpha

Examples

```
solutions <- GE_bias_normal_squaredmis( beta_list=as.list(runif(n=6, min=0, max=1)),
rho_list=as.list(rep(0.3,6)), prob_G=0.3, cov_Z=1, cov_W=1)
GE_bias(beta_list=solutions$beta_list, solutions$cov_list, solutions$cov_mat_list,
solutions$mu_list, solutions$HOM_list)
```

GE_bias_normal_squaredmis

GE_bias_normal_squaredmis.R

Description

A function to calculate the bias in testing for GxE interaction, making many more assumptions than GE_bias(). The additional assumptions are added to simplify the process of calculating/estimating many higher order moments which the user may not be familiar with.

The following assumptions are made:

- (1) All fitted covariates besides G (that is, E, all Z, and all W) have a marginal standard normal distribution with mean 0 and variance 1. This corresponds to the case of the researcher standardizing all of their fitted covariates.
- (2) All G are generated by means of thresholding two independent normal RVs and are centered to have mean 0.
- (3) The joint distributions of E, Z, W, and the thresholded variables underlying G can be described by a multivariate normal distribution.
- (4) The misspecification is of the form $f(E)=h(E)=E^2$, and $M_j=W_j^2$ for all j. In particular, W always has the same length as M here.

Usage

```
GE_bias_normal_squaredmis(beta_list, rho_list, prob_G, cov_Z = NULL,
cov_W = NULL, corr_G = NULL)
```

Arguments

beta_list A list of the effect sizes in the true model. Use the order beta_0, beta_G, beta_E, beta_I, beta_Z, beta_M. If G or Z or M is a vector, then beta_G/beta_Z/beta_M should be vectors. If Z and/or M/W do not exist in your model, then set beta_Z and/or beta_M = 0.

rho_list	A list of expectations (which happen to be covariances if all covariates are centered at 0) in the order specified by GE_enumerate_inputs(). If Z and/or M/W do not exist in your model, then treat them as constants 0. For example, if Z doesn't exist and W includes 2 covariates, then set cov(EZ) = 0 and cov(ZW) = (0,0). If describing expectations relating two vectors, i.e. Z includes two covariates and W includes three covariates, sort by the first term and then the second. Thus in the example, the first three terms of cov(ZW) are cov(Z_1,W_1),cov(Z_1,W_2), cov(Z_1,W_3), and the last three terms are cov(Z_3,W_1), cov(Z_3,W_2), cov(Z_3,W_3).
prob_G	Probability that each allele is equal to 1. Since each SNP has two alleles, the expectation of G is 2*prob_G. Should be a d*1 vector.
cov_Z	Should be a matrix equal to cov(Z) or NULL if no Z.
cov_W	Should be a matrix equal to cov(W) or NULL if no W.
corr_G	Should be a matrix giving the *pairwise correlations* between each SNP in the set, or NULL. Must be specified if G is a vector. For example, the [2,3] element of the matrix would be the pairwise correlation between SNP2 and SNP3.

Value

A list with the elements:

alpha_list	The asymptotic values of the fitted coefficients alpha.
beta_list	The same beta_list that was given as input.
cov_list	The list of all covariances (both input and calculated) for use with GE_nleqslv() and GE_bias().
mu_list	List of calculated means for f(E), h(E), Z, M, and W for use with GE_nleqslv() and GE_bias().
HOM_list	List of calculated Higher Order Moments for use with GE_nleqslv() and GE_bias().

Examples

```
GE_bias_normal_squaredmis( beta_list=as.list(runif(n=6, min=0, max=1)),
rho_list=as.list(rep(0.3,6)), cov_Z=1, cov_W=1, prob_G=0.3)
```

GE_BICS

GE_BICS.R

Description

A function to perform inference on the GxE interaction regression coefficient. Shows better small sample performance than comparable methods.

Usage

```
GE_BICS(outcome, design_mat, num_boots = 1000, desired_coef,
outcome_type, check_singular = FALSE)
```

Arguments

outcome	The outcome vector
design_mat	The design matrix of covariates
num_boots	The number of bootstrap resamples to perform - we suggest 1000
desired_coef	The column in the design matrix holding the interaction covariate
outcome_type	Either 'D' for dichotomous outcome or 'C' for continuous outcome
check_singular	Make sure the design matrix can be inverted for variance estimation

Value

The p-value for the interaction effect

Examples

```
E <- rnorm(n=500)
G <- rbinom(n=500, size=2, prob=0.3)
design_mat <- cbind(1, G, E, G*E)
outcome <- rnorm(500)
GE_BICS(outcome=outcome, design_mat=design_mat, desired_coef=4, outcome_type='C')
```

GE_nleqslv	<i>GE_nleqslv.R #</i> Uses package nleqslv to get a numerical solution to the score equations, which we can use to check our direct solution from GE_bias().
------------	--

Description

GE_nleqslv.R # Uses package nleqslv to get a numerical solution to the score equations, which we can use to check our direct solution from GE_bias().

Usage

```
GE_nleqslv(beta_list, cov_list, cov_mat_list, mu_list, HOM_list)
```

Arguments

beta_list	A list of the effect sizes in the true model. Use the order beta_0, beta_G, beta_E, beta_I, beta_Z, beta_M. If G or Z or M is a vector, then beta_G/beta_Z/beta_M should be vectors. If Z and/or M/W do not exist in your model, then set beta_Z and/or beta_M = 0.
cov_list	A list of expectations (which happen to be covariances if all covariates are centered at 0) in the order specified by GE_enumerate_inputs(). If Z and/or M/W do not exist in your model, then treat them as constants 0. For example, if Z doesn't exist and W includes 2 covariates, then set cov(EZ) = 0 and cov(ZW) = (0,0). If describing expectations relating two vectors, i.e. Z includes two covariates and W includes three covariates, sort by the first term

and then the second. Thus in the example, the first three terms of $\text{cov}(ZW)$ are $\text{cov}(Z_1, W_1), \text{cov}(Z_1, W_2), \text{cov}(Z_1, W_3)$, and the last three terms are $\text{cov}(Z_3, W_1), \text{cov}(Z_3, W_2), \text{cov}(Z_3, W_3)$.

`cov_mat_list` A list of matrices of expectations as specified by `GE_enumerate_inputs()`.
`mu_list` A list of means as specified by `GE_enumerate_inputs()`.
`HOM_list` A list of higher order moments as specified by `GE_enumerate_inputs()`.

Value

A list of the fitted coefficients alpha

Examples

```
solutions <- GE_bias_normal_squaredmis( beta_list=as.list(runif(n=6, min=0, max=1)),
rho_list=as.list(rep(0.3,6)), prob_G=0.3, cov_Z=1, cov_W=1)
GE_nleqslv(beta_list=solutions$beta_list, solutions$cov_list, solutions$cov_mat_list,
solutions$mu_list, solutions$HOM_list)
```

GE_scoreeq_sim *GE_scoreeq_sim.R*

Description

Here we perform simulation to verify that we have solved for the correct alpha values in `GE_bias_norm_squaredmis()`. Make the same assumptions as in `GE_bias_norm_squaredmis()`.

Usage

```
GE_scoreeq_sim(num_sims = 5000, num_sub = 2000, beta_list, rho_list,
prob_G, cov_Z = NULL, cov_W = NULL, corr_G = NULL)
```

Arguments

`num_sims` The number of simulations to run, we suggest 5000.
`num_sub` The number of subjects to generate in every simulation, we suggest 2000.
`beta_list` A list of the effect sizes in the true model. Use the order `beta_0, beta_G, beta_E, beta_I, beta_Z, beta_M`. If G or Z or M is a vector, then `beta_G/beta_Z/beta_M` should be vectors. If Z and/or M/W do not exist in your model, then set `beta_Z` and/or `beta_M = 0`.
`rho_list` A list of expectations (which happen to be covariances if all covariates are centered at 0) in the order specified by `GE_enumerate_inputs()`. If Z and/or M/W do not exist in your model, then treat them as constants 0. For example, if Z doesn't exist and W includes 2 covariates, then set $\text{cov}(EZ) = 0$ and $\text{cov}(ZW) = (0,0)$. If describing expectations relating two vectors, i.e. Z includes two covariates and W includes three covariates, sort by the first term and then the second. Thus in the example, the first three terms of $\text{cov}(ZW)$ are $\text{cov}(Z_1, W_1), \text{cov}(Z_1, W_2), \text{cov}(Z_1, W_3)$, and the last three terms are $\text{cov}(Z_3, W_1), \text{cov}(Z_3, W_2), \text{cov}(Z_3, W_3)$.

prob_G	Probability that each allele is equal to 1. Since each SNP has two alleles, the expectation of G is $2 * \text{prob_G}$. Should be a $d * 1$ vector.
cov_Z	Should be a matrix equal to $\text{cov}(Z)$ or NULL if no Z.
cov_W	Should be a matrix equal to $\text{cov}(W)$ or NULL if no W.
corr_G	Should be a matrix giving the <i>*pairwise correlations*</i> between each SNP in the set, or NULL. Must be specified if G is a vector. For example, the [2,3] element of the matrix would be the pairwise correlation between SNP2 and SNP3.

Value

A list of the fitted values alpha

Examples

```
GE_scoreeq_sim( num_sims=10, num_sub=1000, beta_list=as.list(runif(n=6, min=0, max=1)),  
rho_list=as.list(rep(0.3,6)), prob_G=0.3, cov_Z=1, cov_W=1)
```

Index

GE_bias, 2
GE_bias_normal_squaredmis, 3
GE_BICS, 4
GE_nleqslv, 5
GE_scoreeq_sim, 6