Package ‘GSAgm’

February 19, 2015

Type Package

Title Gene Set Analysis using the Gamma Method

Version 1.0

Date 2013-02-25

Author Rama Raghavan, Alice Wang

Maintainer Rama Raghavan <rraghavan@kumc.edu>

Description GSAgm is an R package that completes a self-contained gene set analysis (GSA) for RNA-seq and SNP data using the Gamma Method.

License GPL-2

Depends survival, edgeR

NeedsCompilation no

Repository CRAN

Date/Publication 2014-04-21 16:51:25

R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSAgm-package</td>
<td>2</td>
</tr>
<tr>
<td>gene_example</td>
<td>2</td>
</tr>
<tr>
<td>PCgamma</td>
<td>3</td>
</tr>
<tr>
<td>RNAgamma</td>
<td>4</td>
</tr>
<tr>
<td>rnaseq_counts</td>
<td>6</td>
</tr>
<tr>
<td>STTtoShapeParameter</td>
<td>7</td>
</tr>
<tr>
<td>testdata</td>
<td>7</td>
</tr>
<tr>
<td>Index</td>
<td>9</td>
</tr>
</tbody>
</table>
Description

This package runs gene set analyses on SNP and RNA data. See individual functions for details.

Details

<table>
<thead>
<tr>
<th>Package:</th>
<th>GSAgm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Package</td>
</tr>
<tr>
<td>Version:</td>
<td>1.0</td>
</tr>
<tr>
<td>Date:</td>
<td>2013-12-04</td>
</tr>
<tr>
<td>License:</td>
<td>GPL-2</td>
</tr>
</tbody>
</table>

Author(s)

Rama Raghavan, Alice Wang
Maintainer: Rama Raghavan <rraghavan@kumc.edu>

References


Description

Dummy genes to test data with.
**Usage**

```
data(gene_example)
```

**Format**

The format is: `int [1:15, 1] 111 111 111 222 222 222 222 222 333 333 ... - attr(*, "dimnames")=List of 2 ..$ : NULL ..$ : chr "x"

**Examples**

```
data(gene_example)
```

---

**Description**

For GSA of SNP data, the following two-step procedure is implemented (see Biernacka et al[1] for more details on the method). Step 1: Principal components analysis for SNPs within a gene is completed with the components needed to explain 80 percent of the variation retained. Using these components, a gene-level association test is completed to determine the association of the gene with the phenotype. Step 2: The gene-level p values for genes within a given gene set are combined using the Gamma Method, a variation of Fisher’s Method, to determine the association of the gene set with the phenotype. The GSA function for SNP data allow quantitative, binary and time-to-event phenotypes (i.e., linear models, logistic models, Cox proportional hazard models).

**Usage**

```
PCgamma(formula, data, snpprefix = "snp", gene, PCpctVar = 80, gammaShape = 1, STT = NULL, phenoType = c("case.control", "quantitative", "survival"), perm = T, n.perm = 1000, seed = 12212012)
```

**Arguments**

- `formula`: formula for model, include phenotype and covars. SNPs will be added by function
- `data`: All data including matrix of genetic markers, each marker represented by the dosage of some allele, could also be CNV, treated as continuous and covariates
- `snpprefix`: prefix for SNP variable, defaults to "snp"
- `gene`: vector designating the gene each marker belongs to, must be in same order as SNPs
- `PCpctVar`: numeric indicating the percent of variation (in percent) in the genetic markers that is to be explained by PCs
- `gammaShape`: numeric indicating the gamma shape parameter to be used for p-value summarization
**STT**
numeric indicating soft truncation threshold to be used, will calculate gamma parameter (must be <= 0.4)

**pheno.type**
type of phenotype, case-control results in logistic regression, quantitative results in OLS, and survival results in cox model

**perm**
boolean indicating whether permutation p-value are to be used for the gamma summary method

**n.perm**
umeric indicating number of permutations to be used

**seed**
umeric to set RNG for reproducability

**Value**

This functions returns a list.

- **gamma.pvalue**  Gamma P value
- **perm.pvalue**  Gamma permutation p value, if specified. Else NA
- **gene.info**  Info for each gene

**Examples**

```r
# Case Control (logistic) example
data(testdata)
data(gene_example)
PCgamma(pheno~strata(study)+age,
data=testdata,gene=gene_example,pheno.type="case.control",
STT = 0.2, gammaShape = NULL,
perm=FALSE, n.perm = 10, seed = 12212012)

# Here is a survival example
set.seed(1234)
time_example  <- rnorm(150, m=50, sd=10)
event_example  <- rbinom(150, 1, 0.3)
testdata  <- cbind(testdata,time_example,event_example)

PCgamma(Surv(time_example,event_example)~strata(study)+age,
data=testdata,gene=gene_example,pheno.type="survival",
STT = 0.2, gammaShape = NULL,
perm=FALSE, n.perm = 10, seed = 12212012)
```

---

**RNA gamma**

```
RNA gamma
```
RNAgamma

Description

For GSA of RNA-seq data, the following procedure, similar to the analysis of SNP data, is imple-
mented (see Fridley et al[2] for more details on the method). Step 1: Association of gene expression
data from RNA-seq (count data) is assessed for differential expression between two groups using
edgeR[3]. Step 2: P-values from the association analysis within edgeR for genes within a given gene
set are combined using the Gamma Method to determine the association of the gene set with the
phenotype. Currently, the RNA-seq GSA allows only a binary phenotype (i.e, treatment, control).

Usage

RNAgamma(formula, data, rnaprefix="ENSG", gammaShape=1, STT=NULL,
pheno.type=c("case.control"), tagwise=F, perm=T, n.perm=1000, seed=12212012)

Arguments

formula formula in R format: phenotype~cov1+cov2
data data frame containing phenotype, covars, and RNA stuff
rnaprefix RNA data prefix, defaults to ENSG ensembl genes
gammaShape numeric indicating the gamma shape parameter to be used for p-value summa-
STT numeric indicating soft truncation threshold to be used, will calculate gamma
pheno.type type of phenotype, case-control results in logistic regression, quantitative results
in OLS, and survival results in cox model
tagwise TRUE or FALSE for estimating tagwise dispersion values by an empirical Bayes
method based on weighted conditional maximum likelihood. Defaults to max-
imizing the negative binomial conditional common likelihood for the common
dispersion across all tags.
perm boolean indicating whether permutation p-value are to be used for the gamma
summary method
n.perm numeric indicating number of permutations to be used
seed numeric to set RNG for reproducability

Examples

data(testdata)
data(rnaseq_counts)
testdata <- cbind(testdata, rnaseq_counts)
RNAgamma(pheno~strata(study)+age, data=testdata, rnaprefix="rnaseqcount",
pheno.type=c("case.control"), tagwise=FALSE, perm=TRUE, n.perm=5)

#No covars, no permutation
RNAgamma(pheno=., data=testdata, rnaprefix="rnaseqcount",
pheno.type=c("case.control"), tagwise=FALSE, perm=FALSE)
rnaseq_counts

RNA Seq Test data

Description

RNA test data for example

Usage

data(rnaseq_counts)

Format

A data frame with 150 observations on the following 15 variables.

rnaseqcount1 a numeric vector
rnaseqcount2 a numeric vector
rnaseqcount3 a numeric vector
rnaseqcount4 a numeric vector
rnaseqcount5 a numeric vector
rnaseqcount6 a numeric vector
rnaseqcount7 a numeric vector
rnaseqcount8 a numeric vector
rnaseqcount9 a numeric vector
rnaseqcount10 a numeric vector
rnaseqcount11 a numeric vector
rnaseqcount12 a numeric vector
rnaseqcount13 a numeric vector
rnaseqcount14 a numeric vector
rnaseqcount15 a numeric vector

Examples

data(rnaseq_counts)
## maybe str(rnaseq_counts) ; plot(rnaseq_counts) ...
**STTtoShapeParameter**  

*Soft Truncation Threshold*

**Description**

In using the Gamma Method\[4\], a soft truncation threshold (STT) must be specified (that is, shape parameter for gamma distribution). For combining p values using Fisher’s method, set STT to \(1/e\). Based on simulation studies, we have found that STT between 0.10 and 0.20 achieve optimal power for a variety of situations. Empirical p values for the gene set association are determined via permutations.

This function is called by the pcgamma one.

**Usage**

```r
STTtoShapeParameter(STT)
```

**Arguments**

- **STT** numeric indicating soft truncation threshold (STT) to convert to gamma parameter (must be \(\leq 0.4\))

**Examples**

```r
STTtoShapeParameter(0.2)
```

---

**testdata**  

*Test data*

**Description**

A dummy dataset for testing these functions. It contains two covariates (age and study), SNP data, and phenotype (coded 0/1).

**Usage**

```r
data(testdata)
```

**Format**

A data frame with 150 observations on the following 18 variables.

- age   a numeric vector
- study a factor with levels AAA BBB CCC
- snp1  a numeric vector
- snp2  a numeric vector
snp3  a numeric vector
snp4  a numeric vector
snp5  a numeric vector
snp6  a numeric vector
snp7  a numeric vector
snp8  a numeric vector
snp9  a numeric vector
snp10 a numeric vector
snp11 a numeric vector
snp12 a numeric vector
snp13 a numeric vector
snp14 a numeric vector
snp15 a numeric vector
pheno a numeric vector

Examples

data(testdata)
## maybe str(testdata) ; plot(testdata) ...
Index

*Topic datasets
  gene_example, 2
  rnaseq_counts, 6
  testdata, 7
*Topic package
  GSAGm-package, 2

gene_example, 2
GSAGm (GSAGm-package), 2
GSAGm-package, 2
PCgamma, 3
RNAgamma, 4
rnaseq_counts, 6
STTtoShapeParameter, 7
testdata, 7