Package `episcan`

September 15, 2018

Title Scan Pairwise Epistasis

Version 0.0.1

Author Beibei Jiang <beibei_jiang@psych.mpg.de> and Benno Pütz <puetz@psych.mpg.de>

Maintainer Beibei Jiang <beibei_jiang@psych.mpg.de>


Depends R (>= 3.5.0)

License GPL (>= 2)

Encoding UTF-8

Suggests testthat, knitr, rmarkdown

VignetteBuilder knitr

LazyData true

RoxygenNote 6.1.0

NeedsCompilation no

Repository CRAN

Date/Publication 2018-09-14 23:02:19 UTC

R topics documented:

  checkchunksize .................................................. 2
  epiblaster1geno .................................................. 3
  epiblaster2genos .................................................. 4
  epiHSIC ............................................................ 5
  epiHSIC1geno ...................................................... 6
  epiHSIC2genos ...................................................... 7
checkchunksize

Description
Check the chunk size whether it is over the given number of variables (variant) in genotype data. If yes, reset the chunk size equal to the number of variables (variant).

Usage
checkchunksize(c, m, n = NULL, ...)

Arguments
c an integer indicating the set chunk size.
m an integer indicating the number of variables (variant) in geno1 if there is only one genotype input.
n an integer indicating the number of variables (variant) in geno2 if there are two genotype inputs. The default is NULL.
...
not used.

Value
an integer indicating the chunk size

Examples
set.seed(123)
geno1 <- matrix(sample(0:2, size = 1000, replace = TRUE, prob = c(0.5, 0.3, 0.2)),
ncol = 10)
geno2 <- matrix(sample(0:2, size = 2000, replace = TRUE, prob = c(0.4, 0.3, 0.3)),
ncol = 20)

# if chunk size is smaller, there is no problem
chunksize <- 10
checkchunksize(chunksize, ncol(geno1))

# if chunk size is bigger than the number of columns in genotype input,
# set chunk size equal to the number of columns in genotype input
chunksize <- 12
epiblaster1geno

checkchunksize(chunksize, ncol(geno1))

# if chunk size is bigger than the number of columns of geno1 and geno2,
# set chunk size equal to the minima number of columns of geno1 and geno2
chunksize <- 50
checkchunksize(chunksize, ncol(geno1), ncol(geno2))

epiblaster1geno  Parallelized calculation of the difference of correlation coefficients
and compute Z test with one genotype input

Description

Calculate the difference of correlation coefficients between cases and controls, conduct Z test for
the differences (values) and choose variant pairs with the significance below the given threshold for
output.

Usage

epiblaster1geno(geno, pheno, chunk = 1000, zpthres = 1e-05,
outfile = "NONE", suffix = ".txt", ...)

Arguments

geno is the normalized genotype data. It can be a matrix or a dataframe, or a big.matrix
object (from bigmemory). The columns contain the information of variables and
the rows contain the information of samples.

pheno a vector containing the binary phenotype information (case/control). The values
are either 0 (control) or 1 (case).

chunk is the number of variants in each chunk. Default: 1000.

zpthres is the significance threshold to select variant pairs for output. Default is 1e-6.

outfile is the base of out filename. Default: 'NONE'.

suffix is the suffix of out filename. Default: '.txt'.

... not used.

Value

null

Examples

# simulate some data
set.seed(123)
geno1 <- matrix(sample(0:2, size = 1000, replace = TRUE, prob = c(0.5, 0.3, 0.2)), ncol = 10)
dimnames(geno1) <- list(row = paste0("IND", 1:nrow(geno1)), col = paste0("rs", 1:ncol(geno1)))
p1 <- c(rep(0, 60), rep(1, 40))
# normalized data
gen01 <- scale(geno1)

# one genotype with case-control phenotype
epiblaster1geno(geno = geno1,
pheno = pheno,
outfile = "episcan_1geno_cc",
suffix = ".txt",
zpthres = 0.9,
chunk = 10)

# take a look at the result
res <- read.table("episcan_1geno_cc.txt",
header = TRUE,
stringsAsFactors = FALSE)
head(res)

epiblaster2genos

Parallelized calculation of the difference of correlation coefficients and compute Z test with two genotype inputs

Description

Calculate the difference of correlation coefficients between cases and controls, conduct Z test for the differences (values) and choose variant pairs with the significance below the given threshold for output.

Usage

epiblaster2genos(geno1, geno2, pheno, chunk = 1000, zpthres = 1e-05,
outfile = "NONE", suffix = ".txt", ...)

Arguments

gen01 is the first normalized genotype data. It can be a matrix or a dataframe, or a big.matrix object from bigmemory. The columns contain the information of variables and the rows contain the information of samples.

gen02 is the second normalized genotype data. It can be a matrix or a dataframe, or a big.matrix object from bigmemory. The columns contain the information of variables and the rows contain the information of samples.

pheno a vector containing the binary phenotype information (case/control). The values are either 0 (control) or 1 (case).

chunk is the number of variants in each chunk.

zpthres is the significance threshold to select variant pairs for output. Default is 1e-6.

outfile is the prefix of out filename.

suffix is the suffix of out filename.

... not used.
epiHSIC

Value
null

Examples

# simulate some data
set.seed(123)
gen1 <- matrix(sample(c(0:2, size = 1000, replace = TRUE, prob = c(0.5, 0.3, 0.2)), ncol = 10)
gen2 <- matrix(sample(c(0:2, size = 2000, replace = TRUE, prob = c(0.4, 0.3, 0.3)), ncol = 20)
dimnames(gen1) <- list(row = paste0("IND", 1:nrow(geno1)), col = paste0("rs", 1:ncol(geno1)))
dimnames(gen2) <- list(row = paste0("IND", 1:nrow(geno2)), col = paste0("exm", 1:ncol(geno2)))
p1 <- c(rep(0, 60), rep(1, 40))

# normalized data
gen1 <- scale(geno1)
gen2 <- scale(geno2)

# two genotypes with quantitative phenotype
epiBlaster2genos(geno1 = geno1,
gen2 = geno2,
pheno = p1, outfile = "episcan_2geno_cc",
suffix = ".txt",
zpthres = 0.9,
chunk = 10)

# take a look at the result
res <- read.table("episcan_2geno_cc.txt",
header = TRUE,
stringsAsFactors = FALSE)
head(res)

epiHSIC

Calculate HSIC values

Description
Calculate HSIC values

Usage
epiHSIC(A = NULL, B = NULL, P = NULL, ...)

Arguments
A is one matrix.
B is one matrix.
P is "phenotype", a vector.
... not used.
epiHSIC1geno

**Value**

a matrix

**Author(s)**

Beibei Jiang <beibei_jiang@psych.mpg.de>

**Examples**

```r
# simulate some data
set.seed(123)
geno1 <- matrix(sample(0:2, size = 1000, replace = TRUE, prob = c(0.5, 0.3, 0.2)), ncol = 10)
geno2 <- matrix(sample(0:2, size = 2000, replace = TRUE, prob = c(0.4, 0.3, 0.3)), ncol = 20)
dimnames(geno1) <- list(row = paste0("IND", 1:nrow(geno1)), col = paste0("rs", 1:ncol(geno1)))
dimnames(geno2) <- list(row = paste0("IND", 1:nrow(geno2)), col = paste0("exm", 1:ncol(geno2)))
epiHSIC(A = scale(geno1),
   B = scale(geno2),
   P = rnorm(100))
```

**Description**

Calculate epistasis using HSIC with one genotype input

**Usage**

```r
epiHSIC1geno(geno = NULL, pheno, chunk = 1000, zpthres = 1e-05,
   outfile = "NONE", suffix = ".txt", ...)
```

**Arguments**

- `geno` is the normalized genotype data. It can be a matrix or a dataframe, or a big.matrix object from bigmemory. The columns contain the information of variables and the rows contain the information of samples.
- `pheno` is a vector containing the normalized phenotype information.
- `chunk` is the number of variants in each chunk.
- `zpthres` is the significance threshold to select variant pairs for output. Default is 1e-6.
- `outfile` is the basename of out filename.
- `suffix` is the suffix of out filename.
- ... not used.
epiHSIC2genos

Value
null

Author(s)
Beibei Jiang <beibei_jiang@psych.mpg.de>

Examples

```r
# simulate some data
set.seed(123)
gen1 <- matrix(sample(0:2, size = 1000, replace = TRUE, prob = c(0.5, 0.3, 0.2)), ncol = 10)
dimnames(gen1) <- list(row = paste0("IND", 1:nrow(gen1)), col = paste0("rs", 1:ncol(gen1))
p2 <- rnorm(100, mean = 5, sd = 10)

# normalized data
gen1 <- scale(gen1)
p2 <- as.vector(unlist(scale(p2)))

# one genotypes with quantitative phenotype
epiHSIC1geno(geno = gen1,
pheno = p2,
outfile = "episcan_1geno_quant",
suffix = ".txt",
zpthres = 0.9,
chunk = 10)

# take a look at the result
res <- read.table("episcan_1geno_quant.txt",
header = TRUE,
stringsAsFactors = FALSE)
head(res)
```

epiHSIC2genos  Calculate epistasis using HSIC with two genotype inputs

Description

Calculate the significance of epistasis according the definition of HSIC, conduct Z test for HSIC values and choose variant pairs with the significance below the given threshold for output.

Usage

```r
epiHSIC2genos(geno1 = NULL, geno2 = NULL, pheno = NULL,
chunk = 1000, zpthres = 1e-05, outfile = "NONE", suffix = ".txt",
...)
```
**Arguments**

- `geno1` is the first normalized genotype data. It can be a matrix or a dataframe, or a big.matrix object from `bigmemory`. The columns contain the information of variables and the rows contain the information of samples.

- `geno2` is the second normalized genotype data. It can be a matrix or a dataframe, or a big.matrix object from `bigmemory`. The columns contain the information of variables and the rows contain the information of samples.

- `pheno` is a vector containing the normalized phenotype information.

- `chunk` is the number of variants in each chunk.

- `zpthres` is the significance threshold for cut-off output of the variant pairs.

- `outfile` is the basename of out filename.

- `suffix` is the suffix of out filename.

- `...` not used

**Value**

- `null`

**Examples**

```r
# simulate some data
set.seed(123)
N = 10; nR = 15; rows = 10
geno1 <- matrix(sample(0:2, size = N*rows, replace = TRUE, prob = c(0.5, 0.3, 0.2)), ncol = N)
geno2 <- matrix(sample(0:2, size = nR*rows, replace = TRUE, prob = c(0.4, 0.3, 0.3)), ncol = nR)
dimnames(geno1) <- list(row = paste("Ind", 1:nrow(geno1)), col = paste("rs", 1:ncol(geno1)))
dimnames(geno2) <- list(row = paste("Ind", 1:nrow(geno2)), col = paste("exm", 1:ncol(geno2)))
p2 <- rnorm(rows, mean = 5, sd = 10)

# normalized data
geno1 <- scale(geno1)
geno2 <- scale(geno2)
p2 <- as.vector(unlist(scale(p2)))

# two genotypes with quantitative phenotype
epiHSIC2genos(geno1 = geno1,
geno2 = geno2,
pheno = p2,
outfile = "episcan_2geno_quant",
suffix = ".txt",
zpthres = 0.9,
chunk = 10)

# take a look at the result
res <- read.table("episcan_2geno_quant.txt",
header = TRUE,
stringsAsFactors = FALSE)
head(res)
```
episcan

Scan pairwise epistasis

Description
Genomic interaction analysis with EPIBLASTER or epistasis-oriented Hilbert–Schmidt Independence Criterion (HSIC).

Usage
episcan(geno1, geno2 = NULL, pheno = NULL,
phetype = c("case-control", "quantitative"), outfile = "episcan",
suffix = ".txt", zpthres = 1e-06, chunksize = 1000, scale = TRUE,
...)

Arguments
- geno1: a data.frame or matrix of the first genotype data. big.matrix object from bigmemory also works. The columns contain the information of variables and the rows contain the information of samples.
- geno2: optional. A data.frame or matrix of the second genotype data. big.matrix object from bigmemory also works. The columns contain the information of variables and the rows contain the information of samples.
- pheno: a vector (named or not). If not provided, the value of geno2 will be used if it is a vector. The values is either case-control phenotype (0, 1) or quantitative phenotype.
- phetype: character string. Either "case-control" or "quantitative".
- outfile: output file name. Default is "episcan".
- suffix: suffix for output file. Default is ".txt". The final result will be stored in outfilesuffix.
- zpthres: is the significance threshold to select variant pairs for output. Default is 1e-6.
- chunksize: the number of variants in each chunk.
- scale: a logical value to define wheter the input data needs to be normalized. Default is TRUE which means, by default, all the genotype data will be normalized and if the phetype is "quantitative", the phenotype will also be normalized.
- ... not used.

Value
null

Author(s)
Beibei Jiang <beibei_jiang@psych.mpg.de>
References


Examples

```r
# simulate some data
set.seed(123)
gen1 <- matrix(sample(0:2, size = 1000, replace = TRUE, prob = c(0.5, 0.3, 0.2)),
nrow = 10)
gen2 <- matrix(sample(0:2, size = 2000, replace = TRUE, prob = c(0.4, 0.3, 0.3)),
nrow = 20)
dimnames(gen1) <- list(row = paste0("IND", 1:nrow(gen1)),
col = paste0("rs", 1:ncol(gen1)))
dimnames(gen2) <- list(row = paste0("IND", 1:nrow(gen2)),
col = paste0("exm", 1:ncol(gen2)))
p1 <- c(rep(0, 60), rep(1, 40))
p2 <- rnorm(100)

# one genotype with case-control phenotype
episca(gen1 = gen1,
gen2 = NULL,
pheno = p1,
phetype = "case-control",
outfile = "episcan_1geno_cc",
suffix = ".txt",
zpthres = 0.9,
chunksize = 10,
scale = TRUE)

# take a look at the result
res <- read.table("episcan_1geno_cc.txt",
header = TRUE,
stringsAsFactors = FALSE)
head(res)

# two genotypes with quantitative phenotype
episca(gen1 = gen1,
gen2 = gen2,
pheno = p2,
phetype = "quantitative",
outfile = "episcan_2geno_quant",
suffix = ".txt",
zpthres = 0.9,
chunksize = 10,
scale = TRUE)
```
getcor

Get correlation matrix

Description

Fast calculation of correlation matrix on CPU (the idea is from WGCNA fast function for pearson correlations)

Usage

getcor(A = NULL, B = NULL, method = "pearson", ...)

Arguments

- `A` is a matrix or data.frame.
- `B` is a matrix or data.frame.
- `method` is a character string indicating which correlation coefficient is to be computed. Current version only supports "pearson" correlation.
- `...` not used.

Value

correlation matrix

Author(s)

Beibei Jiang <beibei_jiang@psych.mpg.de>

Examples

```r
set.seed(123)
A <- matrix(rnorm(100, mean = 5, sd = 10), ncol = 10)
B <- matrix(rnorm(200, mean = 10, sd = 100), ncol = 20)
C <- getcor(A, B)
```

ithChunk

index set for idx-th chunk of size chunk for n elements

Description

For proper use of this function it will return the set of variant indices corresponding to the idx-th chunk of size chunk in n variants, taking care of the case that the last chunk might have less than n elements. If used with an idx-value outside the possible chunks (i.e., negative or larger than ceiling(n/chunk)) an empty vector (numeric(0)) is returned.
WriteSnpPairs

Usage

ithChunk(idx, n, chunk = 1000)

Arguments

idx chunk index (which chunk, first is 1)

n total number of variants

chunk desired chunksize

Value

index range into variants for chunk idx (see details)

WriteSnpPairs Write out epistasis result (normal matrix)

Description

Write out the result of epistasis analysis. Z score matrix is not a symmetric matrix.

Usage

WriteSnpPairs(Zmatrix, indexArr, outfile = "NONE", ...)

Arguments

Zmatrix is the Z score matrix (non-symmetric matrix).

indexArr is the index of Zmarix whose z score is over the given zpthres.

outfile is the SNP pairs file for the second stage.

... not used.

Value

null

Author(s)

Beibei Jiang <beibei_jiang@psych.mpg.de>
**WriteSnppairs_sym**

Write out epistasis result (symmetric matrix)

**Description**
Write out the result of epistasis analysis. Z score matrix is a symmetric matrix.

**Usage**
```
WriteSnppairs_sym(Zmatrix, indexArr, outfile = "NONE", ...)
```

**Arguments**
- **Zmatrix** is the Z score matrix (symmetric matrix).
- **indexArr** is the index of Z matrix whose z score is over the given zpthres.
- **outfile** is the SNP pairs file for the second stage.
- ... not used.

**Value**
null

**Author(s)**
Beibei Jiang <beibei_jiang@psych.mpg.de>

---

**ZtoP**

Convert Z-score to corresponding p-value

**Description**
Convert Z score to corresponding p-values

**Usage**
```
ZtoP(z.score, ...)
```

**Arguments**
- **z.score** Z-score(s) (either scalar or vector).
- ... not used.

**Value**
 corresponding p-value(s).
Note

Due to the IEEE number limits of representing doubles, any Z score over 37.51929999999999765 will be converted to a p-value of 1e-309.

Author(s)

Beibei Jiang <beibei_jiang@psych.mpg.de> and Benno Pütz <puetz@psych.mpg.de>
Index

checkchunksize, 2
epiblaster1geno, 3
epiblaster2genos, 4
epiHSIC, 5
epiHSIC1geno, 6
epiHSIC2genos, 7
episcan, 9
getcor, 11
ithChunk, 11
WriteSnpPairs, 12
WriteSnpPairs_sym, 13
ZtoP, 13