Package ‘MBttest’

February 20, 2024

Type Package
Title Multiple Beta t-Tests
Version 1.30.0
Date 2015-01-04
Author Yuan-De Tan
Maintainer Yuan-De Tan <tanyuande@gmail.com>

Description MBttest method was developed from beta t-test method of Baggerly et al (2003). Compared to baySeq (Hard castle and Kelly 2010), DESeq (Anders and Huber 2010) and exact test (Robinson and Smyth 2007, 2008) and the GLM of McCarthy et al (2012), MBttest is of high work efficiency, that is, it has high power, high conservativeness of FDR estimation and high stability. MBttest is suitable to transcriptomic data, tag data, SAGE data (count data) from small samples or a few replicate libraries. It can be used to identify genes, mRNA isoforms or tags differentially expressed between two conditions.

License GPL-3
Depends R (>= 3.3.0), stats, gplots, gtools, graphics, base, utils, grDevices
Suggests BiocStyle, BiocGenerics
LazyLoad yes

biocViews Sequencing, DifferentialExpression, MultipleComparison, SAGE, GeneExpression, Transcription, AlternativeSplicing, Coverage, DifferentialSplicing

NeedsCompilation no
git_url https://git.bioconductor.org/packages/MBttest
git_branch RELEASE_3_18
git_last_commit 0de6ef2
git_last_commit_date 2023-10-24
Repository Bioconductor 3.18
Date/Publication 2024-02-20
Multiple Beta t-tests

Description

This package is used to perform multiple beta t-test analyses of real data and gives heatmap of differential expressions of genes or differential splicings. The results listing geneid or isoformid, gene name, the other information, t-value, p-value, adjusted p-value, adjusted alpha value, rho, and symb are saved in csv file.

Details

Package: MBttest
Type: Package
Version: 1.0
Date: 2015-01-02
License: GPL-3

Author(s)

Yuan-De Tan
Maintainer: Yuan-De Tan <tanyuande@gmail.com>
betaparametab

References


See Also

betaparametab, betaparametVP, betaparametw, betattest, mbetattest, maplot, myheatmap, oddratio, pratio, simulat, smbetattest, mtprocedure, mtpvadjust

Examples

data(jkttcell)
betat_test(X=jkttcell[1:500,],na=3,nb=3,W=1,alpha=0.05,file="jurkat_NS_48h_tag_mbetattest.csv")

betaparametab

*Estimation of Beta Parameters alpha and beta*

Description

parameters alpha(a) and beta (b) in beta distribution are estimated by using an iteration algorithm.

Usage

betaparametab(xn, w, P, V)

Arguments

xn column vector, a set of library sizes.
w column vector, a set of weights
P proportion of counts of a gene or an isoform
V variance for proportions of counts of a gene or an isoform over m replicate libraries in a condition

Value

return parameters a and b.

Author(s)

Yuan-De Tan <tanyuande@gmail.com>

References

betaparametVP

See Also

betaparametVP, betaparametw

Examples

```r
p<-0.15
V=0.004
w<-c(0.3,0.3,0.3)
betaparametab(xn=XX,w=w,P=p,V=V)
# [1] 1.145868 6.493254
```

betaparametVP  

**Estimation of Binomial Parameters V And P in Count Data of RNA Reads**

Description

This function is used to estimate parameters P and V by optimalizing estimation of parameters: alpha and beta.

Usage

```r
betaparametVP(X, NX)
```

Arguments

- `X`: count dataset derived from m replicate libraries in one condition.
- `NX`: vector of m library sizes. Library size is sum of counts over the whole library.

Details

Count data of RNA reads are assumed to follow binomial distribution with parameters (P) and (V), while P is assumed to follow beta distribution with parameters alpha (a) and beta(b). Parameters P and V are estimated by optimal estimation of parameters a and b. The optimal method is an iteration method drived by weighting proportion of gene or isoform in each replicate library. This is a large-scale method for estimating these parameters. Estimation of parameters P and V is core of the multiple beta t-test method because P and V will be used to calculate t-value.

Value

return a list:

- `P`: N proportions estimated.
- `V`: N variances estimated.

Note

betaparametVP requires functions betaparametab and betaparametw.
Author(s)
Yuan-DE Tan <tanyuande@gmail.com>

References

See Also
betaparametab, betaparametw

Examples
```r
data(jktcell)
X<-jktcell[1:500,]
na<-3
nb<-3
cn<-length(X[,1])
nr<-length(X[,1])
XC<-X[,1:(cn-na-nb)]
XX<-X[, (cn-na-nb+1):cn]
n<-na+nb
XA<-XX[,1:na]
SA<-apply(XA,2,sum)
PA<-betaparametVP(XA,SA)
```

betaparametw Estimation of proportion weights

Description
Function betaparametw is used to calculate weight.

Usage
betaparametw(xn, a, b)

Arguments
- **xn**  
  vector of m library sizes. Library size is sum of counts over the whole library.
- **a**  
  parameter alpha in beta distribution derived from output of function betaparametab
- **b**  
  parameter beta in beta distribution derived from output of function betaparametab
Details

alpha and beta are used to calculate weight. Then weight is in turn used to correct bias of estimation of alpha and beta in betaparametab function.

Value

return weight(W)

Author(s)

Yuan-De Tan <tanyuande@gmail.com>

References


See Also

betaparametab,betaparametVP.

Examples

```r
a<-1.1458
b<-6.4932
betaparametw(xn=XX,a=a,b=b)
# [1] 0.3333333 0.3333333 0.3333333
```

betattest

*Beta t-test*

Description

Beta t-test and degree of freedom for each gene or isoform are calculated in this function.

Usage

`betattest(X, na, nb)`

Arguments

<table>
<thead>
<tr>
<th>X</th>
<th>count data of RNA reads containing N genes (or isoforms).</th>
</tr>
</thead>
<tbody>
<tr>
<td>na</td>
<td>number of replicate libraries in condition A</td>
</tr>
<tr>
<td>nb</td>
<td>number of replicate libraries in condition B</td>
</tr>
</tbody>
</table>
Details

In beta t-test,

\[ t = \frac{(P_A - P_B)}{\sqrt{(V_A + V_B)}} \]

where \( P_A \) and \( P_B \) are proportions of a gene or an isoform in conditions A and B. \( V_A \) and \( V_B \) are variances estimated in conditions A and B. They are outputted by betaparametVP.

Value

return two lists:

- \( t \) t-value list.
- \( df \) df list. df is degree of freedom.

Note

If pooled standard error is zero, then the t-value is not defined and set to be zero.

Author(s)

Yuan-De Tan <tanyuande@gmail.com>

References


See Also

- pratio, oddratio.

Examples

data(jkttcell)
X<-jkttcell[1:1000,]
na<3
nb<3
cn<ncol(X)
rn<nrow(X)
XC<X[,1:(cn-na-nb)]
XX<X[, (cn-na-nb+1):cn]
betattest<-betattest(XX,na=3,nb=3)
The Transcriptomic data and t-test results.

dat

dat

Description

t-value and rho are results output by mbttest.

Usage

data("dat")

Format

A data frame with 13409 observations on the following 16 variables.

- tagid  a numeric vector
- geneid a numeric vector
- name   a string vector
- chr    a string vector
- strand a character vector
- pos    a numeric vector
- anno   a string vector
- Jurk.NS.A a numeric vector
- Jurk.NS.B a numeric vector
- Jurk.NS.C a numeric vector
- Jurk.48h.A a numeric vector
- Jurk.48h.B a numeric vector
- Jurk.48h.C a numeric vector
- beta_t  a numeric vector
- rho     a numeric vector
- symb    a character vector

Details

t-values (beta_t) and means over all replicate libraries in two conditions are used to make MA plot. The count data of DE isoforms are selected by symb = "+" and W(omega) and used to make heatmap using myheatmap function.

Value

ID, information, count data of RNA reads, t-value and rho-value, symbol.
**References**


**Examples**

```r
data(dat)
## maybe str(dat) ; plot(dat) ...
```

---

**jkttcell**  
*Jurkat T-cell Transcritomic Data*

**Description**

The data are transcriptomic count data of *RNA* reads generated by next generation sequencing from Jurkat T-cells.

**Usage**

```r
data("jkttcell")
```

**Format**

A data frame with 13409 observations on the following 13 variables.

- `tagid` a numeric vector
- `geneid` a numeric vector
- `name` a string vector
- `chr` a string vector
- `strand` a character vector
- `pos` a numeric vector
- `anno` a string vector
- `Jurk.NS.A` a numeric vector
- `Jurk.NS.B` a numeric vector
- `Jurk.NS.C` a numeric vector
- `Jurk.48h.A` a numeric vector
- `Jurk.48h.B` a numeric vector
- `Jurk.48h.C` a numeric vector
The data are count data generated by next generation sequencing from Jurkat T-cells. The T-cells were treated by resting and stimulating with CD3/CD28 for 48 hours. The data have 7 columns for the information of poly(A) site: tagid, geneid, gene name, chromosome, strand, poly(A) site position, poly(A) site annotation and 6 columns for data: Jurk.NS.A, Jurk.NS.B, Jurk.NS.C, Jurk.48h.A, Jurk.48h.B, Jurk.48h.C. where NS means Normal state and 48h means 48 hours after CD3/CD28 stimulation of T-cells. 13409 RNA isoforms were detected to have alternative poly(A) sites.

Value

ID, information, count data of RNA reads

Source

Real transcriptomic count data

References


Examples

data(jkttcell)
## maybe str(jkttcell) ; plot(jkttcell) ...

maplot

**MA plot of t-values Against Log Mean**

Description

This function is to display MA plot of t-value against log mean.

Usage

maplot(dat, r1, r2, TT, matitle)

Arguments

dat object outputted by mbetattest containing data ordered by absolution of t-value and rho (ρ).

r1 number of replicate libraries in condition 1.

r2 number of replicate libraries in condition 2.

TT a numeric parameter that gives truncate value of t-values.

matitle string for MA plot title.
Details
In MA plot, t-value is in y-axis and log mean in x-axis; Black points gathered nearby zero along log mean are genes without differential expressions or differential splicings while red points scattered out of black points are those of being differentially expressed or differentially spliced.

Value
no return value

Author(s)
Yuan-De Tan <tanyuande@gmail.com>

Examples
```
data(dat)
maplot(dat=dat, r1=3, r2=3, TT=350, matitle="MA plot")
maplot(dat=dat, r1=3, r2=3, TT=50, matitle="MA plot")
```

---

**mbetatptest**  
Performance of multiple beta t-test on simulated data

Description
This function is to perform multiple beta t-test method on real data. The result lists geneid or isoformid, gene name, the other information, t-value, p-value, adjusted p-value, adjusted alpha value, rho ($\rho$), and symb. All these lists are ordered by absolution of t-values.

Usage
```
mbetatptest(X, na, nb, W, alpha=0.05, file)
```

Arguments
- **X** count data of RNA reads with na replicates in condition A ans nb replicates in condition B.
- **na** number of replicate libraries in condition A.
- **nb** number of replicate libraries in condition B.
- **W** numeric parameter, called omega ($\omega$) that is a constant determined by null simulation.
- **alpha** the probabilistic threshold. User can set alpha ($\alpha$)= 0.05 or 0.01 or the other values. Default value is 0.05
- **file** a csv file. User needs to give file name and specify direction path. But if user uses setwd function, drive is not necessarily specified in file.
Details

t-statistic is defined as t-statistic multiplied by (rho/omega), that is,

\[ T = t \times \frac{\rho}{\omega} \]

where

\[ t = \frac{(P_A - P_B)}{\sqrt{(V_A + V_B)}} \]
\[ \rho = \sqrt{\psi} \zeta \]

where

\[ \psi = \max(\min(X_A), \min(X_B)) \]
\[ \zeta = \log(1 + \frac{X^2 + 1}{X_A^2 + X_B^2 + 1}) \]

\( \omega \) is a constant as threshold estimated from null data.

Value

return a dat list: the data ordered by abs(t) contain information columns, data columns, t-values, rho and symb that are used to make heatmap and MAplot.

Author(s)

Yuan-De Tan <tanyuande@gmail.com>

References


See Also

smbetatest.

Examples

data(jkttcell)

dat<-mbetatest(X=jkttcell[1:1000,],na=3,nb=3,W=1,alpha=0.05,file="jurkat_NS_48h_tag_mbetatest.csv")
mtprocedure  

Multiple-Test Procedures

Description

Similiar to Benjamini-Hochberg multiple-test procedure, alpha is adjusted to be a set of values.

Usage

mtprocedure(alpha, N, C)

Arguments

- **alpha**: probabilistic threshold and is usually set to be 0.05 or 0.01. Default value is 0.05
- **N**: numeric constant, number of genes to be detected in transcriptome.
- **C**: numeric constant, it can be taken from 0 to N. C is used to choose multiple-test procedure. Default value is 0.01. This procedure is single test with C=0, Benjamini-Hochberg procedure with C=1.22 and Bonfroni procedure with C=N.

Details

This is a multiple-test procedure family including Benjamini-Hochberg procedure, Bonferroni procedure and single-test procedure. By choosing C-value, it can generat a multiple-test procedure for controlling the false discovery rate, the expected proportion of false discoveries amongst the rejected hypotheses.

Value

return a list of adjusted alpha values.

Author(s)

Yuan-De Tan <tanyuande@gmail.com>

References


See Also

- p.adjust
Examples

mtprocedure(alpha=0.5,N=200,C=1.22)
# [1] 0.007501404 0.011906423 0.015914688 0.019682621 0.023284917 0.026763656
# [7] 0.030145311 0.033447843 0.036684127 0.039863779 0.042994217 0.046081313
# .....# [175] 0.444073506 0.446322519 0.448570478 0.450817390 0.453063265 0.455308110
# [181] 0.457551933 0.459794741 0.462036542 0.464277343 0.466517153 0.468755977
# [187] 0.470993825 0.473230701 0.475466614 0.477701571 0.479935578 0.482168642
# [193] 0.484400770 0.486631969 0.488862244 0.491091603 0.493320052 0.495547597
# [199] 0.497774244 0.500000000

mtpvadjust  P-value Adjustment for Multiple Comparisons

Description

Given a set of N p-values, it returns a set of N p-values adjusted by choosing C-value

Usage

mtpvadjust(pv, C)

Arguments

pv numeric vector of p-values.
C numeric constant, the value can be taken from any number > 0 or equal to 0. C is used to choose multiple-test procedure.

Details

This is a multiple-test procedure family including Benjamini-Hochberg procedure, Bonferroni procedure and single-test procedure. By choosing C-value, it can generate a multiple-test procedure for controlling the false discovery rate, the expected proportion of false discoveries amongst the rejected hypotheses. Benjamini-Hochberg procedure is given with C=1.22, Bonferroni procedure is given with C = N and single-test procedure can be given with C=0.

Value

return a list of adjusted p-values.

Note

p-value must be ordered from the largest value to the smallest value before executing tan_pvadjust.

Author(s)

Yuan-De Tan <tanyuande@gmail.com>
myheatmap

References

See Also
p.adjust

Examples

```r
set.seed(123)
x <- rnorm(50, mean = c(rep(0, 25), rep(3, 25)))
p <- 2*pnorm(sort(-abs(x)))
round(mtpvadjust(pv=p, C=1.22),4)
# [1] 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000
# [11] 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 0.6875 0.6174 0.4588
# [21] 0.4115 0.3644 0.2216 0.1554 0.1443 0.1249 0.1027 0.0964 0.0763 0.0319
# [31] 0.0166 0.0135 0.0123 0.0096 0.0091 0.0068 0.0045 0.0041 0.0020 0.0007
# [41] 0.0004 0.0003 0.0002 0.0001 0.0001 0.0001 0.0001 0.0000 0.0000 0.0000
```

myheatmap

Description
This function is used to display heatmap of differential expressions of genes or isoforms or differential splicings of genes detected by the multiple beta t-test method in the real data.

Usage

```r
myheatmap(dat, r1, r2, W, colrs, tree, method, rwangle, clangle, maptitle)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dat</td>
<td>data outputted by mbetattest, includes information columns, data columns, t-value, rho and symbol columns;</td>
</tr>
<tr>
<td>r1</td>
<td>numeric argument: number of replicate libraries in condition 1.</td>
</tr>
<tr>
<td>r2</td>
<td>numeric argument: number of replicate libraries in condition 2</td>
</tr>
<tr>
<td>W</td>
<td>numeric argument: threshold for choosing genes or isoforms for heatmap. W value can be set to be 0 to any large number. If user sets W = 0, then the function will select all differentially expressed genes with symb=&quot;+&quot;. To choose a appropriate W, user needs to refere to rho values in the result file. Default W=1.</td>
</tr>
</tbody>
</table>

Heatmap
myheatmap

- **cols**: heatmap colors. User has 5 options: "redgreen", "greenred", "redblue", "bluered" and "heat.colors". Default cols="redgreen".

- **tree**: object of heatmap. User has four options: "both" for row and column trees,"row" for only row tree,"column" for only column tree, and "none" for no tree specified. Default tree="both".

- **method**: method to be chosen to calculate distance between columns or rows. It has four options: "euclidean", "pearson","spearman" and "kendall". The latter three are $d=1-cc$ where cc is correlation coefficients. Default="euclidean".

- **rwangle**: angle of xlab under heatmap. Default value is 30.

- **clangle**: angle of ylab. Default value is 30

- **maptitle**: string for heatmap title.

**Details**

This function uses W (omega) and "symb" to choose genes or isoforms in the data ordered by t-values and then to normalize the selected data by using z-scale. This function has multiple options to select map color, distance, cluster and x- and y-lab angles. The heatmap was designed for publication and presentation, that is, zoom of the figure can be reduced without impacting solution.

**Value**

no return value but create a heatmap.

**Note**

myheatmap requires gplots

**Author(s)**

Yuan-De Tan <tanyuande@gmail.com>

**See Also**

heatmap.2

**Examples**

```r
#require(gplots)
data(dat)

#dat<-mbetattest(X=jkttcell,na=3,nb=3,W=1,alpha=0.05,
#file="C:/mBeta_ttest/R_package/jurkat_NS_48h_tag_mbetattest.csv")

# data(mtcars)
#x <-as.matrix(mtcars)
#myheatmap(dat=x,r1=3,r2=3, maptitle="mtcars_heatmap")

myheatmap(dat=dat,r1=3,r2=3,maptitle="Jurkat T-cell heatmap2")
```
myheatmap(dat=dat,r1=3,r2=3,tree="none",maptitle="Jurkat T-cell heatmap")

oddratio

**Calculation of Zeta(ζ)**

**Description**

Zeta (ζ) is used to measure homogeneity intensity of two subdatasets. If ζ > 1, these two subdatasets have good homogeneity; otherwise, ζ < 1 indicates that two subdatasets have poor homogeneity (big noise).

**Usage**

`oddratio(XX, na, nb)`

**Arguments**

- **XX**: count data of RNA reads generated by next generation sequencing.
- **na**: number of replicate libraries in condition A.
- **nb**: number of replicate libraries in condition B.

**Details**

Zeta is defined as

\[ ζ = \log(1 + \frac{\bar{X}σ^2 + 1}{\bar{X}_Aσ^2_A + \bar{X}_Bσ^2_B + 1}) \]

where ζ is different from ψ. If two subdatasets have big a gap and good homogeneity, then ζ value has much larger than 1.

**Value**

- **oddrat**: list of zeta values

**Author(s)**

Yuan-De Tan <tanyuande@gmail.com>

**References**


**See Also**

- `pratio`
- `mbetattest`
Examples

\[
XX <- \text{matrix}(\text{NA}, 2, 8) \\
XX[1,] <- \text{c}(112, 122, 108, 127, 302, 314, 322, 328) \\
XX[2,] <- \text{c}(511, 230, 754, 335, 771, 842, 1014, 798) \\
\]

# XX
# [1,] [2,] [3,] [4,] [5,] [6,] [7,] [8,]
# [1,] 112 122 108 127 302 314 322 328
# [2,] 511 230 754 335 771 842 1014 798
oddratio(XX=XX, na=4, nb=4)

# [1] 3.9432676 0.8762017

# see example in mbetattest

---

**pratio**

*Calculation of Psi*(\( \psi \))

**Description**

Psi is also called polar ratio.

\[
\psi = \max(\frac{\min(X_A)}{\max(X_B) + 1}, \frac{\min(X_B)}{\max(X_A) + 1})
\]

**Usage**

`pratio(xx, na, nb)`

**Arguments**

- `xx` count data of RNA reads generated by next generation sequencing.
- `na` number of replicate libraries in condition A.
- `nb` number of replicate libraries in condition B.

**Details**

Psi is defined as

\[
\psi = \max(\frac{\min(X_A)}{\max(X_B) + 1}, \frac{\min(X_B)}{\max(X_A) + 1})
\]

It is used to measure overlap of two subdatasets. \( \psi > 1 \), these two subdatasets have a gap, not overlap. \( \psi < 1 \) indicates that two subdatasets overlap.

**Value**

- `pratio` pratio list
**Description**

This function uses negative binomial (NB) pseudorandom generator to create any count datasets of RNA isoform reads based on real data.

**Usage**

`simulat(yy, nci, r1, r2, p, q, A)`

**Arguments**

- `yy` 
  real count data
- `nci` 
  numeric argument: column number of information related to genes or isoforms.
- `r1` 
  numeric argument: number of replicate libraries in condition 1.
- `r2` 
  numeric argument: number of replicate libraries in condition 2.
- `p` 
  numeric argument: proportion of genes or isoforms differentially expressed. The value is in range of 0~1. Default value is 0.
- `q` 
  numeric argument: proportion of genes or isoforms artificially noised. The value is in range of 0~1. Default value is 0.
- `A` 
  numeric argument: conditional effect value. The value is larger than or equal to 0. Default value is 0.
Details

Null count data are created by using R negative binomial pseudorandom generator `rnbinom` with mu and size. Parameters mu and size are given by mean and variance drawn from real read counts of a gene or an isoforms in a condition. Condition (or treatment) effect on differential transcription of isoforms is linearly and randomly assigned to genes or isoforms. The conditional effect = AU where U is uniform variable and A is an input constant. P percent of genes or isoforms are set to be differentially expressed or differentially spliced. Q percent of genes or isoforms have technical noise. If P = 0, then simulation is null simulation, the data are null data or baseline data.

Value

Return count data.

Author(s)

Yuan-De Tan <tanyuande@gmail.com>

References


See Also

NegBinomial

Examples

data(jkttcell)
jknull<-simulat(yy=jkttcell[1:500,],nci=7,r1=3,r2=3,p=0,q=0.2,A=0)

```

skjt Simulated Null Transcriptomic data

```

Description

The dataset generated by using R negative binomial pseudorandom generator `rnbinom` is used as an example for calculating omega.

Usage

data("skjt")
Format

A data frame with 13409 observations on the following 14 variables.

- geneid  a string vector
- tagid   a numeric vector
- geneid.1 a numeric vector
- name    a string vector
- chr     a string vector
- strand  a character vector
- pos     a numeric vector
- anno    a string vector
- Jurk.NS.A a numeric vector
- Jurk.NS.B a numeric vector
- Jurk.NS.C a numeric vector
- Jurk.48h.A a numeric vector
- Jurk.48h.B a numeric vector
- Jurk.48h.C a numeric vector

Details

The dataset skjt was generated by using R negative binomial pseudorandom generator rnbinom with mu and size. Parameters mu and size are given by mean and variance drawn from real Jurkat T cell transcriptomic count data. Condition (or treatment) effect on differential transcription of isoforms was set to zero. The data have 13409 genes and 7 information columns: geneid, tagid, name, chr, strand, pos, anno, and 6 data columns: Jurk.NS.A, Jurk.NS.B, Jurk.NS.C, Jurk.48h.A, Jurk.48h.B, Jurk.48h.C.

Value

ID, information, count data of RNA reads

Source

Simulation.

References


Examples

data(skjt)
## maybe str(skjt) ; plot(skjt) ...
smbetattest

Performance of multiple Beta t-test on simulated data

Description

This function is to perform mBeta t-test with rho=1 and omega=1 on simulated data. The result lists differentially expressed genes or isoforms marked by symbol="+" and their rho values. The rho values are used to calculate omega value for performance of mBeta t-tests on the real data.

Usage

smbetattest(X, na, nb, alpha)

Arguments

- **X**: simulated count data with N genes or isoforms.
- **na**: number of replicate libraries in condition A.
- **nb**: number of replicate libraries in condition B.
- **alpha**: statistical probabilistic threshold, default value is 0.05.

Details

Before performing mbeta t-test on real data, user needs omega (w) value for the threshold of rho(ρ). To determine omega value, user is required to simulate null data having the same gene or isoform number and the same numbers of replicate libraries in two conditions and then performs mbeta t-test on the simulated null data by setting rho =1 and omega =1. To calculate accurately omega value, user needs such performance on 4-6 simulated null datasets. Manual provides method for omega calculation.

Value

Return results from multiple beta t-tests on simulated data.

Author(s)

Yuan-De Tan <tanyuande@gmail.com>

References


See Also

See Also as **mbetattest**
Examples

data(skjt)

mysim<-smbetatext(X=skjt[1:500,],na=3,nb=3,alpha=0.05)
Index

* alpha adjustment
  mtprocedure, 13
* alpha
  betaparametab, 3
* beta distribution
  betatstat, 6
* beta
  betaparametab, 3
* binomial
  simulat, 19
* datasets
  dat, 8
  jktcall, 9
  skjt, 20
* gap
  oddratio, 17
  pratio, 18
* heatmap
  myheatmap, 15
* homogeneity
  oddratio, 17
* maplot
  maplot, 10
* multiple test procedure
  mtprocedure, 13
* multiple
  mbetatstat, 11
* multiple test procedure
  mtpvadjust, 14
* negative
  simulat, 19
* overlap
  pratio, 18
* p-value adjustment
  mtpvadjust, 14
* package
  MBtest-package, 2
* proportion
  betaparametVP, 4
* simulation
  simulat, 19
  smbetatstat, 22
* t-tests
  mbetatstat, 11
* t-test
  mbetatstat, 22
* t-value
  betatstat, 6
* variance
  betaparametVP, 4
* weight
  betaparametw, 5
  betaparametab, 3, 3, 5, 6
  betaparametVP, 3, 4, 4, 6
  betaparametw, 3–5, 5
  betatstat, 3, 6
dat, 8
heatmap.2, 16
jktcall, 9
maplot, 3, 10
mbetatstat (MBtest-package), 2
mbetatstat-package (MBtest-package), 2
mbetatstat, 3, 11, 17, 19, 22
MBtest-package, 2
mtpvadjust, 3, 13
mtprocedure, 3, 14
myheatmap, 3, 15
NegBinomial, 20
oddratio, 3, 7, 17, 19
pratio, 3, 7, 17, 18
p.adjust, 13, 15
simulat, 3, 19
INDEX

skjt, 20
smbetattest, 3, 12, 22