Package ‘BALLI’

April 25, 2019

Type Package
Title Expression RNA-Seq Data Analysis Based on Linear Mixed Model
Version 0.2.0
Author Kyungtaek Park <qkrrudxor147@snu.ac.kr>
Maintainer Kyungtaek Park <qkrrudxor147@snu.ac.kr>
Description Analysis of gene expression RNA-seq data using Bartlett-Adjusted Likelihood-based Linear model (BALLI). Based on likelihood ratio test, it provides comparisons for effect of one or more variables. See Kyungtaek Park (2018) <doi:10.1101/344929> for more information.
Depends R (>= 2.15.0), edgeR, limma, MASS, parallel, stats, methods
License GPL
Encoding UTF-8
LazyData true
RoxygenNote 6.1.1
Suggests knitr, rmarkdown
VignetteBuilder knitr
NeedsCompilation no
Repository CRAN
Date/Publication 2019-04-25 10:40:07 UTC

R topics documented:

balli .................................................. 2
Balli-class ........................................... 3
balliFit ............................................. 3
LargeDataObject-class ............................. 4
tecVarEstim ........................................ 4
TecVarList-class .................................... 5

Index 6
**Description**

DEG analysis using BALLI algorithm

**Usage**

```r
balli(object, intV = 2, logcpm = NULL, tecVar = NULL,
       design = NULL, numCores = NULL, threshold = 1e-06, maxiter = 200)
```

**Arguments**

- `object` - a TecVarList object
- `intV` - numeric vector designating interest variable(s) which is(are) column number(s) of design matrix
- `logcpm` - logcpm values for each gene and each sample
- `tecVar` - estimated technical variance values for each gene and each sample
- `design` - design matrix with samples in row and covariable(s) to be estimated in column
- `numCores` - number of cores to be used for multithreading. If NULL, a single core is used
- `threshold` - threshold for convergence
- `maxiter` - maximum number of iteration to converge of estimated biological variance. If not, biological variance is estimated by using Brent method

**Value**

an Balli object including Result and topGenes list. Following components are shown by Result (same order of genes with input data) and topGenes (ordered by pBALLI in Result) :

- `log2FC` - log2 fold changes of interest variable(s)
- `lLLI` - log-likelihoods estimated by LLI
- `lBALLI` - log-likelihoods estimated by BALLI
- `pLLI` - p-values estimated by LLI
- `pBALLI` - p-values estimated by BALLI
- `BCF` - Bartlett’s correction factor

```r
eexpr <- data.frame(t(sapply(1:1000,function(x)rnbinom(20,mu=500,size=50))))
group <- c(rep("A",10),rep("B",10))
design <- model.matrix(~group, data = expr)
dge <- DGEList(counts=expr, group=group)
dge <- calcNormFactors(dge)
tV <- tecVarEstim(dge,design)
balli(tV,intV=2)
```
**Description**

Class Balli Class Balli holds results from BALLI

**ballifit**  
*balliFit*

**Description**

Estimates likelihood and Bartlett correction factor using BALLI algorithm of each gene

**Usage**

```r
balliFit(y_mat, x_mat, tecVar, intVar = 2, full = T, cfault = 0,
         miter = 200, conv = 1e-06)
```

**Arguments**

- `y_mat` numeric vector containing log-cpm values of each gene and each sample
- `x_mat` design matrix with samples in row and covariable(s) to be estimated in column
- `tecVar` numeric vector containing estimated technical variance of a gene of each sample
- `intVar` numeric vector designating interest variable(s) which is(are) column number(s) of `x_mat`
- `full` logical value designating full model (TRUE) or reduced model (FALSE).
- `cfault` initial value of index showing whether converged (0) or not (1).
- `miter` maximum number of iteration to converge.
- `conv` threshold for convergence

**Value**

Following components are estimated

- `ll` log-likelihoods
- `beta` coefficients of interested variable(s)
- `alpha` coefficients of nuisance variable(s)
- `BCF` Bartlett’s correction factor
- `cfault` index whether converged or not
Examples

```r
expr <- data.frame(t(sapply(1:1000L,function(x)rnbinom(20,mu=500,size=50))))
group <- c(rep("A",10),rep("B",10))
design <- model.matrix(~group, data = expr)
dge <- DGEList(counts=expr, group=group)
dge <- calcNormFactors(dge)
tV <- tecVarEstim(dge,design)
gtv <- tV$tecVar[1,]
gdat <- data.frame(logcpm=tV$logcpm[1,],design,tecVar=gtv)
gy <- matrix(unlist(gdat[,1]),ncol=1)
gx <- matrix(unlist(gdat[,2:(ncol(gdat)-1)]),ncol=ncol(gdat)-2)
bailiFit(y_mat=gy,x_mat=gx,tecVar=gtv,IntVar=2,full=TRUE,cfault=0,miter=200,conv=1e-6)
```

---

**LargeDataObject-class**  
*Class LargeDataObject Class*  
*Largedataobject holds large data such as technical variance and results from BALLI fit*

---

**Description**

Class LargeDataObject Class LargeDataObject holds large data such as technical variance and results from BALLI fit

---

**tecVarEstim**  
*Technical Variance Estimation*

---

**Description**

Estimate technical variance by using voom-trend. The code is derived from voom function in limma package

**Usage**

```
tecVarEstim(counts, design = NULL, lib.size = NULL, span = 0.5, ...)
```

**Arguments**

- `counts` a DGEList object
- `design` design matrix with samples in row and coefficient(s) to be estimated in column
- `lib.size` numeric vector containing total library sizes for each sample
- `span` width of the lowess smoothing window as a proportion
- `...` other arguments are passed to lmFit.
Value

- targets: matrix containing covariables, library sizes and normalization factors of each sample
- design: design matrix with samples in row and covariable(s) to be estimated in column
- logcpm: logcpm values of each gene and each sample
- tecVar: estimated technical variance of each gene and each sample

Examples

```r
expr <- data.frame(t(sapply(1:1000L,function(x)rnbinom(20,mu=500,size=50))))
group <- c(rep("A",10),rep("B",10))
design <- model.matrix(~group, data = expr)
dge <- DEList(counts=expr, group=group)
dge <- calcNormFactors(dge)
tecVarEstim(dge,design)
```

Description

Class TecVarList holds technical variance
Index

balli, 2
Balli-class, 3
balliFit, 3

LargeDataObject-class, 4

tecVarEstim, 4
TecVarList-class, 5