

Package ‘ASpli’

October 17, 2017

Type Package

Title Analysis of alternative splicing using RNA-Seq

Version 1.2.4

Date 2017-07-17

Author Estefania Mancini, Marcelo Yanovsky and Ariel Chernomoretz

License GPL

biocViews GeneExpression, Transcription, AlternativeSplicing,
Coverage, DifferentialExpression, DifferentialSplicing,
TimeCourse, RNASeq, GenomeAnnotation, Sequencing, Alignment

Depends methods, grDevices, stats, utils, parallel

Imports DESeq2, DEXSeq, GenomicRanges, GenomicFeatures, edgeR,
BiocGenerics, IRanges, GenomicAlignments, Gviz, S4Vectors,
AnnotationDbi, limma

Suggests RNAseqData.HNRNPC.bam.chr14, BiocStyle

Maintainer

Estefania Mancini <emancini@leloir.org.ar>, Javier Iserte <jiserte@leloir.org.ar>

Description Integrative pipeline for the analysis of alternative
splicing using RNA-seq.

NeedsCompilation no

R topics documented:

ASpli-package	2
AS accesors	3
AsDiscover	4
ASpliAS-class	5
ASpliCounts	6
ASpliCounts-class	7
ASpliDU-class	8
ASpliFeatures-class	8
binGenome	9
binGenome-methods	10
Counts accesors	10
DU accesors	11
DUreport	12
DUreport_DEXSeq	13

features accesors	14
loadBAM	15
plotTopTags	16
rds	17
readCounts	17
show-methods	18
write	19
write-methods	19
Index	20

ASpli-package	<i>Analysis of alternative splicing using RNAseq</i>
---------------	--

Description

ASpli is an integrative and flexible package that facilitates the characterization of genome-wide changes in AS under different experimental conditions. ASpli analyzes the differential usage of introns, exons, and splice junctions using read counts, and estimates the magnitude of changes in AS by calculating differences in the percentage of exon inclusion or intron retention using splice junctions. This integrative approach allows the identification of changes in both annotated and novel AS events. ASpli allows users to produce self-explanatory intermediate outputs, based on the aim of their analysis. A typical workflow involves parsing the genome annotation into new features called bins, overlapping read alignments against those bins, and inferring differential bin usage based on the number of reads aligning to the bins and junctions.

Details

Package: ASpli

Type: Package

Version: 1.2.3

Date: 2017-07-18

License: GPL

Depends: methods, GenomicRanges, GenomicFeatures, edgeR, methods, BiocGenerics, IRanges, GenomicAlignments,

Author(s)

Estefania Mancini, Marcelo Yanovsky and Ariel Chernomoretz

References

- Acute effects of light on alternative splicing in light-grown plants. *Photochemistry and Photobiology*. Mancini, E, Sanchez, S, Romanowsky, A, Yanovsky, MJ. DOI: 10.1111/php.12550
- GEMIN2 attenuates the effects of temperature on alternative splicing and circadian rhythms in *Arabidopsis thaliana*. *Proceedings of the National Academy of Sciences*. Schlaen, RG, Mancini, E, Sanchez, SE, Perez-Santangelo, S, Rugnone, ML, Simpson, CG, Brown, JWS, Zhang, X, Chernomoretz, A, Yanovsky, MJ. DOI:10.1073/pnas.1504541112

- Genome wide comparative analysis of the effects of PRMT5 and PRMT4/CARM1 arginine methyltransferases on the Arabidopsis thaliana transcriptome. BMC Genomics. Hernando, E, Sanchez, S, Mancini, E, Yanovsky MJ. DOI:10.1186/s12864-015-1399-2
- A role for LSM genes in the regulation of circadian rhythms. Proceedings of the National Academy of Sciences. Perez Santangelo, S, Mancini, E, Francey, LJ, Schlaen, RG, Chernomoretz, A, Hogenesch, JB, Yanovsky MJ. DOI: 10.1073/pnas.1409791111

Examples

```
library(RNAseqData.HNRNPC.bam.chr14)
library(AnnotationDbi)
chr14 <- system.file("extdata", "chr14.sqlite", package="ASpli")
genome <- loadDb(chr14)
features <- binGenome(genome)
targets <- data.frame(bam=RNAseqData.HNRNPC.bam.chr14_BAMFILES,
                     condition=c(rep("CT",4),rep("KD",4)))
bam <- loadBAM(targets)
counts <- readCounts(features, bam, l=100L, targets = targets, maxISize=50000);
group <- factor(c(rep("CT",4),rep("KD",4)))
pair <- c("CT","KD")
du <- DUreport(counts, targets, pair, group)
as <- AsDiscover(counts, targets, features, bam, threshold=5, l=100, pair=pair)
```

AS accesors

Accessors for ASpliAS object

Description

Accessors for ASpliAS object

Usage

```
altPSI(x)
esPSI(x)
irPIR(x)
joint(x)
junctionsPIR(x)
junctionsPSI(x)
```

Arguments

x An ASpliAS object

Value

Returns dataframes with genomic metadata and PSI and PIR metrics

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

Examples

```

library(AnnotationDbi)
chr14 <- system.file("extdata","chr14.sqlite", package="ASpli")
genome <- loadDb(chr14)
features <- binGenome(genome)
library(RNAseqData.HNRNPC.bam.chr14)
targets <- data.frame(bam=RNAseqData.HNRNPC.bam.chr14_BAMFILES,
                     condition=c(rep("CT",4),rep("KD",4)))
bam <- loadBAM(targets)
counts <- readCounts(features, bam, l=100L, targets = targets, maxISize=50000);
group <- factor(c(rep("CT",4),rep("KD",4)))
pair <- c("CT","KD")
as <- AsDiscover(counts, targets, features, bam, threshold=5, l=100, pair=pair)
altPSI(as)
esPSI(as)
irPIR(as)
joint(as)
junctionsPIR(as)
junctionsPSI(as)

```

AsDiscover

*Report PSI and PIR using experimental junctions***Description**

Given a bin, it is possible to calculate PSI/PIR metric using junctions to estimate changes in the use of it along different conditions.

Usage

```

AsDiscover(counts,
           targets,
           features,
           bam,
           l,
           pair,
           threshold,
           cores)

```

Arguments

counts	An object of class ASpliCounts.
targets	A dataframe containing sample, bam and condition columns
features	An object of class ASpliFeatures.
bam	A list with BAM files
l	Read length of sequenced read. Default 100L
pair	Vector of length two, either numeric or character, providing the pair of groups to be compared
threshold	Minimum number of reads supporting junctions. Default=5
cores	Number of procesors to use

Value

An object of class ASpliAS

<code>irPIR</code>	reports: event, <code>e1i</code> counts (J1), <code>ie1</code> counts (J2), <code>j_within</code> (J3), PIR by condition. J1, J2, J3 sum of junctions (J1, J2, J3) by condition.
<code>altPSI</code>	reports: event, J1 (start), J2 (end), J3 (exclusion), PSI. J1, J2, J3 sum of junctions (J1, J2, J3) by condition.
<code>esPSI</code>	reports: event, J1 (start), J2 (end), J3 (exclusion), PSI. J1, J2, J3 sum of junctions (J1, J2, J3) by condition.
<code>junctionsPIR</code>	PIR metric for each experimental junction using <code>e1i</code> and <code>ie2</code> counts. Exclusion junction is the junction itself. This output helps to discover new introns as well as new retention events
<code>junctionsPSI</code>	Given a junction, it is possible to analyze if it shares start, end or both with another junction. If so, is because there is more than one way for/of splicing. Using strand information it is possible to classify those pair of junctions into Alt5'ss, Alt3'ss or ES. Ratio between them along samples is reported.

Author(s)

Estefania Mancini, Marcelo Yanovsky and Ariel Chernomoretz

See Also

Accesors: `irPIR`, `altPSI`, `esPSI`, `junctionsPIR`, `junctionsPSI` Export: `writeAS`

Examples

```
library(RNAseqData.HNRNPC.bam.chr14)
library(AnnotationDbi)
chr14 <- system.file("extdata", "chr14.sqlite", package="ASpli")
genome <- loadDb(chr14)
features <- binGenome(genome)
targets <- data.frame(bam=RNAseqData.HNRNPC.bam.chr14_BAMFILES,
                     condition=c(rep("CT", 4), rep("KD", 4)))
bam <- loadBAM(targets)
counts <- readCounts(features, bam, l=100L, targets = targets, maxISize=50000)
group <- factor(c(rep("CT", 4), rep("KD", 4)))
pair <- c("CT", "KD")
as <- AsDiscover(counts, targets, features, bam, l=100L, pair=pair)
writeAS(as=as, output.dir="only_as")
```

ASpliAS-class

Class "ASpliAS"

Description

Results of PSI and PIR using experimental junctions

Slots

irPIR: Reports: event, e1i counts (J1), ie1 counts (J2), j_within (J3), PIR by condition. J1, J2, J3 sum of junctions (J1, J2, J3) by condition.

altPSI: Reports: event, J1 (start), J2 (end), J3 (exclusion), PSI. J1, J2, J3 sum of junctions (J1, J2, J3) by condition.

esPSI: Reports: event, J1 (start), J2 (end), J3 (exclusion), PSI. J1, J2, J3 sum of junctions (J1, J2, J3) by condition.

join: It is a combination of irPIR, altPSI and esPSI tables

junctionsPIR: PIR metric for each experimental junction using e1i and ie2 counts. Exclusion junction is the junction itself. This output helps to discover new introns as well as new retention events

junctionsPSI: Given a junction, it is possible to analyze if it shares start, end or both with another junction. If so, is because there is more than one way for/of splicing. Using strand information it is possible to classify those pair of junctions into Alt5'ss, Alt3'ss or ES. Ratio between them along samples is reported.

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

See Also

Methods: [ASDiscover](#), Accessors: [irPIR](#), [esPSI](#), [junctionsPIR](#), [junctionsPSI](#)

ASpliCounts

Class "ASpliCounts"

Description

Contains results of read overlaps against all feature levels summarization

Slots

gene.counts

exon.intron.counts

junction.counts

e1i.counts

ie2.counts

gene.rd

bin.rd

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

ASpliCounts-class *Class "ASpliCounts"*

Description

Contains results of read overlaps against all feature levels summarization

Slots

gene.counts: Object of class "data.frame"
exon.intron.counts: Object of class "data.frame"
junction.counts: Object of class "data.frame"
eli.counts: Object of class "data.frame"
ie2.counts: Object of class "data.frame"
gene.rd: Object of class "data.frame"
bin.rd: Object of class "data.frame"

Methods

AsDiscover psi and pir metrics
countsb bin counts accesor
countseli eli counts accesor
countsg gene counts accesor
countsie2 ie2 counts accesor
countsj junction counts accesor
DUreport_DEXSeq differential expression and usage estimation using DEXSeq
DUreport differential expression and usage estimation using DEXSeq
rdsb bin read densities accesor
rdsg gen read densities accesor
rds compute read densities on genes and bins
writeCounts Export count tables
writeRds Export read density tables

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

ASpliDU-class	Class "ASpliDU"
---------------	-----------------

Description

Contains results of differential expression at gene level and differential usage at bin and junction level estimation using DEreport method.

Slots

genes

bins

junctions

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

ASpliFeatures-class	Class "ASpliFeatures"
---------------------	-----------------------

Description

Contains Genomic Ranges of different features extracted from a TxDb

Slots

genes:

bins:

junctions:

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

binGenome

*Feature coordinates extraction***Description**

Exons and introns are subdivided into new features called exon and intron bins and are then classified into exclusively exonic bins, exclusively intronic bins or alternative splicing (AS) bins .

Usage

```
binGenome(genome, md = NULL)
```

Arguments

genome	An object of class transcriptDb (TxDb)
md	A dataframe with symbol (common names) of TxDb genes. If there isn't md file, gene name will be repeated

Details

Exon and intron coordinates are extracted from gene annotation, only those from multi-exonic genes are saved for further evaluation. In case more than one isoform exist, some exons and introns will overlap. Exons and introns are then disjoint into new features called exon and intron bins, and then they are classified into exclusively exonic bins, exclusively intronic bind or alternative splicing bins (AS-bins), which are labeled according to which alternative splicing event are assumed to came from:

- ES: exon skipping
- IR: intron retention
- Alt5l3'ss: alternative five/three prime splicing site
- "*" (ES*, IR*, AltSS*) means this AS bin/region is involved simultaneously in more than one AS event type
- external: from the beginning or the end of a transcript

Subgenic features are labeled as follow (hypothetical GeneAAA):

- GeneAAA:E001: defines first exonic bin
- GeneAAA:I001: defines first intronic bin
- GeneAAA:Io001: defines first intron before disjoint into bins
- GeneAAA:J001: defines first junction

Junctions are defined as the last position of five prime exon (donor position) and first position of three prime exon (acceptor position). Using TxDb object, it is possible to extract annotated/known junctions. This information will be useful for the analysis of "experimental" junctions (reads aligned with gaps). Bins and junctions are labelled always in 5' to 3' sense. This notation is strand independent. It implies that bin / junction with lower numbering is always at 5'.

Value

An ASpliFeatures object. It is a list of features using GRanges format.

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

See Also

[featuresg](#), [featuresb](#) , [featuresj](#)

Examples

```
library(AnnotationDbi)
chr14 <- system.file("extdata","chr14.sqlite", package="ASpli")
genome <- loadDb(chr14)
features <- binGenome(genome)
GeneCoord <- featuresg(features)
BinCoord <- featuresb(features)
JunctionCoord <- featuresj(features)
```

binGenome-methods

Feature coordinates extraction

Description

Feature coordinates extraction from a Transcript Db Database

Methods

signature(genome = "TxDb") An object of class transcriptDb (TxDb)

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

See Also

[featuresg](#), [featuresb](#) , [featuresj](#)

Counts accesors

Accessors for ASpliCounts object

Description

Accessors for ASpliCounts object

Usage

```
countsb(x)
countseli(x)
countsg(x)
countsie2(x)
countsj(x)
rdsg(x)
rdsb(x)
```

Arguments

x An ASpliCounts object

Value

Returns dataframes with counts by sample and genomic metadata

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

Examples

```
library(AnnotationDbi)
chr14 <- system.file("extdata", "chr14.sqlite", package="ASpli")
genome <- loadDb(chr14)
features <- binGenome(genome)
library(RNAseqData.HNRNPC.bam.chr14)
targets <- data.frame(bam=RNAseqData.HNRNPC.bam.chr14_BAMFILES,
                     condition=c(rep("CT",4),rep("KD",4)))
bam <- loadBAM(targets)
counts <- readCounts(features, bam, l=100L, targets = targets, maxISize=50000);
countsb(counts)
countse1i(counts)
countsg(counts)
countsie2(counts)
countsj(counts)
rdsg(counts)
rdsb(counts)
```

DU accesors

Accessors for ASpliDU object

Description

Accessors for ASpliDU object

Usage

```
genesDE(x)
binsDU(x)
junctionsDU(x)
```

Arguments

x An ASpliDU object

Value

Returns dataframes with genomic metadata and logFC and pvalue

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

Examples

```
library(AnnotationDbi)
chr14 <- system.file("extdata", "chr14.sqlite", package="ASpli")
genome <- loadDb(chr14)
features <- binGenome(genome)
library(RNAseqData.HNRNPC.bam.chr14)
targets <- data.frame(bam=RNAseqData.HNRNPC.bam.chr14_BAMFILES,
                      condition=c(rep("CT",4),rep("KD",4)))
bam <- loadBAM(targets)
counts <- readCounts(features, bam, l=100L, targets = targets, maxISize=50000);
group <- factor(c(rep("CT",4),rep("KD",4)))
pair <- c("CT", "KD")
du <- DUreport(counts, targets, pair, group)
genesDE(du)
binsDU(du)
junctionsDU(du)
```

DUreport

Differential gene expression and differential bin/junction usage estimation

Description

Estimate differential expression at gene level and differential usage at bin and junction level.

Usage

```
DUreport(counts, targets, pair, group, minGenReads, minBinReads, minRds, ignoreExternal, threshold)
```

Arguments

counts	An object of class ASpliCounts
targets	A dataframe containing sample, bam and condition columns
pair	vector of length two, either numeric or character, providing the pair of groups to be compared
group	Factorial vector with tags for each sample
minGenReads	Default 10 reads
minBinReads	Default 5 reads
minRds	Default 0.05
ignoreExternal	Ignore Exon Bins at the beginning or end of the transcript. Default TRUE
threshold	Minimum number of junction. Default 5

Value

An ASpliDU object with results at genes, bins and junctions level

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

See Also

DEXSeq, edgeR Accessors: genesDE, binsDU, junctionsDU Export: writeDU

Examples

```
library(RNAseqData.HNRNPC.bam.chr14)
library(AnnotationDbi)
chr14 <- system.file("extdata", "chr14.sqlite", package="ASpli")
genome <- loadDb(chr14)
features <- binGenome(genome)
targets <- data.frame(bam=RNAseqData.HNRNPC.bam.chr14_BAMFILES,
                    condition=c(rep("CT",4),rep("KD",4)))
bam <- loadBAM(targets)
counts <- readCounts(features, bam, l=100L, targets = targets, maxISize=50000)
group <- factor(c(rep("CT",4),
                 rep("KD",4)))
pair <- c("CT", "KD")
du <- DUreport(counts, targets, pair, group)
writeDU(du, output.dir="only_du")
```

DUreport_DEXSeq

Differential gene expression and differential bin/junction usage estimation

Description

Estimate differential expression at gene level and differential usage at bin and junction level.

Usage

```
DUreport_DEXSeq(counts, targets, pair, group, minGenReads, minBinReads, minRds, threshold)
```

Arguments

counts	An object of class ASpliCounts
targets	A dataframe containing sample, bam and condition columns
pair	vector of length two, either numeric or character, providing the pair of groups to be compared
group	Factorial vector with tags for each sample
minGenReads	Default 10 reads
minBinReads	Default 5 reads
minRds	Default 0.05
threshold	Minimum number of junction. Default 5

Value

An ASpliDU object with results at genes, bins and junctions level

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

See Also

DEXSeq, edgeR Accessors: genesDE, binsDU, junctionsDU Export: writeDU

Examples

```
library(RNAseqData.HNRNPC.bam.chr14)
library(AnnotationDbi)
chr14 <- system.file("extdata", "chr14.sqlite", package="ASpli")
genome <- loadDb(chr14)
features <- binGenome(genome)
targets <- data.frame(bam=RNAseqData.HNRNPC.bam.chr14_BAMFILES,
                     condition=c(rep("CT",4),rep("KD",4)))
bam <- loadBAM(targets)
counts <- readCounts(features, bam, l=100L, targets = targets, maxISize=50000)
group <- factor(c(rep("CT",4),
                  rep("KD",4)))
pair <- c("CT", "KD")
du <- DUreport_DEXSeq(counts, targets, pair, group)
writeDU(du, output.dir="only_du")
```

features accesors *Accessors for ASpliFeatures object*

Description

Accessors for ASpliFeatures object

Usage

```
featuresg(x)
featuresb(x)
featuresj(x)
```

Arguments

x An ASpliFeatures object

Value

Returns a GenomicRanges object

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

Examples

```
library(AnnotationDbi)
chr14 <- system.file("extdata", "chr14.sqlite", package="ASpli")
genome <- loadDb(chr14)
features <- binGenome(genome)
featuresg(features)
featuresb(features)
featuresj(features)
```

loadBAM

Load BAM files

Description

Load BAM files into R session using targets object especification

Usage

```
loadBAM(targets, cores)
```

Arguments

targets	A dataframe containing sample, bam and condition columns
cores	Number of procesors to use

Value

A list of GAlignments. Each element of the list correspond to a BAM file (or sample)

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

Examples

```
library(RNAseqData.HNRNPC.bam.chr14)
targets <- data.frame(bam=RNAseqData.HNRNPC.bam.chr14_BAMFILES,
                     condition=c(rep("CT", 4), rep("KD", 4)))
targets
bam <- loadBAM(targets)
```

plotTopTags

Coverage plots

Description

Using genomic coordinates and BAM files this function is useful for make coverage plots

Usage

```
plotTopTags(auxdf, genome, targetsPlot, output.dir)
```

Arguments

auxdf	A data frame: row.names=bin names, gene coordinates, bin coordinates and event name columns
genome	TxDb genome
targetsPlot	A dataframe containing: bam files name, condition (y axe tag), color for each condition
output.dir	Name of directory where plots are supposed to be exported

Value

Coverage plots in png format of selected events

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

Examples

```
library(RNAseqData.HNRNPC.bam.chr14)
library(AnnotationDbi)
chr14 <- system.file("extdata", "chr14.sqlite", package="ASpli")
genome <- loadDb(chr14)
features <- binGenome(genome)
targets <- data.frame(bam=RNAseqData.HNRNPC.bam.chr14_BAMFILES,
                     condition=c(rep("CT",4),rep("KD",4)))
bam <- loadBAM(targets)
counts <- readCounts(features, bam, l=100L, targets = targets, maxISize = 50000)
pair <- c("CT", "KD")
group <- c(rep("CT", 4), rep("KD", 4))
du_HNRNPC <- DUreport(counts, targets, pair, group)
bins <- binsDU(du_HNRNPC)
topTagsBins <- which(bins$bin.fdr <= 0.1 &
                    abs(bins$logFC) >= 0.58)
targetsPlot <- data.frame(bam=targets$bam,
                         sample=targets$condition,
                         color=c(rep("blue", 4), rep("red", 4)),
                         stringsAsFactors=FALSE)

auxdf<-bins[topTagsBins,]
#for simplicity, just one: LRR1:E005
```



```
plotTopTags(auxdf["LRR1:E005",],
            genome,
            targetsPlot,
            output.dir="testPlots")
```

rds

Divides read counts by gene and bin length

Description

Divides read counts by gene and bin length

Usage

```
rds(counts, targets)
```

Arguments

counts	An ASpliCounts object
targets	Target dataframe

Value

Read densities of genes and bins

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

readCounts

Summarize read overlaps

Description

Summarize read overlaps against all feature levels

Usage

```
readCounts(features, bam, cores, 1, targets, maxISize, minAnchor)
```

Arguments

features	An object of class ASpliFeatures. It is a list of GRanges at gene, bin and junction level
bam	List of bam files
l	Read length of sequenced library. It is used for compute E1I and IE2 read summarization
targets	A dataframe containing sample, bam and condition columns
maxISize	maximum intron expected size. Junctions longer than this size will be dicarded
cores	Number of cores to use. Default 1
minAnchor	Percentage of read thath sould be aligned in exon-intron boundary

Value

An object of class ASpliCounts. Each slot is a dataframe containing features metadata and read counts. Summarization is reported at gene, bin, junction and intron flanking regions (E1I, IE2)

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

See Also

Accesors: [countsg](#), [countsb](#), [countsj](#), [countseli](#), [countsie2](#), [rdsj](#), [rdsb](#) Export: [writeCounts](#)

Examples

```
library(RNAseqData.HNRNPC.bam.chr14)
library(AnnotationDbi)
chr14 <- system.file("extdata", "chr14.sqlite", package="ASpli")
genome <- loadDb(chr14)
features <- binGenome(genome)
targets <- data.frame(bam=RNAseqData.HNRNPC.bam.chr14_BAMFILES,
                     condition=c(rep("CT",4),rep("KD",4)))
bam <- loadBAM(targets)
counts <- readCounts(features, bam, l=100L, targets = targets, maxISize=50000)#OK
writeCounts(counts,output.dir="only_counts")
```

show-methods

Display a summary of data contained in ASpliObjects

Description

Display a summary of data contained in ASpliObjects

Details

Display a summary of data contained in ASpliObjects

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

write	<i>Write results</i>
-------	----------------------

Description

Export tab delimited files in structured output

Usage

```
writeCounts(counts, output.dir="counts")
writeRds(counts, output.dir="rds")
writeDU(du, output.dir="du")
writeAS(as, output.dir="as")
writeAll(counts, du, as, output.dir="output")
```

Arguments

counts	An ASpliCounts object
as	An ASpliAS object
du	An ASpliDU object
output.dir	Name of output folder (new or existing)

Value

Tab delimited files are exported in a tidy manner into output folder

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

See Also

[AsDiscover](#), [binGenome](#), [DUreport](#)

write-methods	<i>Write results</i>
---------------	----------------------

Description

Export tab delimited files in structured output

Details

Tab delimited files are exported in a tidy manner into output folder

Author(s)

Estefania Mancini, Marcelo Yanovsky, Ariel Chernomoretz

See Also

[AsDiscover](#), [binGenome](#), [DUreport](#)

Index

*Topic **alternative splicing, RNA-seq, junctions**

- ASpli-package, 2
- altPSI, 5
- altPSI (AS accesor), 3
- altPSI, ASpliAS-method (ASpliAS-class), 5
- AS accesor, 3
- ASDiscover, 4, 6, 19
- ASDiscover, ASpliCounts-method (ASpliCounts-class), 7
- ASpli (ASpli-package), 2
- ASpli-package, 2
- ASpliAS-class, 5
- ASpliCounts, 6
- ASpliCounts-class, 7
- ASpliDU-class, 8
- ASpliFeatures-class, 8

- binGenome, 9, 19
- binGenome, TxDdb-method (binGenome-methods), 10
- binGenome-methods, 10
- binsDU, 13, 14
- binsDU (DU accesor), 11
- binsDU, ASpliDU-method (ASpliDU-class), 8

- Counts accesor, 10
- countsb, 18
- countsb (Counts accesor), 10
- countsb, ASpliCounts-method (ASpliCounts-class), 7
- countseli, 18
- countseli (Counts accesor), 10
- countseli, ASpliCounts-method (ASpliCounts-class), 7
- countsg, 18
- countsg (Counts accesor), 10
- countsg, ASpliCounts-method (ASpliCounts-class), 7
- countsie2, 18
- countsie2 (Counts accesor), 10
- countsie2, ASpliCounts-method (ASpliCounts-class), 7

- countsj, 18
- countsj (Counts accesor), 10
- countsj, ASpliCounts-method (ASpliCounts-class), 7

- DEXSeq, 13, 14
- DU accesor, 11
- DUreport, 12, 19
- DUreport, ASpliCounts-method (ASpliCounts-class), 7
- DUreport_DEXSeq, 13
- DUreport_DEXSeq, ASpliCounts-method (ASpliCounts-class), 7

- edgeR, 13, 14
- esPSI, 5, 6
- esPSI (AS accesor), 3
- esPSI, ASpliAS-method (ASpliAS-class), 5

- features accesor, 14
- featuresb, 10
- featuresb (features accesor), 14
- featuresb, ASpliFeatures-method (ASpliFeatures-class), 8
- featuresg, 10
- featuresg (features accesor), 14
- featuresg, ASpliFeatures-method (ASpliFeatures-class), 8
- featuresj, 10
- featuresj (features accesor), 14
- featuresj, ASpliFeatures-method (ASpliFeatures-class), 8

- genesDE, 13, 14
- genesDE (DU accesor), 11
- genesDE, ASpliDU-method (ASpliDU-class), 8

- irPIR, 5, 6
- irPIR (AS accesor), 3
- irPIR, ASpliAS-method (ASpliAS-class), 5

- joint (AS accesor), 3
- joint, ASpliAS-method (ASpliAS-class), 5
- junctionsDU, 13, 14

- junctionsDU (DU accesors), 11
- junctionsDU, ASpliDU-method (ASpliDU-class), 8
- junctionsPIR, 5, 6
- junctionsPIR (AS accesors), 3
- junctionsPIR, ASpliAS-method (ASpliAS-class), 5
- junctionsPSI, 5, 6
- junctionsPSI (AS accesors), 3
- junctionsPSI, ASpliAS-method (ASpliAS-class), 5

- loadBAM, 15

- plotTopTags, 16

- rds, 17
- rds, ASpliCounts-method (ASpliCounts-class), 7
- rdsb, 18
- rdsb (Counts accesors), 10
- rdsb, ASpliCounts-method (ASpliCounts-class), 7
- rdsg, 18
- rdsg (Counts accesors), 10
- rdsg, ASpliCounts-method (ASpliCounts-class), 7
- readCounts, 17
- readCounts, ASpliFeatures-method (ASpliFeatures-class), 8

- show, ASpliAS-method (show-methods), 18
- show, ASpliCounts-method (show-methods), 18
- show, ASpliDU-method (show-methods), 18
- show, ASpliFeatures-method (show-methods), 18
- show-methods, 18

- write, 19
- write-methods, 19
- writeAll (write), 19
- writeAll, ANY-method (write-methods), 19
- writeAS, 5
- writeAS (write), 19
- writeAS, ASpliAS-method (ASpliAS-class), 5
- writeAS-methods (write-methods), 19
- writeCounts, 18
- writeCounts (write), 19
- writeCounts, ASpliCounts-method (ASpliCounts-class), 7
- writeCounts-methods (write-methods), 19

- writeDU, 13, 14
- writeDU (write), 19
- writeDU, ASpliDU-method (ASpliDU-class), 8
- writeDU-methods (write-methods), 19
- writeRds (write), 19
- writeRds, ASpliCounts-method (ASpliCounts-class), 7
- writeRds-methods (write-methods), 19