TriadSim

Min Shi

2017-05-23

TriadSim is a package that can simulate genotypes for case-parent triads, case-control, and quantitative trait samples with realistic linkage diequilibrium structure and allele frequency distribution. For studies of epistasis one can simulate models that involve specific SNPs at specific sets of loci, which we will refer to as "pathways". TriadSim generates genotype data by resampling triad genotypes from existing data. It takes genotypes in PLINK format as the input files. The genotypes for the mothers, fathers, and children are in separate files. The mothers, fathers, and children must be from the same set of triad families although the ordering of the families can be different for the three sets of data. After reading in the genotypes, a sorting step will reorder the families so that the three individuals in each family can realign.

Main function `TriadSim`

`TriadSim` is the main function to perform the simulations. The example function call below simulates genotype data for 1000 case-parent triads for 4 chromosomes (chromsomes 1, 8 17, 20) under a genetic main effect scenario with a baseline disease prevalence of P0=0.001 and genetic relative risks of 1.5 and 2 for carrying the first and the second pathway respectively. This function call will write output files in PLINK. The output file names and path to the directory are given by the parameter "out.put.file" and the chromosome number. Each set (.bim, .bed and .fam files) of PLINK files contain genotype data for one chromosome for all simulated samples. The name of the file is the concatenation of the value of the input parameter "out.put.file" and chromosome number. For example, if "out.put.file" is set to be "triad", the names of the output files will be triad1, triad8, triad17 and triad20 for our example. See R package documentation for more details.

```
library(TriadSim)
 m.file <- file.path(system.file(package = "TriadSim"),'extdata/pop1_4chr_mom')
 f.file <- file.path(system.file(package = "TriadSim"),'extdata/pop1_4chr_dad')
 k.file <- file.path(system.file(package = "TriadSim"),'extdata/pop1_4chr_kid')
 input.plink.file <- c(m.file, f.file, k.file)

 TriadSim(input.plink.file, out.put.file=file.path(tempdir(),'triad'), fr.desire=0.05,
          pathways=list(1:4,5:8),n.ped=1000, N.brk=3, target.snp=NA,P0=0.001,is.OR=FALSE,
          risk.exposure= 1,risk.pathway.unexposed=c(1.5, 2),risk.pathway.exposed=c(1.5,2),
          is.case=TRUE, e.fr=NA, pop1.frac=NA, P0.ratio=1,rcmb.rate, no_cores=1)
```

```
## [1]  21 118 121 140 155 168 218 383
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad1.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad1.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad1.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad8.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad8.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad8.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad17.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad17.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad17.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
```

```
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad20.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad20.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad20.bed (SNP-major mode)
```

The following call simulates a quantitative trait (by setting "qtl=T"). The function will create 4 sets of plink files, one for each chromosome.

```
TriadSim(input.plink.file, file.path(tempdir(),'qtl'), fr.desire=0.3,
         pathways=list(1:4,5:8),n.ped=1000, N.brk=3, target.snp=NA,P0=0.001,
         is.OR=FALSE,risk.exposure= 1,risk.pathway.unexposed=c(0.5, 1),
         risk.pathway.exposed=c(0.5, 1), is.case=TRUE, e.fr=NA, pop1.frac=NA,
         P0.ratio=1,rcmb.rate, no_cores=1, qtl=T)
```

```
## [1]  72  95 139 145 247 276 279 339
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/qtl1.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/qtl1.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/qtl1.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/qtl8.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/qtl8.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/qtl8.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/qtl17.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/qtl17.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/qtl17.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/qtl20.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/qtl20.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/qtl20.bed (SNP-major mode)
```

The following call simulates a scenario that involves gene-environment interaction. The relative risk for the exposure main effect is 1.2. The relative risks for carrying the first and second pathway SNPs are 1.5 and 2 respectively for the exposed individuals and are 1 and 1 for the unexposed individuals.

```
TriadSim(input.plink.file, file.path(tempdir(),'gxe'), fr.desire=0.3,
         pathways=list(1:4,5:8),n.ped=1000, N.brk=3, target.snp=NA,P0=0.001,
         is.OR=FALSE,risk.exposure= 1.2,risk.pathway.unexposed=c(1,1),
         risk.pathway.exposed=c(1.5, 2), is.case=TRUE, e.fr=0.3, pop1.frac=NA,
         P0.ratio=1,rcmb.rate, no_cores=1, qtl=FALSE)
```

```
## [1]  72  95 139 145 247 276 279 339
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/gxe1.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/gxe1.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/gxe1.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/gxe8.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/gxe8.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/gxe8.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/gxe17.fam
```

```
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/gxe17.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/gxe17.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/gxe20.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/gxe20.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/gxe20.bed (SNP-major mode)
```

The following call simulates a stratified scenario that involves gene-environment interaction. The risk parameters are the same as the scenario above. The "input.plink.file"" is a list of two character vectors. Each vector contains three character strings giving the directory and basename of the PLINK files in one subpopulation.The subpopulations are equally sized (pop1.frac=0.5). The desire allele frequency in the first subpopulation is 0.3 and the desired difference in allele frequencies of the two subpopulations is 0.15 (as set by the parameter fr.desire=c(0.3,0.15)). The baseline disease prevalence (disease prevalence in the unexposed who carries 0 copy of the risk pathway) is 0.001 in the first subpopulation while that in the second subpopulation is 0.003 (0.001*3). The exposure prevalence in the two subpopulations are 0.1 and 0.3 respectively.

```
library(TriadSim)
 m.file <- file.path(system.file(package = "TriadSim"),'extdata/pop1_4chr_mom')
 f.file <- file.path(system.file(package = "TriadSim"),'extdata/pop1_4chr_dad')
 k.file <- file.path(system.file(package = "TriadSim"),'extdata/pop1_4chr_kid')
 m.file2 <- file.path(system.file(package = "TriadSim"),'extdata/pop2_4chr_mom')
 f.file2 <- file.path(system.file(package = "TriadSim"),'extdata/pop2_4chr_dad')
 k.file2 <- file.path(system.file(package = "TriadSim"),'extdata/pop2_4chr_kid')
 input.plink.file2 <- list(c(m.file, f.file, k.file),c(m.file2, f.file2, k.file2))

 TriadSim(input.plink.file2, out.put.file=file.path(tempdir(),'stratified') ,
          fr.desire=c(0.3,0.15),pathways=list(1:4,5:8),n.ped=1000, N.brk=3,
          target.snp=NA,P0=0.001,is.OR=FALSE,risk.exposure= 1.2,
          risk.pathway.unexposed=c(1,1), risk.pathway.exposed=c(1.5, 2),
          is.case=TRUE, e.fr=c(0.1, 0.3), pop1.frac=0.5, P0.ratio=3,
          rcmb.rate,no_cores=1)
```

```
## [1]  17  26  92 147 212 215 273 316
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/stratified1.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/stratified1.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/stratified1.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/stratified8.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/stratified8.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/stratified8.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/stratified17.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/stratified17.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/stratified17.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/stratified20.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/stratified20.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/stratified20.bed (SNP-major mode)
```

To simulate case-control data the function needs to be called twice, calls to simulate cases (is.case=TRUE) and controls (is.case=FALSE) respectively. The script below calls the function to simulate 1000 cases and 1000 controls and writes genotypes of the cases and controls into seperate sets of PLINK files.

```
## cases
 TriadSim(input.plink.file,file.path(tempdir(),'case') , fr.desire=0.05,
          pathways=list(1:4,5:8),n.ped=1000, N.brk=3, target.snp=NA,P0=0.001,
          is.OR=TRUE,risk.exposure= 1,risk.pathway.unexposed=c(1.5, 2),
          risk.pathway.exposed=c(1.5, 2), is.case=TRUE, e.fr=NA, pop1.frac=NA,
          P0.ratio=1,rcmb.rate, no_cores=1)
```

```
## [1]  21 118 121 140 155 168 218 383
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/case1.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/case1.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/case1.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/case8.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/case8.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/case8.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/case17.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/case17.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/case17.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/case20.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/case20.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/case20.bed (SNP-major mode)
```

```
## controls
 TriadSim(input.plink.file, file.path(tempdir(),'ctrl'), fr.desire=0.05,
          pathways=list(1:4,5:8),n.ped=1000, N.brk=3, target.snp=NA,P0=0.001,
          is.OR=TRUE,risk.exposure= 1,risk.pathway.unexposed=c(1.5, 2),
          risk.pathway.exposed=c(1.5, 2), is.case=FALSE, e.fr=NA, pop1.frac=NA,
          P0.ratio=1,rcmb.rate, no_cores=1)
```

```
## [1]  21 118 121 140 155 168 218 383
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/ctrl1.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/ctrl1.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/ctrl1.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/ctrl8.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/ctrl8.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/ctrl8.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/ctrl17.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/ctrl17.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/ctrl17.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/ctrl20.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/ctrl20.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/ctrl20.bed (SNP-major mode)
```

Some additional details

The source data may contain genotyping errors that cause non-Mendelian inheritance patterns. For these non-Mendelian families, genotypes of the three individuals in the family will be set to missing at the corresponding SNPs. We assume nonpaternity and adoption have both been ruled out in QC for the source data.

This function requires at least two pathway SNPs, eithe two SNPs in the one pathway or two pathways each involving one SNP. If the users are interested in a single SNP scenario one can trick the function by setting the number of pathway to 2, each with a single SNP in the pathway but only the SNP in the first pathway carries a risk while that in the second pathway does not change risk. See below for an example.

```
TriadSim(input.plink.file, file.path(tempdir(),'singleSNP'), fr.desire=0.05,
         pathways=list(1,2),n.ped=1000, N.brk=3, target.snp=NA,P0=0.001,
         is.OR=FALSE,risk.exposure= 1,risk.pathway.unexposed=c(1.5, 1),
         risk.pathway.exposed=c(1.5, 1), is.case=TRUE, e.fr=NA, pop1.frac=NA,
         P0.ratio=1,rcmb.rate, no_cores=1)
```

```
## [1] 118 140
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/singleSNP1.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/singleSNP1.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/singleSNP1.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/singleSNP8.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/singleSNP8.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/singleSNP8.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/singleSNP17.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/singleSNP17.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/singleSNP17.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/singleSNP20.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/singleSNP20.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/singleSNP20.bed (SNP-major mode)
```

Facility functions

The following set of functions is provided in case users want to have more control over the simulation parameters. They are called by the main function to generate simulations. Users do not need to call them directly.

`pick_target.snp`

Users can manually pick the target SNPs in the pathway or use the facility function `pick_target.snp` to pick the set of target SNPs in the pathway(s) based on a desired allele frequency. The example below uses the example files that come with the package to select 8 SNPs with allele frequencies close to 0.05. The function returns the selected target SNPs by giving the row numbers (i.e., the order) of the corresponding SNPs among all the SNPs in the associated "bim" file. For example a return of "1084 2044 3285 4016 5117 6067 7077 8187" means the SNPs at rows 1084 2044 3285 4016 5117 6067 7077 8187 are selected to be the target SNPs in the pathway.

```
 m.file <- file.path(system.file(package = "TriadSim"),'extdata/pop1_4chr_mom')
 f.file <- file.path(system.file(package = "TriadSim"),'extdata/pop1_4chr_dad')
 picked.target <- pick_target.snp(c(m.file,f.file),0.05, 8)
```

```
## [1]  21 118 121 140 155 168 218 383
```

```
 cat('Target SNPs picked:',picked.target[[2]],'\n')
```

```
## Target SNPs picked: 21 118 121 140 155 168 218 383
```

get.target.geno

The function **get.target.geno** retrieves genotypes of the target SNPs and returns the genotypes of the triads in a list of three elements: the observed genotypes of the mothers from family 1 to family n repeated twice, genotypes of the fathers from family 1 to family n repeated twice and genotypes of children from family 1 to n followed by (stacking on top of) genotypes of the complements in the same family order.

```
target.snp <- picked.target[[2]]
 m.file <- file.path(system.file(package = "TriadSim"),'extdata/pop1_4chr_mom')
 f.file <- file.path(system.file(package = "TriadSim"),'extdata/pop1_4chr_dad')
 k.file <- file.path(system.file(package = "TriadSim"),'extdata/pop1_4chr_kid')
# the preloaded data frame snp.all2 contains the data frame read from the corresponding .bim file.
target.geno <- get.target.geno(c(m.file,f.file,k.file), target.snp,snp.all2)
```

The output *target.geno* is a list of three elements, each being a matrix of genotypes

```
length(target.geno)
```

```
## [1] 3
```

For this example the genotypes form a 2000 x 8 numerical matrix (2x1000 families and 8 SNPs)

```
mom.target <- target.geno[[1]]
dad.target <- target.geno[[2]]
kid.target <- target.geno[[3]]
str(mom.target)
```

```
##  num [1:2000, 1:8] 1 2 2 2 2 2 2 2 2 0 ...
```

To increase diversity, TriadSim introduces break points at each chromosome, selecting them independently for each triad being simulated. The break points can be picked manually or using the function **get.brks**. The function tends to pick the break points at recombination hotspots if such data are passed in as an input parameter *rcmb.rate*. In the following example the same number of break points (N.brk=3) are selected for each chromosome.

```
found.brks <- get.brks(N.brk=3,n.ped=1000, snp.all2, target.snp,rcmb.rate=rcmb.rate)
breaks <- found.brks[[1]]
family.position <- found.brks[[2]]
```

This function returns a list of two items. The first is a 1000 x 17 matrix of integers showing where the chromosomal breaks are to take place (in terms of the order of the SNPs in the PLINK files) for each individual in the simulated trios. Each chromosome has 3 breaks, adding to that is the number of breaks between chromosomes, i.e., 3, and the first and the last SNPs, and this is where the 17 comes from. Here 1000 denotes the number of triads in the simulated data as defined by the `n.ped` input parameter.

```
dim(breaks)
```

```
## [1] 1000   17
```

```
head(breaks)
```

```
##   [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## 1    0   27   41   99  173  202  263  266  297   300   324   337   355
## 1    0   49   61   65  173  196  226  266  297   300   309   317   355
## 1    0   82   84   99  173  183  197  250  297   317   339   340   355
## 1    0    6   61  114  173  222  232  244  297   300   306   339   355
## 1    0  102  158  162  173  210  248  264  297   300   309   324   355
## 1    0   22   36   76  173  196  201  204  297   300   306   339   355
##   [,14] [,15] [,16] [,17]
## 1   362   369   407   412
## 1   372   394   398   412
## 1   370   381   407   412
## 1   360   375   404   412
## 1   369   398   408   412
## 1   370   377   407   412
```

The second one is a 1000 x 8 matrix showing the chromosomal segments out of which each target SNP is selected for each simulated trio.

```
dim(family.position)
```

```
## [1] 1000    8
```

```
head(family.position)
```

```
##   [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
## 1    1    4    4    4    4    4    6   15
## 1    1    4    4    4    4    4    6   14
## 1    1    4    4    4    4    4    7   15
## 1    2    4    4    4    4    4    5   15
## 1    1    2    2    2    2    4    6   14
## 1    1    4    4    4    4    4    8   15
```

The users can also select different number of break points for different chromosomes by prodiving a vector of integers as the input for N.brk. Note that the number of integers should be of same length as the number of chromosomes, each number giving the number of break points of the corresponding chromosomes.

```
found.brks <- get.brks(N.brk=c(4,3,2,2),n.ped=1000, snp.all2, target.snp,rcmb.rate=rcmb.rate)
breaks <- found.brks[[1]]
family.position <- found.brks[[2]]
dim(breaks)
```

```
## [1] 1000   16
```

```
head(breaks)
```

```
##   [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## 1    0    4   58   85  162  173  196  233  255   297   311   326   355
## 1    0   29   81  130  137  173  246  264  266   297   312   313   355
## 1    0   13   49  102  154  173  197  204  248   297   326   347   355
## 1    0   35   97   99  161  173  189  214  226   297   303   326   355
## 1    0   40   49  136  162  173  226  232  243   297   330   340   355
## 1    0   76   88  133  143  173  225  232  254   297   316   326   355
##   [,14] [,15] [,16]
## 1   376   398   412
## 1   373   408   412
## 1   383   407   412
## 1   360   369   412
## 1   381   383   412
## 1   372   376   412
```

fit.risk.model.par

fit.risk.model.par is a function that resamples families based on the specified risk model. It can simulate a homogenous scenario or a stratified scenario with two subpopulations. The risk model can involve exposure main effects as well as gene by exposure interactions. This function is parallelized to shorten the running time. An example call for simulating a binary phenotype is given below.

```
betas <- c(-6.4, 3.2, 5.8)
pwy <- list(1:4,5:8)
## scenarios of genetic main effects only for a binary phenotype
fitted.model1 <- fit.risk.model.par(n.ped=1000,brks=breaks,target.snp,
                                    fam.pos=family.position, mom.tar=mom.target,
                                    dad.tar=dad.target, kid.tar=kid.target,
                                    pathways=pwy, betas, e.fr=NA, betas,pop1.frac= NA,
                                    rate.beta=NA,qtl= FALSE,
                                    out.put.file=file.path(tempdir(),'riskmodel1'),no_cores=1)
```

This function returns a list of five items. For a scenario involving a binary trait in a homogeneous population and no gene-environment interaction only the first two items contain the data needed. The first is a 1000 x 16 matrix of integers showing which source families are picked for each chromosomal segments in each of the 1000 simulated trios.

```
sel.fam <- fitted.model1[[1]]
colnames(sel.fam) <- paste('seg',1:ncol(sel.fam),sep="_")
rownames(sel.fam) <- paste('fam',1:nrow(sel.fam),sep="_")
dim(sel.fam)
```

```
## [1] 1000   15
```

```
head(sel.fam)
```

```
##       seg_1 seg_2 seg_3 seg_4 seg_5 seg_6 seg_7 seg_8 seg_9 seg_10 seg_11
## fam_1   138  1304    25  1841   532   272    37   236  1008   1123    883
## fam_2   985  1858   508  1050    79  1339  1716    57  1414   1234    271
## fam_3   437   219  1973    59   764  1589   547    92  1086    895    562
## fam_4  1985   496  1901  1568  1427   938   347  1522   672    339   1334
## fam_5  1864  1844  1914   897   521    60   516   225   482    504    921
## fam_6   207  1036  1609  1171  1275  1061  1936  1352    48   1118   1697
##       seg_12 seg_13 seg_14 seg_15
## fam_1   1731   1551   1759   1820
## fam_2    106   1374    862    662
## fam_3    689   1854   1122     35
## fam_4   1873   1028    295    805
## fam_5   1516   1709   1749     24
## fam_6     16   1967   1378   1905
```

The second one is a 1000 x 8 matrix showing the genotypes at the 8 target SNPs.

```
sim.pathway.geno <-  fitted.model1[[2]]
colnames(sim.pathway.geno) <- paste('target.snp',1:ncol(sim.pathway.geno),sep="_")
rownames(sim.pathway.geno) <- paste('fam',1:nrow(sim.pathway.geno),sep="_")
dim(sim.pathway.geno)
```

```
## [1] 3000    8
```

```
head(sim.pathway.geno)
```

```
##       target.snp_1 target.snp_2 target.snp_3 target.snp_4 target.snp_5
## fam_1            2            0            2            2            2
## fam_2            2            2            2            2            2
## fam_3            2            2            2            2            2
## fam_4            2            2            2            2            2
## fam_5            2            2            0            0            2
## fam_6            2            2            2            2            2
##       target.snp_6 target.snp_7 target.snp_8
## fam_1            2            0            2
## fam_2            2            2            2
## fam_3            2            2            2
## fam_4            2            2            0
## fam_5            2            2            2
## fam_6            2            2            2
```

The following is an example call for a scenario involving gene-environment interaction for a binary phenotype.

```
## a scenario of gene-environment interaction  for a binary phenotype
betas.e <- c(-6.4, 3.9, 6.5)

fitted.model2 <- fit.risk.model.par(n.ped=1000,brks=breaks,target.snp,
                                    fam.pos=family.position,mom.tar=mom.target,
                                    dad.tar=dad.target, kid.tar=kid.target,
                                    pathways=pwy,betas, e.fr= 0.2,
                                    betas.e,pop1.frac= NA,rate.beta=NA,qtl= FALSE,
                                    out.put.file=file.path(tempdir(),'riskmodel2'),
                                    no_cores=1)
```

For this scenario the third returned item contains exposure data.

```
exposure <-  fitted.model2[[3]]
table(exposure)
```

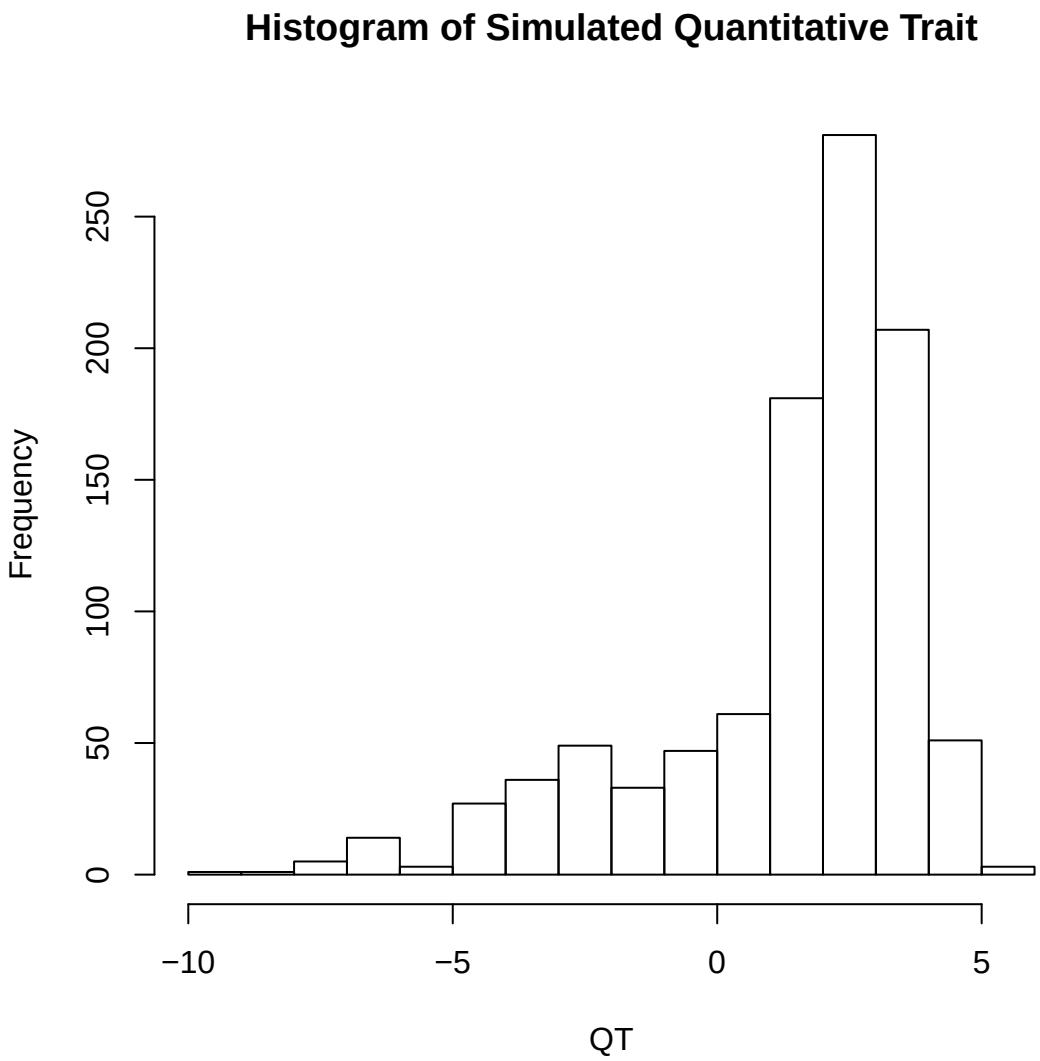
```
## exposure
##   0   1
## 798 202
```

The following is an example call for a quantitative trait scenario.

```
## scenarios of a quantitative trait phenotype
fitted.model3 <- fit.risk.model.par(n.ped=1000,brks=breaks,target.snp,
                                    fam.pos=family.position, mom.tar=mom.target,
                                    dad.tar=dad.target, kid.tar=kid.target,
                                    pathways=pwy,betas, e.fr=NA,
                                    betas,pop1.frac= NA,rate.beta=NA,qtl=TRUE,
                                    out.put.file=file.path(tempdir(),'riskmodel3'),
                                    no_cores=1)
```

For this scenario the fifth returned item contains data for the quantitative phenotype.

```
qt.pheno <-  fitted.model3[[5]]
hist(qt.pheno,main='Histogram of Simulated Quantitative Trait',xlab='QT')
```



glue.chr.segment.par

`glue.chr.segment.par` is a function that splices the triad chromosomal segments into "complete" trios. The spliced trio sets are written into separate plink files chromosome by chromosome. It is parallelized and if no "no_cores"" value is given half of the total number of CPUs available will be used in the parallelization.

```
glue.chr.segment.par(c(m.file,f.file,k.file),file.path(tempdir(),'triad'),breaks,sel.fam,
                     snp.all2,sim.pathway.geno,target.snp,pop.vec=NA,no_cores=1,flip=TRUE)
```

```
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad1.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad1.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad1.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad8.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad8.bim
```

```
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad8.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad17.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad17.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad17.bed (SNP-major mode)
## coercing object of mode  numeric  to SnpMatrix
## Writing FAM file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad20.fam
## Writing extended MAP file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad20.bim
## Writing BED file to C:\Users\shi2\AppData\Local\Temp\2\RtmpwFq6Au/triad20.bed (SNP-major mode)
```