Package ‘amritaPackage’

October 19, 2013

Type Package

Title Quasi-bootstrap association tests

Author Amrita Ray <amray@stanford.edu>

Maintainer Amrita Ray <amray@stanford.edu>

Depends R (>= 2.10.0),kinship2,matrixcalc

Description Provides quasi-bootstrap p-values for provided and user-provided tests.

License GPL (>= 3)

Encoding UTF-8

LazyLoad yes

URL http://stanford.edu/~amray/thepackage/index.html

Version 0.01-1

Date 2013-10-19

Collate ’amritaPackage-package.R’’bootstrap_fn.R’’burden_statistic_fn.R’’C_fn.R’’kernel_statistic_fn.R’’kinship_fn.R’’mb_statistic_fn.R’’p

R topics documented:

`amritaPackage-package` *The amritaPackage Package*

Description

Quasi-bootstrap association tests.

Details

Package:	amritaPackage
Type:	Package
Version:	0.01-1
Date:	2013-10-19
Depends:	R (>= 2.10.0), kinship2, matrixcalc
Encoding:	UTF-8
License:	GPL (>= 3)
LazyLoad:	yes
URL:	http://stanford.edu/~amray/thepackage/index.html

Provides quasi-bootstrap p-values for provided and user-provided tests.

Author(s)

Amrita Ray <amray@stanford.edu>

`bootstrap_fn` *Quasi-Bootstrap*

Description

Compute quasi-bootstrap pvalues for association statistics

Usage

```
bootstrap_fn(N_bootstrap_reps, genotype, ped_object,
  test_statistic_fns, ...)
```

Arguments

`N_bootstrap_reps`
is the number of bootstrap replications

`genotype` is the genotype matrix

`ped_object` is the user input pedigree data

`test_statistic_fns`
is a list of test statistics. This includes the default list of three statistics (Burden, Kernel and Madsen-Browning), and any user specified statistic.

`map_object` is the user input mapfile of the markers

Details

This function returns a list of the observed statistics, number of bootstrap replications and the quasi-bootstrap pvalues. The quasi-bootstrap method can be applied to any genetic data with design (case control, pedigree) to compute association statistics and corresponding pvalues. The idea is to bootstrap from the decorrelated genotype matrix to circumvent the problem that subjects' genotypes at any marker may be correlated.

Author(s)

Ray and Gong

Examples

```
data(example_data)
 genotype = geno_object[,2:ncol(geno_object)]
 test_statistic_fns = list(
     burden = burden_statistic_fn,
     kernel = kernel_statistic_fn,
     mb = mb_statistic_fn)
 print(bootstrap_fn(100, genotype, ped_object, test_statistic_fns, map_object))
```

`burden_statistic_fn` *Multi-locus Burden statistic*

Description

This function returns the multi-locus burden statistic

Usage

```
burden_statistic_fn(genotype, ped_object, Psi, p_hat,
  r_hat, map_object)
```

Arguments

`genotype`	is the genotype matrix
`ped_object`	is the user input pedigree data
`Psi`	is the matrix of twice kinship coefficients between a pair of individuals
`p_hat`	is the vector of estimated minor allele frequency per marker
`r_hat`	is the matrix of estimated inter-marker correlation coefficients
`map_object`	is the user input mapfile of the markers

Burden

pvalue

Author(s)

Ray and Gong

Examples

```
data(example_data)
 genotype = geno_object[,2:ncol(geno_object)]
Psi = 2*kinship_fn(ped_object)
p_hat = p_hat_fn(genotype)
r_hat=r_hat_fn(genotype)
print(burden_statistic_fn(genotype,ped_object, Psi, p_hat, r_hat, map_object))
```

C_fn *Denominator term for two default statistics: Multi-locus Burden and Linear Kernel.*

Description

This function returns the value of $c_s$ term that is part of the denominator for Burden and Kernel statistics.

Usage

```
  C_fn(map_object, p_hat, r_hat)
```

Arguments

map_object is the user input map file of all the markers where markers are rows and 4 columns: marker name, chromosome, base pair, user-specified weights. Internal weights as function of estimated minor alelle frequency will be used if user has not specified weights.

p_hat is the estimated minor allele frequency per marker

r_hat is the estimated inter-marker correlation coefficient matrix

Author(s)

Ray and Gong

Examples

```
data(example_data)
 genotype = geno_object[,2:ncol(geno_object)]
 p_hat = p_hat_fn(genotype)
r_hat=r_hat_fn(genotype)
print(C_fn(map_object,p_hat,r_hat))
```

kernel_statistic_fn *Linear Kernel statistic*

Description

Computes linear kernel statistic and degrees of freedom

Usage

```
kernel_statistic_fn(genotype, ped_object, Psi, p_hat,
  r_hat, map_object)
```

Arguments

genotype	is the genotype matrix with individuals as rows and columns with marker genotypes as number of minor alleles.
ped_object	is the user input pedigree data, where rows are individuals, and 6 columns as pedigree id, individual id, father id, mother id, gender, and affection status
Psi	is the matrix of twice kinship coefficients between a pair of individuals
p_hat	is the vector of estimated minor allele frequency per marker
r_hat	is the matrix of estimated inter-marker correlation coefficients
map_object	is the user input mapfile of the markers

Details

This function returns the linear Kernel statistic (Schaid et al.)

Kernel

Linear

Author(s)

Ray and Gong

References

Schaid (2013)- Ask Alice

Examples

```
data(example_data)
 genotype = geno_object[,2:ncol(geno_object)]
Psi = 2*kinship_fn(ped_object)
p_hat = p_hat_fn(genotype)
r_hat=r_hat_fn(genotype)
print(kernel_statistic_fn(genotype,ped_object, Psi, p_hat, r_hat, map_object))
```

kinship_fn *Kinship matrix*

Description

This function returns a matrix of the kinship coefficients of a pair of individuals

Usage

```
kinship_fn(ped_object)
```

Arguments

ped_object	is the user input pedigree file. This file has individuals as rows, and 6 columns: family id, individual id, father id, mother id, gender, and affection status (0 = unaffected, 1 = affected, NA = missing).

Examples

```
data(example_data)
 kinship_object=kinship_fn(ped_object)
head(kinship_object)
```

mb_statistic_fn *Madsen-Browning statistic*

Description

This function returns the multi-locus burden statistic

Usage

```
mb_statistic_fn(genotype, ped_object, Psi, p_hat, r_hat,
  map_object)
```

Arguments

genotype	is the genotype matrix
ped_object	is the user input pedigree data
Psi	is the matrix of twice kinship coefficients between a pair of individuals
p_hat	is the vector of estimated minor allele frequency per marker
r_hat	is the matrix of estimated inter-marker correlation coefficients
map_object	is the user input mapfile of the markers

Author(s)

Ray and Gail

References

Madsen and Browning (2009) "A Groupwise Association Test for Rare Mutations Using a Weighted Sum Statistic" PLoS Genet 5(2): e1000384

Examples

```
data(example_data)
genotype = geno_object[,2:ncol(geno_object)]
Psi = 2*kinship_fn(ped_object)
p_hat = p_hat_fn(genotype)
r_hat=r_hat_fn(genotype)
print(mb_statistic_fn(genotype,ped_object, Psi, p_hat, r_hat, map_object))
```

p_hat_fn *Minor allele frequency estimate*

Description

This function returns the estimate of minor allele frequency for each marker.

Usage

```
p_hat_fn(genotype, epsilon = 1e-04)
```

Arguments

`Genotype`	is the user input genotype data with rows as individuals and columns as markers with number of minor alleles.
`epsilon`	is a small quantity, if the estimate is less or equal to 0 the function returns epsilon; if the estimate is greater or equal to 1 the function returns 1-epsilon.

Examples

```
data(example_data)
 genotype=geno_object[,2:ncol(geno_object)]
 p_hat_fn(genotype,epsilon)
```

quadratic_kernel_statistic_fn *Quadratic Kernel statistic*

Description

Computes linear kernel statistic and degrees of freedom

Usage

```
quadratic_kernel_statistic_fn(genotype, ped_object, Psi,
  p_hat, r_hat, map_object)
```

Arguments

genotype	is the genotype matrix with individuals as rows and columns with marker genotypes as number of minor alleles.
ped_object	is the user input pedigree data, where rows are individuals, and 6 columns as pedigree id, individual id, father id, mother id, gender, and affection status
Psi	is the matrix of twice kinship coefficients between a pair of individuals
p_hat	is the vector of estimated minor allele frequency per marker
r_hat	is the matrix of estimated inter-marker correlation coefficients
map_object	is the user input mapfile of the markers

Details

This function returns the linear Kernel statistic (Schaid et al.)

Kernel

Linear

Author(s)

Ray and Gong

References

Schaid (2013)- Ask Alice

Examples

```
data(example_data)
 genotype = geno_object[,2:ncol(geno_object)]
Psi = 2*kinship_fn(ped_object)
p_hat = p_hat_fn(genotype)
r_hat=r_hat_fn(genotype)
print(kernel_statistic_fn(genotype,ped_object, Psi, p_hat, r_hat, map_object))
```

---

r_hat_fn *Marker correlation*

Description

This function returns estimate of inter-marker correlation matrix.

Usage

```
  r_hat_fn(genotype, epsilon = 1e-04)
```

Arguments

Genotype	is the user input genotype matrix.
epsilon	is a small quantity that is added or or subtracted from genotype depending on the number of minor alleles per marker. This adjustment #'is done so the genotypic variance at a marker is non-zero.

Examples

```
data(example_data)
 genotype=geno_object[,2:ncol(geno_object)]
 print(r_hat_fn(genotype,epsilon))
```

weight_fn *Weights*

Description

This function assigns weights per marker. Weight = user specified weight in the map file, else function of estimated minor allele frequency $\hatp(1-\hatp)^-0.5$

Usage

```
  weight_fn(map_object, p_hat)
```

Arguments

map_object	Data frame of marker information that user inputs. This file has markers as rows and columns as name, chromosome, base pairs, user-specified weights. Internal weights as function of sample minor allele frequency will be used if user does not specify weights.
p_hat	Estimate of minor allele frequency from the input genotype file.

Examples

```
data(example_data)
genotype = geno_object[,2:ncol(geno_object)]
p_hat = p_hat_fn(genotype)
print(weight_fn(map_object, p_hat))
```

Index