Package 'IMAS'

July 18, 2025

Type Package

Title Integrative analysis of Multi-omics data for Alternative Splicing

Version 1.33.0

Author Seonggyun Han, Younghee Lee

Maintainer Seonggyun Han <hangost@ssu.ac.kr>

Description Integrative analysis of Multi-omics data for Alternative splicing.

License GPL-2

Depends R (> 3.0.0),GenomicFeatures, ggplot2, IVAS

Imports doParallel, lme4, BiocGenerics, GenomicRanges, IRanges, foreach, AnnotationDbi, S4Vectors, GenomeInfoDb, stats, ggfortify, grDevices, methods, Matrix, utils, graphics, gridExtra, grid, lattice, Rsamtools, survival, BiocParallel, GenomicAlignments, parallel

Suggests BiocStyle, RUnit

biocViews ImmunoOncology, AlternativeSplicing, DifferentialExpression, DifferentialSplicing, GeneExpression, GeneRegulation, Regression, RNASeq, Sequencing, SNP, Software, Transcription

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

git_branch devel

git_last_commit 54ecbbd

git_last_commit_date 2025-04-15

Repository Bioconductor 3.22

Date/Publication 2025-07-18

Contents

`IMAS-package` **IMAS***: Integrative analysis of Multi-omics data for Alternative Splicing*

Description

IMAS offers two components. First, `RatioFromReads` estimates PSI values of a given alternatively spliced exon using both of paired-end and junction reads. See the examples at `RatioFromReads`. Second, `CompGroupAlt`, `MEsQTLFinder`, and `ClinicAnalysis` can be used for further analysis using estimated PSI values. We described more detailed information on usage at the package vignette.

Author(s)

Seonggyun Han, Younghee Lee

`ASvisualization` *Visualize the results of the ASdb object.*

Description

This function makes a pdf file consisting of plots for results in the ASdb object.

Usage

```
ASvisualization(ASdb,CalIndex=NULL,txTable=NULL,exon.range=NULL,snpdata=NULL,
  snplocus=NULL,methyldata=NULL,methyllocus=NULL,GroupSam=NULL,
  ClinicalInfo=NULL,out.dir=NULL)
```

Arguments

`ASdb`	A ASdb object.
`CalIndex`	An index number in the ASdb object which will be tested in this function.
`txTable`	A data frame of transcripts including transcript IDs, Ensembl gene names, Ensembl transcript names, transcript start sites, and transcript end sites.
`exon.range`	A list of GRanges objects including total exon ranges in each transcript resulted from the `exonsBy` function in **GenomicFeatures**.
`snpdata`	A data frame of genotype data.
`snplocus`	A data frame consisting of locus information of SNP markers in the snpdata.

methyldata	A data frame consisting of methylation levels.
methyllocus	A data frame consisting of methylation locus.
GroupSam	A list object of a group of each sample.
ClinicalInfo	A data frame consisting of a path of bam file and identifier of each sample.
out.dir	An output directory

Value

This function makes pdf for plots.

Author(s)

Seonggyun Han, Younghee Lee

Examples

```
 data(sampleGroups)
 data(samplemethyl)
 data(samplemethyllocus)
 data(samplesnp)
 data(samplesnplocus)
 data(sampleclinical)
 data(bamfilestest)
 ext.dir <- system.file("extdata", package="IMAS")
samplebamfiles[,"path"] <- paste(ext.dir,"/samplebam/",samplebamfiles[,"path"],".bam",sep="")
 sampleDB <- system.file("extdata", "sampleDB", package="IMAS")
 transdb <- loadDb(sampleDB)
 ASdb <- Splicingfinder(transdb,Ncor=1)
 ASdb <- ExonsCluster(ASdb,transdb)
 ASdb <- RatioFromReads(ASdb,samplebamfiles,"paired",50,40,3,CalIndex="ES3")
 ASdb <- sQTLsFinder(ASdb,samplesnp,samplesnplocus,method="lm")
 ASdb <- CompGroupAlt(ASdb,GroupSam,CalIndex="ES3")
ASdb <- MEsQTLFinder(ASdb,sampleMedata,sampleMelocus,CalIndex="ES3",GroupSam=GroupSam,out.dir=NULL)
 Sdb <- ClinicAnalysis(ASdb,Clinical.data,CalIndex="ES3",out.dir=NULL)
 exon.range <- exonsBy(transdb,by="tx")
 sel.cn <- c("TXCHROM","TXNAME","GENEID","TXSTART","TXEND","TXSTRAND")
 txTable <- select(transdb, keys=names(exon.range),columns=sel.cn,keytype="TXID")
 ASvisualization(ASdb,CalIndex="ES3",txTable,exon.range,samplesnp,samplesnplocus,
     sampleMedata,sampleMelocus,GroupSam,Clinical.data,out.dir="./")
```

Clinical.data *A data frame for clinical data*

Description

A data frame including survival status and time for each sample. This data is a simulated clinical data for 50 samples (half of whom are assigned as PR-positive and the other half PR-negative), which is used in analysis with **IMAS**. The detailed overview of the data is described in the vignette.

Usage

```
data(sampleclinical)
```

Format

A data frame with survival information and times on the 50 samples

Value

A data frame with survival information and times on the 50 samples

---

`ClinicAnalysis` *Analysis for differential clinical outcomes across PSI values*

Description

This function separate a set of samples into two groups (low and high PSI values) using K-means clustering and perform a statistical test to identify differential survival outcomes between the groups. Internally, this function calls the `kmeans` and `survdiff` functions in the **stats** and **survival** packages, respectively.

Usage

```
ClinicAnalysis(ASdb, ClinicalInfo = NULL, CalIndex = NULL,
    display = FALSE, Ncor = 1, out.dir = NULL)
```

Arguments

`ASdb`	An ASdb object containing "SplicingModel" and "Ratio" slots from the `Splicingfinder` and `RatioFromFPKM` functions, respectively.
`ClinicalInfo`	A data frame consisting of a path of bam file and identifier of each sample.
`CalIndex`	An index number in the ASdb object which will be tested in this function.
`display`	The option returns the survival Kaplan-Meier plot. (TRUE = it will return the list object with a ggplot object and table showing the result of this function, FALSE = it will return P-value.)
`Ncor`	The number of cores for multi-threads function.
`out.dir`	An output directory.

Value

ASdb with the slot (labeled by "Clinical") containing results from the `ClinicAnalysis` function. The "Clinical" slot contains a list object and each element of the list object returns the results assigned to three elements, which is of each alternative splicing type (i.e. Exon skipping, Alternative splice site, Intron retention). Three elements are as follows;

`ES`	A data frame for the result of Exon skipping, consisting of the columns named as follows; Index (index number), EnsID (gene name), Nchr (chromosome name), 1stEX (alternatively spliced target exon), 2ndEX (second alternatively spliced target exon which is the other one of the mutually exclusive spliced exons), DownEX (downstream exon range), UpEX (upstream exon range), Types (splicing type), Pvalue (P-value of Kaplan-Meier test for differential survival outcomes between low and high PSI groups), and Fdr.p (FDR values).

ASS — A data frame for the result of Alternative splice sites, consisting of the columns named as follows; Index (index number), EnsID (gene name), Nchr (chromosome name), ShortEX (shorter spliced target exon), LongEX (longer spliced target exon), NeighborEX (neighboring down or upstream exons), Types (splicing type), Pvalue (P-value of Kaplan-Meier test for differential survival outcomes between low and high PSI groups), and Fdr.p (FDR values).

IR — A data frame for the result of Intron retention, consisting of the columns named as follows; Index (index number), EnsID (gene name), Nchr (chromosome name), RetainEX (retained intron exon), DownEX (downstream exon range), UpEX (upstream exon range), Types (splicing type), Pvalue (P-values of Kaplan- Meier test for differential survival outcomes between low and high PSI groups), and Fdr.p (FDR values).

Author(s)

Seonggyun Han, Younghee Lee

See Also

kmeans, survdiff, survfit

Examples

```
  data(bamfilestest)
  data(sampleclinical)
  ext.dir <- system.file("extdata", package="IMAS")
samplebamfiles[,"path"] <- paste(ext.dir,"/samplebam/",samplebamfiles[,"path"],".bam",sep="")
  sampleDB <- system.file("extdata", "sampleDB", package="IMAS")
  transdb <- loadDb(sampleDB)
  ## Not run:
  ASdb <- Splicingfinder(transdb,Ncor=1)
  ASdb <- ExonsCluster(ASdb,transdb)
  ASdb <- RatioFromReads(ASdb,samplebamfiles,"paired",50,40,3,CalIndex="ES3")
  ASdb <- ClinicAnalysis(ASdb,Clinical.data,CalIndex="ES3",out.dir=NULL)

## End(Not run)
```

CompGroupAlt — *Identify alternatively spliced exons with a differential PSIs between the groups*

Description

This function performs a regression test to identify alternatively spliced exons that are differentially expressed between two groups. It will call the lm function to test a linear regression model.

Usage

```
CompGroupAlt(ASdb, GroupSam = NULL, Ncor = 1, CalIndex = NULL, out.dir = NULL)
```

Arguments

ASdb	An ASdb object containing "SplicingModel" and "Ratio" slots from the Splicingfinder and RatioFromFPKM functions, respectively.
GroupSam	A list object of a group of each sample.
Ncor	The number of cores for multi-threads function.
CalIndex	An index number in the ASdb object which will be tested in this function.
out.dir	An output directory.

Value

ASdb with the slot (labeled by "GroupDiff") containing results from the CompGroupAlt function. The "GroupDiff" slot consists of a list object and each element of the list object returns the results assigned to three elements, which is of each alternative splicing type (i.e. Exon skipping, Alternative splice site, Intron retention). Three elements are as follows;

ES	A data frame for the result of Exon skipping, consisting of the columns named as follows; Index (index number), EnsID (gene name), Nchr (chromosome name), 1stEX (alternatively spliced target exon), 2ndEX (second alternatively spliced target exon which is the other one of the mutually exclusive spliced exons), DownEX (downstream exon range), UpEX (upstream exon range), Types (splicing type), Diff.P (P-value of linear regression test for differential expression between groups), and Fdr.p (FDR values).
ASS	A data frame for the result of Alternative splice sites, consisting of the columns named as follows; Index (index number), EnsID (gene name), Nchr (chromosome nam), ShortEX (shorter spliced target exon), LongEX (longer spliced target exon), NeighborEX (neighboring down or upstream exons), Types (splicing type), Diff.P (P-value of linear regression test for differential expression between groups), and Fdr.p (FDR values).
IR	A data frame for the result of Intron retention, consisting of the columns named as follows; Index (index number), EnsID (gene name), Nchr (chromosome name), RetainEX (retained intron exon), DownEX (downstream exon range), UpEX (upstream exon range), Types (splicing type), Diff.P (P-value of linear regression test for differential expression between groups), and Fdr.p (FDR values).

Author(s)

Seonggyun Han, Younghee Lee

References

Chambers, J. M. (1992) Linear models. Chapter 4 of Statistical Models in S eds J. M. Chambers and T. J. Hastie, Wadsworth & Brooks/Cole.

See Also

lm

Examples

```
data(bamfilestest)
data(sampleGroups)
ext.dir <- system.file("extdata", package="IMAS")
samplebamfiles[,"path"] <- paste(ext.dir,"/samplebam/",samplebamfiles[,"path"],".bam",sep="")
sampleDB <- system.file("extdata", "sampleDB", package="IMAS")
transdb <- loadDb(sampleDB)
## Not run:
ASdb <- Splicingfinder(transdb,Ncor=1)
ASdb <- ExonsCluster(ASdb,transdb)
ASdb <- RatioFromReads(ASdb,samplebamfiles,"paired",50,40,3,CalIndex="ES3")
ASdb <- CompGroupAlt(ASdb,GroupSam,CalIndex="ES3")

## End(Not run)
```

`ExonsCluster` *Construct representative Exons*

Description

This function constructs representative Exons.

Usage

```
ExonsCluster(ASdb,GTFdb,Ncor=1,txTable=NULL)
```

Arguments

`ASdb`	An ASdb object containing "SplicingModel" from the `Splicingfinder` funtion.
`GTFdb`	A TxDb object in the **GenomicFeatures** package.
`Ncor`	The number of cores for multi-threads function.
`txTable`	The matrix of transcripts including transcript IDs, Ensembl gene names, Ensembl transcript names, transcript start sites, and transcript end sites.

Value

ASdb containing representative exons.

Author(s)

Seonggyun Han, Younghee Lee

Examples

```
sampleDB <- system.file("extdata", "sampleDB", package="IMAS")
transdb <- loadDb(sampleDB)
## Not run:
ASdb <- Splicingfinder(transdb,Ncor=1)
ASdb <- ExonsCluster(ASdb,transdb)

## End(Not run)
```

GroupSam *Group of each sample.*

Description

A list object comprising sample names belonging to each group, PR-positive and PR-negative. This data is a simulated clinical data for 50 samples (half of whom are assigned as PR-positive and the other half PR-negative). The detailed overview of the data is described in the vignette.

Usage

```
data(sampleGroups)
```

Format

A list object including a group information on the 50 samples

Value

A list object including a group information on the 50 samples

Examples

```
data(sampleGroups)
```

MEsQTLFinder *Identify methylation loci that are significantly associated with alternatively spliced exons*

Description

This function performs a regression test to identify significant association between methylation levels and PSI values using a linear regression model of `lm` function.

Usage

```
MEsQTLFinder(ASdb, Total.Medata = NULL, Total.Melocus = NULL, GroupSam = NULL,
     Ncor = 1, CalIndex = NULL, out.dir = NULL)
```

Arguments

ASdb	An ASdb object including "SplicingModel" and "Ratio" slots from the `Splicingfinder` and `RatioFromFPKM` functions, respectively.
Total.Medata	A data frame consisting of methylation levels.
Total.Melocus	A data frame consisting of methylation locus.
GroupSam	A list object of a group of each sample.
Ncor	The number of cores for multi-threads.
CalIndex	An index number in the ASdb object which will be tested in this function.
out.dir	An output directory.

Value

ASdb with the slot (labeled by "Me.sQTLs") containing the results from the MEsQTLFinder function. The "Me.sQTLs" slot is consists of a list object and each element of the list object returns the results assigned to three elements, which is of each alternative splicing type (i.e. Exon skipping, Alternative splice site, Intron retention). Three elements are as follows;

ES	A data frame for the result of Exon skipping, consisting of the columns named as follows; Index (index number), EnsID (gene name), Nchr (chromosome name), 1stEX (alternatively spliced target exon), 2ndEX (second alternatively spliced target exon which is the other one of the mutually exclusive spliced exons), DownEX (downstream exon range), UpEX (upstream exon range), Types (splicing type), pByMet (P-values of linear regression test for association between methylation levels and PSI values), fdrByMet (FDR values for the pByMet column), pByGroups (P-values of t-test for differential methylation levels between two groups, and fdrByGroups (FDR values for the pByGroups column).
ASS	A data frame for the result of Alternative splice sites, consisting of the columns named as follows; Index (index number), EnsID (gene name), Nchr (chromosome nam), ShortEX (shorter spliced target exon), LongEX (longer spliced target exon), NeighborEX (neighboring down or upstream exons), Types (splicing type), pByMet (P-values of linear regression test for association between methylation levels and PSI values), fdrByMet (FDR values for the pByMet column), pByGroups (P-values of t-test for differential methylation levels between groups, and fdrByGroups (adjust FDR values for the pByGroups column.
IR	A data frame for the result of Intron retention, consisting of the columns named as follows; Index (index number), EnsID (gene name), Nchr (chromosome name), RetainEX (retained intron exon), DownEX (downstream exon range), UpEX (upstream exon range), Types (splicing type), pByMet (P-values of linear regression test for association between methylation levels and PSI values), fdrByMet (adjust FDR values for the pByMet column), pByGroups (P-values of t-test for differential methylation levels between the groups, and fdrByGroups (adjust FDR values for the pByGroups column.

Author(s)

Seonggyun Han, Younghee Lee

References

Chambers, J. M. (1992) Linear models. Chapter 4 of Statistical Models in S eds J. M. Chambers and T. J. Hastie, Wadsworth & Brooks/Cole.

See Also

lm

Examples

```
  data(bamfilestest)
  data(samplemethyl)
  data(samplemethyllocus)
  data(sampleGroups)
  ext.dir <- system.file("extdata", package="IMAS")
samplebamfiles[,"path"] <- paste(ext.dir,"/samplebam/",samplebamfiles[,"path"],".bam",sep="")
```

```
sampleDB <- system.file("extdata", "sampleDB", package="IMAS")
transdb <- loadDb(sampleDB)
## Not run:
ASdb <- Splicingfinder(transdb,Ncor=1)
ASdb <- ExonsCluster(ASdb,transdb)
ASdb <- RatioFromReads(ASdb,samplebamfiles,"paired",50,40,3,CalIndex="ES3")
ASdb <- MEsQTLFinder(ASdb,sampleMedata,sampleMelocus,CalIndex="ES3",GroupSam=GroupSam,out.dir=NULL)

## End(Not run)
```

`RatioFromReads` *Calculate expression ratio (PSI) from bamfiles*

Description

This function extracts reads information from bamfile using **Rsamtools** and calculates expression ratio (denoted as Percent Splice-In, PSI) of each alternatively spliced exon (i.e., exon skipping, intro retention, and 5- and 3- prime splice sites).

Usage

```
RatioFromReads(ASdb=NULL,Total.bamfiles=NULL,readsInfo=c("paired","single"),
    readLen=NULL,inserSize=NULL,minr=3,CalIndex=NULL,Ncor=1,out.dir=NULL)
```

Arguments

`ASdb`	An ASdb object including "SplicingModel" slot from the `Splicingfinder` function.
`Total.bamfiles`	A data frame containing the path and name of a bamfile from RNA-seq
`readsInfo`	Information of RNA-seq types (single- or paired-end reads)
`readLen`	The read length
`inserSize`	The insert size between paired-end reads.
`minr`	A minimum number of testable reads mapping to a given exon.
`CalIndex`	An index number in the ASdb object which will be tested in this function.
`Ncor`	The number of cores for multi-threads.
`out.dir`	An output directory.

Value

ASdb with the slot (labeled by "Ratio") containing results from the the `RatioFromReads` function. The "Ratio" slot contains a list object and each element of the list object returns the results assigned to three elements, which is of each alternative splicing type (i.e. Exon skipping, Alternative splice site, Intron retention). Three elements are as follows;

`ES`	A data frame for the result of Exon skipping, consisting of the columns named as follows; Index (index number), EnsID (gene name), Nchr (chromosome name), 1stEX (alternatively spliced target exon), 2ndEX (second alternatively spliced target exon which is the other one of the mutually exclusive spliced exons), DownEX (downstream exon range), UpEX (upstream exon range), Types (splicing type), and names of individuals.

ASS	A data frame for the result of Alternative splice sites, consisting of the columns named as follows; Index (index number), EnsID (gene name), Nchr (chromosome name), ShortEX (shorter spliced target exon), LongEX (longer spliced target exon), NeighborEX (neighboring down or upstream exons), Types (splicing type), and names of individuals.
IR	A data frame for the result of Intron retention, consisting of the columns named as follows; Index (index number), EnsID (gene name), Nchr (chromosome name), RetainEX (retained intron exon), DownEX (downstream exon range), UpEX (upstream exon range), Types (splicing type), and names of individuals.

Author(s)

Seonggyun Han, Younghee Lee

See Also

SplicingReads

Examples

```
    data(bamfilestest)
    ext.dir <- system.file("extdata", package="IMAS")
  samplebamfiles[,"path"] <- paste(ext.dir,"/samplebam/",samplebamfiles[,"path"],".bam",sep="")
    sampleDB <- system.file("extdata", "sampleDB", package="IMAS")
    transdb <- loadDb(sampleDB)
    ## Not run:
    ASdb <- Splicingfinder(transdb,Ncor=1)
    ASdb <- ExonsCluster(ASdb,transdb)
    ASdb <- RatioFromReads(ASdb,samplebamfiles,"paired",50,40,3,CalIndex="ES3")

## End(Not run)
```

---

samplebamfiles *A data frame for example expression bam files.*

Description

A path and identifier of bam files for 50 samples. For each bam file, mapped reads were randomly generated that came from the genomic region of chr11: 100,933,178 - 100,996,889. With each simulated bam file of 50 samples, PSI level is calculated for the exon that is located in chr11: 100,962,491-100,962,607. The simulated PSI values are in the range of 0.6 to 1.0. The range of 0.9 to 1.0 of PSI values are assigned to PR-positive group and 0.5 to 0.6 to PR-negative group. The detailed overview of the data is described in the vignette.

Usage

```
data(bamfilestest)
```

Format

A data frame with paths and identifiers on the 50 samples

Value

A data frame with paths and identifiers on the 50 samples

Source

The data was provided from **IMAS**

Examples

```
data(bamfilestest)
```

`sampleMedata` *Methylation level data*

Description

Methylation level of 5 loci (beta value), which are located in the PRGA gene for 50 samples. We generated a simulation data set of methylation level (beta value) for each locus, that significantly differs between two groups (PR-positive and PR-negative), while other methylation loci are not different. The detailed overview of the data is described in the vignette.

Usage

```
data(samplemethyl)
```

Format

A data frame with levels of 5 methylation locus on the 50 samples

Value

A data frame with levels of 5 methylation locus on the 50 samples

Examples

```
data(samplemethyl)
```

sampleMelocus *Genomic locus of methylations*

Description

Genomic location of 5 methylation loci located in the PRGA gene for 50 samples, which are matched with methylation level data provided in **IMAS**. The detailed overview of the data is described in the vignette.

Usage

```
data(samplemethyllocus)
```

Format

A data frame with genomic locus of 5 methylations

Value

A data frame with genomic locus of 5 methylations

Examples

```
data(samplemethyllocus)
```

samplesnp *Genotype data*

Description

Genotype data of five SNPs located in the PRGA gene for 50 samples (half of whom are assigned as PR-positive and the other half PR-negative), which is used in analysis with **IMAS**. We generated a simulation data set of genotypes for each SNP. Among five SNPs, three are associated with PSI levels for 50 samples, while two SNPs are not. The detailed overview of the data is described in the vignette.

Usage

```
data(samplesnp)
```

Format

A data frame with genotypes of 5 SNPs on the 50 samples

Value

A data frame with genotypes of 5 SNPs on the 50 samples

Source

The data was provided from **IMAS**

Examples

```
data(samplesnp)
```

`samplesnplocus` *Genomic locus of SNPs*

Description

Genomic locus of five SNPs located in the PRGA gene for 50 samples, which are matched with SNP genotype data provided in **IMAS**. The detailed overview of the data is described in the vignette.

Usage

```
data(samplesnplocus)
```

Format

A data frame with genomic locus of 5 SNPs

Value

A data frame with genomic locus of 5 SNPs

Examples

```
data(samplesnplocus)
```

`SplicingReads` *Count a junction and paired-end reads*

Description

This function counts the reads that are mapped to two separate exons, mapped to either splice site of two exons (called junction reads) or within each of two exons (paired end reads).

Usage

```
SplicingReads(bamfile=NULL,test.exon=NULL,spli.jun=NULL,e.ran=NULL,
    SNPchr=NULL,readsinfo="paired",inse=40)
```

Arguments

`bamfile`	A path of mapped bamfile.
`test.exon`	A data frame containing an alternative target exon and their neighboring exons.
`spli.jun`	A data frame containing spliced junction information.
`e.ran`	A range for parsing reads from a bamfile.
`SNPchr`	A chromosome number
`readsinfo`	Information of RNA-seq types (single- or paired- end reads).
`inse`	An insert size

Value

This function returns the list object providing counts the reads that are mapped to two separate exons, mapped to either splice site of two exons (called junction reads) or within each of two exons (paired end reads).

Author(s)

Seonggyun Han, Younghee Lee

Examples

```
  data(bamfilestest)
  ext.dir <- system.file("extdata", package="IMAS")
samplebamfiles[,"path"] <- paste(ext.dir,"/samplebam/",samplebamfiles[,"path"],".bam",sep="")
  sampleDB <- system.file("extdata", "sampleDB", package="IMAS")
  transdb <- loadDb(sampleDB)
  ## Not run:
  ASdb <- Splicingfinder(transdb,Ncor=1)
  ASdb <- ExonsCluster(ASdb,transdb)
  bamfiles <- rbind(samplebamfiles[,"path"])
  Total.splicingInfo <- ASdb@SplicingModel$"ES"
  each.ES.re <- rbind(ES.fi.result[ES.fi.result[,"Index"] == "ES3",])
each.ranges <- rbind(unique(cbind(do.call(rbind,strsplit(each.ES.re[,"DownEX"],"-"))[,1],
      do.call(rbind,strsplit(each.ES.re[,"UpEX"],"-"))[,2])))
  group.1.spl <- c(split.splice(each.ES.re[,"Do_des"],each.ES.re[,"1st_des"]),
      split.splice(each.ES.re[,"1st_des"],each.ES.re[,"Up_des"]))
  group.2.spl <- split.splice(each.ES.re[,"Do_des"],each.ES.re[,"Up_des"])
  total.reads <- SplicingReads(bamfiles[1],each.ES.re[,c("DownEX","1stEX","UpEX")],
      c(group.1.spl,group.2.spl),each.ranges,each.ES.re[,"Nchr"],"paired")

## End(Not run)
```

Index