

# Package ‘GWAF’

April 17, 2012

**Type** Package

**Title** Genome-wide association/interaction analyses with family data

**Version** 2.0

**Date** 2012-03-14

**Author** Ming-Huei Chen <mhchen@bu.edu> and Qiong Yang <qyang@bu.edu>

**Maintainer** Ming-Huei Chen <mhchen@bu.edu>

**Depends** R(>= 2.9.0), geepack, survival, methods, nlme, lattice

**Description** Functions for genome-wide association analyses and genome-wide interaction analyses on a continuous/dichotomous trait using family data, and for making genome-wide p-value plot and QQ plot.

**License** GPL (>= 2)

**Repository** CRAN

**Date/Publication** 2012-04-17 17:48:31

## R topics documented:

GWAF-package . . . . .	2
auto . . . . .	3
bdsmatrix-class . . . . .	4
gchol . . . . .	6
gchol-class . . . . .	7
gchol-methods . . . . .	7
gchol.bdsmatrix-class . . . . .	8
geepack.lgst . . . . .	9
geepack.lgst.batch . . . . .	10
geepack.lgst.batch.imputed . . . . .	12
geepack.lgst.imputed . . . . .	14
geepack.lgst.int . . . . .	15
geepack.lgst.int.batch . . . . .	16

geepack.lgst.int.batch.imputed . . . . .	19
geepack.lgst.int.imputed . . . . .	21
geepack.quant.batch . . . . .	22
geepack.quant.batch.imputed . . . . .	24
geepack.quant.int.batch . . . . .	26
geepack.quant.int.batch.imputed . . . . .	28
GWplot . . . . .	29
lmekin . . . . .	30
lmepack.batch . . . . .	31
lmepack.batch.imputed . . . . .	33
lmepack.int.batch . . . . .	34
lmepack.int.batch.imputed . . . . .	37
makekinship . . . . .	38
qq . . . . .	39

<b>Index</b>	<b>40</b>
--------------	-----------

---

GWAF-package

*Genome-wide association/interaction analyses with family data*

---

## Description

For continuous traits, GWAF package provides two sets of functions for each of genome-wide association analyses and genome-wide interaction analyses with observed/imputed SNP genotypes for family data. One fits Linear Mixed Effects (LME) model and the other fits Generalized Estimation Equation (GEE) model to accounting for within pedigree correlation. While for dichotomous trait, GWAF package provides functions to fit GEE model for genome-wide association analyses and genome-wide interaction analyses. For LME, previous version of GWAF depends on kinship package that has been archived. In the current version, GWAF includes functions from kinship. In addition, GWAF package also provides functions for making genome-wide p-values plot and QQ plot that contains genomic control parameter estimate and generating scripts for genome-wide association analysis.

## Details

Package: GWAF  
 Type: Package  
 Version: 2.0  
 Date: 2012-02-27  
 License: GPL (>= 2)

## Author(s)

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

Maintainer: Ming-Huei Chen <mhchen@bu.edu>

---

auto *function to generate scripts for genome-wide association/interaction analysis*

---

## Description

Given a path/directory (genopath) that keeps genotype files, phenotype file, pedigree file, phenotype of interest, covariates, analysis of interest (can be 'lme', 'lme.imputed', 'geepack', 'geepack.imputed', 'geepack.quant', 'geepack.quant.imputed', 'lme.int', 'lme.int.imputed', 'geepack.int', 'geepack.int.imputed', 'geepack.quant.int', 'geepack.quant.int.imputed') and other arguments, auto function generates one R script, one shell script that can excute R script, and one list file that can excute all shell scripts in batch mode, for each genotype file. Once the list file (XXXX.lst) is generated, user can use ksh XXXX.lst to submit all jobs to test all SNPs in genopath.

## Usage

```
auto(genopath, phenfile, pedfile, outfile, phen, covars, cov.int, sub="N", analysis, lib.loc, model = N, kinmat = NULL, col.names = F, sep.ped = ",", sep.phe = ",", sep.gen = ",")
```

## Arguments

genopath	a character string indicating the path/directory that keeps genotype files to be analyzed
phenfile	a character string naming the phenotype file for reading (see format requirement in details)
pedfile	a character string naming the pedigree file for reading (see format requirement in details)
outfile	a character string naming the result file for writing
phen	a character string for a phenotype name in phenfile
covars	a character vector for covariates in phenfile
cov.int	a character string naming the covariate for interaction, the covariate has to be included in covars
sub	"N" (default) for no stratified analysis, and "Y" for requesting stratified analyses (only when cov.int is dichotomous)
analysis	a character string indicating the analysis of interest available in GWAF package, can be 'lme', 'lme.imputed', 'gee' or 'gee.imputed'
lib.loc	a character string indicating the location of GWAF package
model	a single character of 'a', 'd', 'g', or 'r', with 'a'=additive, 'd'=dominant, 'g'=general and 'r'=recessive models; Not appropriate/needed for analyzing imputed SNPs
kinmat	a character string naming the file where kinship coefficient matrix is kept; needed for LME analyses

col.names	a logical value indicating whether the output file should contain column names
sep.ped	the field separator character for pedigree file
sep.phe	the field separator character for phenotype file
sep.gen	the field separator character for genotype file

### Details

auto function generates one R script, one shell script that can excute R script, and one list file that can excute all shell scripts in batch mode. These scripts are named based on the phenotype of interest, the analysis of interest and the time these scripts are generated. After generating these scripts, auto function generitates a message telling the user how to submit ALL the jobs (using ksh XXXX.lst). When a submitted job is completed, a log file indicating which genotype file was analyzed will be generated and the R script and the shell script will be removed. The number of log files should equal to the number of genotype files, if all jobs are completed. All the results will be written and appended to the user specified single output file. Different outfile should be assigned for different genopath to avoid over-writing.

### Value

No value is returned. Instead, results are written to outfile.

### Author(s)

Ming-Huei Chen <mhchen@bu.edu> and Qiong Yang <qyang@bu.edu>

---

bdsmatrix-class      *Class "bdsmatrix"*

---

### Description

A bdsmatrix class, included from archived kinship package that can be found at <http://cran.r-project.org/web/packages/kinship/index.html>.

### Objects from the Class

Objects can be created by calls of the form `new("bdsmatrix", ...)`.

### Slots

**blocksize:** Object of class "integer" vector of sizes for the matrices on the diagonal  
**blocks:** Object of class "numeric" contents of the diagonal blocks, strung out as a vector  
**rmat:** Object of class "matrix" the dense portion of the matrix, forming a right and lower border  
**offdiag:** Object of class "numeric" 0s  
**.Dim:** Object of class "integer" dimensions  
**.Dimnames:** Object of class "list" or "NULL" a list of dimension names for the matrix

**Extends**

Class "matrix", directly.

**Methods**

```

%% signature(x = "matrix", y = "bdsmatrix"):
%% signature(x = "numeric", y = "bdsmatrix"):
Math2 signature(x = "bdsmatrix"):
Math signature(x = "bdsmatrix"):
Ops signature(e1 = "bdsmatrix", e2 = "numeric"):
Ops signature(e1 = "bdsmatrix", e2 = "bdsmatrix"):
Ops signature(e1 = "numeric", e2 = "bdsmatrix"):
[ signature(x = "bdsmatrix"):
all signature(x = "bdsmatrix"):
any signature(x = "bdsmatrix"):
coerce signature(from = "bdsmatrix", to = "matrix"):
coerce signature(from = "bdsmatrix", to = "vector"):
diag signature(x = "bdsmatrix"):
diag<- signature(x = "bdsmatrix"):
dim signature(x = "bdsmatrix"):
dimnames signature(x = "bdsmatrix"):
dimnames<- signature(x = "bdsmatrix"):
max signature(x = "bdsmatrix"):
min signature(x = "bdsmatrix"):
prod signature(x = "bdsmatrix"):
range signature(x = "bdsmatrix"):
show signature(object = "bdsmatrix"):
sum signature(x = "bdsmatrix"):
unique signature(x = "bdsmatrix", incomparables = "missing"):

```

---

`gchol`*Generalized Cholesky decomposition*

---

**Description**

Perform the generalized Cholesky decomposition of a real symmetric matrix. The function and R documentation are included from the archived kinship package that can be found at <http://cran.r-project.org/web/packages/kinship/index.html>.

**Usage**

```
gchol(x, tolerance=1e-10)
```

**Arguments**

<code>x</code>	the symmetric matrix to be factored
<code>tolerance</code>	the numeric tolerance for detection of singular columns in <code>x</code> .

**Details**

A symmetric matrix  $A$  can be decomposed as  $LDL'$ , where  $L$  is a lower triangular matrix with 1's on the diagonal,  $L'$  is the transpose of  $L$ , and  $D$  is diagonal. The inverse of  $L$  is also lower-triangular, with 1's on the diagonal. If all elements of  $D$  are positive, then  $A$  must be symmetric positive definite (SPD), and the solution can be reduced to the usual Cholesky decomposition  $U'U$  where  $U$  is upper triangular and  $U = \text{sqrt}(D) L'$ .

The main advantage of the generalized form is that it admits of matrices that are not of full rank:  $D$  will contain zeros marking the redundant columns, and the rank of  $A$  is the number of non-zero columns. If all elements of  $D$  are zero or positive, then  $A$  is a non-negative definite (NND) matrix. The generalized form also has the (quite minor) numerical advantage of not requiring square roots during its calculation. To extract the components of the decomposition, use the `diag` and `as.matrix` functions.

The `solve` has a method for `gchol` decompositions, and there are `gchol` methods for block diagonal symmetric (`bdsmatrix`) matrices as well.

**Value**

an object of class `gchol` containing the generalized Cholesky decomposition. It has the appearance of a lower triangular matrix.

---

gchol-class	<i>Class "gchol"</i>
-------------	----------------------

---

### Description

a class of gchol, included from archived kinship package that can be found at <http://cran.r-project.org/web/packages/kinship/>

### Objects from the Class

Objects can be created by calls of the form `new("gchol", ...)`.

### Slots

`.Data`: Object of class "numeric" ~~  
`.Dim`: Object of class "integer" ~~  
`.Dimnames`: Object of class "list or NULL" ~~  
`rank`: Object of class "integer" ~~

### Methods

**coerce** signature(from = "gchol", to = "matrix"): ...  
**diag** signature(x = "gchol"): ...  
**show** signature(object = "gchol"): ...

---

gchol-methods	<i>Methods for Function gchol in Package 'kinship'</i>
---------------	--

---

### Description

gchol methods, included from archived kinship package that can be found at <http://cran.r-project.org/web/packages/kinship/>

### Methods

**x = "matrix"** an ordinary matrix or bdsmatrix object

---

gchol.bdsmatrix-class *Class "gchol.bdsmatrix"*

---

### Description

a class generated from gchol(bdsmatrix object), included from the archived kinship package that can be found at <http://cran.r-project.org/web/packages/kinship/index.html>.

### Objects from the Class

Objects can be created by calls of the form `new("gchol.bdsmatrix", ...)`. or `gchol(bdsmatrix object)`

### Slots

**blocksize:** Object of class "integer" ~~

**blocks:** Object of class "numeric" ~~

**rmat:** Object of class "matrix" ~~

**rank:** Object of class "integer" ~~

**.Dim:** Object of class "integer" ~~

**.Dimnames:** Object of class "list or NULL" ~~

### Methods

**%\*%** signature(x = "matrix", y = "gchol.bdsmatrix"): ...

**%\*\*%** signature(x = "numeric", y = "gchol.bdsmatrix"): ...

**[** signature(x = "gchol.bdsmatrix"): ...

**coerce** signature(from = "gchol.bdsmatrix", to = "matrix"): ...

**diag** signature(x = "gchol.bdsmatrix"): ...

**dim** signature(x = "gchol.bdsmatrix"): ...

**show** signature(object = "gchol.bdsmatrix"): ...

---

geepack.lgst	<i>function for testing association between a dichotomous trait and a genotyped SNP in family data using Generalized Estimation Equation model</i>
--------------	--

---

## Description

Fit logistic regression via GEE to test association between a dichotomous phenotype and one genotyped SNP in a genotype file with user specified genetic model. Each family is treated as a cluster, with independence working correlation matrix used in the robust variance estimator. The trait-SNP association test is carried out by the `geese` function from package `geepack`. This function is called in `geepack.lgst.batch` function to apply association test to all SNPs in the genotype data.

## Usage

```
geepack.lgst(snp, phen, test.dat, covar = NULL, model = "a")
```

## Arguments

<code>snp</code>	genotype data of a SNP
<code>phen</code>	a character string for a phenotype name in <code>test.dat</code>
<code>test.dat</code>	the product of merging phenotype, genotype and pedigree data, should be ordered by "famid"
<code>covar</code>	a character vector for covariates in <code>test.dat</code>
<code>model</code>	a single character of 'a', 'd', 'g', or 'r', with 'a'=additive, 'd'=dominant, 'g'=general and 'r'=recessive models

## Details

The `geepack.lgst` function tests association between a dichotomous trait and a SNP from a dataset that contains phenotype, genotype and pedigree data (`test.dat`), where the dataset needs to be ordered by `famid`.

## Value

Please see output in `geepack.lgst.batch`.

## Author(s)

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

## References

- Liang, K.Y. and Zeger, S.L. (1986) Longitudinal data analysis using generalized linear models. *Biometrika*, **73** 13–22.
- Zeger, S.L. and Liang, K.Y. (1986) Longitudinal data analysis for discrete and continuous outcomes. *Biometrics*, **42** 121–130.
- Yan, J and Fine, J. (2004) Estimating equations for association structures. *Stat Med*, **23** 859–874.

**See Also**

geese function from package geepack

---

geepack.lgst.batch	<i>function to test genetic association between a dichotomous trait and a batch of genotyped SNPs in families using Generalized Estimation Equation model</i>
--------------------	---

---

**Description**

Fit logistic regression via Generalized Estimation Equation (GEE) to test association between a dichotomous phenotype and all genotyped SNPs in a genotype file in family data with user specified genetic model. Each pedigree is treated as a cluster, with independence working correlation matrix used in the robust variance estimator. This function applies the same trait-SNP association test to all SNPs in the genotype data. The trait-SNP association test is carried out by `geepack.lgst` function where the `geese` function from package `geepack` is used.

**Usage**

```
geepack.lgst.batch(genfile, phenfile, pedigree, outfile, phen, covars = NULL,
model = "a", col.names = T, sep.ped = ",", sep.phe = ",", sep.gen = ",")
```

**Arguments**

genfile	a character string naming the genotype file for reading (see format requirement in details)
phenfile	a character string naming the phenotype file for reading (see format requirement in details)
pedfile	a character string naming the pedigree file for reading (see format requirement in details)
outfile	a character string naming the result file for writing
phen	a character string for a phenotype name in phenfile
covars	a character vector for covariates in phenfile
model	a single character of 'a', 'd', 'g', or 'r', with 'a'=additive, 'd'=dominant, 'g'=general and 'r'=recessive models
col.names	a logical value indicating whether the output file should contain column names
sep.ped	the field separator character for pedigree file
sep.phe	the field separator character for phenotype file
sep.gen	the field separator character for genotype file

## Details

The `geepack.lgst.batch` function first reads in and merges phenotype-covariates, genotype and pedigree files, then tests the association of phen against all SNPs in `genfile`. `genfile` contains unique individual id and genotype data, with the column names being "id" and SNP names. For each genotyped SNP, the genotype data should be coded as 0, 1, 2 indicating the numbers of the coded alleles. The SNP names in genotype file should not have any dash, '-' and other special characters (dots and underscores are OK). `phenfile` contains unique individual id, phenotype and covariates data, with the column names being "id" and phenotype and covariate names. `pedfile` contains pedigree information, with the column names being "famid", "id", "fa", "mo", "sex". In all files, missing value should be an empty space, except missing parental id in `pedfile`. Only phenotypes with two categories are analyzed. A phenotype should be coded as 0 and 1, with 1 denoting affected and 0 unaffected. SNPs with low genotype counts (especially minor allele homozygote) may be omitted or analyzed with dominant model or analyzed with logistic regression. The `geepack.lgst.batch` function fits GEE model using each pedigree as a cluster with `geepack.lgst` function from GWAF package and `geese` function from `geepack` package.

## Value

No value is returned. Instead, results are written to `outfile`. When the genetic model is 'a', 'd' or 'r', the result includes the following columns. When the genetic model is 'g', beta and se are replaced with `beta10`, `beta20`, `beta21`, `se10`, `se20`, and `se21`.

<code>phen</code>	phenotype name
<code>snp</code>	SNP name
<code>n0</code>	the number of individuals with 0 copy of coded alleles
<code>n1</code>	the number of individuals with 1 copy of coded alleles
<code>n2</code>	the number of individuals with 2 copies of coded alleles
<code>nd0</code>	the number of individuals with 0 copy of coded alleles in affected sample
<code>nd1</code>	the number of individuals with 1 copy of coded alleles in affected sample
<code>nd2</code>	the number of individuals with 2 copies of coded alleles in affected sample
<code>miss.0</code>	Genotype missing rate in unaffected sample
<code>miss.1</code>	Genotype missing rate in affected sample
<code>miss.diff.p</code>	P-value of differential missingness test between unaffected and affected samples
<code>beta</code>	regression coefficient of SNP covariate
<code>se</code>	standard error of beta
<code>chisq</code>	Chi-square statistic for testing beta not equal to zero
<code>df</code>	degree of freedom of the Chi-square statistic
<code>model</code>	model actually used in the analysis
<code>remark</code>	warning or additional information for the analysis, 'not converged' indicates the GEE analysis did not converge; 'logistic reg' indicates GEE model is replaced by logistic regression; 'exp count<5' indicates any expected count is less than 5 in phenotype-genotype table; 'not converged and exp count<5', 'logistic reg & exp count<5' are noted similarly; 'collinearity' indicates collinearity exists between SNP and some covariates

pval	p-value of the chi-square statistic
beta10	regression coefficient of genotype with 1 copy of coded allele vs. that with 0 copy
beta20	regression coefficient of genotype with 2 copy of coded allele vs. that with 0 copy
beta21	regression coefficient of genotype with 2 copy of coded allele vs. that with 1 copy
se10	standard error of beta10
se20	standard error of beta20
se21	standard error of beta21

**Author(s)**

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

---

`geepack.lgst.batch.imputed`

*function to test genetic association between a dichotomous trait and a batch of imputed SNPs in families using Generalized Estimation Equation model*

---

**Description**

Fit logistic regression via Generalized Estimation Equation (GEE) to test association between a dichotomous phenotype and all imputed SNPs in a genotype file in family data under additive genetic model. Each family is treated as a cluster, with independence working correlation matrix used in the robust variance estimator. This function applies the same trait-SNP association test to all SNPs in the imputed genotype data. The trait-SNP association test is carried out by `geepack.lgst.imputed` function where the `geese` function from package `geepack` is used.

**Usage**

```
geepack.lgst.batch.imputed(genfile, phenfile, pedfile, outfile, phen,
covars = NULL, col.names = T, sep.ped = ",", sep.phe = ",", sep.gen = ",")
```

**Arguments**

genfile	a character string naming the genotype file for reading (see format requirement in details)
phenfile	a character string naming the phenotype file for reading (see format requirement in details)
pedfile	a character string naming the pedigree file for reading (see format requirement in details)

outfile	a character string naming the result file for writing
phen	a character string for a phenotype name in phenfile
covars	a character vector for covariates in phenfile
col.names	a logical value indicating whether the output file should contain column names
sep.ped	the field separator character for pedigree file
sep.phe	the field separator character for phenotype file
sep.gen	the field separator character for genotype file

### Details

Similar to the details for `geepack.lgst.batch` but here the SNP data contains imputed genotypes (allele dosages) that are continuous and range from 0 to 2. In addition, the user-specified genetic model argument is not available.

### Value

No value is returned. Instead, results are written to `outfile`.

phen	phenotype name
snp	SNP name
N	the number of individuals in analysis
Nd	the number of individuals in affected sample in analysis
AF	imputed allele frequency of coded allele
AFd	imputed allele frequency of coded allele in affected sample
beta	regression coefficient of SNP covariate
se	standard error of beta
remark	warning or additional information for the analysis, note that the genotype counts are based on rounded imputed genotypes; 'not converged' indicates the GEE analysis did not converge; 'logistic reg' indicates GEE model is replaced by logistic regression; 'exp count<5' indicates any expected count is less than 5 in phenotype-genotype table; 'not converged and exp count<5', 'logistic reg & exp count<5' are noted similarly; 'collinearity' indicates collinearity exists between SNP and some covariates
pval	p-value of the association test based on chi-square statistic

### Author(s)

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

---

geepack.lgst.imputed    *function for testing association between a dichotomous trait and an imputed SNP in family data using Generalized Estimation Equation model*

---

### Description

Fit logistic regression via Generalized Estimation Equation (GEE) to test association between a dichotomous phenotype and one imputed SNP in a genotype file in family data under additive genetic model. Each family is treated as a cluster, with independence working correlation matrix used in the robust variance estimator. The trait-SNP association test is carried out by the geese function from package geepack. This function is called in geepack.lgst.batch.imputed function to apply association test to all imputed SNPs in a genotype file.

### Usage

```
geepack.lgst.imputed(snp, phen, test.dat, covar = NULL)
```

### Arguments

snp	imputed genotype data of a SNP
phen	a character string for a phenotype name in test.dat
test.dat	the product of merging phenotype, genotype and pedigree data, should be ordered by "famid"
covar	a character vector for covariates in test.dat

### Details

Similar to the details for geepack.lgst function but here the SNP data contains imputed genotypes (allele dosages) that are continuous and range from 0 to 2. In addition, the user-specified genetic model argument is not available.

### Value

Please see output in geepack.lgst.batch.imputed.

### Author(s)

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

### References

- Liang, K.Y. and Zeger, S.L. (1986) Longitudinal data analysis using generalized linear models. *Biometrika*, **73** 13–22.
- Zeger, S.L. and Liang, K.Y. (1986) Longitudinal data analysis for discrete and continuous outcomes. *Biometrics*, **42** 121–130.
- Yan, J and Fine, J. (2004) Estimating equations for association structures. *Stat Med*, **23** 859–874.

**See Also**

geese function from package geepack

---

geepack.lgst.int	<i>function for testing gene-environment or gene-gene interaction between a dichotomous trait and a genotyped SNP in family data using Generalized Estimation Equation model</i>
------------------	--

---

**Description**

Fit logistic regression via Generalized Estimation Equation (GEE) to test gene-environment or gene-gene interaction between a dichotomous phenotype and one genotyped SNP in a genotype file in family data under additive genetic model. The interaction term is the product of SNP genotype and the covariate for interaction (`cov.int`). The covariate for interaction (`cov.int`) can be SNP genotype (gene-gene interaction) or an environmental factor (gene-environment interaction). Only one interaction term is allowed. When `cov.int` is dichotomous, stratified analyses can be requested by specifying `sub="Y"`. The covariance between the main effect (SNP) and the interaction effect is provided in the output when stratified analysis is not requested. Each family is treated as a cluster, with independence working correlation matrix used in the robust variance estimator. The interaction test is carried out by the `geese` function from package `geepack`. This function is called in `geepack.lgst.int.batch` function to apply interaction test to all SNPs in a genotype file.

**Usage**

```
geepack.lgst.int(snp,phen,test.dat,covar,cov.int,sub="N")
```

**Arguments**

snp	genotype data of a SNP
phen	a character string for a phenotype name in <code>test.dat</code>
test.dat	the product of merging phenotype, genotype and pedigree data, should be ordered by "famid"
covar	a character vector for covariates in <code>test.dat</code>
cov.int	a character string naming the covariate for interaction, the covariate has to be included in <code>covar</code>
sub	"N" (default) for no stratified analysis, and "Y" for requesting stratified analyses (only when <code>cov.int</code> is dichotomous)

**Details**

The `geepack.lgst.int` function tests gene-environment or gene-gene interaction between a dichotomous trait and a SNP from a dataset that contains phenotype, genotype and pedigree data (`test.dat`), where the dataset needs to be ordered by `famid`. Please also see details in details for `geepack.lgst.int.batch` function.

**Value**

Please see value in `geepack.lgst.int.batch` function.

**Author(s)**

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

**References**

- Liang, K.Y. and Zeger, S.L. (1986) Longitudinal data analysis using generalized linear models. *Biometrika*, **73** 13–22.
- Zeger, S.L. and Liang, K.Y. (1986) Longitudinal data analysis for discrete and continuous outcomes. *Biometrics*, **42** 121–130.
- Yan, J and Fine, J. (2004) Estimating equations for association structures. *Stat Med*, **23** 859–874.

**See Also**

`geese` function from package `geepack`

---

`geepack.lgst.int.batch`

*function to test gene-environment or gene-gene interaction between a dichotomous trait and a batch of genotyped SNPs in families using Generalized Estimation Equation model*

---

**Description**

Fit logistic regression via Generalized Estimation Equation (GEE) to test gene-environment or gene-gene interaction for a dichotomous phenotype and all genotyped SNPs in a genotype file in family data under additive genetic model. The interaction term is the product of SNP genotype and a covariate for interaction (`cov.int`). The covariate for interaction (`cov.int`) can be SNP genotype (gene-gene interaction) or an environmental factor (gene-environment interaction). Only one interaction term is allowed. When `cov.int` is dichotomous, stratified analyses can be requested by specifying `sub="Y"`. The covariance between the main effect (SNP) and the interaction effect is provided in the output when stratified analysis is not requested. Each pedigree is treated as a cluster with independence working correlation matrix used in the robust variance estimator. This function applies the same interaction test to all SNPs in a genotype file. The interaction test is carried out by `geepack.lgst.int` function from GWAF where the `geese` function from package `geepack` is used.

**Usage**

```
geepack.lgst.int.batch(genfile,phenfile,pedfile,outfile,phen,covars,cov.int,sub="N",
col.names=T,sep.ped=",",sep.phe=",",sep.gen=",")
```

**Arguments**

genfile	a character string naming the genotype file for reading (see format requirement in details)
phenfile	a character string naming the phenotype file for reading (see format requirement in details)
pedfile	a character string naming the pedigree file for reading (see format requirement in details)
outfile	a character string naming the result file for writing
phen	a character string for a phenotype name in phenfile
covars	a character vector for covariates in phenfile
cov.int	a character string naming the covariate for interaction, the covariate has to be included in covars
sub	"N" (default) for no stratified analysis, and "Y" for requesting stratified analyses (only when cov.int is dichotomous)
col.names	a logical value indicating whether the output file should contain column names
sep.ped	the field separator character for pedigree file
sep.phen	the field separator character for phenotype file
sep.gen	the field separator character for genotype file

**Details**

The `geepack.lgst.int.batch` function first reads in and merges phenotype-covariates, genotype and pedigree files, then tests gene-environment or gene-gen interaction for phen against all SNPs in `genfile`. Only one interaction term is allowed, so is the covariate for interaction (`cov.int`). When `cov.int` is dichotomous, stratified analyses can be requested by specifying `sub="Y"`. The covariance between the main effect (SNP) and the interaction effect is provided in the output when stratified analysis is not requested. `genfile` contains unique individual id and genotype data, with the column names being "id" and SNP names. For each genotyped SNP, the genotype data should be coded as 0, 1, 2 indicating the numbers of the coded alleles. The SNP names in genotype file should not have any dash, '-' and other special characters (dots and underscores are OK). `phenfile` contains unique individual id, phenotype and covariates data, with the column names being "id" and phenotype and covariate names. `pedfile` contains pedigree information, with the column names being "famid", "id", "fa", "mo", "sex". In all files, missing value should be an empty space, except missing parental id in `pedfile`. Only phenotypes with two categories are analyzed. A phenotype should be coded as 0 and 1, with 1 denoting affected and 0 unaffected. SNPs with low genotype counts (especially minor allele homozygote) may be omitted or analyzed with logistic regression. The `geepack.lgst.int.batch` function fits GEE model using each pedigree as a cluster with `geepack.lgst.int` function from GWAF package and `geese` function from `geepack` package.

**Value**

No value is returned. Instead, results are written to `outfile`. If stratified analyses are requested, the result file will include the following columns. Otherwise, `cov_beta_snp_beta_int` will be included instead of the results from stratified analyses, that is, `beta_snp_cov0`, `se_snp_cov0`, `pval_snp_cov0`, `beta_snp_cov1`, `se_snp_cov1`, and `pval_snp_cov1`.

phen	phenotype name
snp	SNP name
covar_int	the covariate for interaction
n	sample size used in analysis
AF	allele frequency of the coded allele
nd	the number of individuals in affected sample
AFd	allele frequency of the coded allele in affected sample
model	genetic model used in analysis, additive model only
beta_snp	regression coefficient of SNP covariate
se_snp	standard error of beta_snp
pval_snp	p-value of testing beta_snp not equal to zero
beta_snp_cov0	regression coefficient of SNP covariate in stratified analysis using the subset where cov.int level is 0
se_snp_cov0	standard error of beta_snp_cov0
pval_snp_cov0	p-value of testing beta_snp_cov0 not equal to zero
beta_snp_cov1	regression coefficient of SNP covariate in stratified analysis using the subset where cov.int level is 1
se_snp_cov1	standard error of beta_snp_cov1
pval_snp_cov1	p-value of testing beta_snp_cov1 not equal to zero
beta_int	regression coefficient of the interaction term
se_int	standard error of beta_int
pval_int	p-value of testing beta_int not equal to zero
remark	warning or additional information for the analysis, 'not converged' indicates the GEE analysis did not converge; 'logistic reg' indicates GEE model is replaced by logistic regression; 'exp count<5' indicates any expected count is less than 5 in phenotype-genotype table; 'not converged and exp count<5', 'logistic reg & exp count<5' are noted similarly; 'collinearity' indicates collinearity exists between SNP and some covariates

**Author(s)**

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

---

```
geepack.lgst.int.batch.imputed
```

*function to test gene-environment or gene-gene interaction between a dichotomous trait and a batch of imputed SNPs in families using Generalized Estimation Equation model*

---

## Description

Fit logistic regression via Generalized Estimation Equation (GEE) to test gene-environment or gene-gene interaction between a dichotomous phenotype and all imputed SNPs in a genotype file in family data under additive genetic model. The interaction term is the product of SNP genotype and a covariate for interaction (`cov.int`). The covariate for interaction (`cov.int`) can be SNP genotype (gene-gene interaction) or an environmental factor (gene-environment interaction). Only one interaction term is allowed. When `cov.int` is dichotomous, stratified analyses can be requested by specifying `sub="Y"`. The covariance between the main effect (SNP) and the interaction effect is provided in the output when stratified analysis is not requested. Each family is treated as a cluster, with independence working correlation matrix used in the robust variance estimator. This function applies the same interaction test to all SNPs in the imputed genotype data. The interaction test is carried out by `geepack.lgst.int.imputed` function from GWAF where the `geese` function from package `geepack` is used.

## Usage

```
geepack.lgst.int.batch.imputed(genfile,phenfile,pedfile,outfile,phen,covars,cov.int,sub="N",
col.names=T,sep.ped=",",sep.phe=",",sep.gen=",")
```

## Arguments

<code>genfile</code>	a character string naming the genotype file for reading (see format requirement in details)
<code>phenfile</code>	a character string naming the phenotype file for reading (see format requirement in details)
<code>pedfile</code>	a character string naming the pedigree file for reading (see format requirement in details)
<code>outfile</code>	a character string naming the result file for writing
<code>phen</code>	a character string for a phenotype name in <code>phenfile</code>
<code>covars</code>	a character vector for covariates in <code>phenfile</code>
<code>cov.int</code>	a character string naming the covariate for interaction, the covariate has to be included in <code>covars</code>
<code>sub</code>	"N" (default) for no stratified analysis, and "Y" for requesting stratified analyses (only when <code>cov.int</code> is dichotomous)
<code>col.names</code>	a logical value indicating whether the output file should contain column names
<code>sep.ped</code>	the field separator character for pedigree file
<code>sep.phe</code>	the field separator character for phenotype file
<code>sep.gen</code>	the field separator character for genotype file

**Details**

Similar to the details for `geepack.lgst.int.batch` but here the SNP data contains imputed genotypes (allele dosages) that are continuous and range from 0 to 2.

**Value**

No value is returned. Instead, results are written to outfile. If stratified analyses are requested, the result file will include the following columns. Otherwise, `cov_beta_snp_beta_int` will be included instead of the results from stratified analyses, that is, `beta_snp_cov0`, `se_snp_cov0`, `pval_snp_cov0`, `beta_snp_cov1`, `se_snp_cov1`, and `pval_snp_cov1`.

<code>phen</code>	phenotype name
<code>snp</code>	SNP name
<code>covar_int</code>	the covariate for interaction
<code>n</code>	sample size used in analysis
<code>AF</code>	allele frequency of the coded allele
<code>nd</code>	the number of individuals in affected sample
<code>AFd</code>	allele frequency of the coded allele in affected sample
<code>model</code>	genetic model used in analysis, additive model only
<code>beta_snp</code>	regression coefficient of SNP covariate
<code>se_snp</code>	standard error of <code>beta_snp</code>
<code>pval_snp</code>	p-value of testing <code>beta_snp</code> not equal to zero
<code>beta_snp_cov0</code>	regression coefficient of SNP covariate in stratified analysis using the subset where <code>cov.int</code> level is 0
<code>se_snp_cov0</code>	standard error of <code>beta_snp_cov0</code>
<code>pval_snp_cov0</code>	p-value of testing <code>beta_snp_cov0</code> not equal to zero
<code>beta_snp_cov1</code>	regression coefficient of SNP covariate in stratified analysis using the subset where <code>cov.int</code> level is 1
<code>se_snp_cov1</code>	standard error of <code>beta_snp_cov1</code>
<code>pval_snp_cov1</code>	p-value of testing <code>beta_snp_cov1</code> not equal to zero
<code>beta_int</code>	regression coefficient of the interaction term
<code>se_int</code>	standard error of <code>beta_int</code>
<code>pval_int</code>	p-value of testing <code>beta_int</code> not equal to zero
<code>remark</code>	warning or additional information for the analysis, 'not converged' indicates the GEE analysis did not converge; 'logistic reg' indicates GEE model is replaced by logistic regression; 'exp count<5' indicates any expected count is less than 5 in phenotype-genotype table; 'not converged and exp count<5', 'logistic reg & exp count<5' are noted similarly; 'collinearity' indicates collinearity exists between SNP and some covariates

**Author(s)**

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

---

geepack.lgst.int.imputed

*function for testing gene-environment or gene-gene interaction between a dichotomous trait and an imputed SNP in family data using Generalized Estimation Equation model*

---

## Description

Fit logistic regression via Generalized Estimation Equation (GEE) to test gene-environment or gene-gene interaction between a dichotomous phenotype and one imputed SNP in a genotype file under additive genetic model. The interaction term is the product of SNP genotype and a covariate for interaction (`cov.int`). The covariate for interaction (`cov.int`) can be SNP genotype (gene-gene interaction) or an environmental factor (gene-environment interaction). Only one interaction term is allowed. When `cov.int` is dichotomous, stratified analyses can be requested by specifying `sub="Y"`. The covariance between the main effect (SNP) and the interaction effect is provided in the output when stratified analysis is not requested. Each family is treated as a cluster, with independence working correlation matrix used in the robust variance estimator. This function is called in `geepack.lgst.int.batch.imputed` function to apply interaction test to all imputed SNPs in a genotype file. The interaction test is carried out by the `geese` function from package `geepack`.

## Usage

```
geepack.lgst.int.imputed(snp,phen,test.dat,covar,cov.int,sub="N")
```

## Arguments

<code>snp</code>	genotype data of a SNP
<code>phen</code>	a character string for a phenotype name in <code>test.dat</code>
<code>test.dat</code>	the product of merging phenotype, genotype and pedigree data, should be ordered by "famid"
<code>covar</code>	a character vector for covariates in <code>test.dat</code>
<code>cov.int</code>	a character string naming the covariate for interaction, the covariate has to be included in <code>covar</code>
<code>sub</code>	"N" (default) for no stratified analysis, and "Y" for requesting stratified analyses (only when <code>cov.int</code> is dichotomous)

## Details

Similar to the details for `geepack.lgst.int.batch` function but here the SNP data contains imputed genotypes (allele dosages) that are continuous and range from 0 to 2.

## Value

Please see value in `geepack.lgst.int.batch.imputed` function.

**Author(s)**

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

**References**

Liang, K.Y. and Zeger, S.L. (1986) Longitudinal data analysis using generalized linear models. *Biometrika*, **73** 13–22.

Zeger, S.L. and Liang, K.Y. (1986) Longitudinal data analysis for discrete and continuous outcomes. *Biometrics*, **42** 121–130.

Yan, J and Fine, J. (2004) Estimating equations for association structures. *Stat Med*, **23** 859–874.

**See Also**

geese function from package geepack

---

geepack.quant.batch	<i>function to test genetic associations between a continuous trait and a batch of genotyped SNPs in families using Generalized Estimation Equation model</i>
---------------------	---

---

**Description**

Fit Generalized Estimation Equation (GEE) model to test association between a continuous phenotype and all genotyped SNPs in a genotype file in family data with user specified genetic model. Each pedigree is treated as a cluster, with independence working correlation matrix used in the robust variance estimator. The proportion of phenotype variation explained by the tested SNP is not provided. This function applies the same trait-SNP association test to all genotyped SNPs in the genotype data. The trait-SNP association test is carried out by using the geese function from package geepack.

**Usage**

```
geepack.quant.batch(phenfile,genfile,pedfile,phen,model="a",covars=NULL,outfile,
col.names=T,sep.ped=" ",sep.phe=" ",sep.gen=" ")
```

**Arguments**

genfile	a character string naming the genotype file for reading (see format requirement in details)
phenfile	a character string naming the phenotype file for reading (see format requirement in details)
pedfile	a character string naming the pedigree file for reading (see format requirement in details)
outfile	a character string naming the result file for writing
phen	a character string for a phenotype name in phenfile

covars	a character vector for covariates in phenfile
model	a single character of 'a', 'd', 'g', or 'r', with 'a'=additive, 'd'=dominant, 'g'=general and 'r'=recessive models
col.names	a logical value indicating whether the output file should contain column names
sep.ped	the field separator character for pedigree file
sep.phen	the field separator character for phenotype file
sep.gen	the field separator character for genotype file

### Details

For a continuous trait, the `geepack.quant.batch` function first reads in and merges phenotype-covariates, genotype and pedigree files, then tests the association of phen against all SNPs in `genfile`. `genfile` contains unique individual id and genotype data, with the column names being "id" and SNP names. For each SNP, the genotype data should be coded as 0, 1, 2 indicating the numbers of the coded alleles. The SNP name in genotype file should not have any dash, '-' and other special characters (dots and underscores are OK). `phenfile` contains unique individual id, phenotype and covariates data, with the column names being "id" and phenotype and covariate names. `pedfile` contains pedigree information, with the column names being "famid", "id", "fa", "mo", "sex". In all files, missing value should be an empty space, except missing parental id in `pedfile`. SNPs with low genotype counts (especially minor allele homozygote) may be omitted or analyzed with dominant model. The `geepack.quant.batch` function fits GEE model using each pedigree as a cluster with `geese` function from `geepack` package.

### Value

No value is returned. Instead, results are written to `outfile`. When the genetic model is 'a', 'd' or 'r', the result includes the following columns. When the genetic model is 'g', beta and se are replaced with `beta10`, `beta20`, `beta21`, `se10`, `se20`, `se21`.

phen	phenotype name
snp	SNP name
n0	the number of individuals with 0 copy of coded alleles
n1	the number of individuals with 1 copy of coded alleles
n2	the number of individuals with 2 copies of coded alleles
beta	regression coefficient of SNP covariate
se	standard error of beta
chisq	Chi-square statistic for testing beta not equal to zero
df	degree of freedom of the Chi-square statistic
model	model actually used in the analysis
pval	p-value of the chi-square statistic
beta10	regression coefficient of genotype with 1 copy of coded allele vs. that with 0 copy

beta20	regression coefficient of genotype with 2 copy of coded allele vs. that with 0 copy
beta21	regression coefficient of genotype with 2 copy of coded allele vs. that with 1 copy
se10	standard error of beta10
se20	standard error of beta20
se21	standard error of beta21

**Author(s)**

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

**References**

- Liang, K.Y. and Zeger, S.L. (1986) Longitudinal data analysis using generalized linear models. *Biometrika*, **73** 13–22.
- Zeger, S.L. and Liang, K.Y. (1986) Longitudinal data analysis for discrete and continuous outcomes. *Biometrics*, **42** 121–130.
- Yan, J and Fine, J. (2004) Estimating equations for association structures. *Stat Med*, **23** 859–874.

---

geepack.quant.batch.imputed

*function to test associations between a continuous trait and a batch of imputed SNPs in families using Generalized Estimation Equation model*

---

**Description**

Fit Generalized Estimation Equation (GEE) model to test association between a continuous phenotype and all imputed SNPs in a genotype file in family data under additive genetic model. Each family is treated as a cluster, with independence working correlation matrix used in the robust variance estimator. The proportion of phenotype variation explained by the tested SNP is not provided. This function applies the same trait-SNP association test to all imputed SNPs in the genotype data. The trait-SNP association test is carried out by using the geese function from package geepack.

**Usage**

```
geepack.quant.batch.imputed(phenfile, genfile, pedfile, phen, covars = NULL,
outfile, col.names = T, sep.ped = ",", sep.phe = ",", sep.gen = ",")
```

**Arguments**

phenfile	a character string naming the phenotype file for reading (see format requirement in details)
genfile	a character string naming the genotype file for reading (see format requirement in details)
pedfile	a character string naming the pedigree file for reading (see format requirement in details)
phen	a character string for a phenotype name in phenfile
covars	a character vector for covariates in phenfile
outfile	a character string naming the result file for writing
col.names	a logical value indicating whether the output file should contain column names
sep.ped	the field separator character for pedigree file
sep.phe	the field separator character for phenotype file
sep.gen	the field separator character for genotype file

**Details**

Similar to the details for `geepack.quant.batch` function but here the SNP data contains imputed genotypes (allele dosages) that are continuous and range from 0 to 2. In addition, the user specified genetic model argument is not available.

**Value**

No value is returned. Instead, results are written to `outfile`.

phen	phenotype name
snp	SNP name
N	the number of individuals in analysis
AF	imputed allele frequency of coded allele
beta	regression coefficient of SNP covariate
se	standard error of beta
pval	p-value of testing beta not equal to zero

**Author(s)**

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

**References**

- Liang, K.Y. and Zeger, S.L. (1986) Longitudinal data analysis using generalized linear models. *Biometrika*, **73** 13–22.
- Zeger, S.L. and Liang, K.Y. (1986) Longitudinal data analysis for discrete and continuous outcomes. *Biometrics*, **42** 121–130.
- Yan, J and Fine, J. (2004) Estimating equations for association structures. *Stat Med*, **23** 859–874.

---

```
geepack.quant.int.batch
```

*function to test gene-environment or gene-gene interaction for a continuous trait and a batch of genotyped SNPs in families using Generalized Estimation Equation model*

---

## Description

Fit Generalized Estimation Equation (GEE) model to test gene-environment or gene-gene interaction for a continuous phenotype and all genotyped SNPs in a genotype file in family data under additive genetic model. The interaction term is the product of SNP genotype and a covariate for interaction (`cov.int`). The covariate for interaction (`cov.int`) can be SNP genotype (gene-gene interaction) or an environmental factor (gene-environment interaction). Only one interaction term is allowed. When `cov.int` is dichotomous, stratified analyses can be requested by specifying `sub="Y"`. The covariance between the main effect (SNP) and the interaction effect is provided in the output when stratified analysis is not requested. Each pedigree is treated as a cluster, with independence working correlation matrix used in the robust variance estimator. This function applies the same interaction test to all genotyped SNPs in the genotype data. In each test for interaction, the `geese` function from `geepack` package is used.

## Usage

```
geepack.quant.int.batch(phenfile, genfile, pedfile, phen, covars, cov.int, sub="N", outfile,
  col.names=T, sep.ped=",", sep.phe=",", sep.gen=",")
```

## Arguments

<code>genfile</code>	a character string naming the genotype file for reading (see format requirement in details)
<code>phenfile</code>	a character string naming the phenotype file for reading (see format requirement in details)
<code>pedfile</code>	a character string naming the pedigree file for reading (see format requirement in details)
<code>outfile</code>	a character string naming the result file for writing
<code>phen</code>	a character string for a phenotype name in <code>phenfile</code>
<code>covars</code>	a character vector for covariates in <code>phenfile</code>
<code>cov.int</code>	a character string naming the covariate for interaction, the covariate has to be included in <code>covars</code>
<code>sub</code>	"N" (default) for no stratified analysis, and "Y" for requesting stratified analyses (only when <code>cov.int</code> is dichotomous)
<code>col.names</code>	a logical value indicating whether the output file should contain column names
<code>sep.ped</code>	the field separator character for pedigree file
<code>sep.phe</code>	the field separator character for phenotype file
<code>sep.gen</code>	the field separator character for genotype file

## Details

For a continuous trait, the `geepack.quant.int.batch` function first reads in and merges phenotype-covariates, genotype and pedigree files, then tests gene-environment or gene-gene interaction and the association of phen against all genotyped SNPs in `genfile`. Only one interaction term is allowed, so is the covariate for interaction (`cov.int`). When `cov.int` is dichotomous, stratified analyses can be requested by specifying `sub="Y"`. The covariance between the main effect (SNP) and the interaction effect is provided in the output when stratified analysis is not requested. `genfile` contains unique individual id and genotype data, with the column names being "id" and SNP names. For each SNP, the genotype data should be coded as 0, 1, 2 indicating the numbers of the coded alleles. The SNP name in genotype file should not have any dash, '-' and other special characters (dots and underscores are OK). `phenfile` contains unique individual id, phenotype and covariate data, with the column names being "id" and phenotype and covariate names. `pedfile` contains pedigree information, with the column names being "famid", "id", "fa", "mo", "sex". In all files, missing value should be an empty space, except missing parental id in `pedfile`. SNPs with low genotype counts (especially minor allele homozygote) may be omitted. The `geepack.quant.int.batch` function fits GEE model using `geese` function from `geepack` package.

## Value

No value is returned. Instead, results are written to `outfile`. If stratified analyses are requested, the result file will include the following columns. Otherwise, `cov_beta_snp_beta_int` will be included instead of the results from stratified analyses, that is, `beta_snp_cov0`, `se_snp_cov0`, `pval_snp_cov0`, `beta_snp_cov1`, `se_snp_cov1`, and `pval_snp_cov1`.

<code>phen</code>	phenotype name
<code>snp</code>	SNP name
<code>covar_int</code>	the covariate for interaction
<code>n</code>	sample size used in analysis
<code>AF</code>	allele frequency of the coded allele
<code>model</code>	genetic model used in analysis, additive model only
<code>beta_snp</code>	regression coefficient of SNP covariate
<code>se_snp</code>	standard error of <code>beta_beta</code>
<code>pval_snp</code>	p-value of testing <code>beta_beta</code> not equal to zero
<code>beta_snp_cov0</code>	regression coefficient of SNP covariate in stratified analysis using the subset where <code>cov.int</code> level is 0
<code>se_snp_cov0</code>	standard error of <code>beta_snp_cov0</code>
<code>pval_snp_cov0</code>	p-value of testing <code>beta_snp_cov0</code> not equal to zero
<code>beta_snp_cov1</code>	regression coefficient of SNP covariate in stratified analysis using the subset where <code>cov.int</code> level is 1
<code>se_snp_cov1</code>	standard error of <code>beta_snp_cov1</code>
<code>pval_snp_cov1</code>	p-value of testing <code>beta_snp_cov1</code> not equal to zero
<code>beta_int</code>	regression coefficient of the interaction term
<code>se_int</code>	standard error of <code>beta_int</code>
<code>pval_int</code>	p-value of testing <code>beta_int</code> not equal to zero

**Author(s)**

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

**References**

Liang, K.Y. and Zeger, S.L. (1986) Longitudinal data analysis using generalized linear models. *Biometrika*, **73** 13–22.

Zeger, S.L. and Liang, K.Y. (1986) Longitudinal data analysis for discrete and continuous outcomes. *Biometrics*, **42** 121–130.

Yan, J and Fine, J. (2004) Estimating equations for association structures. *Stat Med*, **23** 859–874.

---

geepack.quant.int.batch.imputed

*function to test gene-environment or gene-gene interaction between a continuous trait and a batch of imputed SNPs in families using Generalized Estimation Equation model*

---

**Description**

Fit Generalized Estimation Equation (GEE) model to test gene-environment or gene-gene interaction for a continuous phenotype and all imputed SNPs in a genotype file in family data under additive genetic model. The interaction term is the product of SNP genotype (allelic dosage) and a covariate for interaction (cov.int). The covariate for interaction (cov.int) can be SNP genotype (gene-gene interaction) or an environmental factor (gene-environment interaction). Only one interaction term is allowed. When cov.int is dichotomous, stratified analyses can be requested by specifying sub="Y". The covariance between the main effect (SNP) and the interaction effect is provided in the output when stratified analysis is not requested. Each pedigree is treated as a cluster, with independence working correlation matrix used in the robust variance estimator. This function applies the same interaction test to all imputed SNPs in the genotype data. In each test for interaction, the geese function from geepack package is used.

**Usage**

```
geepack.quant.int.batch.imputed(phenfile,genfile,pedfile,phen,covars,cov.int,sub="N",outfile,
col.names=T,sep.ped="," ,sep.phe="," ,sep.gen="," )
```

**Arguments**

phenfile	a character string naming the phenotype file for reading (see format requirement in details)
genfile	a character string naming the (imputed) genotype file for reading (see format requirement in details)
pedfile	a character string naming the pedigree file for reading (see format requirement in details)
outfile	a character string naming the result file for writing

phen	a character string for a phenotype name in phenfile
covars	a character vector for covariates in phenfile
cov.int	a character string naming the covariate for interaction, the covariate has to be included in covars
sub	"N" (default) for no stratified analysis, and "Y" for requesting stratified analyses (only when cov.int is dichotomous)
col.names	a logical value indicating whether the output file should contain column names
sep.ped	the field separator character for pedigree file
sep.phe	the field separator character for phenotype file
sep.gen	the field separator character for genotype file

### Details

Similar to the details for `geepack.quant.int.batch` function but here the SNP data contains imputed genotypes (allele dosages) that are continuous and range from 0 to 2.

### Value

Please see value in `geepack.quant.int.batch` function.

### Author(s)

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

### References

Liang, K.Y. and Zeger, S.L. (1986) Longitudinal data analysis using generalized linear models. *Biometrika*, **73** 13–22.

Zeger, S.L. and Liang, K.Y. (1986) Longitudinal data analysis for discrete and continuous outcomes. *Biometrics*, **42** 121–130.

Yan, J and Fine, J. (2004) Estimating equations for association structures. *Stat Med*, **23** 859–874.

---

GWplot

*function for making genome-wide p-value plot*

---

### Description

GWplot function plots  $-\log_{10}$  p-value based on SNP's chromosomal position in bitmap format.

### Usage

`GWplot(data, pval, pos, chr, chr.plot = c(1:22, "X"), title.text = "", ylim = Inf, outfile, cutoff1 = 5e-`

**Arguments**

data	a dataframe that contains p-values, chromosome number and physical position of SNPs
pval	a character string correspond to the name of the p-value column
pos	a character string correspond to the name of column with SNP physical positions
chr	a character string correspond to the name of column with SNP chromosome number
chr.plot	the chromosomes of interest for GWplot; either 1:22 or c(1:22,"X"), default chr.plot=c(1:22,"X"), "X" for X chromosome
title.text	the title of the genome-wide p-value plot
ylim	the maximum of $-\log_{10}$ p-value to be plotted, useful when not want to plot extremely small p-values
outfile	the file name (xxxx.bmp) for output plot in bitmap format
cutoff1	genome-wide significance; default is $5E-8$ ; p-values below this threshold will be highlighted in red
cutoff2	suggestive genome-wide significance; default is $4E-7$ ; p-values below this threshold but above cutoff1 will be highlighted in blue

**Details**

When the dataset has 0 p-value, GWplot will generate pvalzero.csv that contain the results with 0 p-value and make the genome-wide p-value plot by replacing 0 p-value with  $5E-324$ . P-values that reach genome-wide significance are displayed in red color; P-values that reach suggestive genome-wide significance but not genome-wide significance are displayed in blue color.

**Author(s)**

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

---

lmekin	<i>function for linear mixed effects modeling with a kinship coefficient matrix</i>
--------	---

---

**Description**

A similar function to lmekin from kinship package, but using Wald test, instead of t test. As kinship package has been archived, GWAF package now includes functions from kinship package to support the modified lmekin function. Archived kinship package is available at <http://cran.r-project.org/web/packages/kinship/index.html>.

---

lmePack.batch	<i>function to test genetic associations between a continuous trait and a batch of genotyped SNPs in families using Linear Mixed Effects model</i>
---------------	--

---

### Description

Fit linear mixed effects (LME) model to test association between a continuous phenotype and all SNPs in a genotype file in family data under user specified genetic model. The SNP genotype is treated as a fixed effect, and a random effect correlated according to degree of relatedness within a family is also fitted. In each trait-SNP association test, the `lmeKin` function which is modified from the same named function in package `kinship` is used. As `kinship` package has been archived, `GWAF` package now includes functions from `kinship` package to support the modified `lmeKin` function. Archived `kinship` package can be found at <http://cran.r-project.org/web/packages/kinship/index.html>.

### Usage

```
lmePack.batch(phenfile, genfile, pedfile, phen, kinmat, model = "a", covars = NULL,
  outfile, col.names = T, sep.ped = ",", sep.phe = ",", sep.gen = ",")
```

### Arguments

<code>genfile</code>	a character string naming the genotype file for reading (see format requirement in details)
<code>phenfile</code>	a character string naming the phenotype file for reading (see format requirement in details)
<code>pedfile</code>	a character string naming the pedigree file for reading (see format requirement in details)
<code>outfile</code>	a character string naming the result file for writing
<code>phen</code>	a character string for a phenotype name in <code>phenfile</code>
<code>covars</code>	a character vector for covariates in <code>phenfile</code>
<code>model</code>	a single character of 'a', 'd', 'g', or 'r', with 'a'=additive, 'd'=dominant, 'g'=general and 'r'=recessive models
<code>kinmat</code>	a character string naming the file where kinship coefficient matrix is kept
<code>col.names</code>	a logical value indicating whether the output file should contain column names
<code>sep.ped</code>	the field separator character for pedigree file
<code>sep.phe</code>	the field separator character for phenotype file
<code>sep.gen</code>	the field separator character for genotype file

### Details

The `lmePack.batch` function first reads in and merges phenotype-covariates, genotype and pedigree files, then tests the association of `phen` against all SNPs in `genfile`. `genfile` contains unique individual id and genotype data, with the column names being "id" and SNP names. For each SNP, the genotype data should be coded as 0, 1, 2 indicating the numbers of the coded alleles.

The SNP name in genotype file should not have any dash, '-' and other special characters (dots and underscores are OK). phenfile contains unique individual id, phenotype and covariates data, with the column names being "id" and phenotype and covariate names. pedfile contains pedigree information, with the column names being "famid", "id", "fa", "mo", "sex". In all files, missing value should be an empty space, except missing parental id in pedfile. SNPs with low genotype counts (especially minor allele homozygote) may be omitted or analyzed with dominant model. The `lmePack.batch` function fits LME model using a modified `lmeKin` function from kinship package.

### Value

No value is returned. Instead, results are written to outfile. When the genetic model is 'a', 'd' or 'r', the result includes the following columns. When the genetic model is 'g', beta and se are replaced with beta10, beta20, beta21, se10, se20, se21.

phen	phenotype name
snp	SNP name
n0	the number of individuals with 0 copy of coded alleles
n1	the number of individuals with 1 copy of coded alleles
n2	the number of individuals with 2 copies of coded alleles
h2q	the portion of phenotypic variation explained by the SNP
beta	regression coefficient of SNP covariate
se	standard error of beta
chisq	Chi-square statistic for testing beta not equal to zero
df	degree of freedom of the Chi-square statistic
model	model actually used in the analysis
pval	p-value of the chi-square statistic
beta10	regression coefficient of genotype with 1 copy of coded allele vs. that with 0 copy
beta20	regression coefficient of genotype with 2 copy of coded allele vs. that with 0 copy
beta21	regression coefficient of genotype with 2 copy of coded allele vs. that with 1 copy
se10	standard error of beta10
se20	standard error of beta20
se21	standard error of beta21

### Author(s)

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

## References

kinship package: mixed-effects Cox models, sparse matrices, and modeling data from large pedigrees. Beth Atkinson (atkinson@mayo.edu) for pedigree functions. Terry Therneau (therneau@mayo.edu) for all other functions. 2007. Ref Type: Computer Program <http://cran.r-project.org/>.

Abecasis, G. R., Cardon, L. R., Cookson, W. O., Sham, P. C., & Cherny, S. S. Association analysis in a variance components framework. *Genet Epidemiol*, **21** Suppl 1, S341-S346 (2001).

---

lmePack.batch.imputed *function to test associations between a continuous trait and a batch of imputed SNPs in families using Linear Mixed Effects model*

---

## Description

Fit linear mixed effects (LME) model to test associations between a continuous phenotype and all imputed SNPs in a genotype file in family data under additive genetic model. The SNP genotype is treated as a fixed effect, and a random effect correlated according to degree of relatedness within a family is also fitted. In each trait-SNP association test, the `lmekin` function which is modified from the same named function in package `kinship` is used. As `kinship` package has been archived, `GWAF` package now includes functions from `kinship` package to support the modified `lmekin` function. Archived `kinship` package can be found at <http://cran.r-project.org/web/packages/kinship/index.html>.

## Usage

```
lmePack.batch.imputed(phenfile, genfile, pedfile, phen, kinmat, covars = NULL,
  outfile, col.names = T, sep.ped = ",", sep.phe = ",", sep.gen = ",")
```

## Arguments

phenfile	a character string naming the phenotype file for reading (see format requirement in details)
genfile	a character string naming the genotype file for reading (see format requirement in details)
pedfile	a character string naming the pedigree file for reading (see format requirement in details)
phen	a character string for a phenotype name in phenfile
kinmat	a character string naming the file where kinship coefficient matrix is kept
covars	a character vector for covariates in phenfile
outfile	a character string naming the result file for writing
col.names	a logical value indicating whether the output file should contain column names
sep.ped	the field separator character for pedigree file
sep.phe	the field separator character for phenotype file
sep.gen	the field separator character for genotype file

**Details**

Similar to the details for `lmepack.batch` function but here the SNP data contains imputed genotypes (allele dosages) that are continuous and range from 0 to 2. In addition, the user specified genetic model argument is not available.

**Value**

No value is returned. Instead, results are written to `outfile`.

<code>phen</code>	phenotype name
<code>snp</code>	SNP name
<code>N</code>	the number of individuals in analysis
<code>AF</code>	imputed allele frequency of coded allele
<code>h2q</code>	the portion of phenotypic variation explained by the SNP
<code>beta</code>	regression coefficient of SNP covariate
<code>se</code>	standard error of beta
<code>pval</code>	p-value of testing beta not equal to zero

**Author(s)**

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

**References**

kinship package: mixed-effects Cox models, sparse matrices, and modeling data from large pedigrees. Beth Atkinson (atkinson@mayo.edu) for pedigree functions. Terry Therneau (therneau@mayo.edu) for all other functions. 2007. Ref Type: Computer Program <http://cran.r-project.org/>.

Abecasis, G. R., Cardon, L. R., Cookson, W. O., Sham, P. C., & Cherny, S. S. Association analysis in a variance components framework. *Genet Epidemiol*, **21** Suppl 1, S341-S346 (2001).

---

<code>lmepack.int.batch</code>	<i>function to test gene-environment or gene-gene interaction for a continuous trait and a batch of genotyped SNPs in families using Linear Mixed Effects model</i>
--------------------------------	---

---

**Description**

Fit linear mixed effects model (LME) to test gene-environment or gene-gene interaction for a continuous phenotype and all SNPs in a genotype file in family data under additive genetic model. The interaction term is the product of SNP genotype and a covariate for interaction (`cov.int`). The covariate for interaction (`cov.int`) can be SNP genotype (gene-gene interaction) or an environmental factor (gene-environment interaction). Only one interaction term is allowed. When `cov.int` is dichotomous, stratified analyses can be requested by specifying `sub="Y"`. The covariance between the main effect (SNP) and the interaction effect is provided in the output when stratified analysis

is not requested. The SNP genotype and the interaction are treated as fixed effects, and a random effect correlated according to degree of relatedness within a family is also fitted. In each test for interaction, the `lmekin` function which is modified from the same named function in package `kinship` is used. As `kinship` package has been archived, `GWAF` package now includes functions from `kinship` package to support the modified `lmekin` function. Archived `kinship` package can be found at <http://cran.r-project.org/web/packages/kinship/index.html>.

### Usage

```
lmepack.int.batch(phenfile, genfile, pedfile, phen, kinmat, covars, cov.int, sub="N",
  outfile, col.names=T, sep.ped=" ", sep.phe=" ", sep.gen=" ")
```

### Arguments

<code>genfile</code>	a character string naming the genotype file for reading (see format requirement in details)
<code>phenfile</code>	a character string naming the phenotype file for reading (see format requirement in details)
<code>pedfile</code>	a character string naming the pedigree file for reading (see format requirement in details)
<code>outfile</code>	a character string naming the result file for writing
<code>phen</code>	a character string for a phenotype name in <code>phenfile</code>
<code>covars</code>	a character vector for covariates in <code>phenfile</code>
<code>cov.int</code>	a character string naming the covariate for interaction, the covariate has to be included in <code>covars</code>
<code>sub</code>	"N" (default) for no stratified analysis, and "Y" for requesting stratified analyses (only when <code>cov.int</code> is dichotomous)
<code>kinmat</code>	a character string naming the file where kinship coefficient matrix is kept
<code>col.names</code>	a logical value indicating whether the output file should contain column names
<code>sep.ped</code>	the field separator character for pedigree file
<code>sep.phe</code>	the field separator character for phenotype file
<code>sep.gen</code>	the field separator character for genotype file

### Details

The `lmepack.int.batch` function first reads in and merges phenotype-covariates, genotype and pedigree files, then tests gene-environment or gene-gene interaction for `phen` against all SNPs in `genfile`. Only one interaction term is allowed, so is the covariate for interaction (`cov.int`). When `cov.int` is dichotomous, stratified analyses can be requested by specifying `sub="Y"`. The covariance between the main effect (SNP) and the interaction effect is provided in the output when stratified analysis is not requested. `genfile` contains unique individual id and genotype data, with the column names being "id" and SNP names. For each SNP, the genotype data should be coded as 0, 1, 2 indicating the numbers of the coded alleles. The SNP name in genotype file should not have any dash, '-' and other special characters (dots and underscores are OK). `phenfile` contains unique individual id, phenotype and covariate data, with the column names being "id" and phenotype and covariate names. `pedfile` contains pedigree information, with the column names being

"famid","id","fa","mo","sex". In all files, missing value should be an empty space, except missing parental id in pedfile. SNPs with low genotype counts (especially minor allele homozygote) may be omitted. The `lmepack.int.batch` function fits LME model using a modified `lmekin` function from kinship package.

### Value

No value is returned. Instead, results are written to outfile. If stratified analyses are requested, the result file will include the following columns. Otherwise, `cov_beta_snp_beta_int` will be included instead of the results from stratified analyses, that is, (`beta_snp_cov0`, `se_snp_cov0`, `pval_snp_cov0`, `beta_snp_cov1`, `se_snp_cov1`, and `pval_snp_cov1`).

<code>phen</code>	phenotype name
<code>snp</code>	SNP name
<code>covar_int</code>	the covariate for interaction
<code>n</code>	sample size used in analysis
<code>AF</code>	allele frequency of the coded allele
<code>model</code>	genetic model used in analysis, additive model only
<code>beta_snp</code>	regression coefficient of SNP covariate
<code>se_snp</code>	standard error of <code>beta_snp</code>
<code>pval_snp</code>	p-value of testing <code>beta_snp</code> not equal to zero
<code>beta_snp_cov0</code>	regression coefficient of SNP covariate in stratified analysis using the subset where <code>cov.int</code> level is 0
<code>se_snp_cov0</code>	standard error of <code>beta_snp_cov0</code>
<code>pval_snp_cov0</code>	p-value of testing <code>beta_snp_cov0</code> not equal to zero
<code>beta_snp_cov1</code>	regression coefficient of SNP covariate in stratified analysis using the subset where <code>cov.int</code> level is 1
<code>se_snp_cov1</code>	standard error of <code>beta_snp_cov1</code>
<code>pval_snp_cov1</code>	p-value of testing <code>beta_snp_cov1</code> not equal to zero
<code>beta_int</code>	regression coefficient of the interaction term
<code>se_int</code>	standard error of <code>beta_int</code>
<code>pval_int</code>	p-value of testing <code>beta_int</code> not equal to zero

### Author(s)

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

### References

- kinship package: mixed-effects Cox models, sparse matrices, and modeling data from large pedigrees. Beth Atkinson (atkinson@mayo.edu) for pedigree functions. Terry Therneau (therneau@mayo.edu) for all other functions. 2007. Ref Type: Computer Program <http://cran.r-project.org/>.
- Abecasis, G. R., Cardon, L. R., Cookson, W. O., Sham, P. C., & Cherny, S. S. Association analysis in a variance components framework. *Genet Epidemiol*, **21** Suppl 1, S341-S346 (2001).

lmepack.int.batch.imputed

*function to test gene-environment or gene-gene interaction and associations between a continuous trait and a batch of imputed SNPs in families using Linear Mixed Effects model*

## Description

Fit linear mixed effect model to test gene-environment or gene-gene interaction and genetic association for a continuous phenotype and all imputed SNPs in a genotype file under additive genetic model. The interaction term is the product of SNP genotype (allelic dosage) and a covariate for interaction (cov.int). The covariate for interaction (cov.int) can be SNP genotype (gene-gene interaction) or an environmental factor (gene-environment interaction). Only one interaction term is allowed. When (cov.int) is dichotomous, stratified analyses can be requested by specifying sub="Y". The covariance between the main effect (SNP) and the interaction effect is provided in the output when stratified analysis is not requested. The SNP genotype and the interaction are treated as fixed effect, and a random effect correlated according to degree of relatedness within a family is also fitted. In each test for trait-SNP association or interaction, the `lmekin()` function which is modified from the same named function in package `kinship` is used. As `kinship` package has been archived, `GWAF` package now includes functions from `kinship` package to support the modified `lmekin` function. Archived `kinship` package can be found at <http://cran.r-project.org/web/packages/kinship/index.html>.

## Usage

```
lmepack.int.batch.imputed(phenfile,genfile,pedfile,phen,kinmat,covars,cov.int,sub="N",
  outfile,col.names=T,sep.ped=" ",sep.phe=" ",sep.gen=" ")
```

## Arguments

phenfile	a character string naming the phenotype file for reading (see format requirement in details)
genfile	a character string naming the (imputed) genotype file for reading (see format requirement in details)
pedfile	a character string naming the pedigree file for reading (see format requirement in details)
outfile	a character string naming the result file for writing
phen	a character string for a phenotype name in phenfile
covars	a character vector for covariates in phenfile
cov.int	a character string naming the covariate for interaction, the covariate has to be included in covars
sub	"N" (default) for no stratified analysis, and "Y" for requesting stratified analyses (only when cov.int is dichotomous)
kinmat	a character string naming the file where kinship coefficient matrix is kept

col.names	a logical value indicating whether the output file should contain column names
sep.ped	the field separator character for pedigree file
sep.phe	the field separator character for phenotype file
sep.gen	the field separator character for genotype file

### Details

Similar to the details for 'lmepack.int.batch' function but here the SNP data contains imputed genotypes (allele dosages) that are continuous and range from 0 to 2.

### Value

Please see value in 'lmepack.int.batch' function.

### Author(s)

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

### References

kinship package: mixed-effects Cox models, sparse matrices, and modeling data from large pedigrees. Beth Atkinson (atkinson@mayo.edu) for pedigree functions. Terry Therneau (therneau@mayo.edu) for all other functions. 2007. Ref Type: Computer Program <http://cran.r-project.org/>.

Abecasis, G. R., Cardon, L. R., Cookson, W. O., Sham, P. C., & Cherny, S. S. Association analysis in a variance components framework. *Genet Epidemiol*, **21** Suppl 1, S341-S346 (2001).

---

makekinship

*Create a kinship matrix*

---

### Description

Compute the overall kinship matrix for a collection of families. This function and R documentation are included from the archived kinship package, which is available at <http://cran.r-project.org/web/packages/kinship/index.html>.

### Usage

```
makekinship(famid, id, father.id, mother.id, unrelated=0)
```

### Arguments

famid	a vector of family identifiers
id	a vector of unique subject identifiers
father.id	for each subject, the identifier of their biological father
mother.id	for each subject, the identifier of their biological mother
unrelated	subjects with this family id are considered to be unrelated singletons, i.e., not related to each other or to anyone else.

**Value**

a kinship matrix

**Examples**

```
## Not run:  
> ped <- read.csv("ped.csv")  
> names(ped)  
[1] "id"      "famid"   "fa"      "mo"      "sex"  
> kmat<-makekinship(ped$famid,ped$id,ped$fa,ped$mo)  
  
## End(Not run)
```

---

qq

*function to make Qantile-Qantile (QQ) plot for p-values*

---

**Description**

qq function makes the QQ plot of p-values against a uniform (0,1) distribution. The genomic control parameter for one degree freedom chi-square statistics corresponding to the p-values is also plotted.

**Usage**

```
qq(pvalue, outfile)
```

**Arguments**

pvalue	P-values of interest.
outfile	the file name (xxxx.bmp) for output QQ plot in bitmap format

**Author(s)**

Qiong Yang <qyang@bu.edu> and Ming-Huei Chen <mhchen@bu.edu>

# Index

- \*Topic **array**
  - gchol, 6
- \*Topic **classes**
  - bdsmatrix-class, 4
  - gchol-class, 7
  - gchol.bdsmatrix-class, 8
- \*Topic **methods**
  - gchol-methods, 7
- [,bdsmatrix-method (bdsmatrix-class), 4
- [,gchol.bdsmatrix-method (gchol.bdsmatrix-class), 8
- %%,matrix,bdsmatrix-method (bdsmatrix-class), 4
- %%,matrix,gchol.bdsmatrix-method (gchol.bdsmatrix-class), 8
- %%,numeric,bdsmatrix-method (bdsmatrix-class), 4
- %%,numeric,gchol.bdsmatrix-method (gchol.bdsmatrix-class), 8
  
- all,bdsmatrix-method (bdsmatrix-class), 4
- any,bdsmatrix-method (bdsmatrix-class), 4
- auto, 3
  
- bdsmatrix-class, 4
  
- coerce,bdsmatrix,matrix-method (bdsmatrix-class), 4
- coerce,bdsmatrix,vector-method (bdsmatrix-class), 4
- coerce,gchol,matrix-method (gchol-class), 7
- coerce,gchol.bdsmatrix,matrix-method (gchol.bdsmatrix-class), 8
  
- diag,bdsmatrix-method (bdsmatrix-class), 4
- diag,gchol-method (gchol-class), 7
  
- diag,gchol.bdsmatrix-method (gchol.bdsmatrix-class), 8
- diag<-,bdsmatrix-method (bdsmatrix-class), 4
- dim,bdsmatrix-method (bdsmatrix-class), 4
- dim,gchol.bdsmatrix-method (gchol.bdsmatrix-class), 8
- dimnames,bdsmatrix-method (bdsmatrix-class), 4
- dimnames<-,bdsmatrix-method (bdsmatrix-class), 4
  
- gchol, 6
- gchol,matrix-method (gchol-methods), 7
- gchol-class, 7
- gchol-methods, 7
- gchol.bdsmatrix-class, 8
- geepack.lgst, 9
- geepack.lgst.batch, 10
- geepack.lgst.batch.imputed, 12
- geepack.lgst.imputed, 14
- geepack.lgst.int, 15
- geepack.lgst.int.batch, 16
- geepack.lgst.int.batch.imputed, 19
- geepack.lgst.int.imputed, 21
- geepack.quant.batch, 22
- geepack.quant.batch.imputed, 24
- geepack.quant.int.batch, 26
- geepack.quant.int.batch.imputed, 28
- GWAF (GWAF-package), 2
- GWAF-package, 2
- GWplot, 29
  
- lmeKin, 30
- lmePack.batch, 31
- lmePack.batch.imputed, 33
- lmePack.int.batch, 34
- lmePack.int.batch.imputed, 37

makekinship, [38](#)  
Math, bdsmatrix-method  
    (bdsmatrix-class), [4](#)  
Math2, bdsmatrix-method  
    (bdsmatrix-class), [4](#)  
max, bdsmatrix-method (bdsmatrix-class),  
    [4](#)  
min, bdsmatrix-method (bdsmatrix-class),  
    [4](#)  
  
Ops, bdsmatrix, bdsmatrix-method  
    (bdsmatrix-class), [4](#)  
Ops, bdsmatrix, numeric-method  
    (bdsmatrix-class), [4](#)  
Ops, numeric, bdsmatrix-method  
    (bdsmatrix-class), [4](#)  
  
prod, bdsmatrix-method  
    (bdsmatrix-class), [4](#)  
  
qq, [39](#)  
  
range, bdsmatrix-method  
    (bdsmatrix-class), [4](#)  
  
show, bdsmatrix-method  
    (bdsmatrix-class), [4](#)  
show, gchol-method (gchol-class), [7](#)  
show, gchol.bdsmatrix-method  
    (gchol.bdsmatrix-class), [8](#)  
sum, bdsmatrix-method (bdsmatrix-class),  
    [4](#)  
  
unique, bdsmatrix, missing-method  
    (bdsmatrix-class), [4](#)