Package 'ARTP'

April 6, 2018

Title Gene and Pathway p-Values Computed using the Adaptive Rank Truncated Product

Version 2.0.5

Date 2018-04-05

Author Kai Yu, Qizhai Li and William Wheeler

Description
For calculating gene and pathway p-values using the Adaptive Rank Truncated Product test.

Maintainer Bill Wheeler <wheelerb@imsweb.com>

Depends

License GPL-2

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-04-06 12:39:56 UTC

R topics documented:

ARTP *Gene and pathway p-values using the Adaptive Rank Truncated Product test*

Description

An R package for computing gene and pathway p-values using the Adaptive Rank Truncated test. This package can be used to analyze pathways/genes based on a genetic association study, with either a continuous or a binary case-control outcome.

Details

It is increasingly recognized that pathway analyses-a joint test of association between the outcome and a group of single nucleotide polymorphisms (SNPs) within a biological pathway-could potentially complement single-SNP analysis and provide additional insights for the genetic architecture of complex diseases. Building upon existing P-value combining methods, we propose a class of highly flexible pathway analysis approaches based on an adaptive rank truncated product statistic that can effectively combine evidence of associations over different SNPs and genes within a pathway. The statistical significance of the pathway-level test statistics is evaluated using a highly efficient permutation algorithm that remains computationally feasible irrespective of the size of the pathway and complexity of the underlying test statistics for summarizing SNP- and gene-level associations. We demonstrate through simulation studies that a gene-based analysis that treats the underlying genes, as opposed to the underlying SNPs, as the basic units for hypothesis testing, is a very robust and powerful approach to pathway-based association testing.

The function `ARTP_pathway` is used to compute gene and pathway p-values provided that the observed and permutation p-values for each SNP already exist in files. The input files required for `ARTP_pathway` can be obtained by calling the function `runPermutations`.

Author(s)

Kai Yu <yuka@mail.nih.gov> and William Wheeler <wheelerb@imsweb.com>

References

Yu K, Li Q, Bergen AW, Pfeiffer RM, Rosenberg PS, Caporaso N, Kraft P, Chatterjee N Pathway analysis by adaptive combination of P-values Genet Epidemiol 33(8):700-9; 2009 Dec.

ARTP_pathway *Gene and pathway p-values using ARTP*

Description

Calculate gene and pathway p-values using the Adaptive Rank Truncated Product test

Usage

```
ARTP_pathway(obs.file, perm.file, nperm, temp.dir, gene.list=NULL, op=NULL)
```

Arguments

obs.file	The output file obs.outfile from runPermutations or a file with the SNP ids and p-values (see details).
perm.file	The output file perm.outfile from runPermutations or a files with the SNP ids and p-values (see details).
nperm	The number of permutations in the output file perm.outfile from runPermutations
temp.dir	A folder to keep temporary files that will be created.
gene.list	A list describing the gene-SNP data. See gene.list. If NULL, then it is assumed that all SNPs belong to the same gene. The default value is NULL.
op	List of options. See details.

Details

If the p-values are not computed using runPermutations, then the format for obs.file and perm.file should be as follows. Both files must be uncompressed, comma seperated files with the first row as the SNP ids in the same order. Row 2 of obs.file has the observed p-values, and starting from row 2 in perm.file are the permuted p-values.

A random seed should be set before calling ARTP_pathway in order to reproduce results. The randomness is due to the ranking of p-values, where ties are broken randomly.

Options list:

Below are the names for the options list op. All names have default values if they are not specified.

- inspect.snp.n The number of candidate truncation points to inspect the top SNPs in a gene. The default is 1.
- inspect.snp.percent A value x between 0 and 1 such that a truncation point will be defined at every x percent of the top SNPs. The default is 0 so that the truncation points will be 1:inspect.snp.n.
- inspect.gene.n The number of candidate truncation points to inspect the top genes in the pathway. The default is 10.
- inspect.gene.percent A value x between 0 and 1 such that a truncation point will be defined at every x percent of the top genes. The default is 0.05.

Assume the number of SNPs in a gene is 100. Below are examples of the truncation points for different values of inspect.snp.n and inspect.snp.percent.

inspect.snp.n	inspect.snp.percent	truncation points
1	0	1
1	0.05	5
1	0.25	25
1	1	100
2	0	1, 2

2	0.05	5, 10
2	0.25	25, 50
2	1	100
3	0.2	20, 40, 60

Value

The returned value is a list with names "pathway.pvalue" and "gene.table". pathway.pvalue is the ARTP p-value for the pathway. gene.table is a data frame containing the gene name, number of SNPs in the gene that were included in the analysis, and the ARTP p-value for the gene.

Author(s)

Kai Yu

See Also

runPermutations

Examples

```
 # Get the file of observed p-values
obs_file <- system.file("sampleData", "obs_pvalues.txt", package="ARTP")

# Get the file of permutation p-values
perm_file <- system.file("sampleData", "perm_pvalues.txt", package="ARTP")

# Define the gene-SNP list
gs_file <- system.file("sampleData", "gene_SNP_data.txt", package="ARTP")
gene.list <- list(file=gs_file, delimiter="\t", header=1,
                  snp.var="SNP", gene.var="Gene")

# Call the ARTP function
nperm <- 100  # The number of permutations in perm_file
temp.dir <- "C:/temp/"
set.seed(123)
# ARTP_pathway(obs_file, perm_file, nperm, temp.dir, gene.list=gene.list)

# Now asume that all SNPs belong to the same gene
# ARTP_pathway(obs_file, perm_file, nperm, temp.dir)
```

---

gene.list *List to describe the gene-SNP file*

Description

The list to describe the gene-SNP file for ARTP_pathway

Format

The format is a list:

file Text file containing at least 2 columns, where one column is for the SNPs and the other column is for the gene containing the SNP. No default.

delimiter The delimiter used in `file`.

gene.var Variable name or column number of the gene variable. The default is "Gene".

snp.var Variable name or column number of the SNP variable. The default is "SNP".

chrm.var Variable name or column number of the chromosome variable. The default is "Chr". This option is only used for calling `plot_genes`.

header 0 or 1 to denote if `file` contains a header of variable names. The default is 1.

Details

All the genes and SNPs listed in this file define a single pathway.

---

gene_SNP_data *Gene-SNP data*

Description

Gene-SNP data file for `ARTP_pathway`

Details

gene_SNP_data.txt is a tab delimited file that contains the gene that each SNP belongs to.

Examples

```
# Load and print the first 5 rows
data(gene_SNP_data, package="ARTP")

gene_SNP_data[1:5, ]
```

geno_data *Sample genotype data*

Description

Sample genotype data for `runPermutations`

Details

geno_data.rda is a type 1 data file (see `file.type` in `snp.list`). This data contains 50 SNPs and 500 subjects, and is tab delimited. The first row of the data contains the subject ids. Starting from row 2, are the SNP ids and the genotypes for each subject. The genotypes are coded as AA, AG, GG.

Examples

```
# Load and print a substring the first 5 lines
data(geno_data, package="ARTP")

substring(geno_data[1:5], 1, 50)
```

obs_pvalues *Observed p-values*

Description

Sample file of observed p-values for the example in `ARTP_pathway`

Details

This is a comma delimited file where the first row contains the SNP ids, second row contains the p-values, third row contains the method of p-value computation (see `single.marker.test`), fourth row contains the estimated main efect of the SNP, and fifth row contains the estimated SNP main effect standard error.

Examples

```
# Read in and print the data
f <- system.file("sampleData", "obs_pvalues.txt", package="ARTP")
x <- scan(f, what="character", sep="\n")
substring(x, 1, 50)
```

perm_pvalues *Permutation p-values*

Description

Sample file of permutation p-values for the example in ARTP_pathway

Details

This is a comma delimited file with 101 rows with row 1 containing the SNP ids, and rows 2-101 containing the p-values. Each row represents one permutation.

Examples

```
# Read in and print the data
  f <- system.file("sampleData", "perm_pvalues.txt", package="ARTP")
  x <- scan(f, what="character", sep="\n")
  substring(x[1:5], 1, 50)
```

pheno.list *List to describe the covariate and outcome data*

Description

The list to describe the covariate and outcome data for runPermutations

Format

The format is a list:

file Covariate data file. This file must have variable names, two of which being an id variable and a response variable (see `id.var` and `response.var`). No default.

id.var Name of the id variable. No default.

response.var Name of the response variable. For logistic regression analyses, this variable must be coded as 0 (control) and 1 (case). No default.

main.vars Character vector of variables names for variables in `file` that will be included in the model as main effects. The default is NULL.

delimiter The delimiter in `file`. The default is "".

in.miss Vector of character strings to define the missing values. This option corresponds to the option `na.strings` in `read.table`. The default is "NA".

Details

In this list, `file`, `id.var`, and `response.var` must be specified. The variable `id.var` is the link between the covariate data and the genotype data. For each subject id, there must be the same subject id in the genotype data for that subject to be included in tha analysis.
Missing data: If any of the variables defined in `main.vars`, `int.vars`, or `response.var` contain missing values, then those subjects will be removed from the covariate and outcome data. After the subjects with missing values are removed, the subject ids are matched with the genotype data.

`pheno_data` *Sample covariate and response data*

Description

Sample covariate and response data for `runPermutations`

Details

The file pheno_data.txt is a tab-delimited type 3 data set (see `file.type` in `pheno.list`). It contains the variables:

- `ID` The subject id
- `Y` Case-control status (0, 1)
- `X1` Continuous covariate
- `X2` Continuous covariate

Examples

```
# Load and print the first 5 rows
data(pheno_data, package="ARTP")

pheno_data[1:5, ]
```

`plot_genes` *Gene Plot*

Description

Plot the observed SNP p-values for each gene

Usage

```
plot_genes(obs.outfile, gene.list, op=NULL)
```

Arguments

obs.outfile	The output file of observed p-values from runPermutations
gene.list	See gene.list
op	List of options. See details.

Details

If the option gene.list$chrm.var is not specified, then it is assumed that all the SNPs are on the same chromosome, and the same color will be used in the plot. If gene.list$chrm.var is specified, then the genes will be grouped by chromosome with the same color for the group.

Options list:
Below are the names for the options list op.

- cex.axis See par
- colors Colors to use for each gene in the plot
- maxLabelLen Maximum length of x-axis labels
- chr.text See par
- x.las See par
- x.padj See par

Value

A data frame containing the SNP ids, parameter estimates, genes, etc.

See Also

runPermutations

Examples

```
# Get the file of observed p-values
obs_file <- system.file("sampleData", "obs_pvalues.txt", package="ARTP")

# Define the gene-SNP list
gs_file <- system.file("sampleData", "gene_SNP_data.txt", package="ARTP")
gene.list <- list(file=gs_file, delimiter="\t", header=1,
                  snp.var="SNP", gene.var="Gene")

plot_genes(obs_file, gene.list)
```

`runPermutations` *Calculate observed and permutation p-values for SNPs*

Description

Calculate observed and permutation p-values for SNPs

Usage

```
runPermutations(snp.list, pheno.list, family, op=NULL)
```

Arguments

`snp.list`	A list describing the SNP data. See `snp.list`
`pheno.list`	A list describing the covariate and response data. See `pheno.list`
`family`	1 or 2, 1 = logistic regression, 2 = linear regression.
`op`	List of options. See `details`.

Details

This function first reads the data stored in the files defined by `snp.list` and `pheno.list`. The subject ids in `snp.list$file` and `pheno.list$file` are matched and any subject not in both files will be removed from the analysis. Also, any subject with a missing value for the response or covariate will be removed. The function `single.marker.test` is called for each observed SNP and for each permutation. Depending on the response variable and genotype frequency counts, `single.marker.test` will call `glm.fit`, `fisher.test` or `lm`.
Running a large number of permutations on a single machine could take a long time. However, if the user has access to multiple machines, then the permutations can be broken up across the different machines for faster computation. A different random seed should be set for each machine, and the output permutation files would need to be combined into a single file before calling `ARTP_pathway`.

Options list:
Below are the names for the options list `op`. All names have default values if they are not specified.

- `nperm` Number of permutations. The default is 100.
- `obs.outfile` Output file for the observed results. The default is "obs.txt".
- `perm.outfile` Output file for the permuted results. The default is "perm.txt".
- `perm.method` 1 or 2 for the type of permutation. 1 is to permute the SNPs. 2 is to generate a new response using the base model. For a continuous response, the residuals from the base model are permuted and then added to the linear predictors from the base model to give the new response vector. For a binary response, the new response vector is rbinom(n, 1, vals), where vals are the fitted values from the base model. The default is 2.
- `min.count` See `single.marker.test`. The default is 5.
- `miss.rate` Maximum missing rate to include SNPs. Any SNP with missing rate greater than `miss.rate` will be excluded. The default is 0.20.

obs.outfile will be a comma delimited file containing 5 rows:
Row 1 contains the SNP ids.
Row 2 contains the SNP p-values.
Row 3 contains a value for how the p-value was computed (see the details of single.marker.test).
Row 4 contains the estimate of the SNP main effect.
Row 5 contains the estimated standard error of the SNP main effect.
perm.outfile will be a comma delimited file, where each row are the permutation p-values for all SNPs.

Value

The returned value is NULL, however 2 output files are created as defined by op$obs.outfile and op$perm.outfile.

Author(s)

Kai Yu and William Wheeler

See Also

single.marker.test snp.list pheno.list

Examples

```
 # Define snp.list
geno_file <- system.file("sampleData", "geno_data.txt", package="ARTP")
snp.list <- list(file=geno_file, file.type=2, delimiter="\t")

# Define pheno.list
pheno_file <- system.file("sampleData", "pheno_data.txt", package="ARTP")
pheno.list <- list(file=pheno_file, delimiter="\t", id.var="ID",
                   response.var="Y", main.vars=c("X1", "X2"))

# Options list. Change obs.outfile and perm.outfile if needed.
op <- list(nperm=10, obs.outfile="./obs.txt", perm.outfile="./perm.txt",
           perm.method=2)

# Not run
# runPermutations(snp.list, pheno.list, 1, op=op)
```

---

single.marker.test *Single SNP test*

Description

Perform an association test for 1 SNP

Usage

```
single.marker.test(y, covariates, weights, offset, control, snpcol,
                   min.count=5, y.continuous=FALSE)
```

Arguments

y	The response vector
covariates	A design matrix where the SNP is the last column. The SNP must be coded as 0-1-2.
weights	Vector of weights.
offset	Vector for the offset.
control	List for glm.control
snpcol	Number of columns of the design matrix covariates
min.count	The minimum number of subjects to have in at least 2 of the genotype categories (0-1-2), if y is continuous. If y is binary, then the minimum expected frequency count for cases or controls to use logistic regression; otherwise, Fisher's exact test will be used. The default is 5
y.continuous	TRUE or FALSE for whether or not y is continuous. If FALSE, then y must be coded as 0-1. The default is FALSE.

Details

The input vectors and matrices must not contain missing values. To compute the p-value, either glm.fit, fisher.test or lm is called. The p-value flag is a value for how the p-value was computed:

Value	Genetic Model	Test
0	trend	Wald test from logistic/linear regression
-1	dominant	Fisher's exact test
-2	recessive	Fisher's exact test
1	dominant	Wald test from logistic regression
2	recessive	Wald test from logistic regression

Value

The returned object is a vector of length 4 containing the p-value, p-value flag (see details), SNP main effect estimate, and standard error of the SNP main effect estimate. If Fisher's exact test was used, then the main effect and standard error will be set to NA.

Author(s)

Kai Yu and Qizhai Li

See Also

runPermutations

Examples

```
# Generate data
set.seed(123)
n <- 1000
y <- rbinom(n, 1, 0.5)
snp <- rbinom(n, 2, 0.4)
weights <- rep.int(1, times=n)
offset  <- rep.int(0, times=n)
control  <- glm.control()

# Create a design matrix
x <- matrix(data=NA, nrow=n, ncol=3)
x[, 1] <- 1  # Intercept column
x[, 2] <- runif(n) # Continuous covariate
x[, 3] <- snp

single.marker.test(y, x, weights, offset, control, 3)
```

---

snp.list *List to describe the genotype data*

Description

The list to describe the genotype data for runPermutations

Format

The format is a list:

file File to use. No default.

file.type 2 or 3 (see details).

delimiter The delimiter used in file.

in.miss Vector of values to denote the missing values in file. The default is " " (2 blank spaces).

heter.codes Vector of codes used for the heterozygous genotype. If NULL, then it is assumed that the heterozygous genotype is of the form "AB", "Aa", "CT", ... etc, ie a 2-character string with different characters (case sensitive). The default is NULL.

id.var (Only for file.type = 3) The subject id variable. The default is 1.

Details

In this list, file must be specified. If the SNPs are coded in the standard (0,1,2) coding, then set heter.codes to 1 (the heterozygous genotype).

Type 2 has data in the form:

	subject1	subject2	subject3
snp1	0	2	1
snp2	1	1	0

The first row must contain the subject ids. Starting from row 2, the first delimited field must contain the SNP id. The remaining delimited fields contain the genotypes. Rows are SNPs, columns are the subjects.

`Type 3` has data of the form:

id	snp1	snp2
subject1	0	1
subject2	2	1
subject3	1	0

Examples

```
# Suppose the genotype data is a tab-delimited, type 2 file: c:/temp/data/geno1.txt.
#  Also assume the data has the trend coding 0, 1, 2 with NA as missing values.
# The below list is for processing the file.
snp.list <- list(file="C:/temp/data/geno1.txt", delimiter="\t", file.type=2,
                 heter.codes=1, in.miss=NA)
```

Index