Package 'TSGSIS'

October 12, 2022

Title Two Stage-Grouped Sure Independence Screening

Description
To provide a high dimensional grouped variable selection approach for detection of whole-genome SNP effects and SNP-SNP interactions, as described in Fang et al. (2017, under review).

Version 0.1

Author Yao-Hwei Fang, Jie-Huei Wang, and Chao A. Hsiung

Maintainer Yao-Hwei Fang <yhfang@nhri.org.tw>

Date 2017-04-18

Depends R (>= 3.2.3), glmnet, MASS, stats

License GPL (>= 2)

Encoding UTF-8

LazyData true

Repository CRAN

RoxygenNote 6.0.1

NeedsCompilation no

Date/Publication 2017-04-18 06:43:40 UTC

R topics documented:

`TSGSIS` *Two Stage-Grouped Sure Independence Screening*

Description

The package is a beta version that provides a high-dimensional grouped variable selection approach for detection of whole-genome SNP effects and SNP-SNP interactions, as described in Fang et al. (2017, under review). The proposed TSGSIS is developed to study interactions that may not have marginal effects.

Usage

```
TSGSIS(XA, Y, Gene_list, ntest, lambda, Method)
```

Arguments

XA	The $N \times P$ matrix of XA. There are N individuals and P variables in matrix, with one individual in each row and one genotype in each column.
Y	The $N \times 1$ matrix of Y. It can be real number or binary outcome.
Gene_list	The $a \times d$ matrix of the Gene_list. a is the maximal number of gene size in the Gene_list which other values are denoted by 0. d is the number of genes.
ntest	The ntest ($< N$) is the number of testing data for evaluation of MSE.
lambda	The lambda is the parameter of Lasso regression.
Method	"Reg" for quantitative trait modeling, "LR" for disease trait modeling.

Value

Returns a result of screening

result	First element of the result is the MSE of testing data, the rest elements are the important SNP effects and SNP-SNP interactions after TSGSIS modeling.

Note

The missing value (NA) in the XA and Y is not allowed in this beta version.

References

Yao-Hwei Fang, Jie-Huei Wang and Chao A. Hsiung (2017). TSGSIS: A High-dimensional Grouped Variable Selection Approach for Detection of Whole-genome SNP-SNP Interactions. (Under revision in Bioinformatics)

Examples

```
#We investigate the performance of TS-GSIS under model 1 with intra-gene correlation rho = 0.2,
#trait dispersion sigma^2 = 1, effect size k = 3 and homogeneous MAF.
#Given 100 SNPs with gene size d = 10, 500 unrelated individuals are simulated.
#(Please refer to the Figure 3 of the reference)

library(glmnet)
library(MASS)

set.seed(1)# Set seed
#Parameter setting
ntotal = 500
p = 100
n.pred = 10 #Gene sizes
rho = 0.2 #Intra-gene correlation in block
k = 3 #Effect size
vari = 1 #Sigma2
```

```
lambda = 0.5 #For lasso parameter
ntest = 150 #For evaluation
Method="Reg"#For quantitative trait
#Heterogeneous MAF: randomly set to 0.35, 0.2 or 0.1 with equal likelihood.
MAF = matrix(0,2,3)
MAF[,1] = c(0.1225,0.5775)
MAF[,2] = c(0.04,0.36)
MAF[,3] = c(0.01,0.19)
#Trait Y
modelY = "k*XA[,1] - k*(sqrt(rho))*XA[,5] + k*XA[,31]*XA[,5] + rnorm(ntotal,0,vari)"

PAS1 = function(z){ g = paste("A",z,sep = "")
return(g)
}#Define colname fun.
norm = function(a) (a-mean(a))/sd(a) #Define standardization fun.

#The codes of simulated data for quantitative trait are listed in the following. We use mvrnorm
#function to simulate the genotype data. Y is continuous with normal distribution, all errors are
#assumed to be normally distributed with a mean of zero and a variance of one (vari = 1).
out = array(0, dim=c(n.pred)) #For LOOCV
corrmat = diag(rep(1-rho, n.pred)) + matrix(rho, n.pred, n.pred) #Create covariance matrix with rho
corrmat[,5] = sqrt(rho)
corrmat[5,] = sqrt(rho)
corrmat[5,5] = 1
L = array(0, dim=c(n.pred, n.pred, (p/n.pred)))
L[,,1] = corrmat
for(i in 2:(p/n.pred)){
L[,,i] = diag(rep(1-rho, n.pred)) + matrix(rho, n.pred, n.pred)
}
temp = "bdiag(L[,,1]"
for (i in 2:(p/n.pred)){
temp = paste(temp,",","L[,,",i,"]", sep="")
}
temp = paste(temp,")", sep="")
corrmat2 = eval(parse(text=temp))

beta0 = matrix(0,p,1) #Simulate genotype
X = matrix(0,ntotal,p)
X = mvrnorm(ntotal, beta0, corrmat2 , tol=1e-8, empirical=FALSE)
XA = data.frame(X); colnames(XA) <- c(sapply(1:p,PAS1))
C1 = matrix(0,1,p)
C2 = matrix(0,1,p)
tempMAF = sample(3,1)
for (i in 1:p){
C2[1,i] = quantile(X[,i], MAF[1,tempMAF])
C1[1,i] = quantile(X[,i], MAF[2,tempMAF])
XA[X[,i] > C1[,i],i] = 1
XA[X[,i] <= C1[,i] & X[,i] >= C2[,i],i] = 0
XA[X[,i] < C2[,i],i] = -1
}
XA = apply(XA, 2, norm) #Standardization

Y = eval(parse(text=modelY)) #Simulate gaussian response
```

```
temp = 1:p
Gene_list = matrix(temp,nrow=n.pred) #Create Gene-based SNP set
#Run TSGSIS fun. with XA, Y, Gene_list, ntest (for predicted model), lambda of lasso regression,
#Method types: "Reg" for quantitative trait; "LR" for disease trait.
Screen_result = TSGSIS(XA, Y, Gene_list, ntest, lambda, Method)
```

Index