Package 'GWASbyCluster'

October 11, 2019

Type Package

Title Identifying Significant SNPs in Genome Wide Association Studies (GWAS) via Clustering

Version 0.1.7

Date 2019-10-09

Author Yan Xu, Li Xing, Jessica Su, Xuekui Zhang<UBC.X.Zhang@gmail.com>, Weiliang Qiu <Weiliang.Qiu@gmail.com>

Maintainer Li Xing <sfulxing@gmail.com>

Depends R (>= 3.5.0), Biobase

Imports stats, snpStats, methods, rootSolve, limma

Description
Identifying disease-associated significant SNPs using clustering approach. This package is implementation of method proposed in Xu et al (2019) <DOI:10.1038/s41598-019-50229-6>.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2019-10-11 09:30:06 UTC

R topics documented:

esSim *An ExpressionSet Object Storing Simulated Genotype Data*

Description

An ExpressionSet object storing simulated genotype data. The minor allele frequency (MAF) of cases has the same prior as that of controls.

Usage

```
data("esSim")
```

Details

In this simulation, we generate additive-coded genotypes for 3 clusters of SNPs based on a mixture of 3 Bayesian hierarchical models.

In cluster $+$, the minor allele frequency (MAF) θ_{x+} of cases is greater than the MAF θ_{y+} of controls.

In cluster 0, the MAF θ_0 of cases is equal to the MAF of controls.

In cluster $-$, the MAF θ_{x-} of cases is smaller than the MAF θ_{y-} of controls.

The proportions of the 3 clusters of SNPs are π_+, π_0, and π_-, respectively.

We assume a "half-flat shape" bivariate prior for the MAF in cluster $+$

$$2h_+\left(\theta_{x+}\right)h_+\left(\theta_{y+}\right)I\left(\theta_{x+}>\theta_{y+}\right),$$

where $I(a)$ is hte indicator function taking value 1 if the event a is true, and value 0 otherwise. The function h_+ is the probability density function of the beta distribution $Beta\left(\alpha_+,\beta_+\right)$.

We assume θ_0 has the beta prior $Beta(\alpha_0,\beta_0)$.

We also assume a "half-flat shape" bivariate prior for the MAF in cluster $-$

$$2h_-\left(\theta_{x-}\right)h_-\left(\theta_{y-}\right)I\left(\theta_{x-}>\theta_{y-}\right).$$

The function h_- is the probability density function of the beta distribution $Beta\left(\alpha_-,\beta_-\right)$.

Given a SNP, we assume Hardy-Weinberg equilibrium holds for its genotypes. That is, given MAF θ, the probabilities of genotypes are

$$Pr(geno=2)=\theta^2$$

$$Pr(geno=1)=2\theta\left(1-\theta\right)$$

$$Pr(geno=0)=\left(1-\theta\right)^2$$

We also assume the genotypes 0 (wild-type), 1 (heterozygote), and 2 (mutation) follows a multinomial distribution $Multinomial\left\{1,\left[\theta^2,2\theta\left(1-\theta\right),\left(1-\theta\right)^2\right]\right\}$

We set the number of cases as 100, the number of controls as 100, and the number of SNPs as 1000.

The hyperparameters are $\alpha_+=2$, $\beta_+=5$, $\pi_+=0.1$, $\alpha_0=2$, $\beta_0=5$, $\pi_0=0.8$, $\alpha_-=2$, $\beta_-=5$, $\pi_-=0.1$.

Note that when we generate MAFs from the half-flat shape bivariate priors, we might get very small MAFs or get MAFs > 0.5. In these cased, we then delete this SNP.

So the final number of SNPs generated might be less than the initially-set number 1000 of SNPs.

For the dataset stored in `esSim`, there are 872 SNPs. 83 SNPs are in cluster -, 714 SNPs are in cluster 0, and 75 SNPs are in cluster $+$.

References

Yan X, Xing L, Su J, Zhang X, Qiu W. Model-based clustering for identifying disease-associated SNPs in case-control genome-wide association studies. Scientific Reports 9, Article number: 13686 (2019) https://www.nature.com/articles/s41598-019-50229-6.

Examples

```
data(esSim)
print(esSim)

pDat=pData(esSim)
print(pDat[1:2,])
print(table(pDat$memSubjs))

fDat=fData(esSim)
print(fDat[1:2,])
print(table(fDat$memGenes))
print(table(fDat$memGenes2))
```

---

esSimDiffPriors *An ExpressionSet Object Storing Simulated Genotype Data*

Description

An ExpressionSet object storing simulated genotype data. The minor allele frequency (MAF) of cases has different prior than that of controls.

Usage

```
data("esSimDiffPriors")
```

Details

In this simulation, we generate additive-coded genotypes for 3 clusters of SNPs based on a mixture of 3 Bayesian hierarchical models.

In cluster $+$, the minor allele frequency (MAF) θ_{x+} of cases is greater than the MAF θ_{y+} of controls.

In cluster 0, the MAF θ_0 of cases is equal to the MAF of controls.

In cluster $-$, the MAF θ_{x-} of cases is smaller than the MAF θ_{y-} of controls.

The proportions of the 3 clusters of SNPs are π_+, π_0, and π_-, respectively.

We assume a "half-flat shape" bivariate prior for the MAF in cluster $+$

$$2h_{x+}\left(\theta_{x+}\right)h_{y+}\left(\theta_{y+}\right)I\left(\theta_{x+}>\theta_{y+}\right),$$

where $I(a)$ is hte indicator function taking value 1 if the event a is true, and value 0 otherwise. The function h_{x+} is the probability density function of the beta distribution $Beta\left(\alpha_{x+},\beta_{x+}\right)$. The function h_{y+} is the probability density function of the beta distribution $Beta\left(\alpha_{y+},\beta_{y+}\right)$.

We assume θ_0 has the beta prior $Beta(\alpha_0,\beta_0)$.

We also assume a "half-flat shape" bivariate prior for the MAF in cluster $-$

$$2h_{x-}\left(\theta_{x-}\right)h_{y-}\left(\theta_{y-}\right)I\left(\theta_{x-}>\theta_{y-}\right).$$

The function h_{x-} is the probability density function of the beta distribution $Beta\left(\alpha_{x-},\beta_{x-}\right)$. The function h_{y-} is the probability density function of the beta distribution $Beta\left(\alpha_{y-},\beta_{y-}\right)$.

Given a SNP, we assume Hardy-Weinberg equilibrium holds for its genotypes. That is, given MAF θ, the probabilities of genotypes are

$$Pr(geno=2)=\theta^2$$

$$Pr(geno=1)=2\theta\left(1-\theta\right)$$

$$Pr(geno=0)=\left(1-\theta\right)^2$$

We also assume the genotypes 0 (wild-type), 1 (heterozygote), and 2 (mutation) follows a multinomial distribution $Multinomial\left\{1,\left[\theta^2,2\theta\left(1-\theta\right),\left(1-\theta\right)^2\right]\right\}$

We set the number of cases as 100, the number of controls as 100, and the number of SNPs as 1000.

The hyperparameters are $\alpha_{x+}=2$, $\beta_{x+}=3$, $\alpha_{y+}=2$, $\beta_{y+}=8$, $\pi_+=0.1$,

$\alpha_0=2$, $\beta_0=5$, $\pi_0=0.8$,

$\alpha_{x-}=2$, $\beta_{x-}=8$, $\alpha_{y-}=2$, $\beta_{y-}=3$, $\pi_-=0.1$.

Note that when we generate MAFs from the half-flat shape bivariate priors, we might get very small MAFs or get MAFs >0.5. In these cased, we then delete this SNP.

So the final number of SNPs generated might be less than the initially-set number 1000 of SNPs.

For the dataset stored in `esSim`, there are 838 SNPs. 64 SNPs are in cluster -, 708 SNPs are in cluster 0, and 66 SNPs are in cluster $+$.

References

Yan X, Xing L, Su J, Zhang X, Qiu W. Model-based clustering for identifying disease-associated SNPs in case-control genome-wide association studies. Scientific Reports 9, Article number: 13686 (2019) https://www.nature.com/articles/s41598-019-50229-6.

Examples

```
data(esSimDiffPriors)
print(esSimDiffPriors)

pDat=pData(esSimDiffPriors)
print(pDat[1:2,])
```

```
print(table(pDat$memSubjs))

fDat=fData(esSimDiffPriors)
print(fDat[1:2,])
print(table(fDat$memGenes))
print(table(fDat$memGenes2))
```

---

estMemSNPs *Estimate SNP cluster membership*

Description

Estimate SNP cluster membership. Only update cluster mixture proportions. Assume the 3 clusters have different sets of hyperparameters.

Usage

```
estMemSNPs(es,
           var.memSubjs = "memSubjs",
           eps = 0.001,
           MaxIter = 50,
           bVec = rep(3, 3),
           pvalAdjMethod = "fdr",
           method = "FDR",
           fdr = 0.05,
           verbose = FALSE)
```

Arguments

es	An ExpressionSet object storing SNP genotype data. It contains 3 matrices. The first matrix, which can be extracted by `exprs` method (e.g., `exprs(es)`), stores genotype data, with rows are SNPs and columns are subjects. The second matrix, which can be extracted by `pData` method (e.g., `pData(es)`), stores phenotype data describing subjects. Rows are subjects, and columns are phenotype variables. The third matrix, which can be extracted by `fData` method (e.g., `fData(es)`), stores feature data describing SNPs. Rows are SNPs and columns are feature variables.
var.memSubjs	character. The name of the phenotype variable indicating subject's case-control status. It must take only two values: 1 indicating case and 0 indicating control.
eps	numeric. A small positive number as threshold for convergence of EM algorithm.
MaxIter	integer. A positive integer indicating maximum iteration in EM algorithm.
bVec	numeric. A vector of 2 elements. Indicates the parameters of the symmetric Dirichlet prior for proportion mixtures.
pvalAdjMethod	character. Indicating p-value adjustment method. c.f. `p.adjust`.

method	method to obtain SNP cluster membership based on the responsibility matrix. The default value is "FDR". The other possible value is "max". see details.
fdr	numeric. A small positive FDR threshold used to call SNP cluster membership
verbose	logical. Indicating if intermediate and final results should be output.

Details

In this simulation, we generate additive-coded genotypes for 3 clusters of SNPs based on a mixture of 3 Bayesian hierarchical models.

In cluster $+$, the minor allele frequency (MAF) θ_{x+} of cases is greater than the MAF θ_{y+} of controls.

In cluster 0, the MAF θ_0 of cases is equal to the MAF of controls.

In cluster $-$, the MAF θ_{x-} of cases is smaller than the MAF θ_{y-} of controls.

The proportions of the 3 clusters of SNPs are π_+, π_0, and π_-, respectively.

We assume a "half-flat shape" bivariate prior for the MAF in cluster $+$

$$2h_+\left(\theta_{x+}\right)h_+\left(\theta_{y+}\right)I\left(\theta_{x+}>\theta_{y+}\right),$$

where $I(a)$ is hte indicator function taking value 1 if the event a is true, and value 0 otherwise. The function h_+ is the probability density function of the beta distribution $Beta\left(\alpha_+,\beta_+\right)$.

We assume θ_0 has the beta prior $Beta(\alpha_0,\beta_0)$.

We also assume a "half-flat shape" bivariate prior for the MAF in cluster $-$

$$2h_-\left(\theta_{x-}\right)h_-\left(\theta_{y-}\right)I\left(\theta_{x-}>\theta_{y-}\right).$$

The function h_- is the probability density function of the beta distribution $Beta\left(\alpha_-,\beta_-\right)$.

Given a SNP, we assume Hardy-Weinberg equilibrium holds for its genotypes. That is, given MAF θ, the probabilities of genotypes are

$$Pr(geno=2)=\theta^2$$

$$Pr(geno=1)=2\theta\left(1-\theta\right)$$

$$Pr(geno=0)=\left(1-\theta\right)^2$$

We also assume the genotypes 0 (wild-type), 1 (heterozygote), and 2 (mutation) follows a multinomial distribution $Multinomial\left\{1,\left[\theta^2,2\theta\left(1-\theta\right),\left(1-\theta\right)^2\right]\right\}$

For each SNP, we calculat its posterior probabilities that it belongs to cluster k. This forms a matrix with 3 columns. Rows are SNPs. The 1st column is the posterior probability that the SNP belongs to cluster $-$. The 2nd column is the posterior probability that the SNP belongs to cluster 0. The 3rd column is the posterior probability that the SNP belongs to cluster $+$. We call this posterior probability matrix as responsibility matrix. To determine which cluster a SNP eventually belongs to, we can use 2 methods. The first method (the default method) is "FDR" method, which will use FDR criterion to determine SNP cluster membership. The 2nd method is use the maximum posterior probability to decide which cluster a SNP belongs to.

Value

A list of 12 elements

wMat	matrix of posterior probabilities. The rows are SNPs. There are 3 columns. The first column is the posterior probability that a SNP belongs to cluster - given genotypes of subjects. The second column is the posterior probability that a SNP belongs to cluster 0 given genotypes of subjects. The third column is the posterior probability that a SNP belongs to cluster + given genotypes of subjects.
memSNPs	a vector of SNP cluster membership for the 3-cluster partitionfrom the mixture of 3 Bayesian hierarchical models.
memSNPs2	a vector of binary SNP cluster membership. 1 indicates the SNP has different MAFs between cases and controls. 0 indicates the SNP has the same MAF in cases as that in controls.
piVec	a vector of cluster mixture proportions.
alpha.p	the first shape parameter of the beta prior for MAF obtaind from initial 3-cluster partitions based on GWAS for cluster +.
beta.p	the second shape parameter of the beta prior for MAF obtaind from initial 3-cluster partitions based on GWAS for cluster +.
alpha0	the first shape parameter of the beta prior for MAF obtaind from initial 3-cluster partitions based on GWAS for cluster 0.
beta0	the second shape parameter of the beta prior for MAF obtaind from initial 3-cluster partitions based on GWAS for cluster 0.
alpha.n	the first shape parameter of the beta prior for MAF obtaind from initial 3-cluster partitions based on GWAS for cluster -.
beta.n	the second shape parameter of the beta prior for MAF obtaind from initial 3-cluster partitions based on GWAS for cluster -.
loop	number of iteration in EM algorithm
diff	sum of the squared difference of cluster mixture proportions between current iteration and previous iteration in EM algorithm. if eps < eps, we claim the EM algorithm converges.
res.limma	object returned by limma

Author(s)

Yan Xu <yanxu@uvic.ca>, Li Xing <sfulxing@gmail.com>, Jessica Su <rejas@channing.harvard.edu>, Xuekui Zhang <xuekui@uvic.ca>, Weiliang Qiu <Weiliang.Qiu@gmail.com>

References

Yan X, Xing L, Su J, Zhang X, Qiu W. Model-based clustering for identifying disease-associated SNPs in case-control genome-wide association studies. Scientific Reports 9, Article number: 13686 (2019) https://www.nature.com/articles/s41598-019-50229-6.

Examples

```
data(esSimDiffPriors)
print(esSimDiffPriors)

es=esSimDiffPriors[1:500,]
fDat = fData(es)
print(fDat[1:2,])
print(table(fDat$memGenes))

res = estMemSNPs(
  es = es,
  var.memSubjs = "memSubjs")

print(table(fDat$memGenes, res$memSNPs))
```

`estMemSNPs.oneSetHyperPara` *Estimate SNP cluster membership*

Description

Estimate SNP cluster membership. Only update cluster mixture proportions. Assume all 3 clusters have the same set of hyperparameters.

Usage

```
estMemSNPs.oneSetHyperPara(es,
           var.memSubjs = "memSubjs",
           eps = 1.0e-3,
           MaxIter = 50,
           bVec = rep(3, 3),
           pvalAdjMethod = "none",
           method = "FDR",
           fdr = 0.05,
           verbose = FALSE)
```

Arguments

`es`	An ExpressionSet object storing SNP genotype data. It contains 3 matrices. The first matrix, which can be extracted by `exprs` method (e.g., `exprs(es)`), stores genotype data, with rows are SNPs and columns are subjects. The second matrix, which can be extracted by `pData` method (e.g., `pData(es)`), stores phenotype data describing subjects. Rows are subjects, and columns are phenotype variables. The third matrix, which can be extracted by `fData` method (e.g., `fData(es)`), stores feature data describing SNPs. Rows are SNPs and columns are feature variables.

var.memSubjs	character. The name of the phenotype variable indicating subject's case-control status. It must take only two values: 1 indicating case and 0 indicating control.
eps	numeric. A small positive number as threshold for convergence of EM algorithm.
MaxIter	integer. A positive integer indicating maximum iteration in EM algorithm.
bVec	numeric. A vector of 2 elements. Indicates the parameters of the symmetric Dirichlet prior for proportion mixtures.
pvalAdjMethod	character. Indicating p-value adjustment method. c.f. p.adjust.
method	method to obtain SNP cluster membership based on the responsibility matrix. The default value is "FDR". The other possible value is "max". see details.
fdr	numeric. A small positive FDR threshold used to call SNP cluster membership
verbose	logical. Indicating if intermediate and final results should be output.

Details

We characterize the distribution of genotypes of SNPs by a mixture of 3 Bayesian hierarchical models. The 3 Bayeisan hierarchical models correspond to 3 clusters of SNPs.

In cluster $+$, the minor allele frequency (MAF) θ_{x+} of cases is greater than the MAF θ_{y+} of controls.

In cluster 0, the MAF θ_0 of cases is equal to the MAF of controls.

In cluster $-$, the MAF θ_{x-} of cases is smaller than the MAF θ_{y-} of controls.

The proportions of the 3 clusters of SNPs are π_+, π_0, and π_-, respectively.

We assume a "half-flat shape" bivariate prior for the MAF in cluster $+$

$$2h\left(\theta_{x+}\right)h\left(\theta_{y+}\right)I\left(\theta_{x+}>\theta_{y+}\right),$$

where $I(a)$ is hte indicator function taking value 1 if the event a is true, and value 0 otherwise. The function h is the probability density function of the beta distribution $Beta\left(\alpha,\beta\right)$.

We assume θ_0 has the beta prior $Beta(\alpha,\beta)$.

We also assume a "half-flat shape" bivariate prior for the MAF in cluster $-$

$$2h\left(\theta_{x-}\right)h\left(\theta_{y-}\right)I\left(\theta_{x-}>\theta_{y-}\right).$$

Given a SNP, we assume Hardy-Weinberg equilibrium holds for its genotypes. That is, given MAF θ, the probabilities of genotypes are

$$Pr(geno=2)=\theta^2$$

$$Pr(geno=1)=2\theta\left(1-\theta\right)$$

$$Pr(geno=0)=\left(1-\theta\right)^2$$

We also assume the genotypes 0 (wild-type), 1 (heterozygote), and 2 (mutation) follows a multinomial distribution $Multinomial\left\{1,\left[\theta^2,2\theta\left(1-\theta\right),\left(1-\theta\right)^2\right]\right\}$

For each SNP, we calculat its posterior probabilities that it belongs to cluster k. This forms a matrix with 3 columns. Rows are SNPs. The 1st column is the posterior probability that the SNP belongs

to cluster $-$. The 2nd column is the posterior probability that the SNP belongs to cluster 0. The 3rd column is the posterior probability that the SNP belongs to cluster $+$. We call this posterior probability matrix as responsibility matrix. To determine which cluster a SNP eventually belongs to, we can use 2 methods. The first method (the default method) is "FDR" method, which will use FDR criterion to determine SNP cluster membership. The 2nd method is use the maximum posterior probability to decide which cluster a SNP belongs to.

Value

A list of 10 elements

`wMat`	matrix of posterior probabilities. The rows are SNPs. There are 3 columns. The first column is the posterior probability that a SNP belongs to cluster - given genotypes of subjects. The second column is the posterior probability that a SNP belongs to cluster 0 given genotypes of subjects. The third column is the posterior probability that a SNP belongs to cluster + given genotypes of subjects.
`memSNPs`	a vector of SNP cluster membership for the 3-cluster partitionfrom the mixture of 3 Bayesian hierarchical models.
`memSNPs2`	a vector of binary SNP cluster membership. 1 indicates the SNP has different MAFs between cases and controls. 0 indicates the SNP has the same MAF in cases as that in controls.
`piVec`	a vector of cluster mixture proportions.
`alpha`	the first shape parameter of the beta prior for MAF obtaind from initial 3-cluster partitions based on GWAS.
`beta`	the second shape parameter of the beta prior for MAF obtaind from initial 3-cluster partitions based on GWAS.
`loop`	number of iteration in EM algorithm
`diff`	sum of the squared difference of cluster mixture proportions between current iteration and previous iteration in EM algorithm. if `eps` < `eps`, we claim the EM algorithm converges.
`res.limma`	object returned by limma

Author(s)

Yan Xu <yanxu@uvic.ca>, Li Xing <sfulxing@gmail.com>, Jessica Su <rejas@channing.harvard.edu>, Xuekui Zhang <xuekui@uvic.ca>, Weiliang Qiu <Weiliang.Qiu@gmail.com>

References

Yan X, Xing L, Su J, Zhang X, Qiu W. Model-based clustering for identifying disease-associated SNPs in case-control genome-wide association studies. Scientific Reports 9, Article number: 13686 (2019) https://www.nature.com/articles/s41598-019-50229-6.

Examples

```
data(esSimDiffPriors)
```

```
print(esSimDiffPriors)
fDat = fData(esSimDiffPriors)
print(fDat[1:2,])
print(table(fDat$memGenes))

res = estMemSNPs.oneSetHyperPara(
  es = esSimDiffPriors,
  var.memSubjs = "memSubjs")

print(table(fDat$memGenes, res$memSNPs))
```

`simGenoFunc` *Simulate Genotype Data from a Mixture of 3 Bayesian Hierarchical Models*

Description

Simulate Genotype Data from a Mixture of 3 Bayesian Hierarchical Models. The minor allele frequency (MAF) of cases has the same prior as that of controls.

Usage

```
simGenoFunc(nCases = 100,
            nControls = 100,
            nSNPs = 1000,
            alpha.p = 2,
            beta.p = 5,
            pi.p = 0.1,
            alpha0 = 2,
            beta0 = 5,
            pi0 = 0.8,
            alpha.n = 2,
            beta.n = 5,
            pi.n = 0.1,
            low = 0.02,
            upp = 0.5,
            verbose = FALSE)
```

Arguments

`nCases`	integer. Number of cases.
`nControls`	integer. Number of controls.
`nSNPs`	integer. Number of SNPs.
`alpha.p`	numeric. The first shape parameter of Beta prior in cluster +.
`beta.p`	numeric. The second shape parameter of Beta prior in cluster +.
`pi.p`	numeric. Mixture proportion for cluster +.

alpha0	numeric. The first shape parameter of Beta prior in cluster 0.
beta0	numeric. The second shape parameter of Beta prior in cluster 0.
pi0	numeric. Mixture proportion for cluster 0.
alpha.n	numeric. The first shape parameter of Beta prior in cluster $-$.
beta.n	numeric. The second shape parameter of Beta prior in cluster $-$.
pi.n	numeric. Mixture proportion for cluster $-$.
low	numeric. A small positive value. If a MAF generated from half-flat shape bivariate prior is smaller than low, we will delete the SNP to be generated.
upp	numeric. A positive value. If a MAF generated from half-flat shape bivariate prior is greater than upp, we will delete the SNP to be generated.
verbose	logical. Indicating if intermediate results or final results should be output to output screen.

Details

In this simulation, we generate additive-coded genotypes for 3 clusters of SNPs based on a mixture of 3 Bayesian hierarchical models.

In cluster $+$, the minor allele frequency (MAF) θ_{x+} of cases is greater than the MAF θ_{y+} of controls.

In cluster 0, the MAF θ_0 of cases is equal to the MAF of controls.

In cluster $-$, the MAF θ_{x-} of cases is smaller than the MAF θ_{y-} of controls.

The proportions of the 3 clusters of SNPs are π_+, π_0, and π_-, respectively.

We assume a "half-flat shape" bivariate prior for the MAF in cluster $+$

$$2h_+\left(\theta_{x+}\right)h_+\left(\theta_{y+}\right)I\left(\theta_{x+}>\theta_{y+}\right),$$

where $I(a)$ is hte indicator function taking value 1 if the event a is true, and value 0 otherwise. The function h_+ is the probability density function of the beta distribution $Beta\left(\alpha_+,\beta_+\right)$.

We assume θ_0 has the beta prior $Beta(\alpha_0,\beta_0)$.

We also assume a "half-flat shape" bivariate prior for the MAF in cluster $-$

$$2h_-\left(\theta_{x-}\right)h_-\left(\theta_{y-}\right)I\left(\theta_{x-}>\theta_{y-}\right).$$

The function h_- is the probability density function of the beta distribution $Beta\left(\alpha_-,\beta_-\right)$.

Given a SNP, we assume Hardy-Weinberg equilibrium holds for its genotypes. That is, given MAF θ, the probabilities of genotypes are

$$Pr(geno=2)=\theta^2$$

$$Pr(geno=1)=2\theta\left(1-\theta\right)$$

$$Pr(geno=0)=\left(1-\theta\right)^2$$

We also assume the genotypes 0 (wild-type), 1 (heterozygote), and 2 (mutation) follows a multinomial distribution $Multinomial\left\{1,\left[\theta^2,2\theta\left(1-\theta\right),\left(1-\theta\right)^2\right]\right\}$

Note that when we generate MAFs from the half-flat shape bivariate priors, we might get very small MAFs or get MAFs >0.5. In these cased, we then delete this SNP.

So the final number of SNPs generated might be less than the initially-set number of SNPs.

Value

An ExpressionSet object stores genotype data.

Author(s)

Yan Xu <yanxu@uvic.ca>, Li Xing <sfulxing@gmail.com>, Jessica Su <rejas@channing.harvard.edu>, Xuekui Zhang <xuekui@uvic.ca>, Weiliang Qiu <Weiliang.Qiu@gmail.com>

References

Yan X, Xing L, Su J, Zhang X, Qiu W. Model-based clustering for identifying disease-associated SNPs in case-control genome-wide association studies. Scientific Reports 9, Article number: 13686 (2019) https://www.nature.com/articles/s41598-019-50229-6.

Examples

```
set.seed(2)

esSim = simGenoFunc(
  nCases = 100,
  nControls = 100,
  nSNPs = 500,
  alpha.p = 2, beta.p = 5, pi.p = 0.1,
  alpha0 = 2, beta0 = 5, pi0 = 0.8,
  alpha.n = 2, beta.n = 5, pi.n = 0.1,
  low = 0.02, upp = 0.5, verbose = FALSE
)

print(esSim)
pDat = pData(esSim)
print(pDat[1:2,])
print(table(pDat$memSubjs))

fDat = fData(esSim)
print(fDat[1:2,])
print(table(fDat$memGenes))
print(table(fDat$memGenes2))
```

---

simGenoFuncDiffPriors *Simulate Genotype Data from a Mixture of 3 Bayesian Hierarchical Models*

Description

Simulate Genotype Data from a Mixture of 3 Bayesian Hierarchical Models. The minor allele frequency (MAF) of cases has different priors than that of controls.

Usage

```
simGenoFuncDiffPriors(
    nCases = 100,
    nControls = 100,
    nSNPs = 1000,
    alpha.p.ca = 2,
    beta.p.ca = 3,
    alpha.p.co = 2,
    beta.p.co = 8,
    pi.p = 0.1,
    alpha0 = 2,
    beta0 = 5,
    pi0 = 0.8,
    alpha.n.ca = 2,
    beta.n.ca = 8,
    alpha.n.co = 2,
    beta.n.co = 3,
    pi.n = 0.1,
    low = 0.02,
    upp = 0.5,
    verbose = FALSE)
```

Arguments

nCases	integer. Number of cases.
nControls	integer. Number of controls.
nSNPs	integer. Number of SNPs.
alpha.p.ca	numeric. The first shape parameter of Beta prior in cluster $+$ for cases.
beta.p.ca	numeric. The second shape parameter of Beta prior in cluster $+$ for cases.
alpha.p.co	numeric. The first shape parameter of Beta prior in cluster $+$ for controls.
beta.p.co	numeric. The second shape parameter of Beta prior in cluster $+$ for controls.
pi.p	numeric. Mixture proportion for cluster $+$.
alpha0	numeric. The first shape parameter of Beta prior in cluster 0.
beta0	numeric. The second shape parameter of Beta prior in cluster 0.
pi0	numeric. Mixture proportion for cluster 0.
alpha.n.ca	numeric. The first shape parameter of Beta prior in cluster $-$ for cases.
beta.n.ca	numeric. The second shape parameter of Beta prior in cluster $-$ for cases.
alpha.n.co	numeric. The first shape parameter of Beta prior in cluster $-$ for controls.
beta.n.co	numeric. The second shape parameter of Beta prior in cluster $-$ for controls.
pi.n	numeric. Mixture proportion for cluster $-$.
low	numeric. A small positive value. If a MAF generated from half-flat shape bivariate prior is smaller than low, we will delete the SNP to be generated.

upp	numeric. A positive value. If a MAF generated from half-flat shape bivariate prior is greater than upp, we will delete the SNP to be generated.
verbose	logical. Indicating if intermediate results or final results should be output to output screen.

Details

In this simulation, we generate additive-coded genotypes for 3 clusters of SNPs based on a mixture of 3 Bayesian hierarchical models.

In cluster $+$, the minor allele frequency (MAF) θ_{x+} of cases is greater than the MAF θ_{y+} of controls.

In cluster 0, the MAF θ_0 of cases is equal to the MAF of controls.

In cluster $-$, the MAF θ_{x-} of cases is smaller than the MAF θ_{y-} of controls.

The proportions of the 3 clusters of SNPs are π_+, π_0, and π_-, respectively.

We assume a "half-flat shape" bivariate prior for the MAF in cluster $+$

$$2h_+\left(\theta_{x+}\right)h_+\left(\theta_{y+}\right)I\left(\theta_{x+}>\theta_{y+}\right),$$

where $I(a)$ is hte indicator function taking value 1 if the event a is true, and value 0 otherwise. The function h_+ is the probability density function of the beta distribution $Beta\left(\alpha_+,\beta_+\right)$.

We assume θ_0 has the beta prior $Beta(\alpha_0,\beta_0)$.

We also assume a "half-flat shape" bivariate prior for the MAF in cluster $-$

$$2h_-\left(\theta_{x-}\right)h_-\left(\theta_{y-}\right)I\left(\theta_{x-}>\theta_{y-}\right).$$

The function h_- is the probability density function of the beta distribution $Beta\left(\alpha_-,\beta_-\right)$.

Given a SNP, we assume Hardy-Weinberg equilibrium holds for its genotypes. That is, given MAF θ, the probabilities of genotypes are

$$Pr(geno=2)=\theta^2$$

$$Pr(geno=1)=2\theta\left(1-\theta\right)$$

$$Pr(geno=0)=\left(1-\theta\right)^2$$

We also assume the genotypes 0 (wild-type), 1 (heterozygote), and 2 (mutation) follows a multinomial distribution $Multinomial\left\{1,\left[\theta^2,2\theta\left(1-\theta\right),\left(1-\theta\right)^2\right]\right\}$

Note that when we generate MAFs from the half-flat shape bivariate priors, we might get very small MAFs or get MAFs >0.5. In these cased, we then delete this SNP.

So the final number of SNPs generated might be less than the initially-set number of SNPs.

Value

An ExpressionSet object stores genotype data.

Author(s)

Yan Xu <yanxu@uvic.ca>, Li Xing <sfulxing@gmail.com>, Jessica Su <rejas@channing.harvard.edu>, Xuekui Zhang <xuekui@uvic.ca>, Weiliang Qiu <Weiliang.Qiu@gmail.com>

References

Yan X, Xing L, Su J, Zhang X, Qiu W. Model-based clustering for identifying disease-associated SNPs in case-control genome-wide association studies. Scientific Reports 9, Article number: 13686 (2019) https://www.nature.com/articles/s41598-019-50229-6.

Examples

```
set.seed(2)

esSimDiffPriors = simGenoFuncDiffPriors(
  nCases = 100,
  nControls = 100,
  nSNPs = 500,
  alpha.p.ca = 2, beta.p.ca = 3,
  alpha.p.co = 2, beta.p.co = 8, pi.p = 0.1,
  alpha0 = 2, beta0 = 5, pi0 = 0.8,
  alpha.n.ca = 2, beta.n.ca = 8,
  alpha.n.co = 2, beta.n.co = 3, pi.n = 0.1,
  low = 0.02, upp = 0.5, verbose = FALSE
)

print(esSimDiffPriors)

pDat = pData(esSimDiffPriors)
print(pDat[1:2,])
print(table(pDat$memSubjs))

fDat = fData(esSimDiffPriors)
print(fDat[1:2,])
print(table(fDat$memGenes))
print(table(fDat$memGenes2))
```

Index