Package 'meaca'

October 21, 2020

Type Package

Title Mixed-effects Enrichment Analysis with Correlation Adjusted (MEACA)

Version 0.2.2

Author Bin Zhuo

Maintainer Bin Zhuo <rickyb.zh@gmail.com>, Duo Jiang
<jiangd@stat.oregonstate.edu>

Description This package documents the functions used when performing MEACA gene set enrichment analysis.

Depends R (>= 3.2.1)

License MIT + file LICENSE

imports MASS,
mvtnorm,
tibble,
edgeR,
DESeq2

LazyData true

Encoding UTF-8

RoxygenNote 7.1.1

R topics documented:

btw_gene_corr *Calculate sample (Pearson) correlations among gene clusters*

Description

Calculate sample (Pearson) correlations among gene clusters

Usage

```
btw_gene_corr(expression_data, trt, go_term, standardize = T)
```

Arguments

expression_data	the expressoin matrix.
trt	treatment indicators, 1 for treatment, 0 for control group
go_term	an indicator vector. 1 for genes in the test set, 0 otherwise
standardize	whether the data should be standaridzed

Value

a 1×3 data frame containing values for ρ_1, ρ_3 and ρ_2 respectively.

testSetCor	Average correlation for genes in the test set
interCor	Average correlation between genes in the test set and those not in the test set
backSetCor	Average correlations for genes not in the test set.

estimate_sigma *Estimate sample covariance.*

Description

Estimate sample covariance and calculate the gene-level statistics

Usage

```
estimate_sigma(expression_data, trt)
```

Arguments

expression_data	the expression matrix.
trt	sample labels. 0 for control and 1 for treatment

Value

a list

sigma	a covariance matrix
t_val	a vector of gene level test statistics

meaca_multiple *meaca-multiple.*

Description

meaca for testing multiple gene sets.

Usage

```
meaca_multiple(
  expression_data,
  trt,
  geneset,
  standardize = T,
  min_set_size = 5,
  fdr_method = "BH"
)
```

Arguments

expression_data	the expressoin matrix.
trt	treatment labels.
geneset	gene sets to be tested, an object from `read_gene_set`.
standardize	whether the data should be standaridzed.
min_set_size	the minimum number of genes contained for a gene set to be considered.
fdr_method	which method is ued to adjust the p values. see arguments in function `p.adjust`.

Value

a data frame

meaca_single *meaca-single.*

Description

meaca for single gene set test.

Usage

```
meaca_single(expression_data, trt, go_term, standardize = F)
```

Arguments

expression_data	the expressoin matrix.
trt	treatment indicators, 1 for treatment, 0 for control group.
go_term	an indicator vector. 1 for genes in the test set, 0 otherwise.
standardize	whether the data should be standaridzed.

Value

a list

stat	the test statistic
p1	chi-square test p value
status	"up" or "down", the direction of differential expression
p2	two-sided test p-value using normal distribution

Examples

```
t1 <- simulate_expression_data(size = 50, n_gene = 500, n_test = 100,
                               prop = c(0.1, 0.1), de_mu = 2, de_sd = 1,
                               rho1 = 0.1, rho2 = 0.05, rho3 = -0.05,
                               data_gen_method = "chol", seed = 123)
meaca_single(t1$data, trt = t1$trt, go_term = t1$go_term)
```

---

simulate_expression_data

Simulate expression data.

Description

simulate normally distributed expression data with desired DE probabilities for genes in the test set and for those not in the test set..

Usage

```
simulate_expression_data(
  size,
  n_gene,
  n_test,
  prop,
  de_mu,
  de_sd,
  rho1,
  rho2,
  rho3,
  data_gen_method = "chol",
  seed = 123
)
```

Arguments

size	number of samples to be simulated
n_gene	total number of genes to be simulated
n_test	number of genes in the test set.
prop	a vector of length 2, proportion of DE genes within go term and outside go_term, corresponding to p_t and p_b.
de_mu, de_sd	if the gene is DE, delta ~ N(de_mu, de_sd)

rho1	a scalar, correlation between two test genes (i.e., ρ_1 in the paper)
rho2	a scalar, correlation between two background genes (i.e., ρ_2 in the paper)
rho3	correlation between a test gene and a background gene (i.e., ρ_3 in the paper)
data_gen_method	data generation method; if 'data_gen_method = MASS', then mvrnorm is used, otherwise see function rmvnorm
seed	the seed used for simulation (for reproducibility purpose)

Value

a list

data	a expression matrix of $m \times n$ where m is the number of genes and n is the number of samples.
trt	sample labels of length n, 1 for treatment and 0 for control.
go_term	gene labels of length m, 1 for go_term genes and 0 otherwise.
sigma	true covariance matrix upon which data is simulated.

Examples

```
t1 <- simulate_expression_data(size = 50, n_gene = 500, n_test = 100,
                               prop = c(0.1, 0.1), de_mu = 2, de_sd = 1,
                               rho1 = 0.1, rho2 = 0.05, rho3 = -0.05,
                               data_gen_method = "chol", seed = 123)
```

---

standardize_expression_data

standardize expression data, with method described in the paper.

Description

Standardize the expression data.

Usage

```
standardize_expression_data(expression_data, trt)
```

Arguments

expression_data	the expression matrix.
trt	sample labels. 0 for control and 1 for treatment

Value

a matrix of the same dimension as input data.

`transform_count_edgeR` *Perform count matrix transformation using edgeR procedure*

Description

Perform count matrix transformation using edgeR procedure

Usage

```
transform_count_edgeR(y, group)
```

Arguments

y	the expression count matrix, columns being samples, rows being genes
group	a vector of treatment label (e.g., 0 for control, 1 for treatment).

Value

a matrix of transformed data

See Also

[edgeR::camera.DGEList()]

Examples

```
mu <- matrix(10, 100, 4)
group <- factor(c(0,0,1,1))
design <- model.matrix(~group)
set.seed(123)
library(edgeR)
y0 <- matrix(rnbinom(100*4, mu=mu, size=10),100,4)
y <- DGEList(counts=y0, group=group)
y <- estimateDisp(y, design)

iset1 <- 1:10
camera.DGEList(y, iset1, design)

# the Pvalue should be the same
y2 <- transform_count_edgeR(y = y0, group = group)
camera(y2, iset1, design)
```

`transform_count_vst` *Perform variance stabilizing transformation for the count matrix data*

Description

Perform variance stabilizing transformation for the count matrix data

Usage

```
transform_count_vst(y, group, ...)
```

Arguments

`y`	the expression count matrix, columns being samples, rows being genes
`group`	a vector of treatment label (e.g., 0 for control, 1 for treatment).
`...`	other parameters used in `varianceStabilizingTransformation`

Details

It is absolutely critical that the columns of the count matrix and the rows of the column data (information about samples) are in the same order. For more details, see https://bioconductor.org/packages/release/bioc/vignettes/DESeq2/inst/doc/DESeq2.html

Value

a matrix of transformed data

Examples

```
y <- matrix(rbinom(6000, 20, 0.4), nrow = 1000)
group <- c(0, 0, 0, 1, 1, 1)
yr <- transform_count_vst(y = y, group = group)
```

Index