Package 'WISH'

December 11, 2017

Type Package

Title WISH

Version 1.0

Imports doParallel, foreach,fastcluster, Rcpp, RcppEigen, data.table, corrplot, heatmap3, WGCNA, flashClust, bigmemory, parallel, dynamicTreeCut,ggplot2

Date 06-11-2017

Author Victor A.O. Carmelo, Lisette Kogelman,Maibritt Busk Hansen, Haja Kadarmideen

Maintainer Victor A. O. Carmelo <vaocar@bioinformatics.dtu.dk>

Description

The Weighted Interaction SNP Hub network method uses high-throughput genotype data to detect genome-wide interactions between SNPs and its relation with complex traits. Data dimensionality reduction is achieved by selecting SNPs based on its degree of genome-wide significance and degree of genetic variation in a population. Network construction is based on the epistatic interaction effect between SNP pairs. To identify modules the Topological Overlap Measure is calculated, reflecting the degree of overlap in shared neighbours between SNP pairs. Modules, clusters of highly interconnected SNPs, are defined using a tree-cutting algorithm on the SNP dendrogram created from the dissimilarity TOM.

License GPL-3

RoxygenNote 6.0.1

R topics documented:

`WISH-package` *WISH*

Description

The Weighted Interaction SNP Hub network method uses high-throughput genotype data to detect genome-wide interactions between SNPs and its relation with complex traits. Data dimensionality reduction is achieved by selecting SNPs based on its degree of genome-wide significance and degree of genetic variation in a population. Network construction is based on the epistatic interaction effect between SNP pairs. To identify modules the Topological Overlap Measure is calculated, reflecting the degree of overlap in shared neighbours between SNP pairs. Modules, clusters of highly interconnected SNPs, are defined using a tree-cutting algorithm on the SNP dendrogram created from the dissimilarity TOM.

Author(s)

Victor A.O Caremlo, Lisette Kogelamn and Haja Kadarmideen Maintainer: hajak@sund.ku.dk

References

Lisette J.A. Kogelman and Haja N.Kadarmideen (2014). Weighted Interaction SNP Hub (WISH) network method for building genetic Networks for complex diseases and traits using whole genome genotype data. BMC Systems Biology 8(Suppl 2):S5. http://www.biomedcentral.com/1752-0509/8/S2/S5.

`epistatic.correlation`

Calculate the epistatic interaction effect between SNP pairs using a genotype data frame created from genarate.genotype()

Description

A WISH network can be built based on epistatic interaction effects between SNP pairs. Those interaction effects are calculated using linear models.

Usage

```
epistatic.correlation(phenotype,genotype,threads=1,test=T,simple=T,glm=F)
```

Arguments

`phenotype`	Dataframe with the rows correspinding to the individuals in the analysis,and columns for the different measured phenotypes and fixed/random factors. Only give one phenotype column at a time. Phenotypes should be non-categorical continous or discrete/semi-discrete variables. Make sure that the dataframe contains the same individuals as in the genotype-file, and that those are in the same order.
`genotype`	Dataframe with the genotype information, resulting from the function generate.genotype(). Make sure that the dataframe contains the same individuals as in the phenotype-file, and that those are in the same order.

threads	Number of threads to use for parallel execution in the function registerDoParallel()
test	True or False value indicating if a test run is being perform. If True will calculate the expected time it will take for the full analysis based on calculating 100.000 models with the setting chosen
simple	True or false value indicating if only a major/major and minor/minor directed interaction model are tested (simple=T) or if if interactions on the major/minor minor axis are tested as well, with the best one of the two being selected (simple=F).
glm	If T will use a generelized linear model with a binomial link function instead of a regular linear model. This should be used if your phenotype is binary.

Value

A list of two matrices. The first matrix gives the epistatic interaction effects between all the SNP-pairs which were in the input genotype data) and selected with the pvalue from the GWAS results. The second matrix are the corresponding pvalues of the parameter estimates of the epistatic interactions.

References

Lisette J.A. Kogelman and Haja N.Kadarmideen (2014). Weighted Interaction SNP Hub (WISH) network method for building genetic networks for complex diseases and traits using whole genome genotype data. BMC Systems Biology 8(Suppl 2):S5. http://www.biomedcentral.com/1752-0509/8/S2/S5.

Examples

```
epistatic.correlation(phenotype,genotype,threads,test,simple)
```

---

generate.genotype *Import genotype data in the correct format for network construction*

Description

For network construction based on both genomic correlations as well as epistatic interactions a genotype matrix has to be created, consisting of one numeric value per SNP, per individual. This function takes Plink output (1,2-coding) to create the genotype matrix which can be used to calculate genomic correlations or epistatic interaction effects

Usage

```
generate.genotype(ped,tped,snp.id=NULL, pvalue=0.05,id.select=NULL,
gwas.p=NULL,major.freq=0.95,fast.read=T)
```

Arguments

ped	Input ped file as .ped file or data.frame. The ped file (.ped) is an input file from Plink: The PED file is a white-space (space or tab) delimited file: the first six columns are mandatory: Family ID, Idividual ID, Paternal ID, Maternal ID, Sex (1=male; 2=female;other=unknown) and Phenotype. The IDs are alphanumeric: the combination of family and individual ID should uniquely identify a person. A PED file must have 1 and only 1 phenotype in the sixth column. The phenotype can be either a quantitative trait or an affection status column: PLINK will automatically detect which type (i.e. based on whether a value other than 0, 1, 2 or the missing genotype code is observed). SNPs are 1,2-coded (1 for major allele,2 for minor allele) For more information: http://pngu.mgh.harvard.edu/~purcell/plink/data.shtml#
tped	Input tped file as .tped file or data frame. The tped file (.tped) is a transposed ped file, from Plink. This file contains the SNP and genotype information where one row is a SNP. The first 4 columns of a TPED file are the same as a 4-column MAP file. Then all genotypes are listed for all individuals for each particular SNP on each line. Again, SNPs are 1,2-coded.
snp.id	Input SNP ids to use in analysis if not all snps are to be used
pvalue	A value for the cutoff of the SNPs which should be remained in the matrix, based on the pvalue resulting from the GWAS. Default value is 0.05
id.select	If requested, a subset of individuals can be selected (e.g. extremes). If nothing inserted, all individuals are in the output
gwas.p	A vector of the p-values corresponding to the input SNPs in the ped/tped file or gwas.id vector. If assigned, will select snps based on the pvalue parameter with a default value of 0.05.
major.freq	Maximum major allele frequency allowed in each variant. Default value is 0.95.
fast.read	If true will use fread from the data.table package to read the files. This is much faster than read.table, but requires consistent delimeters in the ped and tped file, and a maximum of approximately 950.000 colums in the ped file. This can be increased by changing the stack size (do this only if you know what you are doing)

Value

A genotype dataframe and the corresponding vector of passing snps in a vector. The genotype data frame has a row for each individual and a column for each SNP. SNPs are 1,1.5,2 coded: 1 for homozygous for the major allele, 1.5 for heterozygous, and 2 for homozygous for the minor allele. Missing values are NA coded.

References

Lisette J.A. Kogelman and Haja N.Kadarmideen (2014). Weighted Interaction SNP Hub (WISH) network method for building genetic networks for complex diseases and traits using whole genome genotype data. BMC Systems Biology 8(Suppl 2):S5. http://www.biomedcentral.com/1752-0509/8/S2/S5.

Examples

```
generate.genotype(ped,tped)
```

`generate.modules` *Function for creation of genomic interaction modules based on WGCNA framework.*

Description

Generate.modules normalizes the epistatic interaction coefficients and performs hierarchical clustering,SNP selection and parameter selection for module construction, and generates modules. It uses the WGCNA workflow modified for epistatic correlations and outputs the module components and analysis parameters.

Usage

```
generate.modules(correlations,values="Coefficients",power=c(seq(1,10,0.1),c(12:2
n.snps=dim(correlations$Coefficients)[1],minClusterSize=50,type="unsigned",threa
```

Arguments

`correlations`	List of epistatic correlations and p-values generated by epistatic.correlation()
`values`	Character which can be "Pvalues" or "Coefficients" Indicating if P-values or Coefficients should be used for analysis. The recomended and defualt values is Coefficients
`power`	Powers to test for creating scale free network. Only change if the default values don't work
`n.snps`	Number of SNPs to select. SNPs are selected by connectivity, so 500 will select the top 500 most connected Snps. Default is to use all
`minClusterSize`	Minimum module (cluster) size. Default, is 50, but changing this may be recommended in case of sparse SNPs
`type`	Type of network to generate. Default is "unsigned", can be "signed" or "signed hybrid"
`threads`	Number of threads to use if parallelization is possible.

Value

Plots the network connectivity and the scale and SNP tree clustering with modules found. Returns a named list with all the data generated:

SNPs SNPs used in the analysis and their correlations

connectivity The connectivity matrix of the SNPs

adjMat The adjacency matrix of the SNPs

dissTom The dissimilarity TOM

genetree The clustering object used for the genetree

modules The module numbers for each SNP, in order of the SNP matrix

modulcolors The colors used in the modules for each SNP

power.estimate The power estimate to generate a scale free network

Examples

```
generate.modules(correlations)
```

`genome.interaction` *Visualization of pairwise chromosome epistatic interactions on a genome wide level*

Description

Visualization of the genome wide chromosome pairwise relative strength of epistatic interaction, ranging from 1 (strongest) to -1 (weakest). The strength is based on the 90th percentile quantile (default) of stastistical significance of epistatic interaction between all interactions in each chromosome pair, scaled to 1 to -1.

Usage

```
genome.interaction(tped,correlations,quantile=0.9)
```

Arguments

`tped`	The tped file used in generate.genotype(). The SNPs must be sorted by chromosome, matching the order of the SNPs in the correlation matrices.
`correlations`	List of epistatic correlations and p-values genrated by epistatic.correlation()
`quantile`	Number from 0 to 1 indicating which quantile to base the visualization on.

Value

Outputs a plot visualizing the chromosome interaction map

Examples

```
genome.interaction(tped,correlations)
```

`LD_blocks` *Function for creating blocks of input genotypes based on LD and selecting tagging variants in each block. Genotypes should be sorted by genomic coordinates and chromosome.*

Description

Given an input set of genotypes from the generate.genotype() function, this function will generate LD blocks based on average LD between all members in each block, using r2 metric of LD. Scanning linearly over the genotype file from the first row, genotypes will be contiously added to blocks as long as the average of all pairwise r2 values of the genotypes in the block are above threshold. When the values goes below the threshold, the last genotype tested will be the base of a new block. For each block the median genotype in block will be selected as a tagging genotype.

Usage

```
LD_blocks(genotype,threshold=0.9,max_block_size=1000)
```

Arguments

genotype	A genotype matrix from generate.genotype()
threshold	The threshold used for generating blocks, indicating the minimum average pairwise r2 value allowed.
max_block_size	The maximum block size allowed.

Value

Returns a named list with the following objects:

genotype The tagging genotypes selected from the blocks

tagging_genotype The genotype selected to represent each block. The median genotype, rounded down is selected

genotype_block_matrix A matrix indicating which block each genotype belongs to

Returns a named list with the following objects:

Examples

```
LD_blocks(genotype,threshold,max_blocksize)
```

---

pairwise.chr.map *Visualization of chromosomal pairwise region epistatic interaction strength, based on statistical significance*

Description

Visualization of chromosome pairwise region epistatic interaction strength, based on statistical significance. The value is based of the most signficant epistatic interaction in each region pair, ranging from 1 (strongest) to 0 (weakest). By defaulty chromosomes are separated into 1 Mb regions, but if SNPs are more spaced out that this it will adjust to the smallest region that fit the data.

Usage

```
pairwise.chr.map(chr1,chr2,tped,correlations,span=10^6)
```

Arguments

chr1	The name of the first chromosome in the comparison, matching the name from the tped file
chr2	The name of the second chromosome in the comparison, matching the name from the tped file
tped	The tped file used in generate.genotype(). The SNPs must be sorted by chromosome and position on the chromosome, matching the order of the SNPs in the correlation matrices.
correlations	List of epistatic correlations and p-values genrated by epistatic.correlation()
span	Region in bp. Default is 1 Mb (10^6)

Value

Outputs a plot visualizing the pairwise chromosome region interaction

Examples

```
pairwise.chr.map("1","2",tped,correlations)
```

`partial_correlations_triangular` ****Internal Use Function*** This function calculates the epistatic correlations in a subset of a matrix space based on coordiantes*

Description

Internal package function for calculating epsitatic correlations in sub-matrices

Usage

```
partial_correlations_triangular(genotype_1,genotype_rev_1,phenotype,coords,
glm=F,model=1)
```

Arguments

`genotype_1`	Dataframe with the genotype information, resulting from the function generate.genotype(). Make sure that the dataframe contains the same individuals as in the phenotype-file, and that those are in the same order.
`genotype_rev_1`	Same as genotpye but with reversed genotype coding
`phenotype`	Dataframe with the rows correspinding to the individuals in the analysis,and columns for the different measured phenotypes and fixed/random factors. Phenotypes should be continous variables.
`coords`	Matrix of row split coordinates for subseting input space
`glm`	setting controlling if is a lm or glm
`model`	Specification controlling if MM or Mm directed interaction model is used.

Value

Epsitatic correlations and P-values for the selected set or subset of the data

Examples

```
partial_correlations <- partial_correlaiton_triangular(genotype_1,genotype_rev_1,
phenotype,coords,model)
```

`pseudo_manhattan` *Function to plot summary pseudo-manhattan plots of variants.*

Description

Visualize summary statistics for interactions based on total sum of -loglikelihoods for each variant across all interactions og the sum of effect sizes.

Usage

```
pseudo_manhattan(tped,correlations,values="p")
```

Arguments

`tped`	Input tped file as .tped file or data frame. The tped file (.tped) is a transposed ped file, from Plink. This file contains the SNP and genotype information where one row is a SNP. The first 4 columns of a TPED file are the same as a 4-column MAP file. Then all genotypes are listed for all individuals for each particular SNP on each line. Again, SNPs are 1,2-coded.#'
`correlations`	List of epistatic correlations and p-values generated by epistatic.correlation()
`values`	Values can either be p or for p-values or c for coefficients depending on if you wan't to use the effects sizes for p-values for plotting

Value

Plots a pseudo manhattan plot

Examples

```
pseudo_manhattan(tped,correlations)
```

`triangular_split` ****Internal Use Function*** This function calculates the row coordinates for splitting triangular sub matrices of quadratic matrices into approximately equally sized partitions for use in in dividing correlation calculations into equal size for parallelization*

Description

Internal function for splitting triangular matrices into approximately equal parts

Usage

```
triangular_split(n,split)
```

Arguments

n	Row and Column length of the n by n matrix the triangular matrix originates from
split	Number of partitions to split the triangular matrix in

Value

A matrix of row coordinates used for splitting

Examples

```
triangular_split(1000,5)
```

Index