Package 'SSPA'

October 8, 2014

Type Package

Title General Sample Size and Power Analysis for Microarray and Next-Generation Sequencing Data

Version 2.4.0

Author Maarten van Iterson

Maintainer Maarten van Iterson <mviterson@gmail.com>

Description General Sample size and power analysis for microarray and next-generation sequencing data.

License GPL (>= 2)

LazyLoad yes

Imports graphics, stats

Depends R (>= 2.12), methods, qvalue, lattice, limma

Suggests BiocStyle, genefilter, edgeR, DESeq

URL http://www.humgen.nl/MicroarrayAnalysisGroup.html

Collate 'zzz.R' 'numericalintegration.R' 'trimmingbinning.R' 'DistributionClass.R' 'PilotDataClass.R' 'SampleSizeClass.R' 'bitriangular.R' 'deconvolution.R' 'conjugategradient.R' 'Ferreira.R' 'tikhonov.R' 'powerandsamplesize.R'

biocViews Microarray, StatisticalMethod

R topics documented:

`dbitri` *Density function for a bi-triangular random variable.*

Description

Density function for a bi-triangular random variable.

Usage

```
dbitri(x, a = log2(1.2), b = log2(4), m = log2(2))
```

Arguments

`x`	vector
`a`	location of point ... Default a = log2(1.2).
`b`	location of point ... Default b = log2(4).
`m`	location of the midpoint of the triangle. Default m = log2(2).

Details

For more details see M. Langaas et al. JRSS B 2005.

Value

Gives the density function.

Author(s)

Maarten van Iterson

Examples

```
curve(dbitri, -4, 4)
```

deepSAGE — *Test statistics derived from a deepSAGE experiment*

Description

follow

Usage

```
data(deepSAGE)
```

Format

A vector of 44882 test statistics.

Vector of test statistics obtained by performing a likelihood ratio test using edgeR

Details

follow

Source

't Hoen, P.A.C. Ariyurek, Y. Thygesen, H.H. Vreugdenhil, E. Vossen, R.H.A.M. de Menezes, R.X. Boer, J.M. van Ommen, G.B. and den Dunnen, J.T., Deep Sequencing-based Expression analysis shows Major Advances in Robustness, Resolution and Inter-lab Portability over Five Microarray Platforms, Nucleic Acids Research, 2008.

Examples

```
data(deepSAGE)
str(deepSAGE)
```

Nutrigenomics — *Test statistics from a Nutrigenomics gene expression profiling experiment*

Description

There are five sets of test statistics each represents a different compound and exposure time. Test statistics were obtained by using an empirical Bayes linear model.

Usage

```
data(Nutrigenomics)
```

Format

A data frame with 16539 test statistics for five experiments.

First row indicates the effective sample size of the experiment. Column names refer to the compound and exposure time (see details).

Details

In this experiment the outcome of specific PPAR-alpha activation on murine small intestinal gene expression was examined using Affymetrix GeneChip Mouse 430 2.0 arrays. PPAR-alpha was activated by several PPAR-alpha-agonists that differed in activating potency. In this paper the data of three agonists were used, namely Wy14,643, fenofibrate and trilinolenin (C18:3). The first two compounds belong to the fibrate class of drugs that are widely prescribed to treat dyslipidemia, whereas trilinolenin is an agonist frequently found in the human diet. For intestinal PPAR-alpha, Wy14,643 is the most potent agonist followed by C18:3 and fenofibrate. Since time of exposure also affects the effect size, intestines were collected 6 hrs (all three agonists) or 5 days (Wy14,643 and fenofibrate only) after exposure.

Source

van Iterson, M. 't Hoen, P.A.C. Pedotti, P. Hooiveld, G.J.E.J. den Dunnen, J.T. van Ommen, G.J.B. Boer, J.M. Menezes, R.X., Relative power and sample size analysis on gene expression profiling data, BMC Genomics, (2009).

Examples

```
data(Nutrigenomics)
str(Nutrigenomics)
```

---

pbitri *Distribution function for a bi-triangular random variable.*

Description

Distribution function for a bi-triangular random variable.

Usage

```
pbitri(q, a = log2(1.2), b = log2(4), m = log2(2))
```

Arguments

q	vector of quantiles.
a	location of point, ... Default a = log2(1.2).
b	location of point, ... Default b = log2(4).
m	location of the midpoint of the triangle. Default m = log2(2).

Details

For more details see M. Langaas et al. JRSS B 2005.

Value

Gives the distribution function.

Author(s)

Maarten van Iterson

Examples

```
curve(pbitri, -4, 4)
```

`pilotData` *User friendly interface to class "PilotData"*

Description

User friendly interface to class "PilotData"

Usage

```
pilotData(statistics = NULL, samplesize = NULL,
  distribution = c("norm", "t", "f", "chisq"), ...)
```

Arguments

`statistics`	vector of test statistics
`samplesize`	total sample size of the pilot-data or effective sample size in two-group case (see Details for more information).
`distribution`	type of the null/alternative distribution, one of 'norm', 't', 'f' or 'chisq'
`...`	additional arguments for the distribution like degrees of freedom

Details

In the two-group case the effective sample size is defined as the square-root of the inverse of 1/n1 + 1/n2.

Value

object of class "PilotData"

Author(s)

Maarten van Iterson

Examples

```
pd <- pilotData(statistics=rnorm(100), samplesize=10, distribution="norm")
pd
plot(pd)
```

`plot-methods` *Methods for Function* `plot` *in Package* **SSPA**

Description

Plot function for objects of class PilotData and SampleSize

Methods

`signature(x = "PilotData")` Diagonstic plots of the PilotData.
`signature(x = "SampleSize")` Plot the estimated density of effect sizes.

`predictpower` *Predict power for given vector of sample sizes*

Description

Predict power for given vector of sample sizes

Usage

```
predictpower(object, samplesizes, alpha = 0.1,
  verbose = FALSE, plot = FALSE)
```

Arguments

`object`	of class 'SampleSize'
`samplesizes`	vector of total sample sizes.
`alpha`	FDR.
`verbose`	TRUE/FALSE
`plot`	TRUE/FALSE

Details

details follow.

Value

predicted power.

Author(s)

Maarten van Iterson

qbitri *Quantile function for a bi-triangular random variable.*

Description

Quantile function for a bi-triangular random variable.

Usage

```
qbitri(p, a = log2(1.2), b = log2(4), m = log2(2))
```

Arguments

p	vector of probabilities.
a	location of point, ... Default a = log2(1.2).
b	location of point, ... Default b = log2(4).
m	location of the midpoint of the triangle. Default m = log2(2).

Details

For more details see M. Langaas et al. JRSS B 2005.

Value

Gives the quantile function.

Author(s)

Maarten van Iterson

Examples

```
curve(qbitri, 0, 1)
```

rbitri *Random generation of bitriangular distributed values.*

Description

Random generation of bitriangular distributed values.

Usage

```
rbitri(n, a = log2(1.2), b = log2(4), m = log2(2))
```

Arguments

n	number of observations.
a	location of point, ... Default a = log2(1.2).
b	location of point, ... Default b = log2(4).
m	location of the midpoint of the triangle. Default m = log2(2).

Details

For more details see M. Langaas et al. JRSS B 2005.

Value

Generates random deviates.

Author(s)

Maarten van Iterson

Examples

```
hist(rbitri(100), freq=FALSE)
curve(dbitri, add=TRUE)
```

---

sampleSize *User friendly interface to class 'SampleSize'*

Description

User friendly interface to class "SampleSize"

Usage

```
sampleSize(PilotData,
  method = c("deconv", "congrad", "tikhonov", "ferreira"),
  control = list(from = -6, to = 6, resolution = 2^9))
```

Arguments

PilotData	object of class 'PilotData'.
method	estimation method one of 'deconv', 'congrad', 'tikhonov' or 'ferreira'. See 'Details'.
control	A list of control parameters. See 'Details'.

Details

The default method is 'deconv' which is an kernel deconvolution density estimator implementated using `fft`. The 'nncg' is a nonnegative conjugate gradient algorithm based on R's implementation see `optim`. 'tikonov' implements ridge-regression with optimal penalty selection using the L-curve approach. Higher order penalties are possible as well using a transformation to standard form (see Hansen).

The 'control' argument is a list that can supply any of the following components. Per method logical checks are performed.

- deconv:
  - method:'deconv', 'ferreira'
  - pi0Method:the pi0 estimation method one of 'Langaas', 'Storey', 'Ferreira', 'Userdefined'
  - pi0:if method = 'ferreira' grid pi0-value need to be suppled e.g. seq(0.1, 0.99, 0.01)
  - adjust:Default TRUE, adjust pi0 esitmate if density of effect size is somewhere negative.
  - a:Adjust pi0 better approach suggested by Efron. Symmetric range around zero of size 0.5.
  - bandwith:Default NULL uses 1/sqrt(log(length(statistics)))
  - kernel:Either 'fan', 'wand', 'sinc' kernels can be used.
  - from:Density of effect sizes should be estimated from = -6
  - to: to = 6
  - resolution:Density of effect sizes should be estimated on 2^9 points.
  - verbose:Default FALSE if TRUE additional information is printed to the console.
- congrad:
  - integration:'midpoint', 'trapezoidal', 'simpson'
  - scale:'pdfstat', 'cdfstat', 'cdfpval'
  - trim:0.01, 0.99
  - symmetric:TRUE
  - bin:'epdf', 'ecdf'
  - from:-6
  - to:6
  - resolution:500
  - verbose:Default FALSE if TRUE additional information is printed to the console.
- tikhonov:
  - integration:'midpoint', 'trapezoidal', 'simpson'
  - scale:'pdfstat', 'cdfstat', 'cdfpval'
  - trim:0.01, 0.99
  - symmetric:TRUE
  - bin:'epdf', 'ecdf'
  - from:-6
  - to:6
  - resolution:500

  - method:'lcurve', 'gcv', 'aic'
  - log:TRUE
  - penalty:0
  - lambda:10^seq(-10, 10, length=100)
  - verbose:Default FALSE if TRUE additional information is printed to the console.
- 'ferreira:'not yet implemeneted

Value

object of class SampleSize.

Author(s)

Maarten van Iterson

References

van Iterson, M., P. 't Hoen, P. Pedotti, G. Hooiveld, J. den Dunnen, G. van Ommen, J. Boer, and R. de Menezes (2009): 'Relative power and sample size analysis on gene expression profiling data,' BMC Genomics, 10, 439–449.

Ferreira, J. and A. Zwinderman (2006a): 'Approximate Power and Sample Size Calculations with the Benjamini-Hochberg Method,' The International Journal of Biostatistics, 2, 1.

Ferreira, J. and A. Zwinderman (2006b): 'Approximate Sample Size Calculations with Microarray Data: An Illustration,' Statistical Applications in Genetics and Molecular Biology, 5, 1.

Hansen, P. (2010): Discrete Inverse Problems: Insight and Algorithms, SIAM: Fun- damentals of algorithms series.

Langaas, M., B. Lindqvist, and E. Ferkingstad (2005): 'Estimating the proportion of true null hypotheses, with application to DNA microarray data,' Journal of the Royal Statistical Society Series B, 67, 555–572.

Storey, J. (2003): 'The positive false discovery rate: A bayesian interpretation and the q-value,' Annals of Statistics, 31, 2013–2035.

See Also

optim

Examples

```
m <- 5000 ##number of genes
J <- 10 ##sample size per group
pi0 <- 0.8 ##proportion of non-differentially expressed genes
m0 <- as.integer(m*pi0)
mu <- rbitri(m - m0, a = log2(1.2), b = log2(4), m = log2(2)) #effect size distribution
data <- simdat(mu, m=m, pi0=pi0, J=J, noise=NULL)
library(genefilter)
stat <- rowttests(data, factor(rep(c(0, 1), each=J)), tstatOnly=TRUE)$statistic
pd <- pilotData(statistics=stat, samplesize=sqrt(J/2), distribution=norm)
ss <- sampleSize(pd, method=deconv)
plot(ss)
```

show-methods *General show method for Classes PilotData and SampleSize*

Description

Methods for function show in package **SSPA**

Methods

`signature(object = "PilotData")` Show the content of a PilotData-object in a userfriendly way.

`signature(object = "SampleSize")` Show the content of a SampleSize-object in a userfriendly way.

simdat *Generate simulated microarray data using the bitriangular distribution.*

Description

Simulated microarray data.

Usage

```
simdat(mu, m, pi0, J, nullX = function(x) rnorm(x, 0, 1),
  nullY = function(x) rnorm(x, 0, 1), noise = 0.01)
```

Arguments

mu	vector of effect sizes drawn from the bitriangular distribution.
m	number of features (genes, tags, ...).
pi0	proportion of nondifferentially expressed features.
J	number of samples per group.
nullX	the distribution of nondifferentially expressed features.
nullY	the distribution of nondifferentially expressed features.
noise	standard deviation of the additive noise.

Details

details follow

Value

Matrix of size m x (2J), containing the simulated values.

Author(s)

Maarten van Iterson

Examples

```
##generate two-group microarray data
m <- 5000 ##number of genes
J <- 10 ##sample size per group
pi0 <- 0.8 ##proportion of non-differentially expressed genes
m0 <- as.integer(m*pi0)
mu <- rbitri(m - m0, a = log2(1.2), b = log2(4), m = log2(2)) #effect size distribution
data <- simdat(mu, m=m, pi0=pi0, J=J, noise=0.01)
```

Index