Package 'srnadiff'

October 16, 2019

Type Package

Title Differential Expression of Small RNA-Seq

Version 1.4.0

Maintainer Matthias Zytnicki <matthias.zytnicki@inra.fr>

Description Differential expression of small RNA-seq
when reference annotation is not given.

License GPL-3

Encoding UTF-8

LazyData true

Depends R (>= 3.5)

SystemRequirements C++11

Imports Rcpp (>= 0.12.8), methods, utils, devtools, BiocStyle,
S4Vectors, GenomeInfoDb, rtracklayer, SummarizedExperiment,
IRanges, GenomicRanges, DESeq2, Rsamtools, GenomicFeatures,
GenomicAlignments, ggplot2, BiocParallel

LinkingTo Rcpp

RoxygenNote 6.0.1

biocViews ImmunoOncology, GeneExpression, Coverage, SmallRNA,
Epigenetics, StatisticalMethod, Preprocessing,
DifferentialExpression

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

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

git_branch RELEASE_3_9

git_last_commit 3f1343e

git_last_commit_date 2019-05-02

Date/Publication 2019-10-15

Author Zytnicki Matthias [aut, cre]

R topics documented:

`buildDataHmm` *Read the coverage and extract expressed regions*

Description

Read the coverage and extract expressed regions

Usage

```
buildDataHmm(object)
```

Arguments

object	An srnadiff object.

Value

The selected values: a list of vectors of integers.

---

computePvalues *Compute p-values of the selected counts.*

Description

Compute p-values of the selected counts.

Usage

```
computePvalues(object, counts)
```

Arguments

object	An srnadiff object.
counts	The counts: a list of vectors or integers.

Value

The p-values: a list of numeric.

---

plotRegion *Plot a region*

Description

Plot a region

Usage

```
plotRegion(object, region)
```

Arguments

object	An srnadiff object.
region	A GenomicRange object.

Value

A ggplot object.

Examples

```
exp <- sRNADiffExample()
exp <- runAll(exp)
plotRegion(exp, regions(exp, 0.05)[1])
```

`rcpp_buildHmm` *Compute unique counts.*

Description

Compute unique counts.

Usage

```
rcpp_buildHmm(lengths, values, chromosomeSizes, minDepth)
```

Arguments

`lengths`	the sizes of the RLEs (one list per chromosome)
`values`	the values of the RLEs (one list per chromosome)
`chromosomeSizes`	the sizes of the chromosomes
`minDepth`	the minimum read coverage

Value

the unique counts

`rcpp_naive` *Compute naive method.*

Description

Compute naive method.

Usage

```
rcpp_naive(lengths, values, chromosomeSizes, depth, distance, size)
```

Arguments

`lengths`	the sizes of the RLEs (one list per chromosome)
`values`	the values of the RLEs (one list per chromosome)
`chromosomeSizes`	the sizes of the chromosomes
`depth`	minimum number of reads per position
`distance`	threshold to merge consecutive regions
`size`	minimum region size

Value

the unique counts

rcpp_normalization *Normalize counts (and changes the input values)*

Description

Normalize counts (and changes the input values)

Usage

```
rcpp_normalization(lengths, values, chromosomeSizes, librarySizes)
```

Arguments

lengths	the sizes of the RLEs (one list per chromosome)
values	the values of the RLEs (one list per chromosome)
chromosomeSizes	the sizes of the chromosomes
librarySizes	number of elements per sample

Value

nothing (but transform the values instead)

rcpp_slice *Compute unique counts.*

Description

Compute unique counts.

Usage

```
rcpp_slice(lengths, values, chromosomeSizes, minDepth, minSize, maxSize,
  minDifference)
```

Arguments

lengths	the sizes of the RLEs (one list per chromosome)
values	the values of the RLEs (one list per chromosome)
chromosomeSizes	the sizes of the chromosomes
minDepth	minimum coverage
minSize	minimum region size
maxSize	maximum region size
minDifference	minimum difference between 2 regions

Value

selected regions

rcpp_viterbi *Run the Viterbi algorithm on the HMM.*

Description

Run the Viterbi algorithm on the HMM.

Usage

```
rcpp_viterbi(chromosomeSizes, transitions, emissions, emissionThreshold, starts,
  counts, pvalues, lengths, values, minDepth, minSize, maxSize)
```

Arguments

chromosomeSizes	the sizes of the chromosomes
transitions	the transition log-probabilities
emissions	the emission log-probabilities
emissionThreshold	the emission threshold
starts	the start log-probabilities
counts	the unique counts
pvalues	the p-values of the counts
lengths	the sizes of the RLEs (one list per chromosome)
values	the values of the RLEs (one list per chromosome)
minDepth	the minimum read coverage
minSize	the minimum size region
maxSize	the maximum size region

Value

a segmentation of the chromosomes

readAnnotation *Segmentation using an annotation file.*

Description

Segmentation using an annotation file.

Usage

```
readAnnotation(fileName, source = NULL, feature = NULL, name = NULL)
```

Arguments

fileName	The annotation file name in GFF/GTF format.
source	If not NULL, only lines with this source (2nd field) are imported.
feature	If not NULL, only lines with this feature (3rd field) are imported.
name	If not NULL, use this tag as annotation name.

Value

A GRanges.

Examples

```
dir        <- system.file("extdata", package="srnadiff", mustWork = TRUE)
gtfFile    <- file.path(dir, "Homo_sapiens.GRCh38.76.gtf.gz")
annotation <- readAnnotation(gtfFile, source="miRNA", feature="gene",
                             name="gene_name")
```

---

readMiRBaseMatureAnnotation

Segmentation using an miRBase annotation file and use mature miRNAs.

Description

Segmentation using an miRBase annotation file and use mature miRNAs.

Usage

```
readMiRBaseMatureAnnotation(fileName)
```

Arguments

fileName	The annotation file name in GFF/GTF format.

Value

A GRanges.

Examples

```
dir        <- system.file("extdata", package="srnadiff", mustWork = TRUE)
gffFile    <- file.path(dir, "mirbase21_GRCh38.gff3")
annotation <- readMiRBaseMatureAnnotation(gffFile)
```

readMiRBasePreAnnotation *Segmentation using an miRBase annotation file and use precursor miRNAs.*

Description

Segmentation using an miRBase annotation file and use precursor miRNAs.

Usage

```
readMiRBasePreAnnotation(fileName)
```

Arguments

fileName	The annotation file name in GFF/GTF format.

Value

A GRanges.

Examples

```
dir        <- system.file("extdata", package="srnadiff", mustWork = TRUE)
gffFile    <- file.path(dir, "mirbase21_GRCh38.gff3")
annotation <- readMiRBasePreAnnotation(gffFile)
```

readWholeGenomeAnnotation *Segmentation using an annotation file that contains every genomic feature; it extracts the miRNAs.*

Description

Segmentation using an annotation file that contains every genomic feature; it extracts the miRNAs.

Usage

```
readWholeGenomeAnnotation(fileName)
```

Arguments

fileName	The annotation file name in GFF/GTF format.

Value

A GRanges.

Examples

```
dir        <- system.file("extdata", package="srnadiff", mustWork = TRUE)
gtfFile    <- file.path(dir, "Homo_sapiens.GRCh38.76.gtf.gz")
annotation <- readWholeGenomeAnnotation(gtfFile)
```

---

regions *Get the output regions*

Description

Get the output regions

Usage

```
regions(object, pvalue = 0.05)

## S4 method for signature 'sRNADiff,numeric'
regions(object, pvalue = 0.05)

## S4 method for signature 'sRNADiff,ANY'
regions(object)
```

Arguments

object	An srnadiff object.
pvalue	A minimum p-value

Value

The selected regions

Examples

```
exp <- sRNADiffExample()
regions(exp)
```

---

runAll *Run the segmentation using 3 different methods, and reconciliate them.*

Description

Run the segmentation using 3 different methods, and reconciliate them.

Usage

```
runAll(object)
```

Arguments

object	An srnadiff object.

Value

A GRanges.

Examples

```
exp        <- sRNADiffExample()
exp        <- runAll(exp)
```

---

runAllAnnotation *Segmentation using an annotation file.*

Description

Segmentation using an annotation file.

Usage

```
runAllAnnotation(object)
```

Arguments

object	An srnadiff object.

Value

A GRanges.

---

runAllHmm *Segmentation of the genome using an HMM.*

Description

Segmentation of the genome using an HMM.

Usage

```
runAllHmm(object)
```

Arguments

object	An srnadiff object.

Value

A GRanges object.

runAllNaive *Segmentation of the genome in a naive way.*

Description

Segmentation of the genome in a naive way.

Usage

```
runAllNaive(object)
```

Arguments

object	An srnadiff object.

Value

A GRanges.

runAllSlice *Segmentation of the genome using a slice method.*

Description

Segmentation of the genome using a slice method.

Usage

```
runAllSlice(object)
```

Arguments

object	An srnadiff object.

Value

A GRanges object.

runHmm *Initialize and run the HMM.*

Description

Initialize and run the HMM.

Usage

```
runHmm(object, counts, pvalues)
```

Arguments

object	An srnadiff object.
counts	The counts: a list of vectors or integers.
pvalues	The p-values: a list of numeric.

Value

A GRanges.

runSlice *Initialize and run the slice method.*

Description

Initialize and run the slice method.

Usage

```
runSlice(object)
```

Arguments

object	An srnadiff object.

Value

A GRanges.

`setEmissionProbabilities` *Set emission probabilities (for the HMM step): probability to have a p-value not less than a threshold in the "not-differentially expressed" state, and a p-value not greater than this threshold in the "differentially expressed" state (supposed equal).*

Description

Set emission probabilities (for the HMM step): probability to have a p-value not less than a threshold in the "not-differentially expressed" state, and a p-value not greater than this threshold in the "differentially expressed" state (supposed equal).

Usage

```
setEmissionProbabilities(object, probability)

## S4 method for signature 'sRNADiff,numeric'
setEmissionProbabilities(object, probability)
```

Arguments

`object`	An `srnadiff` object.
`probability`	The emission probability

Value

The same object

Examples

```
exp <- sRNADiffExample()
exp <- setEmissionProbabilities(exp, 0.9)
```

`setEmissionThreshold` *Set emission threshold (for the HMM step): the emission distribution being binomial, all the p-values less than this threshold belong to one class, and all the p-values greater than this threshold belong to the other class.*

Description

Set emission threshold (for the HMM step): the emission distribution being binomial, all the p-values less than this threshold belong to one class, and all the p-values greater than this threshold belong to the other class.

Usage

```
setEmissionThreshold(object, threshold)

## S4 method for signature 'sRNADiff,numeric'
setEmissionThreshold(object, threshold)
```

Arguments

object	An srnadiff object.
threshold	The emission threshold

Value

The same object

Examples

```
exp <- sRNADiffExample()
exp <- setEmissionThreshold(exp, 0.1)
```

---

setMergeDistance *Set the threshold to merge close regions (in the naive step)*

Description

Set the threshold to merge close regions (in the naive step)

Usage

```
setMergeDistance(object, distance)

## S4 method for signature 'sRNADiff,numeric'
setMergeDistance(object, distance)
```

Arguments

object	An srnadiff object.
distance	The maximum distance

Value

The same object

Examples

```
exp <- sRNADiffExample()
exp <- setMergeDistance(exp, 1000)
```

setMinDepth *Set min minimum depth to localize regions*

Description

Set min minimum depth to localize regions

Usage

```
setMinDepth(object, depth)

## S4 method for signature 'sRNADiff,numeric'
setMinDepth(object, depth)
```

Arguments

object	An srnadiff object.
depth	The minimum depth

Value

The same object

Examples

```
exp <- sRNADiffExample()
exp <- setMinDepth(exp, 3)
```

setMinDifferences *Set the threshold to remove similar regions (in the slice step)*

Description

Set the threshold to remove similar regions (in the slice step)

Usage

```
setMinDifferences(object, differences)

## S4 method for signature 'sRNADiff,numeric'
setMinDifferences(object, differences)
```

Arguments

object	An srnadiff object.
differences	The minimum number of different nt.

Value

The same object

Examples

```
exp <- sRNADiffExample()
exp <- setMinDifferences(exp, 10)
```

`setMinOverlap` *Set minimum overlap (for the last quantification step): all the reads with at least n nucleotides shared with a feature will be used for quantification of this feature.*

Description

Set minimum overlap (for the last quantification step): all the reads with at least n nucleotides shared with a feature will be used for quantification of this feature.

Usage

```
setMinOverlap(object, minOverlap)

## S4 method for signature 'sRNADiff,numeric'
setMinOverlap(object, minOverlap)
```

Arguments

`object`	An `srnadiff` object.
`minOverlap`	The minimum overlap

Value

The same object

Examples

```
exp <- sRNADiffExample()
exp <- setMinOverlap(exp, 10)
```

setNThreads *Set number of threads to use*

Description

Set number of threads to use

Usage

```
setNThreads(object, nThreads)

## S4 method for signature 'sRNADiff,numeric'
setNThreads(object, nThreads)
```

Arguments

object	An srnadiff object.
nThreads	The number of threads

Value

The same object

Examples

```
exp <- sRNADiffExample()
exp <- setNThreads(exp, 4)
```

setSizes *Set min and max sizes of the regions*

Description

Set min and max sizes of the regions

Usage

```
setSizes(object, minSize, maxSize)

## S4 method for signature 'sRNADiff,numeric,numeric'
setSizes(object, minSize, maxSize)
```

Arguments

object	An srnadiff object.
minSize	The minimum size.
maxSize	The maximum size.

Value

The same object

Examples

```
exp <- sRNADiffExample()
exp <- setSizes(exp, 10, 1000)
regions(exp)
```

`setStrategies` *Set the different steps*

Description

Set the different steps

Usage

```
setStrategies(object, annotation, naive, hmm, slice)

## S4 method for signature 'sRNADiff,logical,logical,logical,logical'
setStrategies(object,
  annotation, naive, hmm, slice)
```

Arguments

`object`	An `srnadiff` object.
`annotation`	The annotation step.
`naive`	The naive step.
`hmm`	The HMM step.
`slice`	The slice step.

Value

The same object

Examples

```
exp <- sRNADiffExample()
exp <- setStrategies(exp, TRUE, FALSE, TRUE, TRUE)
```

setTransitionProbabilities *Set transition probabilities (for the HMM step).*

Description

Set transition probabilities (for the HMM step).

Usage

```
setTransitionProbabilities(object, noDiffToDiff, diffToNoDiff)

## S4 method for signature 'sRNADiff,numeric,numeric'
setTransitionProbabilities(object,
  noDiffToDiff, diffToNoDiff)
```

Arguments

object	An srnadiff object.
noDiffToDiff	probability to change from the "not-differentially expressed" state to the "differentially expressed" state
diffToNoDiff	probability to change from the "differentially expressed" state to the "not-differentially expressed" state

Value

The same object

Examples

```
exp <- sRNADiffExample()
exp <- setTransitionProbabilities(exp, 0.001, 0.000001)
```

show,sRNADiff-method *Overloading the show method*

Description

Overloading the show method

Usage

```
## S4 method for signature 'sRNADiff'
show(object)
```

Arguments

object	An srnadiff object.

Value

A description of the object.

Examples

```
exp <- sRNADiffExample()
exp
```

`srnadiff` *srnadiff: A package for differential expression of sRNA-Seq.*

Description

The srnadiff package provides uses four strategies to find differentially expressed loci.

Author(s)

Matthias Zytnicki, <matthias.zytnicki@inra.fr>

`sRNADiff-class` *An S4 class to represent sRNA-Seq data for differential expression.*

Description

An S4 class to represent sRNA-Seq data for differential expression.

Slots

`annotation` The annotation in GRanges format.

`bamFileNames` The name of one read file in BAM format.

`bamFiles` The BAM files in a `BamFileList`.

`chromosomes` The names of the chromosomes.

`chromosomeSizes` The sizes of the chromosomes.

`replicates` The names of the replicates.

`conditions` The condition to which each replicate belongs.

`coverages` The coverages, a vector of `RLE`.

`lengths` The lengths parts of the coverages.

`values` The values parts of the coverages.

`design` Experimental design, a `DataFrame` for `DESeq2`

`regions` A `GenomicRanges` of the possibly differentially expressed region

`minDepth` Minimum depth to consider to find regions

`minSize` Minimum region size

`maxSize` Maximum region size

`mergeDistance` Distance to merge consecutive region

`minDifferences` Minimum number of different nt between two regions

`noDiffToDiff` Transition probability

`diffToNoDiff` Transition probability

`emission` Emission probability

`emissionThreshold` Emission threshold

`skipAnnotation` Whether to skip the annotation strategy step

`skipNaive` Whether to skip the naive strategy step

`skipHmm` Whether to skip the HMM strategy step

`skipSlice` Whether to skip the slice strategy step

`nThreads` Number of threads

`sRNADiffExample` *Example constructor*

Description

Example constructor

Usage

```
sRNADiffExample()
```

Value

An `srnadiff` object

Examples

```
exp <- sRNADiffExample()
```

`sRNADiffExp` *Constructor.*

Description

Constructor.

Usage

```
sRNADiffExp(annotation = NULL, bamFileNames, replicates, conditions,
  lazyload = FALSE)
```

Arguments

annotation	The GRanges annotation
bamFileNames	The name of one read file in BAM format.
replicates	The names of the replicates.
conditions	The condition to which each replicate belongs.
lazyload	Usual for S4 functions.

Value

An sRNADiff object.

Examples

```
dir         <- system.file("extdata", package="srnadiff", mustWork = TRUE)
data        <- read.csv(file.path(dir, "data.csv"))
gtfFile     <- file.path(dir, "Homo_sapiens.GRCh38.76.gtf.gz")
annotation  <- readWholeGenomeAnnotation(gtfFile)
bamFiles    <- file.path(dir, data$FileName)
replicates  <- data$SampleName
conditions  <- factor(data$Condition)
exp         <- sRNADiffExp(annotation, bamFiles, replicates, conditions)
```

Index