Package 'ssizeRNA'

February 19, 2019

Type Package

Title Sample Size Calculation for RNA-Seq Experimental Design

Version 1.3.1

Date 2019-02-18

Maintainer Ran Bi <biranpier@gmail.com>

Description We propose a procedure for sample size calculation while controlling false discovery rate for RNA-seq experimental design. Our procedure depends on the Voom method proposed for RNA-seq data analysis by Law et al. (2014) <DOI:10.1186/gb-2014-15-2-r29> and the sample size calculation method proposed for microarray experiments by Liu and Hwang (2007) <DOI:10.1093/bioinformatics/btl664>. We develop a set of functions that calculates appropriate sample sizes for two-sample t-test for RNA-seq experiments with fixed or varied set of parameters. The outputs also contain a plot of power versus sample size, a table of power at different sample sizes, and a table of critical test values at different sample sizes.
To install this package, please use
'source(``http://bioconductor.org/biocLite.R"); biocLite(``ssizeRNA")'.

Depends R (>= 3.2.3)

Imports MASS, Biobase, edgeR, limma, qvalue, ssize.fdr, graphics, stats

VignetteBuilder knitr

Suggests knitr

License GPL (>= 2)

biocViews GeneExpression, DifferentialExpression, ExperimentalDesign, Sequencing, RNASeq, DNASeq, Microarray

RoxygenNote 5.0.1

NeedsCompilation no

Author Ran Bi [aut, cre],
Peng Liu [aut],
Tim Triche [ctb]

Repository CRAN

Date/Publication 2019-02-19 12:30:03 UTC

R topics documented:

`check.power` *Average Power and True FDR Based on limma/voom RNAseq Analysis Pipeline*

Description

For the limma/voom RNAseq analysis pipeline, when we control false discovery rate by using the Benjamini and Hochberg step-up procedure (1995) and/or Storey and Tibshirani's q-value procedure (Storey et al, 2004), `check.power` calculates average power and true FDR for given sample size, user-specified proportions of non-differentially expressed genes, number of iterations, FDR level to control, mean counts in control group, dispersion, and fold change.

Usage

```
check.power(nGenes = 10000, pi0 = 0.8, m, mu, disp, fc, up = 0.5,
  replace = TRUE, fdr = 0.05, sims = 100)
```

Arguments

`nGenes`	total number of genes, the default value is `10000`.
`pi0`	proportion of non-differentially expressed genes, the default value is `0.8`.
`m`	sample size per treatment group.
`mu`	a vector (or scalar) of mean counts in control group from which to simulate.
`disp`	a vector (or scalar) of dispersion parameter from which to simulate.
`fc`	a vector (or scalar, or a function that takes an integer n and generates a vector of length n) of fold change for differentially expressed (DE) genes.
`up`	proportion of up-regulated genes among all DE genes, the default value is `0.5`.
`replace`	sample with or without replacement from given parameters. See Details for more information.
`fdr`	the false discovery rate to be controlled.
`sims`	number of simulations to run when computing power and FDR.

Value

pow_bh_ave	average power when controlling FDR by Benjamini and Hochberg (1995) method.
fdr_bh_ave	true false discovery rate when controlling FDR by Benjamini and Hochberg (1995) method.
pow_bh_ave	average power when controlling FDR by q-value procedure (Storey et al., 2004).
fdr_bh_ave	true false discovery rate when controlling FDR by q-value procedure (Storey et al., 2004).

Author(s)

Ran Bi <biranpier@gmail.com>, Peng Liu <pliu@iastate.edu>

References

Benjamini, Y. and Hochberg, Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. B*, 57, 289-300.

Storey, J. D., Taylor, J. E. and Siegmund, D. (2004) Strong control, conservative point estimation and simultaneous rates: a unified approach. *J. R. Stat. Soc. B*, 66, 187- 205.

Examples

```
library(limma)
library(qvalue)
m <- 14                    ## sample size per treatment group
mu <- 10                   ## mean read counts in control group
disp <- 0.1                ## dispersion for all genes
fc <- 2                    ## 2-fold change for DE genes

check.power(m = m, mu = mu, disp = disp, fc = fc, sims = 2)
```

---

hammer.eset *RNA-seq data from Hammer, P. et al., 2010*

Description

RNA-seq data structured as an expressionSet, from "mRNA-seq with agnostic splice site discovery for nervous system transcriptomics tested in chronic pain" by Hammer, P. et al. (Genome Res. 2010, 20(6):847-860), http://dx.doi.org/10.1101/gr.101204.109.

Usage

```
data(hammer.eset)
```

Value

RNA-seq data structured as an expressionSet.

Author(s)

Ran Bi <biranpier@gmail.com>, Peng Liu <pliu@iastate.edu>

`sim.counts` *RNA-seq Count Data Simulation from Negative-Binomial Distribution*

Description

This function simulates count data from Negative-Binomial distribution for two-sample RNA-seq experiments with given mean, dispersion and fold change. A count data matrix is generated.

Usage

```
sim.counts(nGenes = 10000, pi0 = 0.8, m, mu, disp, fc, up = 0.5,
  replace = TRUE)
```

Arguments

`nGenes`	total number of genes, the default value is `10000`.
`pi0`	proportion of non-differentially expressed genes, the default value is `0.8`.
`m`	sample size per treatment group.
`mu`	a vector (or scalar) of mean counts in control group from which to simulate.
`disp`	a vector (or scalar) of dispersion parameter from which to simulate.
`fc`	a vector (or scalar, or a function that takes an integer n and generates a vector of length n) of fold change for differentially expressed (DE) genes.
`up`	proportion of up-regulated genes among all DE genes, the default value is `0.5`.
`replace`	sample with or without replacement from given parameters. See Details for more information.

Details

If the total number of genes `nGenes` is larger than length of `mu` or `disp`, `replace` always equals `TRUE`.

Value

`counts`	RNA-seq count data matrix.
`group`	treatment group vector.
`lambda0`	mean counts in control group for each gene.
`phi0`	dispersion parameter for each gene.
`de`	differentially expressed genes indicator: `0` for non-differentially expressed genes, `1` for up-regulated genes, `-1` for down-regulated genes.
`delta`	log2 fold change for each gene between treatment group and control group.

Author(s)

Ran Bi <biranpier@gmail.com>, Peng Liu <pliu@iastate.edu>

Examples

```
m <- 3                    ## sample size per treatment group
mu <- 10                  ## mean counts in control group for all genes
disp <- 0.1               ## dispersion for all genes
fc <- 2                   ## 2-fold change for DE genes

sim <- sim.counts(m = m, mu = mu, disp = disp, fc = fc)
sim$counts                ## count data matrix


## varying fold change
fc1 <- function(x){exp(rnorm(x, log(2), 0.5*log(2)))}
sim1 <- sim.counts(m = m, mu = mu, disp = disp, fc = fc1)
```

---

ssize.twoSampVaryDelta

Sample Size Calculations for Two-Sample Microarray Experiments with Differing Mean Expressions but fixed Standard Deviations Among Genes

---

Description

For given desired power, controlled false discovery rate, and user-specified proportions of non-differentially expressed genes, `ssize.twoSampVaryDelta` calculates appropriate sample sizes for two-sample microarray experiments in which the differences between mean treatment expression levels (*delta.g* for gene *g*) vary among genes. A plot of power versus sample size is generated.

Usage

```
ssize.twoSampVaryDelta(deltaMean, deltaSE, sigma, fdr = 0.05, power = 0.8,
  pi0 = 0.95, maxN = 35, side = "two-sided", cex.title = 1.15,
  cex.legend = 1)
```

Arguments

`deltaMean`	location (mean) parameter of normal distribution followed by each *delta.g*.
`deltaSE`	scale (standard deviation) parameter of normal distribution followed by each *delta.g*.
`sigma`	the common standard deviation of expressions for all genes.
`fdr`	the false discovery rate to be controlled.
`power`	the desired power to be achieved.

pi0	a vector (or scalar) of proportions of non-differentially expressed genes.
maxN	the maximum sample size used for power calculations.
side	options are "two-sided", "upper", or "lower".
cex.title	controls size of chart titles.
cex.legend	controls size of chart legend.

Details

Each *delta.g* is assumed to follow a Normal distribution with mean deltaMean and standard deviation deltaSE. The standard deviations of expressions are assumed identical for all genes.

If a vector is input for pi0, sample size calculations are performed for each proportion.

Value

ssize	sample sizes (for each treatment) at which desired power is first reached.
power	power calculations with corresponding sample sizes.
crit.vals	critical value calculations with corresponding sample sizes.

Author(s)

Ran Bi <biranpier@gmail.com>, Peng Liu <pliu@iastate.edu>

References

Liu, P. and Hwang, J. T. G. (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis. *Bioinformatics* 23(6): 739-746.

Orr, M. and Liu, P. (2009) Sample size estimation while controlling false discovery rate for microarray experiments using ssize.fdr package. *The R Journal*, 1, 1, May 2009, 47-53.

See Also

ssize.twoSamp, ssize.twoSampVary, ssize.oneSamp, ssize.oneSampVary, ssize.F, ssize.Fvary

Examples

```
dm <- 1.2; ds <- 0.1  ## the delta.g's follow a Normal(1.2, 0.1) distribution
s <- 1                ## common standard deviation
fdr <- 0.05           ## false discovery rate to be controlled
pwr <- 0.8            ## desired power
pi0 <- c(0.5, 0.8, 0.99) ## proportions of non-differentially expressed genes
N <- 35               ## maximum sample size for calculations

size <- ssize.twoSampVaryDelta(deltaMean = dm, deltaSE = ds, sigma = s,
                               fdr = fdr, power = pwr, pi0 = pi0,
                               maxN = N, side = "two-sided")
size$ssize               ## first sample size(s) to reach desired power
size$power               ## calculated power for each sample size
size$crit.vals           ## calculated critical value for each sample size
```

ssizeRNA_single *Sample Size Calculations for Two-Sample RNA-seq Experiments with Single Set of Parameters*

Description

This function calculates appropriate sample sizes for two-sample RNA-seq experiments for a desired power in which mean and dispersion parameters are identical for all genes. Sample size calculations are performed at controlled false discovery rates, user-specified proportions of non-differentially expressed genes, mean counts in control group, dispersion, and fold change. A plot of power versus sample size is generated.

Usage

```
ssizeRNA_single(nGenes = 10000, pi0 = 0.8, m = 200, mu, disp, fc,
  up = 0.5, replace = TRUE, fdr = 0.05, power = 0.8, maxN = 35,
  side = "two-sided", cex.title = 1.15, cex.legend = 1)
```

Arguments

nGenes	total number of genes, the default value is 10000.
pi0	proportion of non-differentially expressed genes, the default value is 0.8.
m	pseudo sample size for generated data.
mu	a vector (or scalar) of mean counts in control group from which to simulate.
disp	a vector (or scalar) of dispersion parameter from which to simulate.
fc	a vector (or scalar, or a function that takes an integer n and generates a vector of length n) of fold change for differentially expressed (DE) genes.
up	proportion of up-regulated genes among all DE genes, the default value is 0.5.
replace	sample with or without replacement from given parameters. See Details for more information.
fdr	the false discovery rate to be controlled.
power	the desired power to be achieved.
maxN	the maximum sample size used for power calculations.
side	options are "two-sided", "upper", or "lower".
cex.title	controls size of chart titles.
cex.legend	controls size of chart legend.

Details

If a vector is input for pi0, sample size calculations are performed for each proportion.

If the total number of genes is larger than length of mu or disp, replace always equals TRUE.

Value

ssize	sample sizes (for each treatment) at which desired power is first reached.
power	power calculations with corresponding sample sizes.
crit.vals	critical value calculations with corresponding sample sizes.

Author(s)

Ran Bi <biranpier@gmail.com>, Peng Liu <pliu@iastate.edu>

References

Liu, P. and Hwang, J. T. G. (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis. *Bioinformatics* 23(6): 739-746.

Orr, M. and Liu, P. (2009) Sample size estimation while controlling false discovery rate for microarray experiments using ssize.fdr package. *The R Journal*, 1, 1, May 2009, 47-53.

Law, C. W., Chen, Y., Shi, W., Smyth, G. K. (2014). Voom: precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biology* 15, R29.

See Also

ssizeRNA_vary

Examples

```
mu <- 10                ## mean counts in control group for all genes
disp <- 0.1             ## dispersion for all genes
fc <- 2                 ## 2-fold change for DE genes

size <- ssizeRNA_single(m = 30, mu = mu, disp = disp, fc = fc,
                        maxN = 20)
size$ssize              ## first sample size to reach desired power
size$power              ## calculated power for each sample size
size$crit.vals          ## calculated critical value for each sample size
```

---

ssizeRNA_vary *Sample Size Calculations for Two-Sample RNA-seq Experiments with Differing Mean and Dispersion Among Genes*

Description

This function calculates appropriate sample sizes for two-sample RNA-seq experiments for a desired power in which mean and dispersion vary among genes. Sample size calculations are performed at controlled false discovery rates, user-specified proportions of non-differentially expressed genes, mean counts in control group, dispersion, and fold change. A plot of power versus sample size is generated.

Usage

```
ssizeRNA_vary(nGenes = 10000, pi0 = 0.8, m = 200, mu, disp, fc,
  up = 0.5, replace = TRUE, fdr = 0.05, power = 0.8, maxN = 35,
  side = "two-sided", cex.title = 1.15, cex.legend = 1)
```

Arguments

nGenes	total number of genes, the default value is 10000.
pi0	proportion of non-differentially expressed genes, the default value is 0.8.
m	pseudo sample size for generated data.
mu	a vector (or scalar) of mean counts in control group from which to simulate.
disp	a vector (or scalar) of dispersion parameter from which to simulate.
fc	a vector (or scalar, or a function that takes an integer n and generates a vector of length n) of fold change for differentially expressed (DE) genes.
up	proportion of up-regulated genes among all DE genes, the default value is 0.5.
replace	sample with or without replacement from given parameters. See Details for more information.
fdr	the false discovery rate to be controlled.
power	the desired power to be achieved.
maxN	the maximum sample size used for power calculations.
side	options are "two-sided", "upper", or "lower".
cex.title	controls size of chart titles.
cex.legend	controls size of chart legend.

Details

If a vector is input for pi0, sample size calculations are performed for each proportion.

If the total number of genes is larger than length of mu or disp, replace always equals TRUE.

Value

ssize	sample sizes (for each treatment) at which desired power is first reached.
power	power calculations with corresponding sample sizes.
crit.vals	critical value calculations with corresponding sample sizes.

Author(s)

Ran Bi <biranpier@gmail.com>, Peng Liu <pliu@iastate.edu>

References

Liu, P. and Hwang, J. T. G. (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis. *Bioinformatics* 23(6): 739-746.

Orr, M. and Liu, P. (2009) Sample size estimation while controlling false discovery rate for microarray experiments using ssize.fdr package. *The R Journal*, 1, 1, May 2009, 47-53.

Law, C. W., Chen, Y., Shi, W., Smyth, G. K. (2014). Voom: precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biology* 15, R29.

See Also

ssizeRNA_single

Examples

```
library(edgeR)
library(Biobase)
data(hammer.eset)
## load hammer dataset (Hammer, P. et al., 2010)

counts <- exprs(hammer.eset)[, phenoData(hammer.eset)$Time == "2 weeks"]
counts <- counts[rowSums(counts) > 0,]
trt <- hammer.eset$protocol[which(hammer.eset$Time == "2 weeks")]

mu <- apply(counts[, trt == "control"], 1, mean)
## average read count in control group for each gene

d <- DGEList(counts)
d <- calcNormFactors(d)
d <- estimateCommonDisp(d)
d <- estimateTagwiseDisp(d)
disp <- d$tagwise.dispersion
## dispersion for each gene

## fixed fold change
fc <- 2
size <- ssizeRNA_vary(mu = mu, disp = disp, fc = fc,
                      m = 30, maxN = 15, replace = FALSE)
size$ssize         ## first sample size to reach desired power
size$power         ## calculated power for each sample size
size$crit.vals     ## calculated critical value for each sample size


## varying fold change
## fc1 <- function(x){exp(rnorm(x, log(2), 0.5*log(2)))}
## size1 <- ssizeRNA_vary(pi0 = 0.8, mu = mu, disp = disp, fc = fc1,
##                        m = 30, maxN = 20, replace = FALSE)
```

Index