**StocSum: Stochastic Summary Statistics Version 0.1.1**

**Contents**

1 Introduction 2

2 The model 2

3 Getting started 2

3.1 Downloading StocSum 2

3.2 Installing StocSum 2

4 Input 3

4.1 Object 3

4.2 Genotypes 3

4.3 Group definition file 3

5 Running StocSum 4

5.1 Fitting GLMM 4

5.2 Generate random vectors 4

5.3 Calculate summary statistics 5

5.4 Single-variant tests 5

5.5 Variant set tests 5

5.6 Meta-analysis 6

5.7 Gene-environment tests 6

5.8 LD score regression 6

6 Output 6

7 Advanced options 6

8 Version 6

9 Contact 6

10 Acknowledgments 6

# 1 Introduction

StocSum is an R package for performing association tests using generalized linear mixed models (GLMMs) (Breslow and Clayton, 1993) in genome-wide association studies (GWAS) and sequencing association studies. Compared with existing software programs, StocSum uses informative statistics, which greatly reduces the computational complexity and disk space for storage, to replace traditional pairwise LD matrix.

StocSum first fits a GLMM with covariate adjustment and random effects to account for population structure and family or cryptic relatedness. Then using a resampling method to generate a matrix to replace the traditional pairwise LD matrix ***V***. StocSum performs single variant tests, variant set tests based on user-defined variant sets, and meta-analysis. It performs gene-environment interaction (GEI) tests and joint effects tests based on user-defined variant sets. It performs LD score calculation to estimate heritability.

# 2 The model

For both single-variant tests and variant-set tests, we consider the generalized linear mixed model (GLMM):

|  |  |  |
| --- | --- | --- |
|  |  | (1) |

where is a monotonic link function of , and is the conditional mean of the phenotype given covariates , genotypes and random effects , for subject *i* in the study sample. The phenotype follows a distribution in the exponential family. is a *N* x *p* covariates matrix, is a *p* x 1 column vector of fixed covariate effects including the intercept. The genotype matrix is an *N* x *q* matrix for *q* () genetic variants and is the genotype effect. We assume that is an *N* x1 column vector of random effects and , where are the variance components parameters and are known *N* x *N* relatedness matrices which account for multiple layers of correlation structures, such as genetic relatedness, hierarchical designs, shared environmental effects and repeated measures from longitudinal studies.

# 3 Getting started

## 3.1 Downloading StocSum

StocSum is an open source project and is freely available for download at <https://github.com/NWang-hub/StocSum>. It can also be found as a regular R package and downloaded from CRAN (\*\*\*\*\*\*).

## 3.2 Installing StocSum

The following R packages are required before installing StocSum: Rcpp and RcppArmadillo for R and C++ integration and testthat to run code checks during development. Additionally, StocSum imports from Rcpp, CompQuadForm, Foreach, parallel, Matrix, methods, GMMAT, and Bioconductor packages SeqArray and SeqVarTools. The R package doMC is required to run parallel computing in StocSum.stat, StocSum\_GE.stat, StocSum\_LDSC.stat (doMC is not available on Windows and these functions will switch to a single computer thread).

For optimal computational performance, it is recommended to use an R version configured with the Intel Math Kernel Library (or other fast BLAS.LAPACK libraries). See the instructions on building R with Intel MKL (<https://software.intel.com/en-us/articles/using-intel-mkl-with-r>).

Here is an example for installing StocSum and all its dependencies in an R session (assuming none of the R packages other than the default has been installed):

> install.packages(c(“devtools”, ”RcppArmadillo”, ”CompQuadForm”, “doMC”, “foreach”, “Matrix”, “GMMAT”, “BiocManager”, “testthat”), repos=”https://cran.r-project.org/”)

> BiocManager::install(c(“SeqArray”, “SeqVarTools”))

> devtools::install\_github(“<https://github.com/NWang-hub/StocSum>”)

# 4 Input

StocSum requires an object from fitting the null model using the glmm.kin function from the GMMAT package, and a genotype file in a GDS format. For variant set test, a user-defined marker group file is required. Specified formats of these files are described as follows

## 4.1 Object

StocSum can perform various test including single variant test, variant set test, meta analysis, gene-environment test, and LD score regression. To achieve them, fitting the null model using GMMAT is necessary. To fit the null model, the phenotype and covariates (include the environmental factors of interest) should be saved in a data frame. If the samples are related, the relatedness should be known positive semidefinite matrices Vk as an R matrix (in the case of a single matrix) or an R list (in the case of multiple matrices). Refer to the GMMAT user manual (<https://cran.r-project.org/web/packages/GMMAT/vignettes/GMMAT.pdf>) to learn the method of fitting the null model. The class of the object should be “glmmkin”.

## 4.2 Genotypes

StocSum can take genotype files in the GDS format. Genotypes in Variant Call Format (VCF) and PLINK binary PED format can be converted to the GDS format using seqVCF2GDS and seqBED2GDS functions from the SeqArray package:

> SeqArray::seqVCF2GDS(“VCF\_file\_name”,”GDS\_file\_name”)

> SeqArray::seqBED2GDS(“BED\_file\_name”,”FAM\_file\_name”,”BIM\_file\_name”,”GDS\_file\_name”)

## 4.3 Group definition file

For variant set test, a group definition file with no header and 6 columns (variant set id, variant chromosome, variant position, variant reference allele, variant alternate allele, weight) is required. For example, here we show the first 6 rows of the example group definition file “SetID.withweights.txt”:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Set1 | 1 | 1 | T | A | 1 |
| Set1 | 1 | 2 | A | C | 4 |
| Set1 | 1 | 3 | C | A | 3 |
| Set1 | 1 | 4 | G | A | 6 |
| Set1 | 1 | 5 | A | G | 9 |
| Set1 | 1 | 6 | C | A1 | 9 |

Note that each variant in the group definition file is matched by chromosome, position, reference allele and alternate allele with variants from the GDS file. One genetic variant can be included in different groups with possibly different weights. If no external weights are needed in the analysis, simply replace the 6th column by all 1’s.

# 5 Running StocSum

If StocSum has been successfully installed, you can load it in an R session using

> library (StocSum)

We provide xxxx functions in StocSum: glmmin2randomvec for generating the random vectors, StocSum.stat for generating the compact and informatics statistics, StocSum.pval for running variant set test, StocSum.svt for running single variant test, StocSum.meta for performing variant set test meta-analysis. Details about how to use these functions, their arguments and returned values can be found in the R help document of StocSum. For example, to learn more about glmmin2randomvec in an R session you can type

> ?glmmin2randomvec

## 5.1 Fitting GLMM

StocSum requires a “glmmkin” class object that contains a fitted GLMM null model. The object can be obtained from the glmmkin function from the R package GMMAT. For more examples and details about the glmmkin function, see the GMMAT manual (<https://cran.r-project.org/web/packages/GMMAT/vignettes/GMMAT.pdf>). Below is an example of fitting a GLMM using the glmmkin function from GMMAT:

> library(GMMAT)

> data(example)

> attach(example)

> GRM.file <- system.file(“extdata”, “GRM.txt.bz2”, package = “GMMAT”)

> GRM <- as.matrix(read.table(GRM.file, check.names = FALSE))

> nullmod <- glmmkin(disease ~ age + sex, data = pheno, kins = GRM, id = “id”, family = binomial(link = “logit”))

## 5.2 Generate random vectors

To run StocSum, the user needs to generate the random vectors that having covariance approximate to the projection matrix from above fitted null model.

> obj <- glmmkin2randomvec(nullmod)

The function glmmkin2randomvec returns a list. The element *random.vectors* stores the generated random vectors. The remaining elements *theta*, *scaled.residuals*, and *id\_include* are inherited from the null model generated in 5.1.

If the returned nullmod object in 5.1 does no include the projection matrix **P**, the user needs to set the relation matrix in glmkin2randomvec function. The following is an example showing the setting when only the kinship is considered as the relation matrix.

> kinship.chol <- chol(GRM)

> obj <- glmmkin2randomvec(nullmod, Z = list(t(kinship.chol)))

## 5.3 Calculate summary statistics

To obtain the compact and informatics statics that are required for downstream tests, we need the genotype file in the GDS format as input. After running StocSum.stat, intermediates files containing single variant scores and their compact and informatics statics will be generated. An example is as following:

> out.prefix <- "test"

> gdsfile <- system.file("extdata", "geno.gds", package = "GMMAT")

> StocSum.stat(obj, geno.file = gdsfile, meta.file.prefix = out.prefix, MAF.range=c(1e-7, 0.5), miss.cutoff = 1)

The first argument in StocSum.stat is the returned class glmmkin2randomvec object from 5.2. The argument geno.file is the file of genotype file. The argument meta.file.prefix specifies the prefix of output files. In the example above, a space-delimited file “test.sample.1” will be generated to save the single variant scores, and a binary file “test.ressample.1” will be generated to save the compact and informatics statics. Note that this binary file is not human-readable, but can be loaded by downstream modules/functions.

## 5.4 Single-variant tests

When the intermediate files are generated in section 5.3, the function StocSum.pval can be used to perform single variant test. An example is as following:

> out1<-StocSum.svt(out.prefix, n.file=ncores, MAF.range=c(0,0.5), miss.cutoff = 1, auto.flip=F)

The first argument, “out.prefix”, is the prefix of output files specified in StocSum.stat function.

SNP chr pos ref alt N missrate altfreq SCORE VAR PVAL

1 SNP1 1 1 T A 393 0.0175 0.9745547 -1.9849977 4.588055 0.3540751

2 SNP2 1 2 A C 400 0.0000 0.5000000 3.5103164 49.685470 0.6184822

3 SNP3 1 3 C A 400 0.0000 0.7925000 0.5334004 33.398425 0.9264616

4 SNP4 1 4 G A 400 0.0000 0.7012500 3.1149410 41.128852 0.6271732

5 SNP5 1 5 A G 400 0.0000 0.5937500 -4.0013505 43.791006 0.5454023

6 SNP6 1 6 C A 400 0.0000 0.8887500 -1.6920412 17.296781 0.6841223

It returns a data frame with the first 5 columns information extracted from the GDS file, followed by the sample size N, the allele frequency (AF) of ALT allele, effect size estimate BETA of ALT allele, standard error SE, Wald test P value PVAL.

## 5.5 Variant set tests

The function StocSum.pval can be used to perform variant set test. Different from single-variant tests, a group definition file with no header and 6 columns (variant set id, variant chromosome, variant position, variant reference allele, variant alternative allele, weight) is also required, as described in section 4.3. Here we perform variant set test in single study with an example shown as following:

> group.file <- system.file("extdata", "SetID.withweights.txt", package = "GMMAT")

> out2<-StocSum.pval(out.prefix, group.file = group.file, MAF.range=c(0,0.5), miss.cutoff = 1)

group n.variants B.score B.var B.pval E.pval

1 Set1 20 194.05011 84243.93 0.50377208 0.2001607

2 Set2 20 -82.55532 255018.97 0.87014224 0.9580280

3 Set3 20 184.18465 229741.44 0.70078013 0.4804642

4 Set4 20 296.38607 25970.19 0.06589123 0.1020998

5 Set5 20 446.62340 77028.37 0.10756768 0.3252381

6 Set6 20 260.94738 127859.95 0.46553112 0.5403710

7 Set7 20 186.76450 142536.96 0.62082101 0.5535321

8 Set8 20 -217.12052 119423.47 0.52981787 0.0473496

9 Set9 20 32.51345 185369.47 0.93980348 0.5839853

The first argument, “out.prefix”, is the prefix of output files specified in StocSum.stat function. It returns a data frame with the first 2 columns showing the group (variant set) name, number of variants in each group. For Burden, 3 columns will be included to show the burden test score, variance of the score, and its P value. For the efficient hybrid test, the P value column will be included in the last column.

## 5.6 Meta-analysis

The function StocSum.pval enables us to preform meta-analysis to combine multiple studies. In this case, the out.prefix is a vector of intermediate files’ prefix with length equal to the number of studies.

## 5.7 Gene-environment tests

Variant set tests in a single study can be performed using the function StocSum.GE.

library(GMMAT)

data(example)

attach(example)

GRM.file <- system.file("extdata", "GRM.txt.bz2", package = "GMMAT")

GRM <- as.matrix(read.table(GRM.file, check.names = FALSE))

nullmod <- glmmkin(disease ~ age + sex, data = pheno, kins = GRM, id = "id", family = binomial(link = "logit"))

obj <- StocSum.GE.glmmkin2randomvec(nullmod)

out.prefix <- "test.GE"

gdsfile <- system.file("extdata", "geno.gds", package = "GMMAT")

group.file <- system.file("extdata", "SetID.withweights.txt", package = "GMMAT")

interaction <- c("sex")

StocSum.GE.stat(obj, interaction = interaction, geno.file = gdsfile, meta.file.prefix = out.prefix)

out <- StocSum.GE.pval(out.prefix, group.file = group.file, tests=c("JV", "JF", "JD"))

group n.variants MV.pval MF.pval IV.pval IF.pval JV.pval

1 Set1 20 0.66866731 0.6918987 0.23213681 0.4456700 0.44438545

2 Set2 20 0.88965718 0.8768436 0.28418972 0.4653413 0.60048281

3 Set3 20 0.82124069 0.7831147 0.28631122 0.3026438 0.57550874

4 Set4 20 0.83441926 0.6657576 0.15100509 0.2111586 0.38700886

5 Set5 20 0.59029839 0.6419541 0.69015471 0.8673537 0.77322631

6 Set6 20 0.85633552 0.7649978 0.57634584 0.1649543 0.84205777

7 Set7 20 0.74446151 0.9001541 0.07827834 0.1127818 0.22392707

8 Set8 20 0.06381021 0.1271488 0.27068264 0.4379494 0.08737461

9 Set9 20 0.09841867 0.1887735 0.55285375 0.6714410 0.21281207

JF.pval JD.pval

1 0.6521715 0.6711399

2 0.7796828 0.7737954

3 0.5808404 0.5782147

4 0.4117099 0.4163959

5 0.8764428 0.8828345

6 0.4011440 0.3873984

7 0.3827154 0.3337492

8 0.2050636 0.2165051

9 0.3867977 0.3885576

## 5.8 LD score regression

# 6 Output

# 7 Advanced options

# 8 Version

# 9 Contact

# 10 Acknowledgments

**References**

Breslow, Norman E., and David G. Clayton. "Approximate inference in generalized linear mixed models." Journal of the American statistical Association 88.421 (1993): 9-25.