Package 'methylMnM'

November 12, 2026

Type Package

Title detect different methylation level (DMR)

Version 1.50.0

Date 2013-04-08

Author Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Maintainer Yan Zhou<zhouy1016@163.com>

Description To give the exactly p-value and q-value of MeDIP-seq and MRE-seq data for different samples comparation.

License GPL-3

LazyLoad yes

biocViews Software, DNAMethylation, Sequencing

Depends R (>= 2.12.1), edgeR, statmod

git_url https://git.bioconductor.org/packages/methylMnM

git_branch RELEASE_3_23

git_last_commit bdd3463

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Contents

methylMnM-package 2
calcFactornew 3
calculatecount 4
calculatecount1 5
calculatecountneg 6
CNVnormal 7
countcpgbin 8
countMeDIPbin 9
countMREbin 10

countMREcpgbin 11
cpgcount 12
MnM.qvalue 13
MnM.selectDMR 14
MnM.test 15
normpdf 16
normpdft1 17
pmultinom 18
qvalue.rank 19
removeblacklist 20

Index **21**

`methylMnM-package` *MeDIP-Seq and MRE-Seq data analysis*

Description

M&M was developed for analyzing data derived from methylated DNA immunoprecipitation (MeDIP) experiments followed by sequencing (MeDIP-Seq) and the digestions with the methyl-sensitive restriction enzymes (MRE-Seq). Nevertheless, functionalities like the quality controls may be applied to other types of sequencing data (e.g. ChIP-Seq). MeDIP-MRE (methylMnM) test which combine the two differential techniques (MeDIP-seq and MRE-seq) data to detecting the differentially methylation level of CpG.

Details

Package:	methylMnM
Type:	Package
Version:	1.0
Date:	2012-12-01
License:	GPL-3
LazyLoad:	yes
Depends:	R (>= 2.12.0)

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Maintainer: Yan Zhou <zhouy1016@163.com>

References

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang, 2012

`calcFactornew` *Normalization factor.*

Description

Amends of TMM normalization for our methond.

Usage

```
calcFactornew(obs, ref, m, k, logratioTrim=.3, sumTrim=0.05,
 doWeighting=TRUE, Acutoff=-1e10)
```

Arguments

`obs`	Counts of treatment sample.
`ref`	Counts of control sample.
`m`	The number of CpG in each bin.
`k`	The number of MRE-CpG in each bin.
`logratioTrim`	amount of trim to use on log-ratios ("M" values)
`sumTrim`	amount of trim to use on the combined absolute levels ("A" values)
`doWeighting`	logical, whether to compute (asymptotic binomial precision) weights
`Acutoff`	cutoff on "A" values to use before trimming

Value

A real value larger than 0.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
 d <- matrix( rpois(1000, lambda=5), nrow=200 )
 m<-rep(1,nrow=200 )
 k<-rep(1,nrow=200 )
 f <- calcFactornew(d[,2], d[,1], m, k, logratioTrim=.3, sumTrim=0.05,
doWeighting=TRUE, Acutoff=-1e10)
```

`calculatecount` *Call C programs to R.*

Description

Call C programs to R for calculate MeDIP-seq or CpG count of each bin.

Usage

```
calculatecount(data2, data3, cpg2, cpg3, datalength, cpglength,
count=rep(0,cpglength))
```

Arguments

`data2`	Start position of each tag.
`data3`	End position of each tag.
`cpg2`	Start position of each bin.
`cpg3`	End position of each bin.
`datalength`	The number of tags
`cpglength`	The number of bins
`count`	Read count of each bin.

Value

Read count of each bin.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
 data<-matrix( 1:800, nrow=400 )
 data[,2]<-data[,1]+37
 cpg<-matrix( 1:20, nrow=10)
 cpg[,1]<-seq(0,360,length=10)
 cpg[,2]<-seq(40,400,length=10)
 f <- calculatecount(data[,1], data[,2], cpg[,1], cpg[,2], length(data[,1]),
length(cpg[,2]), count=rep(0,length(cpg[,2])))
```

`calculatecount1` *Call C programs to R.*

Description

Call C programs to R for calculate MRE-seq "+" direction count of each bin.

Usage

```
calculatecount1(data2, data3, cpg2, cpg3, datalength, cpglength,
 count=rep(0,cpglength))
```

Arguments

`data2`	Start position of each tag.
`data3`	End position of each tag.
`cpg2`	Start position of each bin.
`cpg3`	End position of each bin.
`datalength`	The number of tags
`cpglength`	The number of bins
`count`	Count of MRE-seq "+" direction of each bin.

Value

Count of MRE-seq "+" direction of each bin.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
 data<-matrix( 1:400, nrow=200 )
 cpg<-matrix( 1:40, nrow=20)
 cpg[,1]<-seq(0,380,length=20)
 cpg[,2]<-seq(20,400,length=20)
 f <- calculatecount1(data[,1], data[,2], cpg[,1], cpg[,2], length(data[,1]),
length(cpg[,2]), count=rep(0,length(cpg[,2])))
```

`calculatecountneg` *Call C programs to R.*

Description

Call C programs to R for calculate MRE-seq "-" direction count of each bin.

Usage

```
calculatecountneg(data2, data3, cpg2, cpg3, datalength, cpglength,
count=rep(0,cpglength))
```

Arguments

`data2`	Start position of each tag.
`data3`	End position of each tag.
`cpg2`	Start position of each bin.
`cpg3`	End position of each bin.
`datalength`	The number of tags
`cpglength`	The number of bins
`count`	Count of MRE-seq "-" direction of each bin.

Value

Count of MRE-seq "-" direction of each bin.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
 data<-matrix( 1:400, nrow=200 )
 cpg<-matrix( 1:40, nrow=20)
 cpg[,1]<-seq(0,380,length=20)
 cpg[,2]<-seq(20,400,length=20)
 f <-calculatecountneg(data[,1], data[,2], cpg[,1], cpg[,2], length(data[,1]),
length(cpg[,2]), count=rep(0,length(cpg[,2])))
```

CNVnormal *Normalize copy number variation (CNV).*

Description

The function is used to normalize copy number variation.

Usage

```
CNVnormal(CNVfile,bincount)
```

Arguments

CNVfile	The path of copy number variation file.
bincount	Count of all bins.

Value

Count of all bins after CNV normalization.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
datafile<-system.file("extdata",  package = "methylMnM")
filepath<-datafile[1]
file1<-paste(filepath,"/all_CpGsite_chr18.txt",sep="")
CpGsite<-read.table(file1, header=FALSE,skip=0, nrows=200, as.is=TRUE)
winbin<-CpGsite[1:100,1:4]
winbin[,2]<-seq(0,49500,500)
winbin[,3]<-winbin[,2]+500
winbin[,4]<-rpois(100, lambda=5)
cnv<-winbin[1:5,]
cnv[,2]<-c(0,10000,20000,30000,40000)
cnv[,3]<-cnv[,2]+10000
cnv[,4]<-c(1.2,1.6,1,2,1)
CNVfile<-paste(setwd(getwd()), "/CNVfile.bed", sep = "")
write.table(cnv, CNVfile, quote=FALSE, row.names =FALSE,col.names =FALSE)
f<-CNVnormal(CNVfile,winbin)
```

countcpgbin *Compute the total CpG number of each bin with each CpG site.*

Description

The function is used to compute the total CpG number of each bin with each CpG site.

Usage

```
countcpgbin(file.cpgsite,file.blacklist=NULL,file.bin=NULL, writefile=NULL,
 reportfile=NULL, binlength=500)
```

Arguments

file.cpgsite	The path of cpg site file or sequence tag file.
file.blacklist	The path of blacklist file (If we do not use the file, there will be defaulted as NULL).
file.bin	The path of all cpg bin file. For computing the number of sequence tag of each window, we use the file as a normalization window position. (If we do not use the file, there will be defaulted as NULL).
writefile	The path of output results. (If writefile=NULL, there will return the results back to main program.)
reportfile	The path of output results of bin length, the number of bin, total reads before processing and total reads after processing.
binlength	The length of each window.(Defaulted length is 500).

Value

The CpG site should include at least three columns "chromosome", "start position" and "end position". The output file is include four columns, that is "chromosome", "start position", "end position" and "CpG count". Also, the function output a report for some parameters.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
datafile<-system.file("extdata",  package = "methylMnM")
filepath<-datafile[1]
file.cpgsite<-paste(filepath,"/all_CpGsite_chr18.txt",sep="")
f<-countcpgbin(file.cpgsite, binlength=5000)
```

countMeDIPbin *Compute the total MeDIP-seq number of each bin.*

Description

The function is used to compute the total MeDIP-seq number of each bin.

Usage

```
countMeDIPbin (file.Medipsite,file.blacklist=NULL,file.bin=NULL,
file.CNV=NULL, writefile=NULL, reportfile=NULL, binlength=500)
```

Arguments

file.Medipsite	The path of MeDIP-seq site file or sequence tag file.
file.blacklist	The path of blacklist file (If we do not use the file, there will be defaulted as NULL).
file.bin	The path of all bins file. For computing the number of sequence tag of each window, we use the file as a normalization window position. (If we do not use the file, there will be defaulted as NULL).
file.CNV	If need, we should input CNV file to normalize count of each bin.
writefile	The path of output results. (If writefile=NULL, there will return the results back to main program.)
reportfile	The path of output results of bin length, the number of bin, total reads before processing and total reads after processing.
binlength	The length of each window.(Defaulted length is 500).

Value

The MeDIP-seq site should include at least three columns "chromosome", "start position" and "end position". The output file is include four columns, that is "chromosome", "start position", "end position" and "MeDIP-seq count". Also, the function output a report for some parameters.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
datafile<-system.file("extdata",  package = "methylMnM")
filepath<-datafile[1]
file.Medipsite<-paste(filepath,"/all_CpGsite_chr18.txt",sep="")
f<-countMeDIPbin(file.Medipsite, binlength=5000)
```

countMREbin *Compute the total MRE-seq number of each bin.*

Description

The function is used to compute the total MRE-seq number of each bin.

Usage

```
countMREbin(file.MREsite,file.blacklist=NULL, file.bin=NULL,
file.CNV=NULL, cutoff=0,writefile=NULL,
reportfile=NULL, binlength=500)
```

Arguments

file.MREsite	The path of MRE-seq sites file.
file.blacklist	The path of blacklist file (If we do not use the file, there will be defaulted as NULL).
file.bin	The path of all bin file. For computing the number of sequence tag of each window, we use the file as a normalization window position. (If we do not use the file, there will be defaulted as NULL).
file.CNV	If need, we should input CNV file to normalize count of each bin.
cutoff	The critical value of PCR. (If we do not use the critical value, there will be defaulted as 0.)
writefile	The path of output results. (If writefile=NULL, there will return the results back to main program.)
reportfile	The path of output results of bin length, the number of bin, total reads before processing and total reads after processing.
binlength	The length of each window.(Defaulted length is 500).

Value

The MRE-seq sites should include at least three columns "chromosome", "start position" and "end position". The output file is include four columns, that is "chromosome", "start position", "end position" and "MRE-seq count". Also, the function output a report for some parameters.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
datafile<-system.file("extdata",  package = "methylMnM")
filepath<-datafile[1]
file.MREsite<-paste(filepath,"/all_CpGsite_chr18.txt",sep="")
f<-countMREbin(file.MREsite, binlength=5000)
```

countMREcpgbin *Compute the MRE CpG number of each bin with MRE CpG sites.*

Description

The function is used to compute the MRE CpG number of each bin with MRE CpG sites. MRE CpG is some specific CpGs in genome-wide, such as "CCGG", "GCGC" and "CCGC". The specific CpG number is directly bound up with each experiment.

Usage

```
countMREcpgbin(mrecpg.site,file.allcpgsite,file.bin=NULL,
writefile=NULL, binlength=500)
```

Arguments

mrecpg.site	The data of mreCpG site.
file.allcpgsite	The path of all cpg site file or sequence tag file.
file.bin	The path of all bins file. For computing the number of sequence tag of each window, we use the file as a normalize window position. (If we do not use the file, there will be defaulted as NULL).
writefile	The path of output result. (If writefile=NULL, there will return the results back to main program)
binlength	The length of each window. (Defaulted length is 500)

Value

The output file is include four columns, that is "chromosome", "start position", "end position" and "MRE CpG count".

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
  datafile<-system.file("extdata",  package = "methylMnM")
  filepath<-datafile[1]
  file<-paste(filepath,"/three_Mre_CpGsite_chr18.txt",sep="")
  file1<-paste(filepath,"/all_CpGsite_chr18.txt",sep="")
  five_Mre_CpGsite<-read.table(file, header=FALSE, as.is=TRUE)
  f<-countMREcpgbin(mrecpg.site=five_Mre_CpGsite[1:1000,],
file.allcpgsite=file1,binlength=5000)
```

cpgcount *Call C programs to R.*

Description

Call C programs to R for calculate which CpG are contained in MRE-CpG.

Usage

```
cpgcount(data2, data3, cpg2, cpg3, datalength, cpglength,
count=rep(0,cpglength))
```

Arguments

data2	Start position of each MRE-CpG.
data3	End position of each MRE-CpG.
cpg2	Start position of each CpG.
cpg3	End position of each CpG.
datalength	The number of MRE-CpG.
cpglength	The number of MRE-CpG.
count	MRE-CpG count of each CpG.

Value

MRE-CpG count of each CpG.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
 cpg<-matrix( 1:800, nrow=400 )
 cpg[,2]<-cpg[,1]+2
 data<-cpg[3:100,]
 data[,1]<-data[,1]-1
 data[,2]<-data[,2]+1
 f <- cpgcount(data[,1], data[,2], cpg[,1], cpg[,2], length(data[,1]),
length(cpg[,2]), count=rep(0,length(cpg[,2])))
```

MnM.qvalue *Estimate the q-values for a given set of p-values*

Description

The function is used to estimate the q-values for a given set of p-values. The q-value of a test measures the proportion of false positives incurred (called the false discovery rate) when that particular test is called significant.

Usage

```
MnM.qvalue(datafile,writefile=NULL,reportfile=NULL)
```

Arguments

datafile	Input data of p-values file (Including all input)
writefile	The file path of output result. (If writefile=NULL,there will return the results back to main program)
reportfile	The path of output results of bin length, the number of bin, total reads before processing and total reads after processing.

Value

The output file is just add a q-value column to the input file.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
  datafile<-system.file("extdata",  package = "methylMnM")
  filepath<-datafile[1]
  file1<-paste(filepath,"/all_CpGsite_chr18.txt",sep="")
  CpGsite<-read.table(file1, header=FALSE,skip=0, nrows=200, as.is=TRUE)
  winbin<-CpGsite[1:100,1:4]
  winbin[,2]<-seq(0,49500,500)
  winbin[,3]<-winbin[,2]+500
  count<-matrix(rpois(600, lambda=5), nrow=100 )
  count[,6]<-count[,5]
  pvalue<-runif(100, min=0, max=1)
  ts<-rnorm(100, mean=0, sd=1)
  cpgpq<-cbind(winbin[,1:3],count,pvalue,ts)
  colnames(cpgpq)=c("chr", "chrSt","chrEnd","Medip1","Medip2","MRE1",
"MRE2","cg","mrecg","pvalue",'Ts')
  pvaluefile<-paste(setwd(getwd()), "/pvalue.bed", sep = "")
  write.table(cpgpq, pvaluefile,sep="\t", quote=FALSE,row.names =FALSE)
  f<-MnM.qvalue(datafile=pvaluefile)
```

MnM.selectDMR *Select significants of each comparation.*

Description

The function is used to select significants of each comparation.

Usage

```
MnM.selectDMR(frames = NULL, up =1.45, down = 1/1.45, p.value.MM = 0.01,
 p.value.SAGE = 0.01,q.value = 0.01,cutoff="q-value",  quant= 0.6)
```

Arguments

frames	The input dataset and q-values of each bin.
up	The ratio of Medip1/Medip2 should be larger than "up" value if we call it significant.
down	The ratio of Medip1/Medip2 should be smaller than "down" value if we call it significant.
p.value.MM	The p-value of the bin which use MM test should be smaller than "p.value.MM" if we call it significant.
p.value.SAGE	The p-value of the bin which use SAGE test should be smaller than "p.value.SAGE" if we call it significant.
q.value	The q-value of the bin should be smaller than "q.value" if we call it significant.
cutoff	We should use p-value or q-value cutoff to detect DMRs (If cutoff="q-value", then we use q-value to detect DMRs, else we use p-value).
quant	The rank of absolute value of (Medip1-Medip2) should be larger than "quant" value if we call it significant.

Value

The DMRs of the comparation.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
datafile<-system.file("extdata",  package = "methylMnM")
filepath<-datafile[1]
file1<-paste(filepath,"/all_CpGsite_chr18.txt",sep="")
CpGsite<-read.table(file1, header=FALSE,skip=0, nrows=200, as.is=TRUE)
winbin<-CpGsite[1:100,1:4]
winbin[,2]<-seq(0,49500,500)
```

```
winbin[,3]<-winbin[,2]+500
count<-matrix(rpois(600, lambda=5), nrow=100 )
count[,6]<-count[,5]
pvalue<-runif(100, min=0, max=1)
ts<-rnorm(100, mean=0, sd=1)
cpgpq<-cbind(winbin[,1:3],count,pvalue,ts)
colnames(cpgpq)=c("chr", "chrSt","chrEnd","Medip1","Medip2","MRE1",
"MRE2","cg","mrecg","pvalue",'Ts')
f<-MnM.selectDMR(frames=cpgpq, p.value.MM = 0.1,
p.value.SAGE = 0.1,cutoff="p-value")
```

`MnM.test` *Compute p-value of each bin.*

Description

The function is used to compute p-value of each bin.

Usage

```
MnM.test(file.dataset=NULL,chrstring=NULL,file.cpgbin=NULL,
file.mrecpgbin=NULL,writefile=NULL,reportfile=NULL,
mreratio=3/7,method="XXYY", psd=2,mkadded=1,a=1e-16,
cut=100,top=500)
```

Arguments

`file.dataset`	The files path of sample. (datafile should be c(datafile1,datafile2,datafile3,datafile4), where datafile1 and datafile2 are path of Medip-seq data, datafile3 and datafile4 are path of MRE-seq data).
`chrstring`	The chromosome should be test.
`file.cpgbin`	The file path of all CpG number of each bin.
`file.mrecpgbin`	The file path of MRE-CpG number of each window (If NULL, mrecpgfile will equal to cpgfile).
`writefile`	The file path of output result. (If writefile=NULL, there will return the results back to main program)
`reportfile`	The path of output results of bin length, the number of bin, total reads before processing and total reads after processing.
`mreratio`	The ratio of total unmethylation level with total methylation level (Defaulted mreratio is 3/7).
`method`	Option different data for the test.
`psd`	The parameters of pseudo count, which pseudo count added to Medip-seq and MRE-seq count.
`mkadded`	Added to all CpG and MRE CpG (We set psd=2 and mkadded=1 as defaulted for robust)

a	Cut-off for recalculating p-value with multinomial distribution when normal p-values smaller than a and the sum of observations smaller than top.
cut	Cut-off for recalculating p-value with multinomial distribution when the sum of observations smaller than cut.
top	Cut-off for recalculating p-value with multinomial distribution when normal p-values smaller than a and the sum of observations smaller than top.

Value

The output file "writefile" will own eleven columns, that is, "chr", "chrSt", "chrEnd", "Medip1", "Medip2", "MRE1", "MRE2", "cg", "mrecg", "pvalue" and "plus-minus". We also output a report file which will include parameters "s1/s2", "s3/s4", "N1", "N2", "N3", "N4", "c1", "c2", "Number of windows" and "Spend time".

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
datafile<-system.file("extdata",  package = "methylMnM")
filepath<-datafile[1]
file1<-paste(filepath,"/all_CpGsite_chr18.txt",sep="")
CpGsite<-read.table(file1, header=FALSE,skip=0, nrows=200, as.is=TRUE)
winbin<-CpGsite[1:100,1:4]
winbin[,2]<-seq(0,49500,500)
winbin[,3]<-winbin[,2]+500
winbin[,4]<-rpois(100, lambda=5)
winbinfile1<-paste(setwd(getwd()), "/winbinfile1.bed", sep = "")
write.table(winbin, winbinfile1,sep="\t", quote=FALSE, row.names =FALSE)
winbin1<-winbin
winbin1[,4]<-winbin[,4]+20
winbinfile2<-paste(setwd(getwd()), "/winbinfile2.bed", sep = "")
write.table(winbin1, winbinfile2,sep="\t", quote=FALSE, row.names =FALSE)
datafile<-c(winbinfile1,winbinfile2)
cpg<-winbin
cpg[,4]<-rpois(100, lambda=12)
cpgfile<-paste(setwd(getwd()), "/cpgfile.bed", sep = "")
write.table(cpg, cpgfile, sep="\t", quote=FALSE, row.names =FALSE)
f<-MnM.test(file.dataset=datafile,file.cpgbin=cpgfile)
```

---

normpdf *Compute p-value with normal distribution.*

Description

The function is used to compute p-value with normal distribution.

Usage

```
normpdf(t,n,p,c1,c2)
```

Arguments

t	Statistic.
n	The sum of MeDIP-seq count and MRE-seq count of each bin of two samples.
p	The probability in multinomial distribution.
c1	A constant to balance MeDIP-seq of sample 1 and sample 2.
c2	A constant to balance MRE-seq of sample 1 and sample 2.

Value

p-values.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
t<-0.1
n<-200
p<-c(0.25,0.25,0.25,0.25)
c1<-1
c2<-1
f<-normpdf(t,n,p,c1,c2)
```

---

normpdft1 *Compute p-value with normal distribution.*

Description

The function is used to compute p-value with normal distribution.

Usage

```
normpdft1(t,n,p,c1,c2)
```

Arguments

t	Statistic.
n	The sum of MeDIP-seq count and MRE-seq count of each bin of two samples.
p	The probability in multinomial distribution.
c1	A constant to balance MeDIP-seq of sample 1 and sample 2.
c2	A constant to balance MRE-seq of sample 1 and sample 2.

Value

statistic of a bin.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
t<-0.1
n<-200
p<-c(0.25,0.25,0.25,0.25)
c1<-1
c2<-1
f<-normpdft1(t,n,p,c1,c2)
```

`pmultinom` *Call C programs to R.*

Description

Call C programs to R for calculate p-value of each bin with multinomial distribution.

Usage

```
pmultinom(T, SIZE,length, P1, P2, P3, P4, C1, C2,
 pvalue=rep(0,length(T)))
```

Arguments

`T`	Statistic.
`SIZE`	The sum of MeDIP-seq count and MRE-seq count of each bin of two samples.
`length`	The number of bins.
`P1`	The probability of MeDIP-seq of sample 1 in multinomial distribution.
`P2`	The probability of MeDIP-seq of sample 2 in multinomial distribution.
`P3`	The probability of MRE-seq of sample 1 in multinomial distribution.
`P4`	The probability of MRE-seq of sample 2 in multinomial distribution.
`C1`	A constant to balance MeDIP-seq of sample 1 and sample 2.
`C2`	A constant to balance MRE-seq of sample 1 and sample 2.
`pvalue`	p-values of windows.

Value

p-value.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
T<-4
SIZE<-200
p<-c(0.25,0.25,0.25,0.25)
c1<-1
c2<-1
length<-1
f<-pmultinom(T, SIZE,length, p[1], p[2], p[3], p[4], c1,
 c2, pvalue=rep(0,length(T)))
```

---

qvalue.rank *Rank values.*

Description

The function is used to rank values.

Usage

```
qvalue.rank(x)
```

Arguments

x	Value.

Value

Ranked values.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
x<-c(4,2,50,42,80,9)
qvalue.rank(x)
```

removeblacklist *Remove blacklist.*

Description

The function is used to remove blacklist which we are not interest.

Usage

```
removeblacklist(file2,cpg)
```

Arguments

file2	The path of blacklist site file.
cpg	All bins.

Value

All bins except blacklist region.

Author(s)

Yan Zhou, Bo Zhang, Nan Lin, BaoXue Zhang and Ting Wang

Examples

```
 datafile<-system.file("extdata",  package = "methylMnM")
 filepath<-datafile[1]
 file1<-paste(filepath,"/all_CpGsite_chr18.txt",sep="")
 CpGsite<-read.table(file1, header=FALSE,skip=0, nrows=200, as.is=TRUE)
 winbin<-CpGsite[1:100,1:4]
 winbin[,2]<-seq(0,49500,500)
 winbin[,3]<-winbin[,2]+500
 winbin[,4]<-rpois(100, lambda=5)
 blacklist<-winbin[1:5,]
 blacklist[,2]<-c(0,10000,20000,30000,40000)
 blacklist[,3]<-blacklist[,2]+1000
 blacklistfile<-paste(setwd(getwd()), "/blacklist.bed", sep = "")
 write.table(blacklist, blacklistfile, quote=FALSE,
row.names =FALSE,col.names =FALSE)
 f<-removeblacklist(blacklistfile,winbin)
```

Index

∗ **package**
methylMnM-package, 2

calcFactornew, 3
calculatecount, 4
calculatecount1, 5
calculatecountneg, 6
CNVnormal, 7
countcpgbin, 8
countMeDIPbin, 9
countMREbin, 10
countMREcpgbin, 11
cpgcount, 12

methylMnM *(*methylMnM-package*)*, 2
methylMnM-package, 2
MnM.qvalue, 13
MnM.selectDMR, 14
MnM.test, 15

normpdf, 16
normpdft1, 17

pmultinom, 18

qvalue.rank, 19

removeblacklist, 20