Package 'srnadiff'

December 9, 2025

Type Package

Title Finding differentially expressed unannotated genomic regions from RNA-seq data

Version 1.30.0

Date 2025-07-24

Description srnadiff is a package that finds differently expressed regions from RNA-seq data at base-resolution level without relying on existing annotation. To do so, the package implements the identify-then-annotate methodology that builds on the idea of combining two pipelines approachs differential expressed regions detection and differential expression quantification. It reads BAM files as input, and outputs a list differentially regions, together with the adjusted p-values.

License GPL-3

Encoding UTF-8

LazyData false

Depends R (>= 3.6)

SystemRequirements C++11

Imports Rcpp (>= 0.12.8), stats, methods, S4Vectors, Seqinfo, rtracklayer, SummarizedExperiment, IRanges, GenomicRanges, DESeq2, edgeR, Rsamtools, GenomicFeatures, GenomicAlignments, grDevices, Gviz, BiocParallel, BiocManager, BiocStyle

LinkingTo Rcpp

RoxygenNote 7.3.2

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

Suggests knitr, rmarkdown, testthat, BiocManager, BiocStyle

VignetteBuilder knitr

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

git_branch RELEASE_3_22

git_last_commit 809d1c9

git_last_commit_date 2025-10-29

Repository Bioconductor 3.22

Date/Publication 2025-12-08

Author Zytnicki Matthias [aut, cre],
Gonzalez Ignacio [aut]

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

Contents

annotReg 2
bamFiles 3
chromosomeSizes 3
countMatrix 4
coverages 5
normFactors 5
parameters 6
plotRegions 8
readAnnotation 10
regions 11
sampleInfo 12
srnadiff 12
srnadiffDefaultParameters 15
srnadiffExample 15
srnadiffExp 16

Index **19**

`annotReg` *Accessors for the 'annotReg' slot of an srnadiffExp object*

Description

The `annotReg` slot holds the annotated regions as a `GRanges` object.

Usage

```
## S4 method for signature 'srnadiffExp'
annotReg(object)

## S4 replacement method for signature 'srnadiffExp'
annotReg(object) <- value
```

Arguments

`object`	An `srnadiffExp` object.
`value`	Annotated regions as a `GRanges` object.

Value

The regions given by the user, as a `GRanges`.

Examples

```
srnaExp <- srnadiffExample()
annotReg(srnaExp)
```

---

bamFiles *Accessors for the 'bamFiles' slot of an srnadiffExp object*

Description

The bamFiles slot holds the full paths to the BAM files as a BamFileList.

Usage

```
## S4 method for signature 'srnadiffExp'
bamFiles(object)
```

Arguments

object	An srnadiffExp object.

Value

The paths to the BAM files.

Examples

```
require(Rsamtools)

srnaExp <- srnadiffExample()
path(bamFiles(srnaExp))
```

---

chromosomeSizes *Accessors for the 'chromosomeSizes' slot of an srnadiffExp object*

Description

The chromosomeSizes slot holds the sizes of the chromosomes as a named vector with chromosome names.

Usage

```
## S4 method for signature 'srnadiffExp'
chromosomeSizes(object)
```

Arguments

object	An srnadiffExp object.

Value

A list containing the chromosome sizes.

Examples

```
srnaExp <- srnadiffExample()
chromosomeSizes(srnaExp)
```

`countMatrix` *Accessors for the 'countMatrix' slot of an srnadiffExp object*

Description

The `countMatrix` slot holds the count matrix from the DERs found.

Usage

```
## S4 method for signature 'srnadiffExp'
countMatrix(object)
```

Arguments

`object`	An `srnadiffExp` object.

Details

In the last step of an sRNA-diff approach, the potential DERs is assessed. Reads (including fractions of reads) that overlap each expressed region are counted to arrive at a count matrix with one row per region and one column per sample. This matrix to been used for quantify the statistic signification of the finded regions.

Value

A matrix with the number of reads for each regions, and each sample.

Examples

```
srnaExp <- srnadiffExample()
srnaExp <- srnadiff(srnaExp)
countMatrix(srnaExp)
```

coverages *Accessors for the 'coverages' slot of an srnadiffExp object*

Description

The coverages slot holds the coverages as a named RleList object with one coverage vector per seqlevel in the set of genomic alignments from the BAM files.

Usage

```
## S4 method for signature 'srnadiffExp'
coverages(object)
```

Arguments

object	An srnadiffExp object.

Value

The coverages, as a list of RleList.

Examples

```
srnaExp <- srnadiffExample()
coverages(srnaExp)
```

normFactors *Accessors for the 'normFactors' slot of an srnadiffExp object*

Description

The normFactors slot holds the normalization factors as a named vector with sample names.

Usage

```
## S4 method for signature 'srnadiffExp'
normFactors(object)

## S4 replacement method for signature 'srnadiffExp,numeric'
normFactors(object) <- value
```

Arguments

object	An srnadiffExp object.
value	A numeric vector, one size factor for each sample in the coverage data.

Details

The `normFactors` vector assigns to each sample coverage a value, the normalization factor, such that count values in each sample coverage can be brought to a common scale by dividing by the corresponding normalization factor. This step is also called normalization, its purpose is to render coverages (counts) from different samples, which may have been sequenced to different depths, comparable. Normalization factors are estimated using the *median ratio method* described by Equation 5 in Anders and Huber (2010). Alternative normalization factor estimators can also be supplied using the assignment function `sizeFactors<-`.

Value

The normalization factors, in a list.

References

Simon Anders, and Wolfgang Huber (2010). Differential expression analysis for sequence count data. *Genome Biology*, 11:106.

Examples

```
srnaExp <- srnadiffExample()
normFactors(srnaExp)
```

`parameters` *Accessors for the 'parameters' slot of an srnadiffExp object*

Description

The `parameters` slot holds the parameter values used in an sRNA-diff approach as a named `list`. Default values exist for parameters, but these can also be supplied as input values in the `useParameters` argument of the `srnadiff` function or using the assignment function `parameters<-`.

Usage

```
## S4 method for signature 'srnadiffExp'
parameters(object)

## S4 replacement method for signature 'srnadiffExp'
parameters(object) <- value

## S3 method for class 'srnadiff_par'
print(x, ...)
```

Arguments

`object`	An `srnadiffExp` object.
`value`	A named `list` containing valid parameters. See details.
`x`	The first element of the parameters used by an `srnadiff` object
`...`	The other elements of the parameters

Details

Parameters in an sRNA-diff approach.

Global parameters:

`minDepth` The cutoff to filter the base-level coverage. Bases where at least one sample has (normalized) coverage greater than `minDepth` be been retained. Default to `10`.

`minSize` The minimum size (in base-pairs) of the regions to be found. Default to `18`.

`maxSize` The maximum size (in base-pairs) of the regions to be found. Default to `1000000`.

`minOverlap` This parameters is used in the construction of the `countMatrix` matrix. Only reads (ranges) with a minimum of `minOverlap` overlapping each expressed region are considered to be overlapping. Default to `10`.

Parameters for the Naive method:

`minGap` The maximum number of different bases between two regions. Near-identical regions are collapsed. Only regions with at most `minGap` different positions are considered identicals and are collapsed into one single region. Default to `20`.

Parameters for the Naive and IR methods:

`minLogFC` The minimun sliding threshold used in the naive and IR method. Default to `0.5`.

Parameters for the HMM method:

`noDiffToDiff` Initial transition probability from no differentially expressed state to differentially expressed. Default to `0.001`.

`diffToNoDiff` Initial transition probability from differentially expressed state to no differentially expressed. Default to `0.000001`.

`emission` Emission probability. Default to `0.9`.

`emissionThreshold` Emission threshold. A real number between `0` and `1`. Default to `0.1`.

Value

The named list of the parameters used in the analysis.

See Also

`useParameters` argument in `srnadiff` function.

Examples

```
srnaExp <- srnadiffExample()
srnaExp <- srnadiff(srnaExp)
print(parameters(srnaExp))

parameters(srnaExp) <- list("minSize" = 1, "maxSize" = 1500)

srnaExp <- srnadiffExample()
srnaExp <- srnadiff(srnaExp)
print(parameters(srnaExp))
```

plotRegions *Plot the coverage information surrounding genomic regions*

Description

This function plot the coverage information surrounding genomic regions while summarizing the annotation.

Usage

```
plotRegions(
  object,
  region,
  normCvg = TRUE,
  annot = NULL,
  allSignReg = TRUE,
  featureAnnot = NULL,
  flankReg = 10,
  colGroup = c("#0080ff", "#ff00ff"),
  fillReg = c("darkgreen", "darkred"),
  fillAnnot = "#FFD58A",
  type = "a",
  chrTitle = TRUE,
  trNames = c("DER", "coverage"),
  legend = TRUE,
  ...
)
```

Arguments

object	An object of class srnadiffExp as returned by srnadiff function.
region	A GRanges object. By example, ranges in the output from regions used on an object of class srnadiffExp.
normCvg	Boolean. If TRUE (the default), normalized coverages are displayed, else, the raw coverages are used.
annot	A GRanges or data.frame object containing gene and/or sRNA annotation information.
allSignReg	Boolean. If TRUE, all differeltially expressed regions contained in the annotated regions will be displayed.
featureAnnot	Character scalar. Feature annotation to be used, it will be one from column names of mcols(annot) if annot is a GRanges or one from colnames(annot) if annot is a data.frame.
flankReg	Integer value. If flankReg is positive, the displayed genomic interval is extended by flankReg nucleotides from the minimum start and maximum end from the displayed (ranges) regions.
colGroup	Character vector of length 2. The fill colors to be used to indicate samples per group.
fillReg	Character vector of length 2. The fill colors to be used to indicate 'up' or 'down' regulated regions.

fillAnnot	Character scalar. The fill color to be used for annotated regions.
type	Character vector. The plot type, one of ("a", "l", "p", "b", "confint") or combinations thereof. See Details for more information on the individual plotting types.
chrTitle	Boolean or character vector of length one. Defaults TRUE, the chromosome name is added to the plot. If character, this will be used for title.
trNames	Title for the tracks. By default "DER" and "coverage".
legend	Boolean triggering the addition of a legend to the (coverage) data track to indicate groups.
...	Additional display parameters to control the look and feel of the plots. See the "Display Parameters" section for functions GenomeAxisTrack, GeneRegionTrack, AnnotationTrack, DataTrack in the Gviz package.

Details

This function provides a flexible genomic visualization framework by displaying tracks in the sense of the Gviz package. Given a region (or regions), four separate tracks are represented: (1) GenomeAxisTrack, a horizontal axis with genomic coordinate tickmarks for reference location to the displayed genomic regions; (2) GeneRegionTrack, if the annot argument is passed, a track displaying all gene and/or sRNA annotation information in a particular region; (3) AnnotationTrack, regions are plotted as simple boxes if no strand information is available, or as arrows to indicate their direction; and (4) DataTrack, plot the sample coverages surrounding the genomic regions.

The sample coverages can be plotted in various different forms as well as combinations thereof. Supported plotting types are:

p: simple dot plot.

l: lines plot.

b: combination of dot and lines plot.

a: lines plot of the sample-groups average (i.e., mean) values.

confint: confidence intervals for average values. In combination with a type.

Value

A list of GenomeGraph track objects, all inheriting from class GdObject.

Examples

```
srnaExp <- srnadiffExample()
srnaExp <- srnadiff(srnaExp)

plotRegions(srnaExp, regions(srnaExp)[1])
plotRegions(srnaExp, regions(srnaExp)[1], type = c("a", "confint"))
```

readAnnotation *Reads and parses GFF/GTF files*

Description

readAnnotation reads and parses content of GFF/GTF files and stores annotated genomic features (regions) in a GRanges object.

Usage

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

Arguments

fileName	A path, URL or connection to the GFF/GTF annotation file. Compressed files ("gz", "bz2" and "xz") are handled transparently.
feature	NULL (the default) or a character vector of valid feature types. If not NULL, then only the features of the specified type(s) are imported.
source	NULL (the default) or a character vector of valid source types. If not NULL, then only the sources of the specified type(s) are imported.
tagName	NULL (the default). If not NULL, use this tag as systematic name for the elements of the GRanges object.

Details

feature and source can be NULL. In this case, no selection is performed and all content into the file is imported. If tagName is NULL, then a systematic name (annot.N) is given to elements of the GRanges object.

Value

A GRanges object with annotated regions information.

See Also

GFFFile-class

Examples

```
##------------------------------------------------------------------------
## Extraction of miRNAs using an GTF annotation file
##------------------------------------------------------------------------
basedir  <- system.file("extdata", package="srnadiff", mustWork = TRUE)
gtfFile  <- file.path(basedir, "Homo_sapiens.GRCh38.76.gtf.gz")
annotReg <- readAnnotation(gtfFile, feature="gene", source="miRNA")
annotReg

##------------------------------------------------------------------------
## Extraction of mature miRNAs using a miRBase-formatted file
##------------------------------------------------------------------------
basedir  <- system.file("extdata", package="srnadiff", mustWork = TRUE)
gffFile  <- file.path(basedir, "mirbase21_GRCh38.gff3")
```

```
annotReg <- readAnnotation(gffFile, feature="miRNA", tagName="miRNA")
annotReg

##------------------------------------------------------------------------
## Extraction of precursor miRNAs using a miRBase-formatted file
##------------------------------------------------------------------------
basedir  <- system.file("extdata", package="srnadiff", mustWork = TRUE)
gffFile  <- file.path(basedir, "mirbase21_GRCh38.gff3")
annotReg <- readAnnotation(gffFile, feature="miRNA_primary_transcript")
annotReg
```

`regions` *Extracts differentially expressed regions of an srnadiffExp object*

Description

This function extracts the differentially expressed regions from `srnadiffExp` object ranked by p-value.

Usage

```
## S4 method for signature 'srnadiffExp'
regions(object, pvalue = 1)
```

Arguments

`object`	An `srnadiffExp` object.
`pvalue`	Numeric cutoff value for adjusted p-values. Only regions with adjusted p-values equal or lower than specified are returned. Default to 1, all regions are returned.

Value

A `GenomicRanges` object of the selected differentially expressed regions.

Examples

```
srnaExp <- srnadiffExample()
srnaExp <- srnadiff(srnaExp)
regions(srnaExp, pvalue = 0.05)
```

sampleInfo *Accessors for the 'sampleInfo' slot of an srnadiffExp object*

Description

The sampleInfo slot holds the sample information as a data.frame with three columns labelled FileName, SampleName and Condition. The first column is the BAM file name (without extension), the second column the sample name, and the third column the condition to which sample belongs. Each row describes one sample.

Usage

```
## S4 method for signature 'srnadiffExp'
sampleInfo(object)
```

Arguments

object	An srnadiffExp object.

Value

A table containing information on the samples

Examples

```
srnaExp <- srnadiffExample()
sampleInfo(srnaExp)
```

srnadiff *Finding differentially expressed unannotated genomic regions from RNA-seq data*

Description

srnadiff is a package that finds differently expressed regions from RNA-seq data at *base-resolution* level without relying on existing annotation. To do so, the package implements the *identify-then-annotate* methodology that builds on the idea of combining two pipelines approach: differential expressed regions detection and differential expression quantification.

This is the main wrapper for running several key functions from this package. It is meant to be used after that a srnadiffExp object has been created. srnadiff implement four methods to produce potential DERs (see Details). Once DERs are detected, the second step in srnadiff is to quantify the statistic signification of these.

Usage

```
srnadiff(
  object,
  segMethod = c("hmm", "IR"),
  diffMethod = "DESeq2",
  useParameters = srnadiffDefaultParameters,
  nThreads = 1
)
```

Arguments

object	An srnadiffExp object.
segMethod	A character vector. The segmentation methods to use, one of 'annotation', 'naive', 'hmm', 'IR' or combinations thereof. Default 'all', all methods are used. See Details.
diffMethod	A character. The differential expression testing method to use, one of 'DESeq2', 'edgeR'. See Details.
useParameters	A named list containing the methods parameters to use. If missing, default parameter values are supplied. See parameters for details.
nThreads	integer(1). Number of workers. Defaults to all cores available as determined by multicoreWorkers.

Details

The srnadiff package implements two major methods to produce potential differentially expressed regions: the HMM and IR method. Briefly, these methods identify contiguous base-pairs in the genome that present differential expression signal, then these are regrouped into genomic intervals called differentially expressed regions (DERs).

Once DERs are detected, the second step in a sRNA-diff approach is to quantify the statistic signification of these. To do so, reads (including fractions of reads) that overlap each expressed region are counted to arrive at a count matrix with one row per region and one column per sample. Then, this count matrix is analyzed using the standard workflow of DESeq2 for differential expression of RNA-seq data, assigning a p-value to each candidate DER. Alternatively, edgeR can be used.

The main functions for finds differently expressed regions are srnadiffExp and srnadiff. The first one creats an S4 class providing the infrastructure (slots) to store the input data, methods parameters, intermediate calculations and results of an sRNA-diff approach. The second one implement four methods to find candidate differentially expressed regions and quantify the statistic signification of the finded regions. Details about the implemented methods are further described in the vignette and the manual page of the srnadiff function.

Implemented methods to produce potential differentially expressed regions in srnadiff are:

annotation: This method simply provides the genomic regions corresponding to the annotation file that is optionally given by the user. It can be a set of known miRNAs, siRNAs, piRNAs, genes, or a combination thereof.

hmm: This approach assumes that continuous regions of RNA along the chromosome are either "differentially expressed" or "not". This is captured with a hidden Markov model (HMM) with binary latent state of each nucleotide: *differentially expressed* or *not differentially expressed*. The observations of the HMM are then the empirical p-values arising from the differential expression analysis corresponding to each nucleotide position. The HMM approach normally needs emission, transition, and starting probabilities values (see parameters). They can be

tuned by the user. In order to finding the most likely sequence of states from the HMM, the Viterbi algorithm is performed. This essentially segments the genome into regions, where a region is defined as a set of consecutive bases showing a common expression signature.

`IR`: In this approach, for each base, the average from the normalized coverage is calculated across all samples into each condition. This generates a vector of (normalized) mean coverage expression per condition. These two vectors are then used to compute per-nucleotide log-ratios (in absolute value) across the genome. For the computed log-ratio expression, the method uses a sliding threshold *h* that run across the log-ratio levels identifying bases with log-ratio value above of *h*. Regions of contiguous bases passing this threshold are then analyzed using an adaptation of Aumann and Lindell algorithm for irreducibility property (Aumann and Lindell (2003)).

`naive`: This method is the simplest, gived a fixed threshold *h*, contiguous bases with log-ratio expression (in absolute value) passing this threshold are then considered as candidate differentially expressed regions.

Value

An `srnadiffExp` object containing additional slots for:

- `regions`
- `parameters`
- `countMatrix`

Author(s)

Matthias Zytnicki and Ignacio González

References

Aumann Y. and, Lindell Y. (2003). A Statistical Theory for Quantitative Association Rules. *Journal of Intelligent Information Systems*, 20(3):255-283.

See Also

`regions`, `parameters`, `countMatrix` and `srnadiffExp`

Examples

```
## A typical srnadiff session might look like the following.

## Here we assume that 'bamFiles' is a vector with the full
## paths to the BAM files and the sample and experimental
## design information are stored in a data frame 'sampleInfo'.

## Not run:

#-- Data preparation
srnaExp <- srnadiffExp(bamFiles, sampleInfo)

#-- Detecting DERs and quantifying differential expression
srnaExp <- srnadiff(srnaExp)

#-- Visualization of the results
plotRegions(srnaExp, regions(srnaExp)[1])
```

```
## End(Not run)

srnaExp <- srnadiffExample()
srnaExp <- srnadiff(srnaExp)
srnaExp
```

`srnadiffDefaultParameters` *List of srnadiff default parameters.*

Description

List of srnadiff default parameters.

Usage

```
srnadiffDefaultParameters
```

Format

An object of class `list` of length 10.

Examples

```
srnaExp <- srnadiffExample()
srnaExp <- srnadiff(srnaExp, useParameters=srnadiffDefaultParameters)
```

`srnadiffExample` *Example constructor*

Description

This function provides an example of a `srnadiffExp` object which contains mapped reads of small RNAs extracted from SLK cells infected with latent KSHV, compared to uninfected SLK cells.

Usage

```
srnadiffExample()
```

Details

Raw data have been downloaded from the GEO data set GSE62830, provided in Viollet et *al.* (2015). Adapters were removed with fastx_clipper and mapped with bowtie2 (Salzberg and Langmead, 2012) on the human genome version GRCh38.

This example is restricted to a small locus on chr14. It uses the whole genome annotation (with coding genes, etc.) and extracts miRNAs.

Value

An srnadiffExp object called 'srnaExp'.

References

Viollet, Coralie, David A. Davis, Martin Reczko, Joseph M. Ziegelbauer, Francesco Pezzella, Jiannis Ragoussis, and Robert Yarchoan (2015). "Next-Generation Sequencing Analysis Reveals Differential Expression Profiles of MiRNA-mRNA Target Pairs in KSHV-Infected Cells." *PLOS ONE*, 10:1–23.

Salzberg, Steven, and Ben Langmead (2012). "Fast gapped-read alignment with Bowtie 2." *Nature Methods*, 9:357–59.

Examples

```
## The 'srnadiffExp' object in this example was constructed by:

## Not run:

basedir             <- system.file("extdata", package="srnadiff", mustWork = TRUE)
sampleInfo          <- read.csv(file.path(basedir, "dataInfo.csv"))
gtfFile             <- file.path(basedir, "Homo_sapiens.GRCh38.76.gtf.gz")
annotReg            <- readAnnotation(gtfFile, feature="gene", source="miRNA")
bamFiles            <- file.path(basedir, sampleInfo$FileName)
srnaExp             <- srnadiffExp(bamFiles, sampleInfo, annotReg)
parameters(srnaExp) <- srnadiffDefaultParameters
save(srnaExp, file = file.path(basedir, "srnadiffExample.rda"))

## End(Not run)

srnaExp <- srnadiffExample()
srnaExp
```

---

srnadiffExp *Infrastructure for sRNA-Seq experiment and differential expression*

Description

srnadiffExp is an S4 class providing the infrastructure (slots) to store the input data, methods parameters, intermediate calculations and results of a sRNA-diff approach.

Usage

```
srnadiffExp(
  bamFiles = NULL,
  sampleInfo = NULL,
  annotReg = NULL,
  diffMethod = NULL,
  normFactors = NULL
)

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

Arguments

bamFiles	A vector with the full paths to the BAM files.
sampleInfo	A data.frame with three columns labelled FileName, SampleName and Condition. The first column is the BAM file name (without extension), the second column the sample name, and the third column the condition to which sample belongs. Each row describes one sample.
annotReg	Optional annotation information. Annotated regions as a GRanges object. By example, ranges in the output from readAnnotation.
diffMethod	A character storing the name of the method used to compute the p-values. Can be DESeq2 (default), or edgeR.
normFactors	A numeric vector, one size factor for each sample in the data.
object	An srnadiffExp object.

Details

srnadiffExp load and summarize sample BAM files into base-resolution coverage and estimate the size factors (the effective library size) from the coverage data.

To facilitate programming pipelines, NULL values are input for regions, parameters and countMatrix slots, in which case the default value is used as if the argument had been missing. These slots will be updated after differential expression (srnadiff) approach.

Value

srnadiffExp constructor returns an srnadiffExp object of class S4.

The show method informatively display object contents.

Slots

bamFiles A BamFileList object with the full paths to the BAM files.

sampleInfo A data.frame with sample and experimental design information. Each row describes one sample.

annotReg A GRanges with annotation information.

diffMethod A character storing the name of the method used to compute the p-values. Can be DESeq2, or edgeR.

chromosomeSizes A named vector with the sizes of the chromosomes.

coverages The sample coverages, a named RleList object.

normFactors A vector of normalization factors.

regions A GenomicRanges of the candidate differentially expressed regions.

countMatrix A matrix of non-negative integer count values, one row per region and one column per sample.

parameters An named list. The parameters for the segmentation methods. See parameters.

Examples

```
basedir    <- system.file("extdata", package="srnadiff", mustWork = TRUE)
sampleInfo <- read.csv(file.path(basedir, "dataInfo.csv"))
gtfFile    <- file.path(basedir, "Homo_sapiens.GRCh38.76.gtf.gz")
annotReg   <- readAnnotation(gtfFile, feature="gene", source="miRNA")
bamFiles   <- file.path(basedir, sampleInfo$FileName)

srnaExp <- srnadiffExp(bamFiles, sampleInfo, annotReg)
srnaExp
```

Index

∗ **data**
srnadiffDefaultParameters, 15
∗ **package**
srnadiff, 12

AnnotationTrack, *9*
annotReg, 2
annotReg,srnadiffExp-method *(*annotReg*)*, 2
annotReg<- *(*annotReg*)*, 2
annotReg<-,srnadiffExp-method *(*annotReg*)*, 2

BamFileList, *3*, *17*
bamFiles, 3
bamFiles,srnadiffExp-method *(*bamFiles*)*, 3

chromosomeSizes, 3
chromosomeSizes,srnadiffExp-method *(*chromosomeSizes*)*, 3
countMatrix, 4, *7*, *14*
countMatrix,srnadiffExp-method *(*countMatrix*)*, 4
coverages, 5
coverages,srnadiffExp-method *(*coverages*)*, 5

DataTrack, *9*

GeneRegionTrack, *9*
GenomeAxisTrack, *9*

multicoreWorkers, *13*

normFactors, 5
normFactors,srnadiffExp-method *(*normFactors*)*, 5
normFactors<- *(*normFactors*)*, 5
normFactors<-,srnadiffExp,numeric-method *(*normFactors*)*, 5

parameters, 6, *13*, *14*, *17*
parameters,srnadiffExp-method *(*parameters*)*, 6
parameters<- *(*parameters*)*, 6
parameters<-,srnadiffExp-method *(*parameters*)*, 6
plotRegions, 8
print.srnadiff_par *(*parameters*)*, 6

readAnnotation, 10, *17*
regions, *8*, 11, *14*
regions,srnadiffExp-method *(*regions*)*, 11
RleList, *5*, *17*

sampleInfo, 12
sampleInfo,srnadiffExp-method *(*sampleInfo*)*, 12
show,srnadiffExp-method *(*srnadiffExp*)*, 16
srnadiff, *6–8*, *12*, 12, *13*, *17*
srnadiff-package *(*srnadiff*)*, 12
srnadiffDefaultParameters, 15
srnadiffExample, 15
srnadiffExp, *8*, *11–14*, 16
srnadiffExp-class *(*srnadiffExp*)*, 16