Package 'mosaics'

October 9, 2013

Type Package

Title MOSAiCS (MOdel-based one and two Sample Analysis and Inference for ChIP-Seq)

Version 1.8.0

Depends R (>= 2.11.1), methods, graphics, Rcpp

Imports MASS, splines, lattice, IRanges

Suggests mosaicsExample

Enhances parallel

LinkingTo Rcpp

SystemRequirements Perl

Date 2012-09-24

Author Dongjun Chung, Pei Fen Kuan, Sunduz Keles

Maintainer Dongjun Chung <chungdon@stat.wisc.edu>

Description

This package provides functions for fitting MOSAiCS, a statistical framework to analyze one-sample or two-sample ChIP-seq data.

License GPL (>= 2)

URL http://groups.google.com/group/mosaics_user_group

LazyLoad yes

biocViews ChIPseq, Sequencing, Transcription, Genetics, Bioinformatics

R topics documented:

`mosaics-package` *MOSAiCS (MOdel-based one and two Sample Analysis and Inference for ChIP-Seq)*

Description

This package provides functions for fitting MOSAiCS, a statistical framework to analyze one-sample or two-sample ChIP-seq data.

Details

Package:	mosaics
Type:	Package
Version:	1.5.3
Date:	2012-09-24
License:	GPL (>= 2)
LazyLoad:	yes

This package contains three main classes, `BinData`, `MosaicsFit`, and `MosaicsPeak`, which represent bin-level ChIP-seq data, MOSAiCS model fit, and MOSAiCS peak calling results, respectively. This package contains four main methods, `constructBins`, `readBins`, `mosaicsFit`, and `mosaicsPeak`. `constructBins` method constructs bin-level files from the aligned read file. `readBins` method imports bin-level data and construct `BinData` class object. `mosaicsFit` method fits a MOSAiCS model using `BinData` class object and constructs `MosaicsFit` class object. `mosaicsPeak` method calls peaks using `MosaicsFit` class object and construct `MosaicsPeak` class object. `MosaicsPeak` class object can be exported as text files or transformed into data frame, which can be used for the downstream analysis. This package also provides methods for simple exploratory analysis.

The `mosaics` package companion website, http://www.stat.wisc.edu/~keles/Software/mosaics/, provides preprocessing scripts, preprocessed files for diverse reference genomes, and easy-to-follow

instructions. We encourage questions or requests regarding mosaics package to be posted on our Google group, http://groups.google.com/group/mosaics_user_group. Please check the vignette for further details on the mosaics package and these websites.

Author(s)

Dongjun Chung, Pei Fen Kuan, Sunduz Keles

Maintainer: Dongjun Chung <chungdon@stat.wisc.edu>

References

Kuan, PF, D Chung, G Pan, JA Thomson, R Stewart, and S Keles (2011), "A Statistical Framework for the Analysis of ChIP-Seq Data", *Journal of the American Statistical Association*, Vol. 106, pp. 891-903.

See Also

constructBins, readBins, mosaicsFit, mosaicsPeak , BinData, MosaicsFit, MosaicsPeak.

Examples

```
## Not run:
library(mosaicsExample)

exampleBinData <- readBins( type=c("chip","input"),
    fileName=c( system.file("extdata/chip_chr21.txt", package="mosaicsExample"),
    system.file("extdata/input_chr21.txt", package="mosaicsExample") ) )
exampleBinData
print(exampleBinData)[1:10, ]
plot(exampleBinData)
plot( exampleBinData, plotType="input" )

exampleFit <- mosaicsFit( exampleBinData, analysisType="IO" )
exampleFit
plot(exampleFit)
estimates(exampleFit)

examplePeak <- mosaicsPeak( exampleFit, signalModel = "2S", FDR = 0.05 )
examplePeak
print(examplePeak)[1:10, ]
export( examplePeak, type = "txt", filename = "./TSpeakList.txt" )
export( examplePeak, type = "bed", filename = "./TSpeakList.bed" )
export( examplePeak, type = "gff", filename = "./TSpeakList.gff" )

## End(Not run)
```

BinData-class *Class "BinData"*

Description

This class represents bin-level ChIP-seq data.

Objects from the Class

Objects can be created by calls of the form `new("BinData", ...)`.

Slots

`chrID`: Object of class `"character"`, a vector of chromosome IDs.

`coord`: Object of class `"numeric"`, a vector of genomic coordinates.

`tagCount`: Object of class `"numeric"`, a vector of tag counts of ChIP sample.

`mappability`: Object of class `"numeric"`, a vector of mappability score.

`gcContent`: Object of class `"numeric"`, a vector of GC content score.

`input`: Object of class `"numeric"`, a vector of tag counts of matched control sample.

`dataType`: Object of class `"character"`, indicating how reads were processed. Possible values are "unique" (only uniquely aligned reads were retained) and "multi" (reads aligned to multiple locations were also retained).

Methods

mosaicsFit `signature(object = "BinData")`: fit a MOSAiCS model using a bin-level ChIP-seq data.

plot `signature(x = "BinData", y = "missing", plotType = NULL )`: provide exploratory plots of mean ChIP tag counts. This method plots mean ChIP tag counts versus mappability score, GC content score, and Control tag counts, with 95% confidence intervals, for `plotType="M"`, `plotType="GC"`, and `plotType="input"`, respectively. `plotType="M|input"` and `plotType="GC|input"` provide plots of mean ChIP tag counts versus mappability and GC content score, respectively, conditional on Control tag counts. If `plotType` is not specified, this method plots histogram of ChIP tag counts.

print `signature(x = "BinData")`: return bin-level data in data frame format.

show `signature(object = "BinData")`: provide brief summary of the object.

Author(s)

Dongjun Chung, Pei Fen Kuan, Sunduz Keles

References

Kuan, PF, D Chung, G Pan, JA Thomson, R Stewart, and S Keles (2011), "A Statistical Framework for the Analysis of ChIP-Seq Data", *Journal of the American Statistical Association*, Vol. 106, pp. 891-903.

See Also

readBins, mosaicsFit.

Examples

```
showClass("BinData")
## Not run:
library(mosaicsExample)
data(exampleBinData)

exampleBinData
print(exampleBinData)[1:10,]
plot(exampleBinData)
plot( exampleBinData, plotType="M" )
plot( exampleBinData, plotType="GC" )
plot( exampleBinData, plotType="input" )
plot( exampleBinData, plotType="M|input" )
plot( exampleBinData, plotType="GC|input" )

exampleFit <- mosaicsFit( exampleBinData, analysisType="IO" )

## End(Not run)
```

---

constructBins *Construct bin-level ChIP-sep data from an aligned read file*

Description

Preprocess and construct bin-level ChIP-sep data from an aligned read file.

Usage

```
constructBins( infile=NULL, fileFormat=NULL, outfileLoc="./",
    byChr=FALSE, useChrfile=FALSE, chrfile=NULL, excludeChr=NULL,
    PET=FALSE, fragLen=200, binSize=200, capping=0, perl = "perl" )
```

Arguments

infile	Name of the aligned read file to be processed.
fileFormat	Format of the aligned read file to be processed. Currently, constructBins permits the following aligned read file formats for SET data (PET = FALSE): "eland_result" (Eland result), "eland_extended" (Eland extended), "eland_export" (Eland export), "bowtie" (default Bowtie), "sam" (SAM), "bed" (BED), and "csem" (CSEM). For PET data (PET = TRUE), the following aligned read file formats are allowed: "eland_result" (Eland result) and "sam" (SAM).
outfileLoc	Directory of processed bin-level files. By default, processed bin-level files are exported to the current directory.

byChr	Construct separate bin-level file for each chromosome? Possible values are TRUE or FALSE. If byChr=FALSE, bin-level data for all chromosomes are exported to one file. If byChr=TRUE, bin-level data for each chromosome is exported to a separate file. Default is FALSE.
useChrfile	Is the file for chromosome info provided? Possible values are TRUE or FALSE. If useChrfile=FALSE, it is assumed that the file for chromosome info is not provided. If useChrfile=TRUE, it is assumed that the file for chromosome info is provided. Default is FALSE.
chrfile	Name of the file for chromosome info. In this file, the first and second columns are ID and size of each chromosome, respectively.
excludeChr	Vector of chromosomes that will be excluded from the analysis. This argument is ignored if useChrfile=TRUE.
PET	Is the file paired-end tag (PET) data? If PET=FALSE, it is assumed that the file is SET data. If PET=TRUE, it is assumed that the file is PET data. Default is FALSE (SET data).
fragLen	Average fragment length. Default is 200. This argument is ignored if PET=TRUE.
binSize	Size of bins. Default is 200.
capping	Maximum number of reads allowed to start at each nucleotide position. To avoid potential PCR amplification artifacts, the maximum number of reads that can start at a nucleotide position is capped at capping. Capping is not applied if non-positive value is used for capping. Default is 0 (no capping).
perl	Name of the perl executable to be called. Default is "perl".

Details

Bin-level files are constructed from the aligned read file and exported to the directory specified in outfileLoc argument. If byChr=FALSE, bin-level files are named as [infileName]_fragL[fragLen]_bin[binSize].txt. If byChr=TRUE, bin-level files are named as [infileName]_fragL[fragLen]_bin[binSize]_[chrID].txt, where chrID is chromosome IDs that reads align to. These chromosome IDs are extracted from the aligned read file.

If the file for chromosome information is provided (useChrfile=TRUE and chrfile is not NULL), only the chromosomes specified in the file will be considered. Chromosomes that are specified in excludeChr will not be included in the processed bin-level files. excludeChr argument is ignored if useChrfile=TRUE. Constructed bin-level files can be loaded into the R environment using the method readBins.

constructBins currently supports the following aligned read file formats for SET data (PET = FALSE): Eland result ("eland_result"), Eland extended ("eland_extended"), Eland export ("eland_export"), default Bowtie ("bowtie"), SAM ("sam"), BED ("bed"), and CSEM ("csem"). For PET data (PET = TRUE), the following aligned read file formats are allowed: "eland_result" (Eland result) and "sam" (SAM).

If input file format is neither BED nor CSEM BED, this method retains only reads mapping uniquely to the reference genome.

Value

Processed bin-level files are exported to the directory specified in outfileLoc.

Author(s)

Dongjun Chung, Pei Fen Kuan, Sunduz Keles

References

Kuan, PF, D Chung, JA Thomson, R Stewart, and S Keles (2011), "A Statistical Framework for the Analysis of ChIP-Seq Data", *Journal of the American Statistical Association*, Vol. 106, pp. 891-903.

See Also

`readBins`, `BinData`.

Examples

```
## Not run:
constructBins( infile="/scratch/eland/STAT1_eland_results.txt",
    fileFormat="eland_result", outfileLoc="/scratch/eland/",
    byChr=FALSE, excludeChr="chrM", PET = FALSE,
    fragLen=200, binSize=200, capping=0 )

## End(Not run)
```

---

estimates *Extract estimates of the fitted MOSAiCS model*

Description

Extract estimates from `MosaicsFit` class object, which is a fitted MOSAiCS model.

Usage

```
estimates( object, ... )
## S4 method for signature 'MosaicsFit'
estimates( object )
```

Arguments

`object`	Object of class `MosaicsFit`, which represents a fitted MOSAiCS model obtained using method `mosaicsFit`.
`...`	Other parameters to be passed through to generic `estimates`.

Value

Returns a list with components:

pi0	Mixing proportion of background component.
a	Parameter for background component.
betaEst	Parameter for background component (coefficient estimates).
muEst	Parameter for background component.
b	Parameter for one-signal-component model.
c	Parameter for one-signal-component model.
p1	Parameter for two-signal-component model (mixing proportion of signal components).
b1	Parameter for two-signal-component model (the first signal component).
c1	Parameter for two-signal-component model (the first signal component).
b2	Parameter for two-signal-component model (the second signal component).
c2	Parameter for two-signal-component model (the second signal component).
analysisType	Analysis type. Possible values are "OS" (one-sample analysis), "TS" (two-sample analysis using mappability and GC content), and "IO" (two-sample analysis without using mappability and GC content).

Author(s)

Dongjun Chung, Pei Fen Kuan, Sunduz Keles

References

Kuan, PF, D Chung, JA Thomson, R Stewart, and S Keles (2011), "A Statistical Framework for the Analysis of ChIP-Seq Data", *Journal of the American Statistical Association*, Vol. 106, pp. 891-903.

See Also

mosaicsFit, MosaicsFit.

Examples

```
## Not run:
library(mosaicsExample)
data(exampleFit)

estimates(exampleFit)

## End(Not run)
```

export *Export peak calling results to text files*

Description

Export peak calling results to text files in TXT, BED, or GFF file formats.

Usage

```
export(object, ...)
## S4 method for signature 'MosaicsPeak'
export( object, type=NA, filename=NA )
```

Arguments

object	Object of class MosaicsPeak, peak calling results obtained using method mosaicsPeak.
type	Format of the exported file. Possible values are "txt", "bed", and "gff". See Details.
filename	Name of the exported file.
...	Other parameters to be passed through to generic export.

Details

TXT file format (type="txt") exports peak calling results in the most informative way. Columns include chromosome ID, peak start position, peak end position, peak width, average posterior probability, minimum posterior probability, average ChIP tag count, maximum ChIP tag count (always), average input tag count, average input tag count scaled by sequencing depth, average log base 2 ratio of ChIP over input tag counts (if matched control sample is also provided), average mappability score, and average GC content score (when mappability and GC content scores are used in the analysis) in each peak. type="bed" and type="gff" export peak calling results in standard BED and GFF file formats, respectively, where score is the average ChIP tag counts in each peak. If no peak is detected, export method will not generate any file.

Author(s)

Dongjun Chung, Pei Fen Kuan, Sunduz Keles

References

Kuan, PF, D Chung, JA Thomson, R Stewart, and S Keles (2011), "A Statistical Framework for the Analysis of ChIP-Seq Data", *Journal of the American Statistical Association*, Vol. 106, pp. 891-903.

See Also

mosaicsPeak, MosaicsPeak.

Examples

```
## Not run:
library(mosaicsExample)
data(exampleFit)

examplePeak <- mosaicsPeak( exampleFit, signalModel = "2S", FDR = 0.05 )
export( examplePeak, type = "txt", filename = "./TSpeakList.txt" )
export( examplePeak, type = "bed", filename = "./TSpeakList.bed" )
export( examplePeak, type = "gff", filename = "./TSpeakList.gff" )

## End(Not run)
```

---

generateWig *Construct wiggle files from an aligned ChIP-sep read file*

Description

Construct wiggle files from an aligned ChIP-sep read file.

Usage

```
generateWig( infile=NULL, fileFormat=NULL, outfileLoc="./",
    byChr=FALSE, useChrfile=FALSE, chrfile=NULL, excludeChr=NULL,
    PET=FALSE, fragLen=200, span=200, capping=0, normConst=1, perl = "perl" )
```

Arguments

infile	Name of the aligned read file to be processed.
fileFormat	Format of the aligned read file to be processed. Currently, generateWig permits the following aligned read file formats for SET data (PET = FALSE): "eland_result" (Eland result), "eland_extended" (Eland extended), "eland_export" (Eland export), "bowtie" (default Bowtie), "sam" (SAM), "bed" (BED), and "csem" (CSEM). For PET data (PET = TRUE), the following aligned read file formats are allowed: "eland_result" (Eland result) and "sam" (SAM).
outfileLoc	Directory of processed wiggle files. By default, processed wiggle files are exported to the current directory.
byChr	Construct separate wiggle file for each chromosome? Possible values are TRUE or FALSE. If byChr=FALSE, all chromosomes are exported to one file. If byChr=TRUE, each chromosome is exported to a separate file. Default is FALSE.
useChrfile	Is the file for chromosome info provided? Possible values are TRUE or FALSE. If useChrfile=FALSE, it is assumed that the file for chromosome info is not provided. If useChrfile=TRUE, it is assumed that the file for chromosome info is provided. Default is FALSE.
chrfile	Name of the file for chromosome info. In this file, the first and second columns are ID and size of each chromosome, respectively.

excludeChr	Vector of chromosomes that will be excluded from the analysis. This argument is ignored if useChrfile=TRUE.
PET	Is the file paired-end tag (PET) data? If PET=FALSE, it is assumed that the file is SET data. If PET=TRUE, it is assumed that the file is PET data. Default is FALSE (SET data).
fragLen	Average fragment length. Default is 200. This argument is ignored if PET=TRUE.
span	Span used in wiggle files. Default is 200.
capping	Maximum number of reads allowed to start at each nucleotide position. To avoid potential PCR amplification artifacts, the maximum number of reads that can start at a nucleotide position is capped at capping. Capping is not applied if non-positive value is used for capping. Default is 0 (no capping).
normConst	Normalizing constant to scale values in each position.
perl	Name of the perl executable to be called. Default is "perl".

Details

Wiggle files are constructed from the aligned read file and exported to the directory specified in outfileLoc argument. If byChr=FALSE, wiggle files are named as [infileName]_fragL[fragLen]_span[span].wig. If byChr=TRUE, wiggle files are named as [infileName]_fragL[fragLen]_span[span]_[chrID].wig, where chrID is chromosome IDs that reads align to. These chromosome IDs are extracted from the aligned read file.

If the file for chromosome information is provided (useChrfile=TRUE and chrfile is not NULL), only the chromosomes specified in the file will be considered. Chromosomes that are specified in excludeChr will not be included in the processed wiggle files. excludeChr argument is ignored if useChrfile=TRUE.

generateWig currently supports the following aligned read file formats for SET data (PET = FALSE): Eland result ("eland_result"), Eland extended ("eland_extended"), Eland export ("eland_export"), default Bowtie ("bowtie"), SAM ("sam"), BED ("bed"), and CSEM ("csem"). For PET data (PET = TRUE), the following aligned read file formats are allowed: "eland_result" (Eland result) and "sam" (SAM).

If input file format is neither BED nor CSEM BED, this method retains only reads mapping uniquely to the reference genome.

Value

Processed wig files are exported to the directory specified in outfileLoc.

Author(s)

Dongjun Chung, Pei Fen Kuan, Sunduz Keles

References

Kuan, PF, D Chung, JA Thomson, R Stewart, and S Keles (2011), "A Statistical Framework for the Analysis of ChIP-Seq Data", *Journal of the American Statistical Association*, Vol. 106, pp. 891-903.

Examples

```
## Not run:
generateWig( infile="/scratch/eland/STAT1_eland_results.txt",
    fileFormat="eland_result", outfileLoc="/scratch/eland/",
    byChr=FALSE, excludeChr="chrM", PET = FALSE,
    fragLen=200, span=200, capping=0, normConst=1 )

## End(Not run)
```

---

mosaicsFit *Fit MOSAiCS model*

Description

Fit one-sample or two-sample MOSAiCS models with one signal component and two signal components.

Usage

```
mosaicsFit( object, ... )
## S4 method for signature 'BinData'
mosaicsFit( object, analysisType="automatic", bgEst="automatic",
    k=3, meanThres=NA, s=2, d=0.25, truncProb=0.999, parallel=FALSE, nCore=8 )
```

Arguments

object	Object of class BinData, bin-level ChIP-seq data imported using method readBins.
analysisType	Analysis type. Possible values are "OS" (one-sample analysis), "TS" (two-sample analysis using mappability and GC content), and "IO" (two-sample analysis without using mappability and GC content). If analysisType="automatic", this method tries to make the best guess for analysisType, based on the data provided.
bgEst	Parameter to determine background estimation approach. Possible values are "matchLow" (estimation using bins with low tag counts) and "rMOM" (estimation using robust method of moment (MOM)). If bgEst="automatic", this method tries to make the best guess for bgEst, based on the data provided.
k	Parameter for estimating background distribution. It is not recommended for users to change this value.
meanThres	Parameter for estimating background distribution. Default is 1 for analysisType="TS" and 0 for analysisType="OS". Not relevant when analysisType="IO".
s	Parameter for estimating background distribution. Relevant only when analysisType="TS". Default is 2.
d	Parameter for estimating background distribution. Relevant only when analysisType="TS" or analysisType="IO". Default is 0.25.
truncProb	Parameter for estimating background distribution. Relevant only when analysisType="IO".

parallel	Utilize multiple CPUs for parallel computing using "parallel" package? Possible values are TRUE (utilize multiple CPUs) or FALSE (do not utilize multiple CPUs). Default is FALSE (do not utilize multiple CPUs).
nCore	Number of CPUs when parallel computing is utilized.
...	Other parameters to be passed through to generic mosaicsFit.

Details

The imported data type constraints the analysis that can be implemented. If only data for ChIP sample and matched control sample (i.e., either type=c("chip", "input") or type=c("chip", "input", "N") was used in method readBins), only two-sample analysis without using mappability and GC content (analysisType="IO") is allowed. If matched control data is available with mappability score, GC content score, and sequence ambiguity score, (i.e., type=c("chip", "input", "M", "GC", "N") was used in method readBins), user can do all of three analysis types (analysisType="OS", analysisType="TS", or analysisType="IO"). If there is no data for matched control sample (i.e., type=c("chip", "M", "GC", "N") was used in method readBins), only one-sample analysis (analysisType="OS") is permitted.

Parallel computing can be utilized for faster computing if parallel=TRUE and parallel package is loaded. nCore determines number of CPUs used for parallel computing. meanThres, s, d, and truncProb are the tuning parameters for estimating background distribution. The vignette and Kuan et al. (2011) provide further details about these tuning parameters. Please do not try different value for k argument.

Value

Construct MosaicsFit class object.

Author(s)

Dongjun Chung, Pei Fen Kuan, Sunduz Keles

References

Kuan, PF, D Chung, JA Thomson, R Stewart, and S Keles (2011), "A Statistical Framework for the Analysis of ChIP-Seq Data", *Journal of the American Statistical Association*, Vol. 106, pp. 891-903.

See Also

readBins, MosaicsFit.

Examples

```
## Not run:
library(mosaicsExample)
data(exampleBinData)

exampleFit <- mosaicsFit( exampleBinData, analysisType="IO", bgEst="automatic" )

## End(Not run)
```

`MosaicsFit-class` *Class "MosaicsFit"*

Description

This class represents MOSAiCS model fit.

Objects from the Class

Objects can be created by calls of the form `new("MosaicsFit", ...)`.

Slots

`mosaicsEst`: Object of class `"MosaicsFitEst"`, representing estimates of MOSAiCS model fit.

`mosaicsParam`: Object of class `"MosaicsFitParam"`, representing tuning parameters for fitting MOSAiCS model.

`chrID`: Object of class `"character"`, a vector of chromosome IDs.

`coord`: Object of class `"numeric"`, a vector of genomic coordinates.

`tagCount`: Object of class `"numeric"`, a vector of tag counts of ChIP sample.

`bic1S`: Object of class `"numeric"`, Bayesian Information Criterion (BIC) value of one-signal-component model.

`bic2S`: Object of class `"numeric"`, Bayesian Information Criterion (BIC) value of two-signal-component model.

Methods

estimates `signature(object = "MosaicsFit")`: extract estimates from MOSAiCS model fit.

mosaicsPeak `signature(object = "MosaicsFit")`: call peaks using MOSAiCS model fit.

plot `signature(x = "MosaicsFit", y = "missing")`: draw Goodness of Fit (GOF) plot.

print `signature(x = "MosaicsFit")`: (not supported yet)

show `signature(object = "MosaicsFit")`: provide brief summary of the object.

Author(s)

Dongjun Chung, Pei Fen Kuan, Sunduz Keles

References

Kuan, PF, D Chung, JA Thomson, R Stewart, and S Keles (2010), "A Statistical Framework for the Analysis of ChIP-Seq Data", To appear in *Journal of the American Statistical Association* (http://pubs.amstat.org/doi/abs/10.1198/jasa.2011.ap09706).

See Also

`mosaicsFit`, `mosaicsPeak`, `estimates`.

Examples

```
showClass("MosaicsFit")
## Not run:
library(mosaicsExample)
data(exampleFit)

exampleFit
plot(exampleFit)
estimates(exampleFit)

examplePeak <- mosaicsPeak( exampleFit, signalModel = "2S", FDR = 0.05 )

## End(Not run)
```

`mosaicsPeak` *Call peaks using fitted MOSAiCS model*

Description

Call peaks using `MosaicsFit` class object, which is a fitted MOSAiCS model.

Usage

```
mosaicsPeak( object, ... )
## S4 method for signature 'MosaicsFit'
mosaicsPeak( object, signalModel="2S", FDR=0.05,
    binsize=NA, maxgap=200, minsize=50, thres=10 )
```

Arguments

`object`	Object of class `MosaicsFit`, a fitted MOSAiCS model obtained using function `mosaicsFit`.
`signalModel`	Signal model. Possible values are "1S" (one-signal-component model) and "2S" (two-signal-component model). Default is "2S".
`FDR`	False discovery rate. Default is 0.05.
`binsize`	Size of each bin. Value should be positive integer. If `binsize=NA`, `mosaicsPeak` function calcuates the value from data. Default is `NA`.
`maxgap`	Initial nearby peaks are merged if the distance (in bp) between them is less than `maxgap`. Default is 200.
`minsize`	An initial peak is removed if its width is narrower than `minsize`. Default is 50.
`thres`	A bin within initial peak is removed if its ChIP tag counts are less than `thres`. Default is 10.
`...`	Other parameters to be passed through to generic `mosaicsPeak`.

Details

When peaks are called, proper signal model needs to be specified. The optimal choice for the number of signal components depends on the characteristics of ChIP-seq data. In order to support users in the choice of optimal signal model, Bayesian Information Criterion (BIC) values and Goodness of Fit (GOF) plot are provided for the fitted MOSAiCS model. BIC values and GOF plot can be obtained by applying `show` and `plot` methods, respectively, to the `MosaicsFit` class object, which is a fitted MOSAiCS model. `maxgap`, `minsize`, and `thres` are for refining initial peaks called using specified `signalModel` and `FDR`.

If you use a bin size shorter than the average fragment length of the experiment, we recommend to set `maxgap` to the average fragment length and `minsize` to the bin size. If you set the bin size to the average fragment length or if bin size is larger than the average fragment length, set `maxgap` to the average fragment length and `minsize` to a value smaller than the average fragment length. See the vignette for further details.

Value

Construct `MosaicsPeak` class object.

Author(s)

Dongjun Chung, Pei Fen Kuan, Sunduz Keles

References

Kuan, PF, D Chung, G Pan, JA Thomson, R Stewart, and S Keles (2011), "A Statistical Framework for the Analysis of ChIP-Seq Data", *Journal of the American Statistical Association*, Vol. 106, pp. 891-903.

See Also

`mosaicsFit` , `MosaicsPeak`, `MosaicsFit`.

Examples

```
## Not run:
library(mosaicsExample)
data(exampleFit)

examplePeak <- mosaicsPeak( exampleFit, signalModel = "2S", FDR = 0.05 )

## End(Not run)
```

MosaicsPeak-class *Class "MosaicsPeak"*

Description

This class represents peak calling results.

Objects from the Class

Objects can be created by calls of the form `new("MosaicsPeak", ...)`.

Slots

`peakList`**:** Object of class `"data.frame"`, representing peak list.

`peakParam`**:** Object of class `"MosaicsPeakParam"`, representing parameters for peak calling.

`bdBin`**:** Object of class `"numeric"`, representing a vector of bounded bins.

`empFDR`**:** Object of class `"numeric"`, representing empirical FDR.

Methods

export `signature(object = "MosaicsPeak")`: export peak list into text files.

print `signature(x = "MosaicsPeak")`: return peak list in data frame format.

show `signature(object = "MosaicsPeak")`: provide brief summary of the object.

Author(s)

Dongjun Chung, Pei Fen Kuan, Sunduz Keles

References

Kuan, PF, D Chung, G Pan, JA Thomson, R Stewart, and S Keles (2011), "A Statistical Framework for the Analysis of ChIP-Seq Data", *Journal of the American Statistical Association*, Vol. 106, pp. 891-903.

See Also

`mosaicsPeak`, `export`.

Examples

```
showClass("MosaicsPeak")
## Not run:
library(mosaicsExample)
data(exampleFit)
examplePeak <- mosaicsPeak( exampleFit, signalModel = "2S", FDR = 0.05 )

examplePeak
```

```
print(examplePeak)[1:10, ]
export( examplePeak, type = "txt", filename = "./TSpeakList.txt" )
export( examplePeak, type = "bed", filename = "./TSpeakList.bed" )
export( examplePeak, type = "gff", filename = "./TSpeakList.gff" )

## End(Not run)
```

`mosaicsRunAll` *Analyze ChIP-seq data using the MOSAiCS framework*

Description

Construct bin-level ChIP-sep data from aligned read files of ChIP and matched control samples, fit a MOSAiCS model, call peaks, export peak calling results, and generate reports for diagnostics.

Usage

```
mosaicsRunAll(
    chipFile=NULL, chipFileFormat=NULL,
    controlFile=NULL, controlFileFormat=NULL,
    binfileDir=NULL,
    peakFile=NULL, peakFileFormat=NULL,
    reportSummary=FALSE, summaryFile=NULL,
    reportExploratory=FALSE, exploratoryFile=NULL,
    reportGOF=FALSE, gofFile=NULL,
    PET=FALSE, byChr=FALSE, useChrfile=FALSE, chrfile=NULL, excludeChr=NULL,
    FDR=0.05, fragLen=200, binSize=200, capping=0, bgEst="automatic", d=0.25,
    signalModel="BIC", maxgap=200, minsize=50,
    thres=10, parallel=FALSE, nCore=8 )
```

Arguments

`chipFile`	Name of the aligned read file of ChIP sample to be processed.
`chipFileFormat`	Format of the aligned read file of ChIP sample to be processed. Currently, `mosaicsRunAll` permits the following aligned read file formats: `"eland_result"` (Eland result), `"eland_extended"` (Eland extended), `"eland_export"` (Eland export), `"bowtie"` (default Bowtie), `"sam"` (SAM), `"bed"` (BED), and `"csem"` (CSEM BED). Note that `"csem"` does not mean CSEM output file format, but CSEM BED file format.
`controlFile`	Name of the aligned read file of matched control sample to be processed.
`controlFileFormat`	Format of the aligned read file of matched control sample to be processed. Currently, `mosaicsRunAll` permits the following aligned read file formats: `"eland_result"` (Eland result), `"eland_extended"` (Eland extended), `"eland_export"` (Eland export), `"bowtie"` (default Bowtie), `"sam"` (SAM), `"bed"` (BED), and `"csem"` (CSEM BED). Note that `"csem"` does not mean CSEM output file format, but CSEM BED file format.

binfileDir	Directory to store processed bin-level files.
peakFile	Name of the peak list generated from the analysis.
peakFileFormat	Format of the peak list generated from the analysis. Possible values are "txt", "bed", and "gff".
reportSummary	Report the summary of model fitting and peak calling? Possible values are TRUE (YES) and FALSE (NO). Default is FALSE (NO).
summaryFile	File name of the summary report of model fitting and peak calling. The summary report is a text file.
reportExploratory	Report the exploratory analysis plots? Possible values are TRUE (YES) and FALSE (NO). Default is FALSE (NO).
exploratoryFile	Name of the file for exploratory analysis plots. The exploratory analysis results are exported as a PDF file.
reportGOF	Report the goodness of fit (GOF) plots? Possible values are TRUE (YES) and FALSE (NO). Default is FALSE (NO).
gofFile	Name of the file for goodness of fit (GOF) plots. The GOF plots are exported as a PDF file.
PET	Is the file paired-end tag (PET) data? If PET=FALSE, it is assumed that the file is SET data. If PET=TRUE, it is assumed that the file is PET data. Default is FALSE (SET data).
byChr	Analyze ChIP-seq data for each chromosome separately or analyze it genome-wide? Possible values are TRUE (chromosome-wise) and FALSE (genome-wide). Default is FALSE (genome-wide analysis).
useChrfile	Is the file for chromosome info provided? Possible values are TRUE or FALSE. If useChrfile=FALSE, it is assumed that the file for chromosome info is not provided. If useChrfile=TRUE, it is assumed that the file for chromosome info is provided. Default is FALSE.
chrfile	Name of the file for chromosome info. In this file, the first and second columns are ID and size of each chromosome, respectively.
excludeChr	Vector of chromosomes that will be excluded from the analysis.
FDR	False discovery rate. Default is 0.05.
fragLen	Average fragment length. Default is 200.
binSize	Size of bins. Default is 200.
capping	Maximum number of reads allowed to start at each nucleotide position. To avoid potential PCR amplification artifacts, the maximum number of reads that can start at a nucleotide position is capped at capping. Capping is not applied if non-positive capping is used. Default is 0 (no capping).
bgEst	Parameter to determine background estimation approach. Possible values are "matchLow" (estimation using bins with low tag counts) and "rMOM" (estimation using robust method of moment (MOM)). If bgEst="automatic", this method tries to make the best guess for bgEst, based on the data provided.
d	Parameter for estimating background distribution. Default is 0.25.

signalModel	Signal model. Possible values are "BIC" (automatic model selection using BIC), "1S" (one-signal-component model), and "2S" (two-signal-component model). Default is "BIC".
maxgap	Initial nearby peaks are merged if the distance (in bp) between them is less than maxgap. Default is 200.
minsize	An initial peak is removed if its width is narrower than minsize. Default is 50.
thres	A bin within initial peak is removed if its ChIP tag counts are less than thres. Default is 10.
parallel	Utilize multiple CPUs for parallel computing using "parallel" package? Possible values are TRUE (use multiple CPUs) or FALSE (do not use multiple CPUs). Default is FALSE (do not use multiple CPUs).
nCore	Number of maximum number of CPUs used for the analysis. Default is 8.

Details

This method implements the work flow for the two-sample analysis of ChIP-seq data using the MOSAiCS framework (without using mappability and GC content scores). It imports aligned read files of ChIP and matched control samples, processes them into bin-level files, fits MOSAiCS model, calls peaks, exports the peak lists to text files, and generates reports for diagnostics. This method is a wrapper function of constructBins, readBins, mosaicsFit, mosaicsPeak, export functions, and methods of BinData, MosaicsFit, and MosaicsPeak classes.

See the vignette of the package for the illustration of the work flow and the description of employed methods and their options. Exploratory analysis plots and goodness of fit (GOF) plots are generated using the methods plot of the classes BinData and MosaicsFit, respectively. See the help of constructBins for details of the options PET, chipFileFormat, controlFileFormat, byChr, useChrfile, chrfile, excludeChr, fragLen, binSize, and capping. See the help of mosaicsFit for details of the options bgEst and d. See the help of mosaicsPeak for details of the options FDR, signalModel, maxgap, minsize, and thres. See the help of export for details of the option peakFileFormat.

When the data contains multiple chromosomes, parallel computing can be utilized for faster preprocessing and model fitting if parallel=TRUE and parallel package is loaded. nCore determines number of CPUs used for parallel computing.

Value

Processed bin-level files are exported to the directory specified in binfileDir argument. If byChr=FALSE (genome-wide analysis), one bin-level file is generated for each of ChIP and matched control samples, where file names are [chipFile]_fragL[fragLen]_bin[binSize].txt and [controlFile]_fragL[fragLen]_bin[respectively. If byChr=TRUE (chromosome-wise analysis), bin-level files are generated for each chromosome of each of ChIP and matched control samples, where file names are [chipFile]_fragL[fragLen]_bin[binSiz and [controlFile]_fragL[fragLen]_bin[binSize]_[chrID].txt ([chrID] is chromosome IDs that reads align to). The peak list generated from the analysis are exported to the file with the name specified in peakFile. If reportSummary=TRUE, the summary of model fitting and peak calling is exported to the file with the name specified in summaryFile (text file). If reportExploratory=TRUE, the exploratory analysis plots are exported to the file with the name specified in exploratoryFile (PDF file). If reportGOF=TRUE, the goodness of fit (GOF) plots are exported to the file with the name specified in gofFile (PDF file).

Author(s)

Dongjun Chung, Pei Fen Kuan, Sunduz Keles

References

Kuan, PF, D Chung, G Pan, JA Thomson, R Stewart, and S Keles (2011), "A Statistical Framework for the Analysis of ChIP-Seq Data", *Journal of the American Statistical Association*, Vol. 106, pp. 891-903.

See Also

constructBins, readBins, mosaicsFit, mosaicsPeak, export, BinData, MosaicsFit, MosaicsPeak.

Examples

```
## Not run:
# minimal input (without any reports for diagnostics)

mosaicsRunAll(
    chipFile = "/scratch/eland/STAT1_eland_results.txt",
    chipFileFormat = "eland_result",
    controlFile = "/scratch/eland/input_eland_results.txt",
    controlFileFormat = "eland_result",
    binfileDir = "/scratch/bin/",
    peakFile = "/scratch/peak/STAT1_peak_list.bed",
    peakFileFormat = "bed" )

# generate all reports for diagnostics

mosaicsRunAll(
    chipFile = "/scratch/eland/STAT1_eland_results.txt",
    chipFileFormat = "eland_result",
    controlFile = "/scratch/eland/input_eland_results.txt",
    controlFileFormat = "eland_result",
    binfileDir = "/scratch/bin/",
    peakFile = "/scratch/peak/STAT1_peak_list.bed",
    peakFileFormat = "bed",
    reportSummary = TRUE,
    summaryFile = "/scratch/reports/mosaics_summary.txt",
    reportExploratory = TRUE,
    exploratoryFile = "/scratch/reports/mosaics_exploratory.pdf",
    reportGOF = TRUE,
    gofFile = "/scratch/reports/mosaics_GOF.pdf",
    PET = FALSE, byChr = FALSE, excludeChr = "chrM", FDR = 0.05,  fragLen = 200,
    capping = 0, bgEst="automatic", parallel = FALSE, nCore = 8 )

## End(Not run)
```

readBins *Import bin-level ChIP-sep data*

Description

Import and preprocess all or subset of bin-level ChIP-sep data, including ChIP data, matched control data, mappability score, GC content score, and sequence ambiguity score.

Usage

```
readBins( type = c("chip", "input"), fileName = NULL,
    dataType = "unique", rounding = 100, parallel=FALSE, nCore=8 )
```

Arguments

type	Character vector indicating data types to be imported. This vector can contain "chip" (ChIP data), "input" (matched control data), "M" (mappability score), "GC" (GC content score), and "N" (sequence ambiguity score). Currently, readBins permits only the following combinations: c("chip", "input"), c("chip", "input", "N"), c("chip", "input", "M", "GC", "N"), and c("chip", "M", "GC", "N"). Default is c("chip", "input").
fileName	Character vector of file names, each of which matches each element of type. type and fileName should have the same length and corresponding elements in two vectors should appear in the same order.
dataType	How reads were processed? Possible values are either "unique" (only uniquely aligned reads were retained) or "multi" (reads aligned to multiple locations were also retained).
rounding	How are mappability score and GC content score rounded? Default is 100 and this indicates rounding of mappability score and GC content score to the nearest hundredth.
parallel	Utilize multiple CPUs for parallel computing using "paralle" package? Possible values are TRUE (use multiple CPUs) or FALSE (do not use multiple CPUs). Default is FALSE (do not use multiple CPUs).
nCore	Number of CPUs when parallel computing is utilized.

Details

Bin-level ChIP and matched control data can be generated from the aligned read files for your samples using the method constructBins. In mosaics package companion website, http://www.stat.wisc.edu/~keles/Software/mosaics/, we provide preprocessed mappability score, GC content score, and sequence ambiguity score files for diverse reference genomes. Please check the website and the vignette for further details.

The imported data type constraints the analysis that can be implemented. If type=c("chip", "input") or c("chip", "input", "N"), only two-sample analysis without using mappability and GC content is allowed. For type=c("chip", "input", "M", "GC", "N"), user can do the one- or

two-sample analysis. If `type=c("chip", "M", "GC", "N")`, only one-sample analysis is permitted. See help page of `mosaicsFit`.

When the data contains multiple chromosomes, parallel computing can be utilized for faster preprocessing if `parallel=TRUE` and `parallel` package is loaded. `nCore` determines number of CPUs used for parallel computing.

Value

Construct `BinData` class object.

Author(s)

Dongjun Chung, Pei Fen Kuan, Sunduz Keles

References

Kuan, PF, D Chung, G Pan, JA Thomson, R Stewart, and S Keles (2011), "A Statistical Framework for the Analysis of ChIP-Seq Data", *Journal of the American Statistical Association*, Vol. 106, pp. 891-903.

See Also

`constructBins`, `mosaicsFit`, `BinData`.

Examples

```
## Not run:
library(mosaicsExample)
exampleBinData <- readBins( type=c("chip","input"),
    fileName=c( system.file("extdata/chip_chr21.txt", package="mosaicsExample"),
    system.file("extdata/input_chr21.txt", package="mosaicsExample") ) )

## End(Not run)
```

Index