Package ‘ENmix’

May 29, 2024

Type    Package
Title   Quality control and analysis tools for Illumina DNA methylation BeadChip
Version 1.40.1
Description Tools for quanlity control, analysis and visualizion of Illumina DNA methylation array data.
License Artistic-2.0
biocViews DNAmethylation, Preprocessing, QualityControl, TwoChannel, Microarray, OneChannel, MethylationArray, BatchEffect, Normalization, DataImport, Regression, PrincipalComponent, Epigenetics, MultiChannel, DifferentialMethylation, ImmunoOncology
Depends parallel, doParallel, foreach, SummarizedExperiment, stats, R (>= 3.5.0)
Imports grDevices, graphics, matrixStats, methods, utils, irlba, geneplotter, impute, minfi, RPMI, illuminaio, dynamicTreeCut, IRanges, gtools, Biobase, ExperimentHub, AnnotationHub, genefilter, gplots, quadprog, S4Vectors
Suggests minfiData, RUnit, BiocGenerics, BiocStyle, knitr, rmarkdown
VignetteBuilder knitr
Encoding UTF-8
URL  https://github.com/Bioconductor/ENmix
BugReports https://github.com/Bioconductor/ENmix/issues
NeedsCompilation no
git_url https://git.bioconductor.org/packages/ENmix
git_branch RELEASE_3_19
git_last_commit 2c48368
git_last_commit_date 2024-05-15
Repository Bioconductor 3.19
Date/Publication 2024-05-29
Author  Zongli Xu [cre, aut],
         Liang Niu [aut],
         Jack Taylor [ctb]

Maintainer Zongli Xu <xuz@niehs.nih.gov>

Contents

B2M ................................................................. 3
calcdetP .......................................................... 4
combp ............................................................... 5
crlsva ............................................................. 6
dupicc ............................................................... 7
estimateCellProp ................................. 8
freqpoly ............................................................ 10
getB ................................................................. 11
getCGinfo ........................................................... 12
getmeth ............................................................. 12
ipdmr ................................................................. 13
M2B ................................................................. 14
methDataSet-class ................................. 15
methscore ........................................................... 16
methyAge ............................................................. 18
mhtplot .............................................................. 20
mpreprocess .................................................. 21
multifreqpoly ................................................ 23
nmode ................................................................. 24
norm.quantile .................................................. 25
oxBS.MLE .......................................................... 26
p.qqplot ............................................................. 27
pcrplot ............................................................... 28
plotCtrl .............................................................. 29
predSex .............................................................. 30
preprocessENmix ........................................ 31
qcfilter ............................................................. 33
QCinfo ............................................................... 34
rcp ................................................................. 36
rcp2 ................................................................. 37
readidat ............................................................ 38
readmanifest ............................................... 40
relic ................................................................. 41
repicc ............................................................... 42
rgDataSet-class ........................................... 44
rm.cgsuffix ................................................... 44
simubed ............................................................ 45

Index  47
B2M

Converting methylation beta value to M value.

Description

Convert methylation beta value to M value.

Usage

B2M(x)

Arguments

x

An numeric matrix with values between 0 and 1

Details

Methylation beta value is calculated as beta=M/(M+U+a). M is methylated intensity, U is unmethylated intensity, and a is a constant offset (by default, a=100). M value is calculated as M=log2((M+a)/(U+a)). M or U is usually greater than 1000, so a is negligible for most probes. if a=0, then M=log2(beta)/(1-beta).

Value

A matrix of M values

Author(s)

Zongli Xu

Examples

if (require(minfiData)){
  path <- file.path(find.package("minfiData"),"extdata")
  rgSet <- readidat(path = path,recursive = TRUE)
  mdat=getmeth(rgSet)
  beta=getB(mdat,"Illumina")
  m=B2M(beta)
}
calcdetP

To calculate detection P values

Description

Calculation of detection P values based on negative internal control probes or out of the band (oob) probes

Usage

calcdetP(rgSet,detPtype = "negative")

Arguments

rgSet An object of class rgDataSet
detPtype Calculation of detection P values based on negative internal control ("negative") probes or out of the band ("oob") probes

Value

An numerical matrix of detection P values, with row for CpGs and column for samples

Author(s)

Zongli Xu

References

Wanding Zhou et al. SeSAMe: reducing artifactual detection of DNA methylation by Infinium BeadChips in genomic deletions, Nucleic Acids Research, 2018

Examples

path <- file.path(find.package("minfiData"),"extdata")
rgSet <- readidat(path = path,recursive = TRUE)
detp=calcdetP(rgSet,detPtype = "negative")
detp2=calcdetP(rgSet,detPtype = "oob")
Identification of differentially methylated regions

Description
To identify differentially methylated regions using a modified comb-p method

Usage
```r
combp(data, dist.cutoff=1000, bin.size=310, seed=0.01,
      region_plot=TRUE, mht_plot=TRUE, nCores=10, verbose=TRUE)
```

Arguments
- `data`: A data frame with column names "chr", "start", "end", "p", and "probe", indicating chromosome (1, 2, 3,..., X, Y), chromosome start and end position, P value and probe name.
- `dist.cutoff`: Maximum distance in base pair to combine adjacent DMRs.
- `bin.size`: Bin size for autocorrelation calculation.
- `seed`: FDR significance threshold for initial selection of DMR region.
- `region_plot`: If TRUE, regional plots will be generated.
- `mht_plot`: If TRUE, mahattan plot will be generated.
- `nCores`: Number of computer cores will be used in calculation.
- `verbose`: If TRUE, detailed running information will be printed.

Details
The input should be a data frame with column names "chr", "start", "end", "p", and "probe", indicating chromosome number, start position, end position, P value and probe name. The function uses a modified comb-p method to identify differentially methylated regions. DMR results will be stored in a file with name resu_combp.csv. If plot options were selected, two figure files will be generated: mht.jpg and region_plot.pdf.

Author(s)
Liang Niu, Zongli Xu

References
- Pedersen BS1, Schwartz DA, Yang IV, Kechris KJ. Comb-p: software for combining, analyzing, grouping and correcting spatially correlated P-values. Bioinformatics 2012
- Zongli Xu, Changchun Xie, Jack A. Taylor, Liang Niu, ipDMR: Identification of differentially methylated regions with interval p-values, Bioinformatics 2020
 ctrlsva

Non-negative internal control surrogate variables

Description

Surrogate variables derived based on intensity data for non-negative internal control probes.

Usage

```r
ctrlsva(rgSet, percvar = 0.95, npc = 1, flag = 1)
```

Arguments

- `rgSet`: An object of class `rgDataSet` or `RGChannelSet`.
- `percvar`: Minimum percentage of data variations can be explained by surrogate variables, range from 0 to 1, default is 0.95.
- `npc`: Number of surrogate variables, default is 1.
- `flag`: 1: select number of surrogate variables based on argument `percvar`; 2: select number of surrogate variables based on argument `npc`.

Value

The function will return a numerical matrix with columns indicating surrogate variables and rows corresponding to samples. These variables can be used in association analysis to adjust for experimental batch effects.

Author(s)

Zongli Xu

References


Examples

```r
if (require(minfiData)) {
  path <- file.path(find.package("minfiData"), "extdata")
  rgSet <- readidat(path = path, recursive = TRUE)
  sva <- ctrlsva(rgSet)
}
```
**dupicc**

**Evaluation of measurement reliability using duplicate samples**

**Description**

The function can be used to evaluate duplicate samples by calculating: 1) centered/un-centered Pearson’s correlation coefficient between duplicates; 2) absolute difference between duplicates; 3) ICC for each CpG probes using oneway or two-way model.

**Usage**

```
dupicc(dat, dupid, mvalue=FALSE, center=TRUE, nCores=2, qcflag=FALSE, qc=NULL, 
detPthre=0.05, nbthre=3, skipicc=FALSE, corfig=FALSE, model="oneway")
```

**Arguments**

- **dat**  
  Methylation beta value matrix

- **dupid**  
  A data frame with two variables, id1 and id2. The two ids in each row indicate a duplicate pair. These ids should be the same with column names of the input methylation matrix

- **mvalue**  
  If TRUE, the beta value will be converted to M value for calculation of ICC

- **center**  
  If TRUE, the methylation beta values will be centered for each CpG before calculation of ICC or correlation

- **nCores**  
  Number of cores will be used for calculation of ICC

- **qcflag**  
  Whether to perform QC before calculation of ICC

- **qc**  
  QC object from function QCinfo

- **detPthre**  
  If qcflag=TRUE, the methylation values with detection P value higher than the threshold will be removed before calculation

- **nbthre**  
  If qcflag=TRUE, the methylation values with number of bead smaller than the threshold will be removed

- **skipicc**  
  If TRUE, ICC calculation will be skipped

- **corfig**  
  If TRUE, a figure will be generated to demonstrate correlations within duplicates or within non-duplicates

- **model**  
  Using "oneway" or "two-way" model to calculate ICC

**Value**

- **icc**  
  a data frame containing ICC and P values for each probe

- **dupcor**  
  a data frame containing Pearson’s correlation and averaged absolute difference between duplicates.

**Author(s)**

Zongli Xu
estimateCellProp

**Cell type proportion estimator**

**Description**
To estimates relative proportion of underlying cell types in a sample based on reference methylation data of pure cell types.

**Usage**
```r
estimateCellProp(userdata, refdata="FlowSorted.Blood.450k", cellTypes=NULL, nonnegative = TRUE, nProbes=50, normalize=TRUE, refplot=FALSE)
```

**Arguments**
- `userdata` The input can be `rgDataSet`, `methDataSet`, `MethylSet`, `RGChannelSet` or methylation beta value matrix.
- `cellTypes` Specific set of cell type data in reference data will be used for deconvolution. if "NULL" all cell types data will be used. see details for possible cell types
- `normalize` TRUE or FALSE, if TRUE, quantile normalization on methylated and unmethylated intensities will be performed.
- `nonnegative` TRUE or FALSE. If TRUE, the estimated proportions will be constrained to nonnegative values
- `nProbes` Number of best probes for each cell types will be used for the estimation.
- `refplot` TRUE or FALSE. IF TRUE, refdata distribution and heatmap will be plotted for inspection of reference dataset.

**Examples**
```r
if (require(minfiData)){
  path <- file.path(find.package("minfiData"),"extdata")
  rgSet <- readidat(path = path, recursive = TRUE)
  mdat=getmeth(rgSet)
  beta=getB(mdat,"Illumina")
  dupidx=data.frame(id1=c("5723646052_R02C02","5723646052_R04C01","5723646052_R05C02"), id2=c("5723646053_R04C02","5723646053_R05C02","5723646053_R06C02"))
  iccresu<-dupicc(dat=beta,dupid=dupidx)
}
```

**References**
estimateCellProp

Details

This function use the method of Houseman et al (2012) to estimate cell type proportions based on reference DNA methylation data.

The following reference datasets can be used to assist the estimation. User should select a reference most resemble to user's data in tissue, age, and array type.

FlowSorted.Blood.450k: consisting of 450K methylation data for 60 blood samples from 6 male adults. Six samples for each of the cell types: Bcell CD4T CD8T Eos Gran Mono Neu NK PBMC WBC; See Reinius et al. 2012 for details.


FlowSorted.DLPFC.450k: consisting of 450K methylation data for 58 brain tissue samples from 29 individuals. 15 females and 14 males, 6 Africans and 23 Caucasians, age range from 13 to 79. 29 samples for each of the cell types: NeuN_neg and NeuN_pos. See Guintivano et al. 2013 for details.


Value

The output is a data frame composed of the estimates of cell type proportions with columns indicate cell types and rows indicate samples.

Author(s)

Zongli Xu

References


Examples

```r
require(minfiData)
path <- file.path(find.package("minfiData"),"extdata")
#based on rgDataset
rgSet <- readidat(path = path,recursive = TRUE)
celltype=estimateCellProp(userdata=rgSet,refdata="FlowSorted.Blood.450k", nonnegative = TRUE,normalize=TRUE)
```
# using methDataSet
qc=QCinfo(rgSet)
mdat<-preprocessENmix(rgSet, bgParaEst="oob", dyeCorr="RELIC",
QCinfo=qc, nCores=6)
celltype=estimateCellProp(userdata=mdat,refdata="FlowSorted.Blood.450k",
nonnegative = TRUE,normalize=TRUE)
mdat<-norm.quantile(mdat, method="quantile1")
# using beta value
beta<-rcp(mdat,qcscore=qc)
celltype=estimateCellProp(userdata=beta,refdata="FlowSorted.Blood.450k",
nonnegative = TRUE)

freqpoly

**Description**

Similar to histogram, frequency polygon plot can be used to inspect data distribution.

**Usage**

```
freqpoly(mat, nbreaks=15, col="black", xlab="", ylab="Frequency",
type="l",append=FALSE,...)
```

**Arguments**

- **mat** A numeric vector
- **nbbreaks** Number of bins for frequency counting
- **col** color code
- **xlab** x-axis label
- **ylab** y-axis label
- **type** character indicating the type of plotting; actually any of the 'type's as in 'plot.default'.
- **append** TRUE or FALSE, whether to create a new figure or append to the current figure.
- **...** Further arguments that get passed to the function "plot"

**Value**

Frequency polygon plot.

**Author(s)**

Zongli Xu

**References**

**getB**

*Extract methylation Beta values.*

**Examples**

```r
freqpoly(rnorm(1000))
```

**Description**

Extract Methylation Beta value, Beta = Meth / (Meth + Unmeth + offset)

**Usage**

```r
getB(mdat,type="Illumina",offset=100)
```

**Arguments**

- `mdat`: An object of class `methDataSet` or `MethylSet`.
- `type`: `type="Illumina"` sets `offset=100` as per Genome Studio software.
- `offset`: Regularization factor in calculating beta ratio, 100 in default

**Value**

Methylation Beta value = Meth / (Meth + Unmeth + offset). Meth is methylated intensity matrix, Unmeth is unmethylated intensity matrix.

**Author(s)**

Zongli Xu

**Examples**

```r
if (require(minfiData)){
  path <- file.path(find.package("minfiData"),"extdata")
  rgSet <- readidat(path = path,recursive = TRUE)
  mdat=getmeth(rgSet)
  beta=getB(mdat,"Illumina")
}
```
getCGinfo

CpG probe annotation information

Description

Extract CpG probe annotation information from an rgDataSet

Usage

getCGinfo(rgSet, type="IandII")

Arguments

rgSet
An object of class rgDataSet

type
One of the following options "I","II","IandII","ctrl", indicating type I, type II
type I & II or control probes type

Value

An object of data frame class

Author(s)

Zongli Xu

Examples

require(minfiData)
path <- file.path(find.package("minfiData"),"extdata")
#based on rgDataset
rgSet <- readidat(path = path,recursive = TRUE)
cginfo=getCGInfo(rgSet,type="IandII")

getmeth

Create a methDataSt

Description

To create a methDataSt based on a rgDataset

Usage

gmeth(rgSet)
**Arguments**

- **rgSet**: An object of class `rgDataSets`

**Details**

CpG annotation information is incorporated in the output `methDataSets` object, and can be extracted using command `rowData()`.

**Value**

An object of class `methDataSets`

**Author(s)**

Zongli Xu

**Examples**

```r
require(minfiData)
path <- file.path(find.package("minfiData"),"extdata")
# based on rgDataset
rgSet <- readidat(path = path,recursive = TRUE)
meth=getmeth(rgSet)
meth
cginfo=rowData(meth)
```

---

**Description**

To identify differentially methylated regions using an interval P value method

**Usage**

```
ipdmr(data,include.all.sig.sites=TRUE,dist.cutoff=1000,bin.size=50,
seed=0.05,region_plot=TRUE,mht_plot=TRUE,verbose=TRUE)
```

**Arguments**

- **data**: A data frame with colname name "chr","start","end","p" and "probe", indicating chromosome (1,2,3,...,X,Y), chromosome start and end position, P value and probe names
- **include.all.sig.sites**: Whether to use CpG singletons in calculation of FDR
dist.cutoff: Maximum distance in base pair to combine adjacent DMRs, and the maximum distance between CpGs where auto-correlation will be calculated.

bin.size: bin size for autocorrelation calculation.

seed: FDR threshold for initial selection of DMR regions.

region_plot: If TRUE, regional plots will be produced for each DMR.

mht_plot: If TRUE, a p-value mahattan plot with marked DMRs will be produced.

verbose: Whether to output detailed information.

Details

The input should be a data frame with column names "chr", "start", "end", "p" and "probe", indicating chromosome, start and end position, P value and probe name. The function will use a novel interval p value method to identify differentially methylated regions. DMR results will be stored in a file with name resu_ipdmr.csv. If plot options were selected, two figure files will be generated: mht.jpg and region_plot.pdf.

Author(s)

Liang Niu, Zongli Xu

References

Zongli Xu, Changchun Xie, Jack A. Taylor, Liang Niu, ipDMR: Identification of differentially methylated regions with interval p-values, Bioinformatics 2020

Examples

dat=simubed()
names(dat)
#seed=0.1 is only for demonstration purpose, it should be smaller than 0.05 or 0.01 in actual study.
ipdmr(data=dat,seed=0.1) #seed=0.1

M2B

Converting methylation M value to beta value.

Description

Converting methylation M value to methylation beta value.

Usage

M2B(x)

Arguments

x: An numeric matrix.
Details

Methylation beta value is calculated as \( \beta = M/(M+U+a) \). \( M \) is methylated intensity, \( U \) is unmethylated intensity, and \( a \) is a constant offset (by default, \( a=100 \)). \( M \) value is calculated as \( M = \log_2((M+a)/(U+a)) \). \( M \) or \( U \) is usually greater than 1000, so \( a \) is negligible for most probes. If \( a=0 \), then \( \beta = 2^M/(2^M+1) \).

Value

A matrix of methylation Beta values.

Author(s)

Zongli Xu

Examples

```r
if (require(minfiData)){
  path <- file.path(find.package("minfiData"),"extdata")
  rgSet <- readidat(path = path,recursive = TRUE)
  mdat=getmeth(rgSet)
  beta=getB(mdat,"Illumina")
  m=B2M(beta)
  b=M2B(m)
}
```

methDataSet-class

Class "methDataSet"

Description

A class for storing Illumina methylation array methylated and unmethylated intensity data, and CpG annotation information.

Usage

```r
methDataSet(Meth = new("matrix"), Unmeth = new("matrix"),
            rowData=new("DataFrame"),...)
```

Arguments

- **Meth**: A matrix of methylated intensity values with row for CpGs and column for samples
- **Unmeth**: A matrix of unmethylated intensity values with row for CpGs and column for samples
- **rowData**: A dataframe contains CpG annotation information
- **...**: Other arguments for class SummarizedExperiment
methscore

Details

CpG annotation information is incorporated in the object, and can be extracted using command rowData

Value

An object of class methDataSet

Examples

showClass("methDataSet")

methscore

DNA Methylation predictors

Description

To calculate various methylation predictors, including DNA methylation age, plasma protein levels, exposures etc.

Usage

methscore(datMeth,datPheno=NULL,fastImputation=FALSE,normalize=TRUE,
GrimAgeComponent=NULL,UserRef=NULL,ForceUserRef=FALSE)

Arguments

datMeth Methylation beta value matrix with CpG names(row names) and sample ids(column names).

datPheno Phenotype data, must include columns SampleID, Age (year) and Female (0:male,1:female). Age and Female information are required to calculate DNAmFitAge and PC-Clocks

fastImputation If "TRUE" mean methylation values will used for imputation, if "FALSE", KNN nearest neighbor method will be used.

normalize TRUE or FALSE, if TRUE, user data will be normalized to a reference data using a modified RCP method

GrimAgeComponent A data frame of grimage component from methylation age online calculator (https://dnamage.clockfoundation.org/user/login). It must include the following variables, ("SampleID","DNAmADM","DNAmB2M","DNAmCystatinC","DNAmGDF15","DNAmLeptin","DNAmPACKYRS","DNAmPAI1","DNAmTIMP1","DNAmGrimAge"). If the file is not provided, bAge and DNAmFitAge will be calculated using PC Grimage components

UserRef User provided methylation reference data will be used for some predictors (see details), must include variables cg and meth_mean

ForceUserRef If TRUE, UserRef will be forced to use in normalization and imputation for all estimates
Details

The original publications (see references) provided reference methylation mean values for most methylation predictors, and thus these values were used directly for normalization and imputations. For the following predictors without reference value in their original publications, PEDBE, EpiToc, EpiToc2, Zhang10CpG, Horvath2, MiAge, DNAmTL, PEDBE, GACPC, GARPC, GARRPC, Bohlin and Knight, we used Sister Study data (~5000 samples) to derive a set of reference methylation values. If UserRef was provided, it will be used to replace the Sister Study reference values for these predictors. If set ForceUserRef=TRUE, UserRef will be used for all predictors in normalization and CpG imputation.

Value

A data frame with rows for samples and columns for methylation predictors. Output file "summary_CpG_count.csv" has information about CpG counts and reference for each predictor.

Author(s)

Zongli Xu

References


---

**Examples**

```r
require(minfiData)
pth <- file.path(find.package("minfiData"),"extdata")
#based on rgDataset
rgSet <- readidat(pth,recursive = TRUE)
meth=getmeth(rgSet)
beta=getB(meth)
pheno=data.frame(SampleID=colnames(beta),Age=c(23,45,52,36,58,43),Female=c(1,0,1,1,0,1))
mage=methscore(datMeth=beta,datPheno=pheno)
```

---

**methyAge**

* Methylation Age estimator

**Description**

To calculate Methylation Age using Hovath, Hannum or PhenoAge methods and pace of aging DunedinPACE.

**Usage**

`methyAge(beta,fastImputation=FALSE,normalize=TRUE,nCores=2)`
methyAge

Arguments

beta Methylation beta value matrix with CpG names (row names) and sample ids (column names).

fastImputation If "TRUE" reference methylation values will used for imputation, if "FALSE", KNN nearest neighbor method will be used.

normalize TRUE or FALSE, if TRUE, Hovath modified BMIQ method will be used to perform normalization.

nCores Number of cores will be used for normalization

Value

A data frame with rows for sample and columns for types of methylation age.

Author(s)

Zongli Xu

References


Examples

```
require(minfiData)
path <- file.path(find.package("minfiData"),"extdata")
# based on rgDataset
rgSet <- readidat(path = path, recursive = TRUE)
meth=getmeth(rgSet)
beta=getB(meth)
mage=methyAge(beta)
```
mhtplot  

Description

P value manhattan plot

Usage

mhtplot(probe=NULL, chr=NULL, pos=NULL, p=NULL, color="bg", sigthre=NULL, sigthre2=NULL, threlty=c(2,1), markprobe=NULL, markcolor="red", outf="mht", outfmt="jpg", reducesize=0,...)

Arguments

probe  probe name  
chr   Chromosome, 1,2,...,22,X,Y  
pos  Chromosome positions  
p    P values  
color  Color scheme of manhattan plot, "bg" indicate "black and gray"  
sigthre  P value of significant threshold line  
sigthre2  P value of second significant threshold line  
threlty  Threshold line type, default is c(2,1)  
markprobe  A list of CpGs to be marked out  
markcolor  Color code for marked probe, "red" in default  
outf  figure file name, default "mht"  
outfmt  Output figure file format, "jpg" or "eps"  
reducesize  A positive interger, larger the value, smaller the eps file size. Smaller file size is achieved by skipping some densely packed data points  
... Arguments pass to plot

Author(s)

Zongli Xu

Examples

dat=simubed()

thre1=1E-100

dat$fdr=p.adjust(mrgd$P, method="BH")

if(sum(dat$fdr<0.05)>0){thre1=max(dat$p[dat$fdr<0.05])}
description

The pipeline performs background correction, dye bias correction, inter-array normalization and probe type bias correction for HumanMethylation 450 and MethylationEPIC BeadChip data. It removes or mitigates background noise and systematic experimental bias. It also performs quality controls, identifying and excluding low quality samples and probes, removing low quality and outlier values, and performing imputation.

usage

```r
mpreprocess(rgSet,nCores=2,bgParaEst="oob",dyeCorr="RELIC", qc=TRUE,qnorm=TRUE,qmethod="quantile1", fqcfilter=FALSE,rmcr=FALSE,impute=FALSE)
```

arguments

- **rgSet**: An object of class `rgDataSet`, `methDataSet`, `RGChannelSetExtended`, `RGChannelSet` or `MethylSet`.
- **nCores**: Number of cores will be used for computation.
- **bgParaEst**: Method to estimate background normal distribution parameters. Possible options: "oob", "est", or "neg".
- **dyeCorr**: Dye bias correction, "mean": correction based on averaged red/green ratio; or "RELIC": correction with RELIC method; or "none": no dye bias correction. The default is RELIC
- **qc**: If TRUE, QC will be performed. Low quality samples and CpGs will be excluded before background correction.
- **qnorm**: If TRUE, inter-array quantile normalization will be performed.
- **qmethod**: Quantile normalization method. This should be one of the following strings: "quantile1", "quantile2", or "quantile3". See details in function `norm.quantile`.
- **fqcfilter**: If TRUE, outlier and low quality values will be filtered out.
- **rmcr**: TRUE: excluded rows and columns with more than 5% of missing values. FALSE is in default
- **impute**: Whether to impute missing values. If TRUE, k-nearest neighbor’s methods will be used for imputation. FALSE is in default.
Details

Fuction `mpreprocess` is a pipeline that perform methylation data preprocessing and quality controls using functions: preprocessENmix, norm.quantile, rcp, QCinfo and qcfilter. More customized preprocessing steps can be achieved using the individual functions, see user’s guide.

Value

A methylation beta value matrix with rows for CpGs and columns for samples.

Author(s)

Zongli Xu

References


See Also

Package `minfi` for classes `RGChannelSet` and `MethylSet`

Examples

```r
if (require(minfiData)) {
  #rgDataSet as input
  path <- file.path(find.package("minfiData"), "extdata")
  rgSet <- readidat(path = path, recursive = TRUE)
  beta=mpreprocess(rgSet,nCores=6,qc=TRUE,fqcfilter=TRUE,rmcr=TRUE,impute=TRUE)

  #methDataSet as input
  mdat=getmeth(rgSet)
  beta=mpreprocess(mdat,nCores=6)

  #RGChannelSet as input
  beta=mpreprocess(RGsetEx,nCores=6)

  #RGChannelSetExtended as input
  sheet <- read.metharray.sheet(file.path(find.package("minfiData"), "extdata"),
                              pattern = "csv$")
  rgSet <- read.metharray.exp(targets = sheet, extended = TRUE)
  beta=mpreprocess(rgSet,nCores=6,qc=TRUE,fqcfilter=TRUE,rmcr=TRUE,impute=TRUE)

  #MethylSet as input
  mdat=preprocessRaw(rgSet)
```
Multifreqpoly

Description

Produce Frequency polygon plot for each column of a numeric data matrix. Similar to multidensity function, the plot can be used to inspect data distribution but with much faster speed.

Usage

```
multifreqpoly(mat, nbreaks=100, col=1:ncol(mat), xlab="", ylab="Frequency", legend = list(x = "top", fill=col, legend = if(is.null(colnames(mat))) paste(1:ncol(mat)) else colnames(mat)),append=FALSE,...)
```

Arguments

- **mat**: A numeric matrix
- **nbreaks**: The number of bins for frequency counting
- **col**: Line plot color code, the length should be equal to the number of columns in mat
- **xlab**: x-axis lable
- **ylab**: y-axis lable
- **legend**: A list of arguments that get passed to the function "legend"
- **append**: TRUE or FALSE, whether to create a new figure or append to the current figure.
- **...**: Further arguments that get passed to the function "plot"

Value

Frequency polygon plots.

Author(s)

Zongli Xu

References

nmode

Examples

```r
x = matrix(rnorm(10000), nrow = 2000, ncol = 5)
multifreqpoly(x, nbreaks = 15, legend = list(x = "topright", fill = 1:ncol(x), legend = paste("V", 1:5, sep = ""))

if (require(minfiData)) {
  path <- file.path(find.package("minfiData"), "extdata")
  rgSet <- readidat(path = path, recursive = TRUE)
  mraw <- getmeth(rgSet)
  beta <- getB(mraw)
  jpeg("dist_raw.jpg")
multifreqpoly(beta, col = 1:ncol(beta))
  dev.off()
}
```

nmode  

Estimating number of mode for each row of data

Description

Due to SNPs in CpG probe region or other unknow factors, methylation beta values for some CpGs
have multimodal distribution. This function is to identify this type of probes (so called gap probes)
with obvious multimoal distribution.

Usage

```r
nmode(x, minN = 3, modedist = 0.2, nCores = 1)
```

Arguments

- **x**: A methylation beta value matrix with row for probes and column for samples.
- **minN**: Minimum number of data points at each cluster
- **modedist**: Minimum distance between adjacent modes
- **nCores**: Number of cores used for computation

Details

This function uses an empirical approach to estimate number of modes in methylation beta value for
each CpG probe. By default, the function requires the distance between modes have to be greater
than 0.2 in methylation beta value, and each mode clusters should has at least 3 data points or 5%
of data points whichever is greater.

Value

A vector of integers indicating number of modes. Gap probes are probes with number of mode
greater than 1.
norm.quantile

Author(s)
Zongli Xu

References

Examples

```r
if (require(minfiData)) {
  mdat <- preprocessRaw(RGsetEx)
  beta=getBeta(mdat, "Illumina")
  nmode=nmode(beta, minN = 3, modedist=0.2, nCores = 5)

  path <- file.path(find.package("minfiData"), "extdata")
  rgSet <- readidat(path = path, recursive = TRUE)
  mdat <- getmeth(rgSet)
  beta=getB(mdat)
  nmode=nmode(beta, minN = 3, modedist=0.2, nCores = 5)
}
```

---

**norm.quantile**

*Quantile normalization.*

**Description**

Quantile normalization of methylation intensity data across samples for Illumina Infinium Human-Methylation 450 and MethylationEPIC BeadChip.

**Usage**

```r
norm.quantile(mdat, method = "quantile1")
```

**Arguments**

- `mdat`: An object of class `methDataSet` or `MethylSet`.
- `method`: Quantile normalization method: "quantile1", "quantile2", or "quantile3".

**Details**

By default, `method = "quantile1"`, which will separately quantile normalize Methylated or Unmethylated intensities for Infinium I or II probes. The "quantile2" will quantile normalize combined Methylated or Unmethylated intensities for Infinium I or II probes. The "quantile3" will quantile normalize combined Methylated or Unmethylated intensities for Infinium I and II probes together.
Value

The output is an an object of class methDataSet or MethylSet.

Author(s)

Zongli Xu

References


Examples

```r
# for methDataSet
path <- file.path(find.package("minfiData"),"extdata")
rgSet <- readidat(path = path,recursive = TRUE)
mdat<-preprocessENmix(rgSet, bgParaEst="oob",nCores=6)
mdatq<-norm.quantile(mdat, method="quantile1")

# for MethySet
if (require(minfiData)) {
  mdat=preprocessENmix(RGsetEx,bgParaEst="oob",nCores=6)
  mdatq=norm.quantile(mdat,method="quantile1")
}
```

Description

Maximum Likelihood Estimate (MLE) of 5-methylcytosine (5mC) and 5-hydroxymethylcytosine (5hmC) using sequencing/array data from paired bisulfite and oxidative bisulfite treated DNA experiments.

Usage

`oxBS.MLE(beta.BS,beta.oxBS,N.BS,N.oxBS)`

Arguments

- `beta.BS`: A matrix of methylation beta values (proportion of methylated sites estimated as methylated intensity over total intensity) obtained from bisulfite (BS) experiments
- `beta.oxBS`: A matrix of methylation beta values obtained from oxidative bisulfite (oxBS) experiments
N. BS  A matrix of total signals (sum of methylated and unmethylated intensity values) from BS experiments
N. oxBS A matrix of total signals from oxBS experiments

Details
For all the inputs (beta.BS, beta.oxBS, N.BS and N.oxBS), the rows should be corresponding to CpG loci and the columns should be corresponding to samples. The row/column names in all four matrices should be the same. For a specific CpG of a sample, if any one of the four values (beta.BS, beta.oxBS, N.BS and N.oxBS) is NA, or N.BS is zero, or N.oxBS is zero, the MLE of both 5mC and 5hmC levels will be set as NA.

Value
The output is a list with two elements: 5mC: a matrix of estimated 5mC levels. 5hmC: a matrix for estimated 5hmC levels.

Author(s)
Liang Niu and Zongli Xu

References
Zongli Xu, Jack A. Taylor, Yuet-Kin Leung, Shuk-Mei Ho and Liang Niu, oxBS-MLE: an efficient method to estimate 5-methylcytosine and 5-hydroxymethylcytosine in paired bisulfite and oxidative bisulfite treated DNA, Bioinformatics. 2016

Examples
# load example data
load(system.file("oxBS.MLE.RData",package="ENmix"))
# run oxBS.MLE
resu<-oxBS.MLE(beta.BS,beta.oxBS,N.BS,N.oxBS)
dim(resu[["5mC"]])
dim(resu[["5hmC"]])

---

p.qqplot

P value Q-Q plot

Description
P value Q-Q plot with optional confidence interval

Usage
p.qqplot(pvalues,outf="qq",outfmt="jpg",draw.conf=TRUE,
conf.col="lightgray",conf.alpha=.95,pch=20,col="black",reducesize=0,...)
Arguments

- **pvalues**: A numeric vector of P values
- **outf**: Figure file name, default "qq"
- **outfmt**: Output figure file format, "jpg" or "eps"
- **draw.conf**: Whether to draw confidence interval of expected P values under NULL hypothesis
- **conf.col**: Color code of confidence interval
- **conf.alpha**: Confidence interval range, 0.95 in default
- **pch**: Point type code
- **col**: Point color code
- **reducesize**: A positive integer, larger the value, smaller the eps file size. Smaller file size is achieved by skipping some densely packed data points
- ... Arguments pass to plot

Details

P value Q-Q plot with optional confidence interval

Author(s)

Zongli Xu

Examples

```r
dat=simubed()
p.qqplot(pvalues=dat$p,draw.conf=TRUE,outf="qq_try",outfmt="jpg")
```

Description

First, principal component analysis will be performed in the standadized input data matrix (standardized for each row/CpG), and then the specified number of top principal components (which explain most data variation) will be used to perform linear regression with each specified variable. Regression P values will be plotted for exploration of methylation data variance structure or identification of possible confounding variables in association analysis.

Usage

```r
cprplot(beta, cov, npc=50)
```
Arguments

beta: A methylation beta value matrix with rows for probes and columns for samples. The input matrix should not contain any missing value.

cov: A data frame of covariates. Categorical variables should be converted to factors. The number of rows should equal to the number of columns in beta matrix.

npc: The number of top ranked principal components to be plotted.

Value

A jpeg figure "svdscreeplot.jpg" to show the variations explained by each principal component. A jpeg figure "pcr_diag.jpg" to show association strength between principal components and covariates with cell colors indicating different levels of association P values.

Author(s)

Zongli Xu

References


Examples

```r
if (require(minfiData)) {
  mdat <- preprocessRaw(RGsetEx)
  beta=getBeta(mdat, "Illumina")
  group=pData(mdat)$Sample_Group
  slide=factor(pData(mdat)$Slide)
  cov=data.frame(group,slide)
  pcrplot(beta,cov,npc=6)
}
```

Description

The function will generate a series of internal control plots that are similar to the plots from Illumina GenomeStudio software. Users should refer to GenomeStudio online guide to interpret these figures. These figures can be used to check data quality and experimental procedures.

Usage

`plotCtrl(rgSet,IDorder=NULL)`
predSex

Estimating sample sex

Description

Estimating sample sex based on methylation data

Usage

predSex(mdat, cutoff = 2)

Arguments

mdat An object of class MethData or rgDataSet.
cutoff The difference in log2 total intensity between X and Y chromosomes
**Details**

Estimation of sex is based on the difference of log2 median total intensity measures on the X and Y chromosomes.

**Author(s)**

Zongli Xu

**Examples**

```r
if (require(minfiData)) {
  path <- file.path(find.package("minfiData"),"extdata")
  rgSet <- readidat(path = path, recursive = TRUE)
  sex = predSex(rgSet)
}
```

**Description**

The ENmix background correction for HumanMethylation 450 and MethylationEPIC BeadChip. ENmix models methylation signal intensities with a flexible exponential-normal mixture distribution, and models background noise with a truncated normal distribution. ENmix will split BeadChip intensity data into 6 parts and separately model methylated and unmethylated intensities, 2 different color channels and 2 different probe types.

**Usage**

```r
preprocessENmix(rgSet, bgParaEst = "oob", dyeCorr="RELIC", QCinfo=NULL, exQCsample=TRUE, exQCcpg=TRUE, exSample=NULL, exCpG=NULL, nCores = 2)
```

**Arguments**

- `rgSet`: An object of class `rgDataSet`, `methDataSet`, `RGChannelSetExtended`, `RGChannelSet` or `MethylSet`.
- `bgParaEst`: Method to estimate background normal distribution parameters. Options are: "oob", "est", or "neg".
- `dyeCorr`: Dye bias correction method, "mean": correction based on averaged red/green ratio, or "RELIC": correction with RELIC method (default method), or "none": no dye bias correction.
- `QCinfo`: If QCinfo object from function QCinfo() is provided, low quality samples (if `exQCsample=TRUE`) and CpGs (if `exQCcpg=TRUE`) will be excluded before background correction.
- `exQCsample`: If TRUE, low quality samples listed in QCinfo will be excluded.
preprocessENmix

- **exQCcpg**: If TRUE, low quality CpGs listed in QCinfo will be excluded.
- **exSample**: User specified samples to be excluded before background correction
- **exCpG**: User specified probes to be excluded before background correction
- **nCores**: Number of cores will be used for computation

**Details**

By default, ENmix will use out-of-band Infinium I intensities ("oob") to estimate normal distribution parameters for modeling background noise. Option "est" will use combined methylated and unmethylated intensities to estimate background distribution parameters separately for each color channel and each probe type. Option "neg" will use 600 chip internal controls probes to estimate background distribution parameters. If rgSet if a MethylSet, then only option "est" can be selected.

**Value**

An object of class same with input data

**Author(s)**

Zongli Xu and Liang Niu

**References**


**Examples**

```r
if (require(minfiData)) {
  #rgDataSet as input
  path <- file.path(find.package("minfiData"),"extdata")
  rgSet <- readidat(path = path,recursive = TRUE)
  #quality control information
  qc<-QCinfo(rgSet)
  #further excluding samples which are not in the qc$badsample list
  ex_id=c("5723646053_R04C02")
  #further excluding cpgs which are not in the qc$badCpG list
  ex_cg=c("cg00000622", "cg00001245", "cg00001261")
  mdat=preprocessENmix(rgSet,QCinfo=qc,exSample=ex_id,exCpG=ex_cg,nCores=6)
  
  #RGChannelSet as input
  mdat=preprocessENmix(RGsetEx,nCores=6)
}
```
**qcfilter**

Filtering out low quality and outlier data

**Description**

Outlier was defined as values smaller than 3 times IQR from the lower quartile or greater than 3 times IQR from the upper quartile. If data quality information were provided, low quality data points will be set as missing data first before looking for outliers. All outliers and low quality data will be set as miss in output matrix. If set imput=TRUE, imputation will be performed using k-nearest neighbors method to impute all missing values.

**Usage**

```r
cqfilter(mat,qcscore=NULL,rmoutlier=TRUE,byrow=TRUE,detPthre=0.000001,nbthre=3,
          rmcr=FALSE,rthre=0.05,cthre=0.05,impute=FALSE,imputebyrow=TRUE,fastimpute=FALSE,...)
```

**Arguments**

- **mat**
  - An numeric matrix containing methylation beta values
- **qcscore**
  - If the data quality information (the output from function QCinfo) were provided, low quality data points as defined by detection p value threshold (detPthre) and number of bead threshold (nbthre) will be set as missing value.
- **rmoutlier**
  - if TRUE, outliers data points will be set as missing data NA.
- **byrow**
  - TRUE: Looking for outliers row by row, or FALSE: column by column.
- **detPthre**
  - Detection P value threshold to define low quality data points, detPthre=0.000001 in default.
- **nbthre**
  - Number of beads threshold to define low quality data points, nbthre=3 in default.
- **rmcr**
  - TRUE: exclude rows and columns with too many missing values as defined by rthre and cthre. FALSE is in default
- **rthre**
  - Minimum of percentage of missing values for a row to be excluded
- **cthre**
  - Minimum of percentage of missing values for a column to be excluded
- **impute**
  - If TRUE, k-nearest neighbors methods will used for imputation.
- **imputebyrow**
  - TRUE: impute missing values using similar values in row, or FALSE: in column
- **fastimpute**
  - If TRUE, probe median will be used for fast imputation.
- **...**
  - Arguments to be passed to the function impute.knn in R package "impute"

**Value**

- The output is an numeric matrix.

**Author(s)**

Zongli Xu
References


Examples

```r
if (require(minfiData)) {
  path <- file.path(find.package("minfiData"), "extdata")
  rgSet <- readidat(path = path, recursive = TRUE)
  qc = QCinfo(rgSet)
  mdat = preprocessENmix(rgSet, QCinfo = qc, nCores = 6)
  beta = rcp(mdat)
  # filter out outliers data points only
  b1 = qcfilter(beta)
  # filter out low quality and outlier data points
  b2 = qcfilter(beta, qcscore = qc)
  # filter out low quality and outlier values, remove rows and columns with
  # too many missing values
  b3 = qcfilter(beta, qcscore = qc, rmcr = TRUE)
  # filter out low quality and outlier values, remove rows and columns with
  # too many missing values, and then do imputation
  b3 = qcfilter(beta, qcscore = qc, rmcr = TRUE, impute = TRUE)
}
```

QCinfo

**Extract QC information**

**Description**

Extract information for data quality control: detection P values, number of beads and averaged bisulfite conversion intensity. The function can also identify low quality samples and probes, as well as outlier samples in total intensity or beta value distribution.

**Usage**

```r
QCinfo(rgSet, detPthre=0.000001, detPtype="negative", nbthre=3, samplethre=0.05,
       CpGthre=0.05, bisulthre=NULL, outlier=TRUE, distplot=TRUE)
```

**Arguments**

- `rgSet` An object of class `rgDataSet`, or `RGChannelSetExtended`
- `detPthre` Detection P value threshold to identify low quality data point
- `detPtype` Calculate detection P values based on negative internal control ("negative") probes or out of the band ("oob") probes
- `nbthre` Number of bead threshold to identify data point of low quality
**samplethre**  
Threshold to identify samples with low data quality, the percentage of low quality methylation data points across probes for each sample

**CpGthre**  
Threshold to identify probes with low data quality, percentage of low quality methylation data points across samples for each probe

**bisulthre**  
Threshold of bisulfite intensity for identification of low quality samples. By default, Mean - 3 x SD of sample bisulfite control intensities will be used as a threshold.

**outlier**  
If TRUE, outlier samples in total intensity or beta value distribution will be identified and classified as bad samples.

**distplot**  
TRUE or FALSE, whether to produce beta value distribution plots before and after QC.

**Value**

- detP: a matrix of detection P values
- nbead: a matrix for number of beads
- bisul: a vector of averaged intensities for bisulfite conversion controls per sample
- badsample: a list of low quality or outlier samples
- badCpG: a list of low quality CpGs
- outlier_sample: a list of outlier samples in methylation beta value or total intensity distribution.

**Figure "qc_sample.jpg"**: scatter plot of Percent of low quality data per sample vs. Average bisulfite conversion intensity

**Figure "qc_CpG.jpg"**: histogram for Percent of low quality data per CpG.

**Figure "freqpolygon_beta_beforeQC.jpg"**: distribution plot of input data, samples colored in red are "bad" samples, list in badsample, including samples with low data quality and outlier in methylation beta value or total intensity.

**Figure "freqpolygon_beta_afterQC.jpg"**: distribution plot input data after filtering "bad" samples.

**Author(s)**

Zongli Xu

**References**


**Examples**

```r
if (require(minfiData)) {
  #rgDataSet as input
  path <- file.path(find.package("minfiData"),"extdata")
  rgSet <- readidat(path = path,recursive = TRUE)
  qc=QCinfo(rgSet)
}


```
#RGChannelSetExtended as input
sheet <- read.metharray.sheet(file.path(find.package("minfiData"),"extdata"),
pattern = "csv")
rgSet <- read.metharray.exp(targets = sheet,extended = TRUE)
qc<-QCinfo(rgSet)
```

---

**rcp**

*Illumina methylation array probe type bias correction*

**Description**

Probe design type bias correction using Regression on Correlated Probes (RCP) method

**Usage**

```r
crp(mdat, dist=25, quantile.grid=seq(0.001,0.999,by=0.001), qcscore = NULL,
    nbthre=3, detPthre=0.000001)
```

**Arguments**

- `mdat` An object of class `methDataSet` or `MethylSet`.
- `dist` Maximum distance in base pair between type I and type II probe pairs for regression calibration
- `quantile.grid` Quantile grid used in linear regression
- `qcscore` Data quality information object, the output from function QCinfo. If the object is provided, low quality data points as defined by detection p value threshold (`detPthre`) or number of bead threshold (`nbthre`) will be set as missing values.
- `detPthre` Detection P value threshold to define low quality data points
- `nbthre` Number of beads threshold to define low quality data points, `nbthre=3` in default.

**Details**

The function will first identify type I and type II probe pairs within a specified distance, and then perform linear regression calibration between the probe types. With the estimates the function will then adjust type II data using type I data as references.

**Value**

A beta value matrix

**Author(s)**

Liang Niu, Zongli Xu
**rcp2**

*Modified RCP method*

**Description**

Modified RCP method to normalize user's data to a list of reference values

**Usage**

```r
rcp2(datMeth, reference, quantile.grid=seq(0.001,0.999,by=0.001))
```

**Arguments**

- **datMeth**: A matrix with row for probes and column for samples
- **reference**: A data frame with two columns, "cg" for CpG names and "meth_mean" for reference values
- **quantile.grid**: Quantile grid used in linear regression

**Details**

The function will normalize user data to a reference value distribution based on common set of probe between user data and reference data.

**References**


**Examples**

```r
if (require(minfiData)) {
  #methDataSet as input
  path <- file.path(find.package("minfiData"),"extdata")
  rgSet <- readidat(path = path,recursive = TRUE)
  qc=QCinfo(rgSet)
  mdat=preprocessENmix(rgSet,QCinfo=qc,nCores=6)
  mdat=norm.quantile(mdat,method="quantile1")
  beta=rcp(mdat)

  #methylset as input
  sheet <- read.metharray.sheet(file.path(find.package("minfiData"),"extdata"),
                              pattern = "csv$")
  rgSet <- read.metharray.exp(targets = sheet,extended = TRUE)
  qc=QCinfo(rgSet)
  mdat=preprocessENmix(rgSet,QCinfo=qc,nCores=6)
  mdat=norm.quantile(mdat,method="quantile1")
  beta=rcp(mdat)
}
```
Value

A matrix with same dimension of user data

Author(s)

Zongli Xu

References


Examples

```r
require(minfiData)
path <- file.path(find.package("minfiData"),"extdata")
#based on rgDataset
rgSet <- readidat(path = path, recursive = TRUE)
meth=getmeth(rgSet)
beta=getB(meth)
reference=data.frame(cg=rownames(beta),goldstandard=beta[,3])
reference=reference[sample(1:nrow(reference),2000),]
beta2=rcp2(beta,reference)
```

readidat Parsing IDAT files for Illumina methylation arrays.

Description

Read in IDAT files and create a rgDataSet with probe annotation

Usage

```r
readidat(path = NULL,manifestfile=NULL,recursive = TRUE, verbose = FALSE, force=FALSE)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>path</td>
<td>Directory where idat files are located</td>
</tr>
<tr>
<td>manifestfile</td>
<td>Array manifestfile, which can be downloaded from Illumina website. Biocon-</td>
</tr>
<tr>
<td></td>
<td>ductor manifest package will be used if not provided</td>
</tr>
<tr>
<td>recursive</td>
<td>if TRUE, idat files in the subdirectories will also be read in</td>
</tr>
<tr>
<td>verbose</td>
<td>if TRUE, detailed running info will be printed on screen</td>
</tr>
<tr>
<td>force</td>
<td>if TRUE, arrays with different sizes will be merged together</td>
</tr>
</tbody>
</table>
Details

Some array types and corresponding manifest files can be guessed by the program based on the number of probes per array. However, we recommend to provide correct manifest file directly, which can be easily downloaded from Illumina website, see below for some examples.

Probe annotation info can be extracted using command rowData

Value

An object of class rgDataSet,

Author(s)

Zongli Xu

Examples

# Illumina methylation array manifest file

# Infinium Mouse Methylation Manifest File (CSV)
 system("unzip infinium-mouse-methylation-manifest-file-csv.zip")
 mf="infinium-mouse-methylation-manifest-file.csv"

# for MethylationEPIC v1.0 B5
 system("unzip infinium-methylationepic-v-1-0-b5-manifest-file-csv.zip")
 mf="infinium-methylationepic-v-1-0-b5-manifest-file.csv"
 manifest=readmanifest(mf)

# for MethylationEPIC v1.0 B4
 system("wget https://webdata.illumina.com/downloads/productfiles/methylationEPIC/infinium-methylationepic-v-1-0-b4-manifest-file-csv.zip")
 system("unzip infinium-methylationepic-v-1-0-b4-manifest-file-csv.zip")
 mf="MethylationEPIC_v-1-0_B4.csv"
 manifest=readmanifest(mf)

# for HumanMethylation450
 mf="HumanMethylation450_15017482_v1-2.csv"
 mf="HumanMethylation450_15017482_v1-2.csv"
 if(require(minfiData)){
readmanifest

Parsing Illumina methylation arrays manifest file.

Description
Parsing Illumina methylation arrays manifest file.

Usage
readmanifest(file)

Arguments
file Illumina methylation array manifest file, downloaded from Illumina website

Value
An object of dataframe containing probe annotation information

Author(s)
Zongli Xu

Examples
# for MethylationEPIC v1.0 B5
system("wget https://webdata.illumina.com/downloads/productfiles/methylationEPIC/
infinium-methylationepic-v-1-0-b5-manifest-file-csv.zip")
system("unzip infinium-methylationepic-v-1-0-b5-manifest-file-csv.zip")
mf="infinium-methylationepic-v-1-0-b5-manifest-file.csv"
manifest=readmanifest(mf)

# for MethylationEPIC v1.0 B4
system("wget https://webdata.illumina.com/downloads/productfiles/methylationEPIC/
infinium-methylationepic-v-1-0-b4-manifest-file-csv.zip")
system("unzip infinium-methylationepic-v-1-0-b4-manifest-file-csv.zip")
mf="MethylationEPIC_v-1-0_B4.csv"
manifest=readmanifest(mf)

# for HumanMethylation450
system("wget https://webdata.illumina.com/downloads/productfiles/
humanmethylation450/humanmethylation450_15017482_v1-2.csv")
REgression on Logarithm of Internal Control probes (RELIC)

Description

REgression on Logarithm of Internal Control probes (RELIC) correct for dye bias on whole array by utilizing the intensity values of paired internal control probes that monitor the two color channels.

Usage

relic (mdat, at_red, cg_grn)

Arguments

mdat
An object of class methDataSet or MethylSet.
at_red
an intensity matrix for Illumina control probes "NORM_A" and "NORM_T"
cg_grn
an intensity matrix for Illumina control probes "NORM_C" and "NORM_G"

Details

The Illumina MethylationEPIC BeadChip contains 85 pairs of internal normalization control probes (name with prefix NORM_A, NORM_T, NORM_G or NORM_C), while its predecessor, Illumina HumanMethylation450 BeadChip contains 93 pairs. RELIC first performs a regression on the logarithms of the intensity values of the normalization control probes to derive a quantitative relationship between red and green channels, and then uses the relationship to correct for dye-bias on intensity values for whole array.

Value

An object of class methDataSet or MethylSet depends on input class.

Author(s)

Zongli Xu and Liang Niu

References


See Also

Package preprocessENmix
Examples

```r
if (require(minfiData)) {

# background correction and dye bias correction
# rgDataSet as input
path <- file.path(find.package("minfiData"),"extdata")
rgSet <- readidat(path = path,recursive = TRUE)
mdat <- preprocessENmix(rgSet,bgParaEst="oob",nCores=6,dyeCorr ="RELIC")

# RGChannelSet as input
mdat=preprocessENmix(RGsetEx,bgParaEst="oob",nCores=6,dyeCorr ="RELIC")

# dye bias correction only
# methDataSet as input
path <- file.path(find.package("minfiData"),"extdata")
rgSet <- readidat(path = path,recursive = TRUE)
ctrls <- getCGinfo(rgSet,type="ctrl")
ctrls <- ctrls[ctrls$Address %in% rownames(rgSet),]

ctrl_r <- assays(rgSet)$Red[ctrls$Address,]
ctrl_g <- assays(rgSet)$Green[ctrls$Address,]
CG.controls <- ctrls$Type %in% c("NORM_C", "NORM_G")
AT.controls <- ctrls$Type %in% c("NORM_A", "NORM_T")

cg_grn = ctrl_g[CG.controls,]
at_red = ctrl_r[AT.controls,]
rownames(cg_grn) = ctrls$ExtendedType[CG.controls]
rownames(at_red) = ctrls$ExtendedType[AT.controls]
mdat=getmeth(rgSet)
mdat <- relic(mdat,at_red,cg_grn)

# MethylSet as input
ctrls <- getProbeInfo(RGsetEx,type="Control")
ctrls <- ctrls[ctrls$Address %in% featureNames(RGsetEx),]
ctrl_r <- getRed(RGsetEx)[ctrls$Address,]
ctrl_g <- getGreen(RGsetEx)[ctrls$Address,]
CG.controls <- ctrls$Type %in% c("NORM_C","NORM_G")
AT.controls <- ctrls$Type %in% c("NORM_A","NORM_T")

cg_grn = ctrl_g[CG.controls,]
at_red = ctrl_r[AT.controls,]
rownames(cg_grn) = ctrls$ExtendedType[CG.controls]
rownames(at_red) = ctrls$ExtendedType[AT.controls]
mdat <- preprocessRaw(RGsetEx)
mdat <- relic(mdat,at_red,cg_grn)
}
```

---

**repicc**

Calculating intraclass correlation coefficient using replicate samples
Description

The function can be used to calculate ICC for each CpG probe using balanced or unbalanced replicate samples.

Usage

repicc(dat, repid, mvalue=FALSE, nCores=2, qcflag=FALSE, qc=NULL, detPthre=0.05, nbthre=3)

Arguments

dat Methylation beta value matrix
repid A data frame with two variables, id and idx. id should be the same with column name of "dat", idx is a variable to show the relationship between samples with same value for samples from same individual.
mvalue If TRUE, the beta value will be converted to M value for calculation of ICC
nCores Number of cores will be used for calculation of ICC
qcflag Whether to perform QC before calculation of ICC
qc QC object from function QCinfo
detPthre If qcflag=TRUE, the methylation values with detection P value higher than the threshold will be removed before calculation
nbthre If qcflag=TRUE, the methylation values with number of bead smaller

Value

A data frame containing ICC for each probe

Author(s)

Zongli Xu

References


Examples

```r
if (require(minfiData)){
  path <- file.path(find.package("minfiData"),"extdata")
  rgSet <- readidat(path = path, recursive = TRUE)
  mdat=getmeth(rgSet)
  beta=getB(mdat,"Illumina")
  repid=data.frame(id=c("5723646052_R02C02","5723646052_R04C01","5723646052_R05C02", "5723646053_R04C02","5723646053_R05C02","5723646053_R06C02"),idx=c(1,1,2,2,2,2))
  iccresu<-repicc(dat=beta,repid=repid)
}
```
rgDataSet-class  

**Description**

A class for storing Illumina methylation array raw intensity data of two color channels, and probe annotation information.

**Usage**

`rgDataSet(Red = new("matrix"), Green = new("matrix"),
            NBeads = new("matrix"), rowData = new("DataFrame"), ictrl = new("DataFrame"), ...)`

**Arguments**

- **Red**: A matrix of Red channel intensity values with row for methylation probes and column for samples
- **Green**: A matrix of Green channel intensity values with row for methylation probes and column for samples
- **NBeads**: A matrix contains the number of beads used to generate intensity values on the Red and Green channels.
- **rowData**: A dataframe contains probe annotation information
- **ictrl**: A dataframe contains detailed information for Illumina internal control probes
- `...`: other arguments for class SummarizedExperiment

**Value**

An object of class `rgDataSet`

**Examples**

`showClass("rgDataSet")`

---

rm.cgsuffix  

**Description**

Remove suffix from CpG names and combine duplicated CpGs

**Usage**

`rm.cgsuffix(datMeth)`
Arguments
datMeth A methylation data matrix with row names for CpG id

Details
Remove suffix from CpG names for EPIC v2 BeadChips and combined values for duplicated CpGs

Value
A matrix with number of rows equal or less than input data.

Author(s)
Zongli Xu

Examples

```r
# beta matrix with row for CpGs and column for samples
beta2 = rm.cgsuffix(beta)
```

Description
Simulation of bed format example file.

Usage
```
simubed(nprobe=1000)
```

Arguments
nprobe Number of probes on each chromosome, default is 1000

Details
Simulation of bed format example file.

Value
A data frame

Author(s)
Zongli Xu
Examples

simubed(nprobe=1000)
## Index

<table>
<thead>
<tr>
<th>B2M, 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>calcdetP, 4</td>
</tr>
<tr>
<td>combp, 5</td>
</tr>
<tr>
<td>ctrlsva, 6</td>
</tr>
<tr>
<td>dupicc, 7</td>
</tr>
<tr>
<td>estimateCellProp, 8</td>
</tr>
<tr>
<td>freqpoly, 10</td>
</tr>
<tr>
<td>getB, 11</td>
</tr>
<tr>
<td>getCGinfo, 12</td>
</tr>
<tr>
<td>getmeth, 12</td>
</tr>
<tr>
<td>ipdmr, 13</td>
</tr>
<tr>
<td>M2B, 14</td>
</tr>
<tr>
<td>methDataSet (methDataSet-class), 15</td>
</tr>
<tr>
<td>methDataSet-class, 15</td>
</tr>
<tr>
<td>methscore, 16</td>
</tr>
<tr>
<td>methyAge, 18</td>
</tr>
<tr>
<td>MethylSet, 22</td>
</tr>
<tr>
<td>mhtplot, 20</td>
</tr>
<tr>
<td>mpreprocess, 21</td>
</tr>
<tr>
<td>multifreqpoly, 23</td>
</tr>
<tr>
<td>nmode, 24</td>
</tr>
<tr>
<td>norm.quantile, 25</td>
</tr>
<tr>
<td>oxBs.MLE, 26</td>
</tr>
<tr>
<td>p.qqplot, 27</td>
</tr>
<tr>
<td>pcrplot, 28</td>
</tr>
<tr>
<td>plotCtrl, 29</td>
</tr>
<tr>
<td>predSex, 30</td>
</tr>
<tr>
<td>preprocessENmix, 31</td>
</tr>
<tr>
<td>qcfiler, 33</td>
</tr>
<tr>
<td>QCInfo, 34</td>
</tr>
<tr>
<td>rcp, 36</td>
</tr>
<tr>
<td>rcp2, 37</td>
</tr>
<tr>
<td>readidat, 38</td>
</tr>
<tr>
<td>readmanifest, 40</td>
</tr>
<tr>
<td>relic, 41</td>
</tr>
<tr>
<td>repicc, 42</td>
</tr>
<tr>
<td>RGChannelSet, 22</td>
</tr>
<tr>
<td>rgDataSet (rgDataSet-class), 44</td>
</tr>
<tr>
<td>rgDataSet-class, 44</td>
</tr>
<tr>
<td>rm.cgsuffix, 44</td>
</tr>
<tr>
<td>show, methDataSet-method</td>
</tr>
<tr>
<td>(methDataSet-class), 15</td>
</tr>
<tr>
<td>show, rgDataSet-method</td>
</tr>
<tr>
<td>(rgDataSet-class), 44</td>
</tr>
<tr>
<td>simubed, 45</td>
</tr>
</tbody>
</table>