Package 'MultiSKAT'

December 1, 2017

Type Package

Title Sequence Kernel Association Tests with Multiple phenotypes

Version 1.0

Date 2017-03-19

Author Diptavo Dutta and Seunggeun Lee

Maintainer Diptavo Dutta <diptavo@umich.edu>

Description Kernel Regression based association tests for Multiple phenotypes. The functions aggregate variant-phenotype score statistic in a particular region and computes corresponding p-values efficiently.

Depends SKAT, nlme, copula

License GPL (>=2)

Suggests knitr,rmarkdown,R.rsp

VignetteBuilder knitr

R topics documented:

Close_SSD *Close SNP set data file (SSD)*

Description

Close the opened SNP Set data file (SSD). After using the SSD file, it must be closed

Usage

```
Close_SSD()
```

Generate_SSD_SetID *Generate SNP set data file (SSD)*

Description

Generate a SNP set data file (SSD) from binary plink data files using user specified SNP sets.

Usage

```
Generate_SSD_SetID(File.Bed, File.Bim, File.Fam,
File.SetID, File.SSD, File.Info, Is.FlipGenotype = TRUE)
```

Arguments

File.Bed	name of the binary ped file (BED).
File.Bim	name of the binary map file (BIM).
File.Fam	name of the FAM file (FAM).
File.SetID	name of the SNP set ID file that defines SNP sets. The first column must be Set ID, and the second column must be SNP ID. There should be no header.
File.SSD	name of the SSD file generated.
File.Info	name of the SSD info file generated
Is.FlipGenotype	internal use only, please do not change

Details

The SetID file is a white-space (space or tab) delimitered file with 2 columns: SetID and SNP_ID. Please keep in mind that there should be no header! The SNP_IDs and SetIDs should be less than 50 characters, otherwise, it will return an error message. The SSD file is a binary formated file with genotypes. The SSD info file is a text file with general information on data and SNP sets (first 6 rows), and information on each set (after 8th row).

`Genotype.Kernels` *Kernelize the Genotype matrix for the region to be tested.*

Description

This function kernelizes the genotype matrix

Usage

```
Genotype.Kernels(Z, obj.res, kernel = "linear.weighted", Is.Common = FALSE,
weights.beta = c(1,25), weights = NULL, impute.method = "fixed",
r.corr = 0, is_check_genotype = TRUE, is_dosage = FALSE,
missing_cutoff = 0.15, estimate_MAF = 1,max_maf=1,verbose = TRUE)
```

Arguments

`Z`	a numeric genotype matrix with each row as a different individual and each column as a separate gene/snp. Each genotype should be coded as 0, 1, 2, and 9 (or NA) for AA, Aa, aa, and missing, where A is a major allele and a is a minor allele. Missing genotypes will be imputed by the simple Hardy-Weinberg equilibrium (HWE) based imputation.
`obj.res`	an output object of the MultiSKAT_NULL function.
`kernel`	a type of kernel (default= "linear.weighted"). See detail section.
`Is.Common`	a binary variable indiciating whether a variant has the same effect on all the phenotypes (default=FALSE).
`weights.beta`	a numeric vector of parameters for the beta weights for the weighted kernels. If you want to use your own weights, please use the "weights" parameter. It will be ignored if "weights" parameter is not null.
`weights`	a numeric vector of weights for the weighted kernels.
`impute.method`	a method to impute missing genotypes (default= "fixed"). "bestguess" imputes missing genotypes as most likely values (0,1,2), "random" imputes missing genotypes by generating binomial(2,p) random variables (p is the MAF), and "fixed" imputes missing genotypes by assigning the mean genotype values (2p).
`r.corr`	the ρ parameter for the compound symmetric correlation structure kernels (default=0).
`is_check_genotype`	a logical value indicating whether to check the validity of the genotype matrix Z (default= TRUE). If Z has non-SNP data, please set it FALSE, otherwise you will get an error message. If it is FALSE and you use weighted kernels, the weights should be given through the "weights" parameter.
`is_dosage`	a logical value indicating whether the matrix Z is a dosage matrix. If it is TRUE, the function will ignore "is_check_genotype".

`missing_cutoff`
: a cutoff of the missing rates of SNPs (default=0.15). Any SNPs with missing rates higher than the cutoff will be excluded from the analysis.

`estimate_MAF`
: a numeric value indicating how to estimate MAFs for the weight calculation and the missing genotype imputation. If estimate_MAF=1 (default), it uses all samples to estimate MAFs. If estimate_MAF=2, only samples with non-missing phenotypes and covariates are used to estimate MAFs

`max_maf`
: a cutoff of the maximum minor allele frequencies (MAF) (default=1, no cutoff). Any SNPs with MAF > cutoff will be excluded from the analysis

`verbose`
: a binary indicator to display messages (default=TRUE, displays messages)

Details

There are 6 types of pre-specified kernels: "linear", "linear.weighted", "IBS", "IBS.weighted", "quadratic" and "2wayIX". Among them, "2wayIX" is a product kernel consisting of main effects and SNP-SNP interaction terms. If users want to use dosage values instead of genotypes, set is_dosage=TRUE.

The r.corr represents a ρ parameter of the unified test, $Q_\rho = (1-\rho)Q_S + \rho Q_B$, where Q_S is a SKAT test statistic, and Q_B is a weighted burden test statistic. Therefore, $\rho = 0$ results in the original weighted linear kernel SKAT, and $\rho = 1$ results in the weighted burden test (default: $\rho = 0$).

If users want to silent the messages from the function, set verbose=FALSE.

By default, SKAT uses impute.method="fixed" that imputes missing genotypes as the mean genotype values (2p). When variates are very rare and missing rates between cases and controls are highly unbalanced, impute.method="fixed" can yield inflated type I error rate. In this case, we recommend to use impute.method="bestguess", which does not suffer the same problem.

Value

This function returns the Kernelized Genotype matrix.

Author(s)

Seunuggeun Lee

References

Lee, S., Wu, M. C., and Lin, X. (2012) Optimal tests for rare variant effects in sequencing association studies. *Biostatistics*, 13, 762-775.

Wu, M. C.*, Lee, S.*, Cai, T., Li, Y., Boehnke, M., and Lin, X. (2011) Rare Variant Association Testing for Sequencing Data Using the Sequence Kernel Association Test (SKAT). *American Journal of Human Genetics*, 89, 82-93. \ * contributed equally.

Examples

```
data(MultiSKAT.example.data)
attach(MultiSKAT.example.data)
```

```
obj.null <- MultiSKAT_NULL(Phenotypes,Cov)
G.Kernel <- Genotype.Kernels(Genotypes,obj.null)
```

`Get_Genotypes_SSD` *Get Genotype data from SSD file*

Description

Read a SSD file and return a genotype matrix.

Usage

```
Get_Genotypes_SSD(SSD_INFO, Set_Index, is_ID = FALSE)
```

Arguments

`SSD_INFO`	SSD_INFO object returned from Open_SSD.
`Set_Index`	a numeric value of Set index. The set index of each set can be found from SetInfo object of SSD.INFO.
`is_ID`	a logical value indicating whether to read SNP ID (default=FALSE). If TRUE, it reads SNP IDs and use them as column names.

Value

A genotype matrix with n rows and m columns, where n is the number of samples and m is the number of SNPs.

Author(s)

Seunggeun Lee

`minP` *Minimum p-value using copula*

Description

Function to compute the p-value for the minimum p-value test statistic using Copula

Usage

```
minP(obj.res, obj.list, Z,resample = 200,
weights.beta=c(1,25),kernel = "linear.weighted",Is.Common=FALSE,
weights = NULL,r.corr = 0, impute.method = "fixed",
is_check_genotype=TRUE, is_dosage = FALSE,
missing_cutoff=0.15, estimate_MAF=1,max_maf=1,verbose = TRUE)
```

Arguments

obj.res	a NULL object from MultiSKAT_NULL or MultiSKAT_NULL_Kins
obj.list	A list containing output from MultiSKAT function
Z	A genotype matrix
resample	Number of resampling iterations to obtain the null p-values(default = 200)
kernel	a type of kernel (default= "linear.weighted"). See detail section.
Is.Common	a binary variable indiciating whether a variant has the same effect on all the phenotypes (default=FALSE).
weights.beta	a numeric vector of parameters for the beta weights for the weighted kernels. If you want to use your own weights, please use the "weights" parameter. It will be ignored if "weights" parameter is not null.
r.corr	the ρ parameter for the compound symmetric correlation structure kernels (default=0).
weights	a numeric vector of weights for the weighted kernels.
impute.method	a method to impute missing genotypes (default= "fixed"). "bestguess" imputes missing genotypes as most likely values (0,1,2), "random" imputes missing genotypes by generating binomial(2,p) random variables (p is the MAF), and "fixed" imputes missing genotypes by assigning the mean genotype values (2p).
is_check_genotype	a logical value indicating whether to check the validity of the genotype matrix Z (default= TRUE). If Z has non-SNP data, please set it FALSE, otherwise you will get an error message. If it is FALSE and you use weighted kernels, the weights should be given through the "weights" parameter.
is_dosage	a logical value indicating whether the matrix Z is a dosage matrix. If it is TRUE, the function will ignore "is_check_genotype".
missing_cutoff	a cutoff of the missing rates of SNPs (default=0.15). Any SNPs with missing rates higher than the cutoff will be excluded from the analysis.
estimate_MAF	a numeric value indicating how to estimate MAFs for the weight calculation and the missing genotype imputation. If estimate_MAF=1 (default), it uses all samples to estimate MAFs. If estimate_MAF=2, only samples with non-missing phenotypes and covariates are used to estimate MAFs
max_maf	a cutoff of the maximum minor allele frequencies (MAF) (default=1, no cutoff). Any SNPs with MAF > cutoff will be excluded from the analysis
verbose	a binary indicator to display messages (default=TRUE, displays messages)

Details

This is a copula based test of association for minimum p-value by combining the MultiSKAT tests of obj.list. Users should not try to combine more than 4 MultiSKAT tests due to numerical instability. The output p-value should be between the minimum p-value of the tests and the same multiplied by the number of tests to be combined.

Value

The function returns a numeric value which is the p-value of the minP test.

Author(s)

Diptavo Dutta

Examples

```
data(MultiSKAT.example.data)
attach(MultiSKAT.example.data)

obj.null <- MultiSKAT_NULL(Phenotypes,Cov)
out1 <- MultiSKAT(obj.null,Genotypes,cov(Phenotypes))
out2 <- MultiSKAT(obj.null,Genotypes,diag(5))
out3 <- MultiSKAT(obj.null,Genotypes,matrix(1,ncol = 5,nrow = 5))

### Combining PhC, Het and Hom
obj.list = list(out1,out2,out3)
obj.minP = minP(obj.null,obj.list,Genotypes)
str(obj.minP)

### Combining Het and Hom
obj.list = list(out2,out3)
obj.minP = minP(obj.null,obj.list,Genotypes)
str(obj.minP)
```

`minPcom` *Compute the p-value for ComCop test*

Description

This function combines the choices for ΣG and Σp kernel in a single grand omnibus test. The p-values are obtained by the minimum p-value statistic using copula.

Usage

```
minPcom(obj.res,Sigma_Ps,Z,resample = 200,
weights.beta=c(1,25),kernel = "linear.weighted", Is.Common=FALSE,
weights = NULL, impute.method = "fixed",
is_check_genotype=TRUE, is_dosage = FALSE,
missing_cutoff=0.15, estimate_MAF=1,max_maf=1,verbose = TRUE)
```

Arguments

obj.res	a NULL object output from MultiSKAT_NULL or MultiSKAT_NULL_Kins
Sigma_Ps	A list of Σp kernels to be combined
Z	Genotype matrix
resample	Number of resampling iterations to obtain the null p-values(default = 200)
kernel	a type of kernel (default= "linear.weighted"). See detail section.
Is.Common	a binary variable indiciating whether a variant has the same effect on all the phenotypes (default=FALSE).
weights.beta	a numeric vector of parameters for the beta weights for the weighted kernels. If you want to use your own weights, please use the "weights" parameter. It will be ignored if "weights" parameter is not null.
weights	a numeric vector of weights for the weighted kernels.
impute.method	a method to impute missing genotypes (default= "fixed"). "bestguess" imputes missing genotypes as most likely values (0,1,2), "random" imputes missing genotypes by generating binomial(2,p) random variables (p is the MAF), and "fixed" imputes missing genotypes by assigning the mean genotype values (2p).
is_check_genotype	a logical value indicating whether to check the validity of the genotype matrix Z (default= TRUE). If Z has non-SNP data, please set it FALSE, otherwise you will get an error message. If it is FALSE and you use weighted kernels, the weights should be given through the "weights" parameter.
is_dosage	a logical value indicating whether the matrix Z is a dosage matrix. If it is TRUE, the function will ignore "is_check_genotype".
missing_cutoff	a cutoff of the missing rates of SNPs (default=0.15). Any SNPs with missing rates higher than the cutoff will be excluded from the analysis.
estimate_MAF	a numeric value indicating how to estimate MAFs for the weight calculation and the missing genotype imputation. If estimate_MAF=1 (default), it uses all samples to estimate MAFs. If estimate_MAF=2, only samples with non-missing phenotypes and covariates are used to estimate MAFs
max_maf	a cutoff of the maximum minor allele frequencies (MAF) (default=1, no cutoff). Any SNPs with MAF > cutoff will be excluded from the analysis
verbose	a binary indicator to display messages (default=TRUE, displays messages)

Details

This is a copula based test of association for minimum p-value by combining the MultiSKAT tests with phenotype kernel corresponding to Sigma_Ps and SKAT and burden as genotype kernels. It is not advisable to have more than 4 kernels in the Sigma_Ps list.

Value

`p.value`	p-value of the minPcom test
`obj.SKAT`	list of output objects from MultiSKAT test with the kernels in Sigma_Ps and SKAT as genotype kernel (r.corr = 0).
`obj.Burden`	list of output objects from MultiSKAT test with the kernels in Sigma_Ps and burden as genotype kernel (r.corr = 1).

Author(s)

Diptavo Dutta

Examples

```
data(MultiSKAT.example.data)
attach(MultiSKAT.example.data)

obj.null <- MultiSKAT_NULL(Phenotypes,Cov)
out1 <- MultiSKAT(obj.null,Genotypes,cov(Phenotypes))
out2 <- MultiSKAT(obj.null,Genotypes,diag(5))
out3 <- MultiSKAT(obj.null,Genotypes,matrix(1,ncol = 5,nrow = 5))

### Getting minPcom p-value combining PhC, Het and Hom kernels
Sigma_Ps = list(cov(Phenotypes),diag(5),matrix(1,ncol = 5,nrow = 5))
obj.com = minPcom(obj.null,Sigma_Ps,Genotypes)
str(obj.com)
```

`MultiSKAT` *Sequence kernel association test for Multiple Phenotypes*

Description

Test for association between a set of SNPs/genes and a battery of continuous correlated phenotypes using kernel regression framework.

Usage

```
MultiSKAT(obj.res, Z, Sigma_p, kernel = "linear.weighted", Is.Common=FALSE,
weights.beta=c(1,25), weights = NULL, impute.method = "fixed", r.corr=0,
is_check_genotype=TRUE,is_dosage = FALSE, missing_cutoff=0.15,
estimate_MAF=1,max_maf=1,verbose =TRUE)
```

Arguments

obj.res	a NULL object output from MultiSKAT_NULL
Z	a genotype matrix corresponding to the region to be tested
Sigma_p	a phenotype kernel. Denotes the kernel for the effect sizes of a particular variant in the gene on the set of phenotypes.
kernel	a type of kernel (default= "linear.weighted"). See detail section.
Is.Common	a binary variable indiciating whether a variant has the same effect on all the phenotypes (default=FALSE).
weights.beta	a numeric vector of parameters for the beta weights for the weighted kernels. If you want to use your own weights, please use the "weights" parameter. It will be ignored if "weights" parameter is not null.
weights	a numeric vector of weights for the weighted kernels.
impute.method	a method to impute missing genotypes (default= "fixed"). "bestguess" imputes missing genotypes as most likely values (0,1,2), "random" imputes missing genotypes by generating binomial(2,p) random variables (p is the MAF), and "fixed" imputes missing genotypes by assigning the mean genotype values (2p).
r.corr	the ρ parameter for the compound symmetric correlation structure kernels (default=0).
is_check_genotype	a logical value indicating whether to check the validity of the genotype matrix Z (default= TRUE). If Z has non-SNP data, please set it FALSE, otherwise you will get an error message. If it is FALSE and you use weighted kernels, the weights should be given through the "weights" parameter.
is_dosage	a logical value indicating whether the matrix Z is a dosage matrix. If it is TRUE, the function will ignore "is_check_genotype".
missing_cutoff	a cutoff of the missing rates of SNPs (default=0.15). Any SNPs with missing rates higher than the cutoff will be excluded from the analysis.
estimate_MAF	a numeric value indicating how to estimate MAFs for the weight calculation and the missing genotype imputation. If estimate_MAF=1 (default), it uses all samples to estimate MAFs. If estimate_MAF=2, only samples with non-missing phenotypes and covariates are used to estimate MAFs
max_maf	a cutoff of the maximum minor allele frequencies (MAF) (default=1, no cutoff). Any SNPs with MAF > cutoff will be excluded from the analysis
verbose	a binary indicator to display messages (default=TRUE, displays messages)

Value

The function returns a list containing

Q	Value of the test statistic
p.value	P-value of MultiSKAT test

W	Phenotype adjusted regional information. The eigen values of this matrix are used as the mixing parameters for the null distribution
R	The phenotype kernel that was used for the test
Lambda	If MultiSKAT_Fast function is used, instead of phenotype adjusted regional information it returns the eigen values of the same, which are used for p-value calculation.

Examples

```
data(MultiSKAT.example.data)
attach(MultiSKAT.example.data)

obj.null <- MultiSKAT_NULL(Phenotypes,Cov)

out <- MultiSKAT(obj.null,Genotypes,cov(Phenotypes))
out$p.value
```

MultiSKAT.example.data

Example data for MultiSKAT

Description

This is a dataset of 5 phenotypes for 5000 individuals genotyped across 56 variants. The correlation between any two variant is 0.7. For this dataset, variants at 16, 39 and 51 are set to be causal. Phenotypes 1, 3 and 4 are associated with the causal SNPs. The effect sizes of individual SNPs are proportional to logarithm of their minor allele frequencies.

Usage

```
data("MultiSKAT.example.data")
```

Format

MultiSKAT.example.data contains the following objects:

- Phenotypes : A matrix of 5 phenotypes for 5000 individuals with individuals as rows and phenotypes as columns.
- Genotypes : A matrix of Genotype values for 5000 individuals across 56 variants with individuals as rows and variants as columns.
- Cov : A vector of intercept covariates for 5000 individuals.

Examples

```
data(MultiSKAT.example.data)
attach(MultiSKAT.example.data)
Phenotypes[1:4,1:5]
```

`MultiSKAT.Kinship.example.data` *Example data for MultiSKAT with related samples*

Description

This is a dataset of 5 phenotypes for 500 individuals genotyped across 20 variants. For this dataset, variants at 2, 5, 11 and 20 are set to be causal. Phenotypes 1, 2 and 3 are associated with the causal SNPs. The effect sizes of individual SNPs are proportional to logarithm of their minor allele frequencies.

Usage

```
data("MultiSKAT.Kinship.example.data")
```

Format

MultiSKAT.example.data contains the following objects:

- Phenotypes : A matrix of 5 phenotypes for 500 individuals with individuals as rows and phenotypes as columns.
- Genotypes : A matrix of Genotype values for 500 individuals across 20 variants with individuals as rows and variants as columns
- Cov : A vector of intercept covariates for 500 individuals.
- Kinship: A Kinship matrix for the 500 individuals
- V_g: A co-heritability matrix for 5 phenotypes
- V_e: A residual variance matrix for the 5 phenotypes
- U: Spectral vector matrix of the Kinship matrix
- D: Spectral values matrix of the Kinship matrix

Examples

```
data(MultiSKAT.Kinship.example.data)
attach(MultiSKAT.Kinship.example.data)
Phenotypes[1:4,1:5]
```

`MultiSKAT_NULL` *Construct and get parameters and residuals from the null model.*

Description

Compute the null model parameters and residuals for MultiSKAT. The values and residuals from here are used to compute the MultiSKAT test.

Usage

```
MultiSKAT_NULL(y.mat, X,is.fast=TRUE,method = "nlminb")
```

Arguments

`y.mat`	A numeric matrix of phenotype values with rows as individuals and columns as phenotypes.
`X`	A numeric matrix of covariate values with rows as individuals and columns as covariates. The first column should be the intercept (a vector of 1's).
`is.fast`	A binary variable indicating whether a fast approximation should be used or not (default = TRUE).
`method`	Indicates the optimizer to use for the generalized least squares procedure (default = "nlminb". Other option is "optim"). Works only with is.fast = FALSE

Details

For most of the scenarios with high sample size (more than 2000), is.fast= TRUE should be used. For lower sample size, is.fast=FALSE should be set. There might be cases where the default optimizer "nlminb" fails to converge due to numerical issue. Users can switch to another optimzer "optim" in that case by setting method="optim".

Value

This function returns an null object of the class SKAT_Multiphen_Null_PhenCorr. This object contains details, values and residuals for computing MultiSKAT test statistic and the null distribution.

Author(s)

Diptavo Dutta

Examples

```
data(MultiSKAT.example.data)
attach(MultiSKAT.example.data)

obj.null <- MultiSKAT_NULL(Phenotypes,Cov)
```

`MultiSKAT_NULL_Kins` *Construct the null MultiSKAT model with kinship adjustment*

Description

Compute the null model parameters and residuals for MultiSKAT_Kins. The values and residuals from here are used to compute the MultiSKAT_Kins test (MultiSKAT with kinship adjustment).

Usage

```
MultiSKAT_NULL_Kins(y.mat, X, U, D, V_g,V_e)
```

Arguments

`y.mat`	A numeric matrix of phenotype values with rows as individuals and columns as phenotypes.
`X`	A numeric matrix of covariate values with rows as individuals and columns as covariates. The first column should be the intercept (a vector of 1's)
`U`	Spectral vector matrix of the Kinship matrix.
`D`	Spectral values matrix of the Kinship matrix
`V_g`	Co-heritability matrix
`V_e`	Residual variance matrix

Details

Estimation of the coheritability and residual variance matrices should be as accurate as possible, otherwise it can hamper the false positive rates of the corresponding MultiSKAT tests.

Value

This function returns an null object of the class MultiSKAT_NULL_related. This object contains details, values and residuals for computing MultiSKAT test statistic and the null distribution.

Author(s)

Diptavo Dutta

Examples

```
data(MultiSKAT.Kinship.example.data)
attach(MultiSKAT.Kinship.example.data)

obj.null = MultiSKAT_NULL_Kins(Phenotypes,Cov,U,D,V_g,V_e)
```

Open_SSD *Open SNP set data file (SSD)*

Description

Open a SNP Set data file (SSD). After finishing using the SSD file, you must close the file by callinsg Close_SSD function.

Usage

```
Open_SSD(File.SSD, File.Info)
```

Arguments

File.SSD	name of the SSD file .
File.Info	name of the SSD info file.

Value

a list object of SSD.INFO.

Author(s)

Seunggeun Lee

Read_Plink_FAM *Read Plink FAM and covariates files*

Description

Read Plink FAM and covariates files.

Usage

```
Read_Plink_FAM(Filename, Is.binary = TRUE, flag1 = 0)
```

Arguments

Filename	input file name of plink FAM file
Is.binary	if TRUE, the phenotype is binary. If phenotype is continuous, it should be FALSE
flag1	0 represents the default coding of unaffected/affected (1/2) (default=0), and 1 represents 0/1 coding. flag1=1 is the same as -1 flag in plink. Please see the plink manual.

Value

A dataframe of Family ID (FID), Individual ID (IID), Paternal ID (PID), Maternal ID(MID), Sex, and Phenotype.

Author(s)

Seunggeun Lee

`Read_SNP_WeightFile`

Read a file with custom weights

Description

Read a file with custom weights

Usage

```
Read_SNP_WeightFile(FileName)
```

Arguments

`FileName`	input file name of a custom weight.

Details

The file should be a white-space (space or tab) delimitered file with 2 columns: SNP_ID and weight value. Please keep in mind that there should be no header!!

Value

Output object has a hash table of SNP IDs and weights.

Author(s)

Seunggeun Lee

`resampling.pvalues` *Get null resampling p-values*

Description

Function to calculate the null p-values for a list of MultiSKAT objects

Usage

```
resampling.pvalues(obj.res, obj.list, Z1, X,
resample = 200, Method = "Normal.Appx")
```

Arguments

`obj.res`	a NULL object output from MultiSKAT_NULL
`obj.list`	a list containing output from MultiSKAT or MultiSKAT_Fast function
`Z1`	A kernelized genotype matrix
`X`	a matrix of covariates
`resample`	Number of resampling iterations to obtain the null p-values(default = 200)
`Method`	A string denoting the method to use for the resampling p-values. Currently the two methods that are available are "Normal.Appx" and "Permutation".

Details

The output from this function is used in minP and minPcom tests. "Permutation" method should be used for unrelated individuals only.

Value

This function returns a list containing

`p.values`	A numeric matrix containing the resampling p-values
`Sigma_Ps`	A list containing the Σp values corresponding to the objects in *obj.list*

Author(s)

Diptavo Dutta

WSS *Weighted Sum of Squares*

Description

Computes the Weighted Sum of squares omnibus test for a list of individual MultiSKAT test

Usage

```
WSS(obj.res, obj.list, Z, weights.beta=c(1,25),
kernel = "linear.weighted",Is.Common=FALSE,
weights = NULL,r.corr = 0, impute.method = "fixed",
is_check_genotype=TRUE, is_dosage = FALSE,
missing_cutoff=0.15, estimate_MAF=1,max_maf=1,verbose = TRUE)
```

Arguments

obj.res	a NULL object from MultiSKAT_NULL or MultiSKAT_NULL_Kins
obj.list	A list containing output from MultiSKAT or MultiSKAT_Fast function
Z	A kernelized genotype matrix
kernel	a type of kernel (default= "linear.weighted"). See detail section.
Is.Common	a binary variable indiciating whether a variant has the same effect on all the phenotypes (default=FALSE).
weights.beta	a numeric vector of parameters for the beta weights for the weighted kernels. If you want to use your own weights, please use the "weights" parameter. It will be ignored if "weights" parameter is not null.
r.corr	the ρ parameter for the compound symmetric correlation structure kernels (default=0).
weights	a numeric vector of weights for the weighted kernels.
impute.method	a method to impute missing genotypes (default= "fixed"). "bestguess" imputes missing genotypes as most likely values (0,1,2), "random" imputes missing genotypes by generating binomial(2,p) random variables (p is the MAF), and "fixed" imputes missing genotypes by assigning the mean genotype values (2p).
is_check_genotype	a logical value indicating whether to check the validity of the genotype matrix Z (default= TRUE). If Z has non-SNP data, please set it FALSE, otherwise you will get an error message. If it is FALSE and you use weighted kernels, the weights should be given through the "weights" parameter.
is_dosage	a logical value indicating whether the matrix Z is a dosage matrix. If it is TRUE, the function will ignore "is_check_genotype".
missing_cutoff	a cutoff of the missing rates of SNPs (default=0.15). Any SNPs with missing rates higher than the cutoff will be excluded from the analysis.

estimate_MAF	a numeric value indicating how to estimate MAFs for the weight calculation and the missing genotype imputation. If estimate_MAF=1 (default), it uses all samples to estimate MAFs. If estimate_MAF=2, only samples with non-missing phenotypes and covariates are used to estimate MAFs
max_maf	a cutoff of the maximum minor allele frequencies (MAF) (default=1, no cutoff). Any SNPs with MAF > cutoff will be excluded from the analysis
verbose	a binary indicator to display messages (default=TRUE, displays messages)

Details

Weighted sum of squares is implemented with weights inversely proportional to the variance.

Value

The function returns a list containing

Q	Value of the test statistic
p.value	P-value for the weighted sum of squares test
Lambda	Mixing parameter for the null distribution of the WSS test statistic

Author(s)

Diptavo Dutta

References

Ionita-Laza, I.*, Lee, S.*, Makarov, V., Buxbaum, J. Lin, X. (2013). Sequence kernel association tests for the combined effect of rare and common variants. *American Journal of Human Genetics*, 92, 841-853. * contributed equally.

Examples

```
data(MultiSKAT.example.data)
attach(MultiSKAT.example.data)

obj.null <- MultiSKAT_NULL(Phenotypes,Cov)
out1 <- MultiSKAT(obj.null,Genotypes,cov(Phenotypes))
out2 <- MultiSKAT(obj.null,Genotypes,diag(5))
out3 <- MultiSKAT(obj.null,Genotypes,matrix(1,ncol = 5,nrow = 5))

### Combining PhC, Het and Hom
obj.list = list(out1,out2,out3)
obj.wss = WSS(obj.null,obj.list,Genotypes)
str(obj.wss)

### Combining PhC and Het
obj.list = list(out2,out1)
obj.wss = WSS(obj.null,obj.list,Genotypes)
str(obj.wss)
```

Index