Package 'metaRNASeq'

February 20, 2015

Type Package

Title Meta-analysis of RNA-seq data

Version 1.0.2

Date 2015-01-26

Author Guillemette Marot, Florence Jaffrezic, Andrea Rau

Maintainer Guillemette Marot <guillemette.marot@inria.fr>

Depends R (>= 2.15.0)

Suggests HTSFilter (>= 0.1.1), DESeq (>= 1.8.3), DESeq2 (>= 1.0.17), VennDiagram (<= 1.6.7)

Description Implementation of two p-value combination techniques (inverse normal and Fisher methods). A vignette is provided to explain how to perform a meta-analysis from two independent RNA-seq experiments.

License GPL

LazyLoad yes

biocViews HighThroughputSequencing, RNAseq, DifferentialExpression

Repository CRAN

Repository/R-Forge/Project htsfilter

Repository/R-Forge/Revision 103

Repository/R-Forge/DateTimeStamp 2015-01-26 14:19:14

Date/Publication 2015-01-26 20:33:13

NeedsCompilation no

R topics documented:

`metaRNASeq-package` *Meta-analysis for RNA-seq data.*

Description

Implementation of two p-value combination techniques (inverse normal and Fisher methods). A vignette is provided to explain how to perform a meta-analysis from two independent RNA-seq experiments.

Details

Package:	metaRNASeq
Type:	Package
Version:	1.0.2
Date:	2015-01-26
License:	GPL

Author(s)

Andrea Rau, Guillemette Marot, Florence Jaffrezic

Maintainer: Guillemette Marot <guillemette.marot@inria.fr>

References

A. Rau, G. Marot and F. Jaffrezic (2014). Differential meta-analysis of RNA-seq data. *BMC Bioinformatics* **15**:91

See Also

`invnorm` `fishercomb`

Examples

```
#An User's guide with detailed examples can be downloaded in interactive R sessions
if(interactive()){
vignette("metaRNASeq")
}
```

`dispFuncs` *Gamma regression parameters describing the mean-dispersion relationship for two real datasets.*

Description

Gamma regression parameters describing the mean-dispersion relationship for each of the two real datasets considered in the associated paper, as estimated using the DESeq package version 1.8.3 (Anders and Huber, 2010).

Usage

```
data(dispFuncs)
```

Format

List of length 2, where each list is a vector containing the two estimated coefficients (α_0 and α_1) for the gamma regression in each study (see details below).

Details

The `dispFuncs` object contains the estimated coefficients from the parametric gamma regressions describing the mean-dispersion relationship for the two real datasets considered in the associated paper. The gamma regressions were estimated using the DESeq package version 1.8.3 (Anders and Huber, 2010).

Briefly, after estimating a per-gene mean expression and dispersion values, the DESeq package fits a curve through these estimates. These fitted values correspond to an estimation of the typical relationship between mean expression values μ and dispersions α within a given dataset. By default, this relationship is estimated using a gamma-family generalized linear model (GLM), where two coefficients α_0 and α_1 are found to parameterize the fit as $\alpha = \alpha_0 + \alpha_1/\mu$.

For the first dataset (F078), the estimated mean-dispersion relationship is described using the following gamma-family GLM:

$$\alpha = 0.024 + 14.896/\mu.$$

For the second dataset (F088), the estimated mean-dispersion relationship is described using the following gamma-family GLM:

$$\alpha = 0.00557 + 1.54247/\mu.$$

These gamma-family GLMs describing the mean-dispersions relationship in each of the two datasets are used in this package to simulate data using dispersion parameters that are as realistic as possible.

References

A. Rau, G. Marot and F. Jaffrezic (2014). Differential meta-analysis of RNA-seq data. *BMC Bioinformatics* **15**:91

S. Anders and W. Huber (2010). Differential expression analysis for sequence count data. *Genome Biology*, 11:R106.

See Also

sim.function

Examples

```
data(dispFuncs)
```

`fishercomb` *P-value combination using Fisher's method*

Description

Combines one sided p-values using Fisher's method.

Usage

```
fishercomb(indpval, BHth = 0.05)
```

Arguments

`indpval`	List of vectors of one sided p-values to be combined.
`BHth`	Benjamini Hochberg threshold. By default, the False Discovery Rate is controlled at 5%.

Details

The test statistic for each gene *g* is defined as

$$F_g = -2\sum_{s=1}^{S} ln(p_{gs})$$

where p_{gs} corresponds to the raw *p*-value obtained for gene *g* in a differential analysis for study *s* (assumed to be uniformly distributed under the null hypothesis). Under the null hypothesis, the test statistic F_g follows a χ^2 distribution with *2S* degrees of freedom. Classical procedures for the correction of multiple testing, such as that of Benjamini and Hochberg (1995) may subsequently be applied to the obtained *p*-values to control the false discovery rate at a desired rate α.

Value

`DEindices`	Indices of differentially expressed genes at the chosen Benjamini Hochberg threshold.
`TestStatistic`	Vector with test statistics for differential expression in the meta-analysis.
`rawpval`	Vector with raw p-values for differential expression in the meta-analysis.
`adjpval`	Vector with adjusted p-values for differential expression in the meta-analysis.

References

Y. Benjamini and Y. Hochberg (1995). Controlling the false discovery rate: a pratical and powerful approach to multiple testing. *JRSS B* (57): 289-300.

M. Brown (1975). A method for combining non-independent, one-sided tests of significance. *Biometrics* **31**(4): 987-992.

A. Rau, G. Marot and F. Jaffrezic (2014). Differential meta-analysis of RNA-seq data. *BMC Bioinformatics* **15**:91

See Also

metaRNASeq

Examples

```
data(rawpval)
fishcomb <- fishercomb(rawpval, BHth = 0.05)
DE <- ifelse(fishcomb$adjpval<=0.05,1,0)
hist(fishcomb$rawpval,nclass=100)

## A more detailed example is given in the vignette of the package:
## vignette("metaRNASeq")
```

IDD.IRR *Integration-driven Discovery and Integration-driven Revision Rates*

Description

Calculates the gain or the loss of differentially expressed genes due to meta-analysis compared to individual studies.

Usage

```
IDD.IRR(meta_de, ind_de)
```

Arguments

meta_de	Vector of differentially expressed tags (or indices of these tags) with the meta-analysis
ind_de	List of vectors storing differentially expressed tags (or indices of these tags) in each individual study

Value

DE	Number of Differentially Expressed (DE) genes
IDD	Integration Driven Discoveries: number of genes that are declared DE in the meta-analysis that were not identified in any of the individual studies alone.
Loss	Number of genes that are declared DE in individual studies but not in meta-analysis.
IDR	Integration-driven Discovery Rate: proportion of genes that are identified as DE in the meta-analysis that were not identified in any of the individual studies alone.
IRR	Integration-driven Revision Rate: percentage of genes that are declared DE in individual studies but not in meta-analysis.

Author(s)

Guillemette Marot

References

Marot, G., Foulley, J.-L., Mayer, C.-D., Jaffrezic, F. (2009) Moderated effect size and p-value combinations for microarray meta-analyses. Bioinformatics. 25 (20): 2692-2699.

Examples

```
data(rawpval)
adjpval<-lapply(rawpval, FUN=function(x) p.adjust(x, method="BH"))
ind_smalladjp<-lapply(adjpval, FUN=function(x) which(x <= 0.05))
#indicators corresponding to the inverse normal p-value combination
invnormcomb <- invnorm(rawpval,nrep=c(8,8), BHth = 0.05)
IDD.IRR(invnormcomb$DEindices,ind_smalladjp)
#indicators corresponding to the p-value combination with Fisher's method
fishcomb <- fishercomb(rawpval, BHth = 0.05)
IDD.IRR(fishcomb$DEindices,ind_smalladjp)
```

---

invnorm *P-value combination using the inverse normal method*

Description

Combines one sided p-values using the inverse normal method.

Usage

```
invnorm(indpval, nrep, BHth = 0.05)
```

Arguments

indpval	List of vectors of one sided p-values to be combined.
nrep	Vector of numbers of replicates used in each study to calculate the previous one-sided p-values.
BHth	Benjamini Hochberg threshold. By default, the False Discovery Rate is controlled at 5%.

Details

For each gene g, let

$$N_g = \sum_{s=1}^{S} \omega_s \Phi^{-1}(1 - p_{gs}),$$

where p_{gs} corresponds to the raw *p*-value obtained for gene g in a differential analysis for study s (assumed to be uniformly distributed under the null hypothesis), Φ the cumulative distribution function of the standard normal distribution, and ω_s a set of weights. We define the weights ω_s as in Marot and Mayer (2009):

$$\omega_s = \sqrt{\frac{\sum_c R_{cs}}{\sum_\ell \sum_c R_{c\ell}}},$$

where $\sum_c R_{cs}$ is the total number of biological replicates in study s. This allows studies with large numbers of biological replicates to be attributed a larger weight than smaller studies.

Under the null hypothesis, the test statistic N_g follows a N(0,1) distribution. A unilateral test on the righthand tail of the distribution may then be performed, and classical procedures for the correction of multiple testing, such as that of Benjamini and Hochberg (1995), may subsequently be applied to the obtained *p*-values to control the false discovery rate at a desired level α.

Value

DEindices	Indices of differentially expressed genes at the chosen Benjamini Hochberg threshold.
TestStatistic	Vector with test statistics for differential expression in the meta-analysis.
rawpval	Vector with raw p-values for differential expression in the meta-analysis.
adjpval	Vector with adjusted p-values for differential expression in the meta-analysis.

Note

This function resembles the function directpvalcombi in the *metaMA* R package; there is, however, one important difference in the calculation of *p*-values. In particular, for microarray data, it is typically advised to separate under- and over-expressed genes prior to the meta-analysis. In the case of RNA-seq data, differential analyses from individual studies typically make use of negative binomial models and exact tests, which lead to one-sided, rather than two-sided, p-values. As such, we suggest performing a meta-analysis over the full set of genes, followed by an a posteriori check, and if necessary filter, of genes with conflicting results (over vs. under expression) among studies.

References

Y. Benjamini and Y. Hochberg (1995). Controlling the false discovery rate: a pratical and powerful approach to multiple testing. *JRSS B* (57): 289-300.

Hedges, L. and Olkin, I. (1985). Statistical Methods for Meta-Analysis. Academic Press.

Marot, G. and Mayer, C.-D. (2009). Sequential analysis for microarray data based on sensitivity and meta-analysis. *SAGMB* 8(1): 1-33.

A. Rau, G. Marot and F. Jaffrezic (2014). Differential meta-analysis of RNA-seq data. *BMC Bioinformatics* **15**:91

See Also

metaRNASeq

Examples

```
data(rawpval)
## 8 replicates simulated in each study
invnormcomb <- invnorm(rawpval,nrep=c(8,8), BHth = 0.05)
DE <- ifelse(invnormcomb$adjpval<=0.05,1,0)
hist(invnormcomb$rawpval,nclass=100)

## A more detailed example is given in the vignette of the package:
## vignette("metaRNASeq")
```

---

param *Mean simulation parameters*

Description

Mean simulation parameters obtained from the analysis of a real dataset

Usage

```
data(param)
```

Format

A data frame with 26408 observations on the following 3 variables.

`mucond1` a numeric vector with mean parameters for condition 1

`mucond2` a numeric vector with mean parameters for condition 2

`DE` a logical vector indicating which tags are differentially expressed (value 1)

Details

Mean parameters provided in this package are empirical means (obtained after normalization for library size differences) of real data described in the following references.

Source

Supplementary material of (Dillies et al., 2013) paper.

References

M.A. Dillies, A. Rau, J. Aubert, C. Hennequet-Antier, M. Jeanmougin, N. Servant, C. Keime, G. Marot, D. Castel, J. Estelle, G. Guernec, B. Jagla, L. Jouneau, D. Laloe, C. Le Gall, B. Schaeffer, S. Le Crom, M. Guedj and F. Jaffrezic, on behalf of the French StatOmique Consortium (2013) A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis. *Briefings in Bioinformatics* **14**(6):671-83 .

T. Strub, S. Giuliano, T. Ye, et al. (2011) Essential role of microphthalmia transcription factor for DNA replication, mitosis and genomic stability in melanoma. emphOncogene **30**:2319-32.

Examples

```
data(param)
```

---

rawpval *Simulated p-values*

Description

The p-values provided here result from the following procedure: 1) simulation of two RNA-seq experiments with four replicates in each condition via the `sim.function`, 2) analysis of differentially expressed tags using the DESeq package.

Usage

```
data(rawpval)
```

Format

List of length 2, where each list is a vector containing the raw p-values for 14,456 genes from individual differential analyses (obtained using DESeq v1.8.3) of each of the simulated RNA-seq datasets.

Details

It is possible to reproduce these p-values using the procedure described in the package vignette.

References

A. Rau, G. Marot and F. Jaffrezic (2014). Differential meta-analysis of RNA-seq data. *BMC Bioinformatics* **15**:91

Examples

```
data(rawpval)
## Maybe str(rawpval)
```

`sim.function` *Simulation of multiple RNA-seq experiments*

Description

Simulate data arising from multiple independent RNA-seq experiments

Usage

```
sim.function(param, dispFuncs, nrep = 4, classes = NULL, inter.sd = 0.3)
```

Arguments

`param`	Mean expression levels: `param` must be a data frame containing at least two columns named "mucond1" and "mucond2" and one row per gene.
`dispFuncs`	List of length equal to the number of studies to be simulated, containing the gamma regression parameters describing the mean-dispersion relationship for each one; these are the mean-dispersion functions linking mean and intra-study variability for each independent experiment.
`nrep`	Number of replicates to be simulated in each condition in each study. Ignored if `classes` is filled.
`classes`	List of class memberships, one per study (maximum 20 studies). Each vector or factor of the list can only contain two levels which correspond to the two conditions studied. If NULL, `classes` is built as a list of two vectors with `nrep` labels 1 (for condition 1) and `nrep` labels 2 (for condition 2).
`inter.sd`	Inter-study variability. By default, `inter.sd` is set to 0.3, which corresponds to a moderate inter-study variability in the case where `param` and `dispFuncs` parameters are used to simulate data.

Details

Details about the simulation procedure are given in the following paper:

Value

A matrix with simulated expression levels, one row per gene and one column per replicate. Names of studies are given in the column names of the matrix.

Note

If the `param` data provided in this package are not used to simulate data, one should check that the per-condition means in `param` are reasonable. Note also that for genes to be simulated as non-differentially expressed, the values of "mucond1" and "mucond2" in `param` should be equal.

References

A. Rau, G. Marot and F. Jaffrezic (2014). Differential meta-analysis of RNA-seq data. *BMC Bioinformatics* **15**:91

See Also

metaRNASeq

Examples

```
## Load simulation parameters
data(param)
data(dispFuncs)

## Simulate data
matsim <- sim.function(param = param, dispFuncs = dispFuncs)
sim.conds <- colnames(matsim)
rownames(matsim) <-paste("tag", 1:dim(matsim)[1],sep="")

# extract simulated data from one study
simstudy1 <- extractfromsim(matsim,"study1")
head(simstudy1$study)
simstudy1$pheno
```

Index