Package 'MeDEStrand'

August 7, 2017

Type Package

Title A method to infer absolute methylation from enrichment-based methylation profiling data

Version 0.0.0.9000

Author Jingting Xu, Shimeng Liu, Ping Yin, Serdar Bulun, Yang Dai

Maintainer Jingting XU <jxu43@uic.edu>

Description MeDEStrand was developped to infer absolute methylation levels from enrichment-based methylation profiling data that includes MeDIP-seq, MethylCap-seq/MBD-seq etc. MeDEStrand acquires its strength by utilizing a sigmoid model to estimate and correct CpG bias, as well as utilizing asymmetric bin counts from reads mapped to positive and negative DNA strand to further improve the accuracy.

License GPL (>=2)

Encoding UTF-8

LazyData true

Depends MEDIPS

RoxygenNote 6.0.1

Suggests knitr, rmarkdown

VignetteBuilder knitr

R topics documented:

`MeDEStrand.binMethyl` *Function to infer bin-based absolute methylation levels from enrichment signals*

Description

Function normalizes MeDIP-seq signals by estimating and subsequently correcting CpG bias from the raw signals (i.e. observed bin counts.)

Usage

```
MeDEStrand.binMethyl(MSetInput = NULL, CSet = NULL, ccObj = NULL,
  Granges = FALSE)
```

Arguments

`MSetInput`	MEDIPSset objects created by function 'MeDEStrand.createSet()', reads mapped to the positive and negative DNA strand are processed separately.
`CSet`	A COUPLINGset object.
`ccObj`	Default NULL. Return of internal function call to `MeDEStrand.calibrationCurve`
`Granges`	Default FALSE. Return a vector of inferred absolute methylation levels at user-specified bin size. If TRUE, return a GRanges object with each bin's genomic coordinate and its absolute methylation levels.

Value

Absolute methylation levels in a vector

Examples

```
file.name = "ENCFF002BKV.bam"; BSgenome="BSgenome.Hsapiens.UCSC.hg19"; uniq=1; extend=200; shift=0; ws=100;
chr.select="chr10"
MeDIPSset = MeDEStrand.createSet(file=file.name, BSgenome=BSgenome, extend=extend, shift=shift, uniq=uniq,
window_size=ws, chr.select=chr.select, paired = F)
Bin_methylation = MeDEStrand.binMethyl(MSet = MeDIPSset, CSet = CS )
```

`MeDEStrand.calibrationCurve`

Function takes a MEDIPSset and finds means of bin counts

Description

Function takes MEDIPS SET object, re-organize bins into groups of same CpG counts and finds mean bin counts for each group. This is an internal function used by function "MeDEStrand.plotCalibrationCurve()" and function "MeDEStrand.binMethyl()"

Usage

```
MeDEStrand.calibrationCurve(MSet = NULL, CSet = NULL, input = FALSE)
```

Arguments

MSet	MEDIPSset objects created by function 'MeDEStrand.createSet()'.
CSet	A COUPLINGset object.

Value

means of bin counts for each group (categorized by bin CpG counts); estimated upper asymptote for the sigmoidal logistic regression model that estimates CpG bias.

MeDEStrand.cor *Function calculates correlation (Pearson or Spearman) coefficient between inferred bin-based absolute methylation levels with supplied RRBS data*

Description

Function calculates correlation (Pearson or Spearman) coefficient between a vector of inferred bin-based absolute methylation levels with supplied RRBS data. Since RRBS data provides single-resolution CpG cytosine methylation levels, thus RRBS CpGs are binned and the means of CpG methylation in the bins are assigned as the 'true' absolute methylation levels for the bins.

Usage

```
MeDEStrand.cor(MSetInput = NULL, CS = NULL, RRBS = NULL, minRRBS = NULL,
  method = "pearson")
```

Arguments

MSetInput	MEDIPSset objects.
RRBS	A methylRaw object (object returned by function "processBismarkAln()" from package 'MethylKit') or A GRanges object with a metadata column 'methylation' to specify the methylation level of each CpG or a '.bed' file with at least these columns: 'chr','start','end','methylation' for CpGs.
minRRBS	to filter out bins with RRBS CpGs less than (<) 'minRRBS' number.
method	"pearson" or "spearman". To find the Pearson or Spearman correlation coefficient.
CSet	A COUPLINGset object.

Value

Pearson/Spearman correlation coefficient value

Examples

```
# Generate MEDIPS SET objects from .bam file
file.name = "ENCFF002BKV.bam"; BSgenome="BSgenome.Hsapiens.UCSC.hg19"; uniq=1; extend=200; shift=0; ws=100;
chr.select="chr10"
MeDIPSset = MeDEStrand.createSet(file=file.name, BSgenome=BSgenome, extend=extend, shift=shift, uniq=uniq,
window_size=ws, chr.select=chr.select, paired = F)
# Load RRBS data (a 'methylRaw' object) saved in the package
```

```
fpath <- system.file("data", 'GM12878.RRBS.RData', package="MeDEStrand")
load(fpath)
# Find the Pearson correlation coefficient
correlation = MeDEStrand.cor( MSetInput= MeDIP_single, CS = CS, RRBS = RRBS, minRRBS = 4 , method = "pearson" )
```

`MeDEStrand.countCG` *The function counts the frequency of the specified sequence pattern exist in each bin.*

Description

The function calculates the local densities(i.e. bin level) of a defined sequence pattern (e.g. 'CG' for CpGs) and returns a COUPLINGset object.

Usage

```
MeDEStrand.countCG(pattern = "CG", refObj = NULL)
```

Arguments

`pattern`	defines the sequence pattern, e.g. 'CG' for CpGs.
`refObj`	A MEDIPSset object or an object returned by function `MeDEStrand.createSet`

Value

the counts of specified pattern e.g.'CG' frequencies in bins

Examples

```
file.name = "ENCFF002BKV.bam"; BSgenome="BSgenome.Hsapiens.UCSC.hg19"; uniq=1; extend=200; shift=0; ws=100;
chr.select="chr10"
MeDIPSset = MeDEStrand.createSet(file=file.name, BSgenome=BSgenome, extend=extend, shift=shift, uniq=uniq,
window_size=ws, chr.select=chr.select,paired = F)
# count CG contents in each bin
CS = MeDEStrand.countCG(pattern="CG", refObj=MeDIPSset)
```

`MeDEStrand.createSet` *Function to create a list of two MEDIPS SET objects for reads mapped to the positive and negative DNA strand*

Description

Function creates MEDIPSset objects from input data (i.e. aligned short reads) mapped to the positive and negative DNA strand respectively.

Input format: bam file or tab (|) separated txt file "chr | start | stop | strand"

Usage

```
MeDEStrand.createSet(file = NULL, extend = 0, shift = 0,
  window_size = 300, BSgenome = NULL, uniq = 0.001, chr.select = NULL,
  paired = F, sample_name = NULL, isSecondaryAlignment = FALSE,
  simpleCigar = TRUE)
```

Arguments

file	Path and file name of the input data.
extend	defines the number of bases by which the region will be extended before the genome vector is calculated.Regions will be extended along the plus or the minus strand as defined by their provided strand information.
shift	As an alternative to the extend parameter, the shift parameter can be specified. Here, the reads are not extended but shifted by the specified number of nucleotides with respect to the given strand infomation.One of the two parameters extend or shift has to be 0.
window_size	i.e. bin size. This parameter defines the genomic resolution by which short read coverage is calculated.
BSgenome	The reference genome name as defined by BSgenome.
uniq	The uniq parameter determines, if all reads mapping to exactly the same genomic position should be kept (uniq = 0), replaced by only one representative (uniq = 1), or if the number of stacked reads should be capped by a maximal number of stacked reads per genomic position determined by a poisson distribution of stacked reads genome wide and by a given p-value (1 > uniq > 0) (deafult: 1e-3). The smaller the p-value, the more reads at the same genomic position are potentially allowed.
chr.select	only data at the specified chromosomes will be processed.
paired	option for paired end reads.
sample_name	name of the sample to be stored with the MEDIPS SET.
isSecondaryAlignment	option to import only primary alignments.
simpleCigar	option to import only alignments with simple Cigar string.

Value

MEDIPSset objects created for reads mapped to the positive and negative DNA strand respectively

Examples

```
# Set parameters
file.name = "ENCFF002BKV.bam"; BSgenome="BSgenome.Hsapiens.UCSC.hg19"; uniq=1; extend=200; shift=0; ws=100;
chr.select="chr10"
# Generate a list of two MEDIP SET objects from reads mapped to the positive and negative DNA strand
,respectively.
MeDIPSset = MeDEStrand.createSet(file=file.name, BSgenome=BSgenome, extend=extend, shift=shift, uniq=uniq,
window_size=ws, chr.select=chr.select, paired = F)
```

MeDEStrand.meth *Function calculates genome wide absolute methylation levels for each provided set and uses t test for differential methylated loci between two groups of MEDIPS SETs (if provided)*

Description

The function summarizes coverage profiles (bin counts) as well as infers absolute methylation levels for given MEDIPS SETs. In case two groups of MEDIPS SETs are provided at MSet1 and MSet2, the function finds differential methylated loci between MSet1 and MSet2 based on the inferred absolute methylation levels. For each bin/loci, t test is conducted and multiple test correction (genome-wide/chromosome-wide) can be specified by user.

Usage

```
MeDEStrand.meth(MSet1 = NULL, MSet2 = NULL, CSet = NULL, chr = NULL,
  p.adj = "bonferroni", minRowSum = 10)
```

Arguments

MSet1	first group of MEDIPS SETs. Each group contains several MEDIPSset objects.
MSet2	second group of MEDIPS SETs. Each group contains several MEDIPSset objects. Differential methylated coverage will be calculated by t test and returned, if MSet1 and MSet2 are not empty.
CSet	A COUPLINGset object. (i.e. corresponding bin CpG density for the provided MSet1 or MSet2). This object is return of function MeDEStrand.countCG
chr	specifies the chromosome(s) for t test for differential methylated loci between MSet1 and MSet2. Depending on chromosome(s) or all chromosomes are provided, multiple test correction are conducted at chromosome-wide or genome-wide levels. The latter one is more stringent.
p.adj	in order to correct p.values for multiple testing, MeDEStrand uses R's 'p.adjust()' function. Therefore, the following methods are available: holm, hochberg, hommel, bonferroni, BH, BY, fdr, none.
minRowSum	threshold for a minimum sum of counts across all samples bins (default=10). Bins with lower coverage will not be tested for differential methylation coverage.

Value

result table of summary statistics and t test results

Examples

```
result = MeDEStrand.meth(MSet1=NSCLC_N.MeDIP, MSet2=NSCLC_T.MeDIP, CSet=CS, chr=c('chr20'),
p.adj="fdr", minRowSum= 12)
```

MeDEStrand.plotCalibrationCurve

Function plots the 'calibration plot', which reveals the CpG density bias

---

Description

Visualizes the dependency between MeDIP-seq signals and CpG densities by 'calibration curve'. 'calibration curve' estimates CpG bias by fitting a sigmoidal logistic regression curve.

Usage

```
MeDEStrand.plotCalibrationCurve(MSet = NULL, CSet = NULL,
  Forward_Strand = T, plot_chr = "all", main = "Calibration plot",
  xrange = T)
```

Arguments

MSet	MEDIPSset objects created by function 'MeDEStrand.createSet()' for reads mapped to the positive and negatie DNA strand respectively.
CSet	A COUPLINGset object. If NULL, it can be calculated and supplied from MSet.
main	The title of the calibration plot.
xrange	The signal range of the calibration curve typically falls into a low signal range. By setting the xrange parameter to TRUE, the calibration plot will visualize the low signal range only.
Forward_strand	if TRUE, plot the 'calibration plot' for the reads mapped to the positive DNA strand(i.e. estimating CpG density bias for reads mapped to the positive DNA strand); if FALSE, plot the 'calibration plot' for the reads mapped to the negative DNA strand (i.e. estimating CpG density bias for reads mapped to the negative DNA strand).
default	= "all". It is recommended to call a graphics device (e.g. png("calibrationPlot.png")) before calling the plot command, because R might not be able to plot the full amount of data in reasonable time.

Value

Calibration plot, i.e. Visualization of the CpG bias curve.

Examples

```
file.name = "ENCFF002BKV.bam"; BSgenome="BSgenome.Hsapiens.UCSC.hg19"; uniq=1; extend=200; shift=0; ws=100;
chr.select="chr10"
MeDIPSset = MeDEStrand.createSet(file=file.name, BSgenome=BSgenome, extend=extend, shift=shift, uniq=uniq,
window_size=ws, chr.select=chr.select, paired = F)
# plot the 'calibration plot'
MeDEStrand.plotCalibrationCurve( MSet=MeDIPset, CSet=NULL,  Forward_Strand = T, main = NULL, xrange=TRUE)
```

MeDEStrand.selectSig *Selects putative DMRs (Differential Methylated Regions) from the result table returned by function 'MeDEStrand.meth()'*

Description

Based on the results table returned by function 'MeDEStrand.meth()', this function selects bins which show significant differential methylation between two groups of MEDIPS SETs. Selection of significant bins follows according to the specified parameters.

Usage

```
MeDEStrand.selectSig(results = NULL, p.value = 0.1, adj = T,
  ratio = NULL, bg.counts = NULL, merge.within.distance = NULL)
```

Arguments

results	results table returned by function MeDEStrand.meth
p.value	threshold for significant differential methylation. default 0.1
adj	TRUE or FALSE. Whether multiple test correction is conducted based on the methods: holm, hochberg, hommel, bonferroni, BH, BY, fdr, none.
bg.counts	bin filtering parameter. The parameter requires a minimal number of reads for each of the MEDIPS SET groups. To apply this condition, mean of the bin counts per group is considered.
merge.within.distance	merges significant differential methylated bins within certain bp distance. default is NULL (do not merge).

Value

result table of bins that are significantly differentially methylated between two groups MSet1 and MSet2.

Examples

```
result = MeDEStrand.meth(MSet1=NSCLC_N.MeDIP, MSet2=NSCLC_T.MeDIP, CSet=CS, chr=c('chr20'),
p.adj="fdr", minRowSum= 12)
result.sig = MeDEStrand.selectSig(results = results, p.value = 0.1, adj = T, ratio = NULL, bg.counts = 1,
merge.within.distance = NULL )
```

Index