Package ‘IVAS’

May 10, 2024

Type Package

Title Identification of genetic Variants affecting Alternative Splicing

Version 2.24.0

Author Seonggyun Han, Sangsoo Kim

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

Description Identification of genetic variants affecting alternative splicing.

License GPL-2

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

Imports doParallel, lme4, BiocGenerics, GenomicRanges, IRanges, foreach, AnnotationDbi, S4Vectors, GenomeInfoDb, ggfortify, grDevices, methods, Matrix, BiocParallel, utils, stats

Suggests BiocStyle

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

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

git_branch RELEASE_3_19

git_last_commit 9312880

git_last_commit_date 2024-04-30

Repository Bioconductor 3.19

Date/Publication 2024-05-10

Contents

IVAS-package .......................................................... 2
ASdb-class ............................................................ 2
calSignificant ....................................................... 3
CalSigSNP ............................................................. 3
chrseparate .......................................................... 4
IVAS-package

IVAS: Identification of genomic variants affecting Alternative Splicing

Description

The tool is to detect genomic variants affecting the alternative splicing using genotypic and gene expression data (RNA-seq).

ASdb-class

ASdb s4 class - a container for results from functions of the IVAS package.

Description

This class is the main object for storing results of the present package.

Note

An ASdb object stores information of alternative splicing patterns, expression ratios between transcripts with and without alternative target exons, and significant sQTLs from the functions of the present package. This ASdb object can be populated further slots during the analysis using functions for the analysis. Typically, an ASdb object can be created when the function Splicingfinder completes to define alternative splicing patterns. After creation, the ASdb contains the slot labeled as "SplicingModel", and the slot includes a list object named by "ES", "ASS", and "IR" (alternative splicing exons are saved separately in each element of the list based on their splicing pattern types; "ES": Exon skipping, "ASS": Alternative splice site, and "IR": Intron retention). In the next analysis step, further result slots can be added. The function RatioFromFPKM can add the "Ratio" slot containing expression ratio for each alternative splicing pattern based on the "SplicingModel" slot of the present class and for each individual from a matrix of FPKM values. Then, the result of the sQTLsFinder function can be saved by adding the "sQTLs" slot including significance of association between the expression ratios, which is stored in the "Ratio" slot of the present class, and SNPs for each alternative splicing exon.
See Also

Splicingfinder, RatioFromFPKM, sQTLsFinder

Examples

```r
sampleDB <- system.file("extdata", "sampleDB", package="IVAS")
sample.Txdb <- loadDb(sampleDB)
data(sampleexp)
data(samplesnp)
data(samplesnlocus)
ASdb <- Splicingfinder(sample.Txdb)
ASdb <- RatioFromFPKM(sample.Txdb, ASdb, sampleexp)
ASdb <- sQTLsFinder(ASdb, samplesnp, samplesnlocus, method="lm")
ASdb
```

---

calSignificant

*Deprecated*

Description

This function is deprecated and will be made defunct. Instead, use Splicingfinder.

CalSigSNP

*Calculate significance SNPs*

Description

This function performs linear regression test to identify significance associations between expression ratio and genotypes using the `lm` function.

Usage

```r
CalSigSNP(ratio.mat=NULL, snp.mat=NULL, overlapsnp=NULL, each.snplocus=NULL, chr, each.gene=NULL, GroupSam=NULL, method="lm")
```

Arguments

- `ratio.mat`: A data frame consisting of expression ratio of an alternatively spliced exon.
- `snp.mat`: A data frame of genotype data.
- `overlapsnp`: A data frame containing SNPs which is within an alternatively spliced exon and its flanking introns.
- `each.snplocus`: A data frame consisting of locus information of SNP markers in the snpdata.
- `chr`: The chromosome number that you would like to test in this function.
- `each.gene`: The gene name that you would like to test in this function.
GroupSam

A list object of a group of each sample.

method

The option for statistical models and boxplot. ("lm" : analysis using linear regression model, "glm" : analysis using generalized linear mixed model, "both" : "lm" and "glm", and "boxplot" : for writing boxplot).

Value

The lm or glm method returns matrix including; SNP marker IDs, P values, information of differential median values of expression ratio among genotypes ("sig" if differential median > 0.1 and "not sig" otherwise), gene names, and methods ("lm" or "glm"). The boxplot method returns matrix with relative ratio values and genotypes of samples.

Author(s)

Seonggyun Han, Sangsoo Kim

References


See Also

lm, glmer

Examples

```r
sampleDB <- system.file("extdata", "sampleDB", package="IVAS")
sample.Txdb <- loadDb(sampleDB)
data(sampleexp)
data(samplesnp)
data(samplesnplocus)
ASdb <- Splicingfinder(sample.Txdb)
ASdb <- RatioFromFPKM(sample.Txdb, ASdb, sampleexp, CalIndex="ASS")
ratio.mat <- slot(ASdb, "Ratio")$ASS
ratio.mat <- rbind(ratio.mat[, grep("NA", colnames(ratio.mat))])
each.snp <- rbind(samplesnp[rrownames(samplesnp) == "rs3810232",])
each.snplocus <- rbind(samplesnplocus[samplesnplocus[,"SNP"] == "rs3810232",])
overlapsnp <- rbind(c(snp="rs3810232", locus="54704760"))
CalSigSNP(ratio.mat, as.matrix(each.snp), overlapsnp, each.snplocus, "19", "ENSG00000170889", method="lm")
```

chrseparate

Separate a TxDb object based on a chromosome.

Description

With the isActiveSeq method in GenomicFeatures package, this function filters the TxDb object in the GenomicFeatures package based on a single chromosome.
findAlternative

Usage

chrseparate(GTFdb = NULL, chrname = NULL)

Arguments

GTFdb The TxDb object in the GenomicFeatures package.
chrname The chromosome number you would like to select from TxDb

Value

This function returns the TxDb limited to the chromosome number that you want.

Author(s)

Seonggyun Han, Sangsoo Kim

References


See Also

isActiveSeq, seqinfo

Examples

sampleDB <- system.file("extdata", "sampleDB", package="IVAS")
sample.Txdb <- loadDb(sampleDB)
filtered.txdb <- chrseparate(sample.Txdb,19)

findAlternative Find alternative exons of a gene.

Description

Search alternative exons among transcript isoforms from a single gene.

Usage

findAlternative(geneid = NULL, txTable = NULL, totalExrange = NULL, totalInrange = NULL, one.chr = NULL)
Arguments

geneid Ensembl gene name.
txTable The matrix of transcripts including transcript IDs, Ensembl gene names, Ensembl transcript names, transcript start sites, and transcript end sites.
totalExrange A list of GRanges objects including total exon ranges in each transcript resulted from the exonsBy function in GenomicFeatures.
totalInrange A list of GRanges objects including total intron ranges in each transcript resulted from the intronsByTranscript function in GenomicFeatures.
one.chr The chromosome number that you would like to test

Value

alterIntron A GRanges object with flanking introns of alternative exons
tableBygene An information table of transcripts including transcript IDs, Ensembl gene names, Ensembl transcript names, transcript start sites, and transcript end sites.
exonRange All exons locus of a gene
intronRange All intron locus of a gene

Author(s)

Seonggyun Han, Sangsoo Kim

References


See Also

GRanges, IRanges

Examples

```r
sampleDB <- system.file("extdata", "sampleDB", package="IVAS")
sample.Txdb <- loadDb(sampleDB)
filtered.txdb <- chrseparate(sample.Txdb,19)
trans.exon.range <- exonsBy(filtered.txdb,by="tx")
trans.intron.range <- intronsByTranscript(filtered.txdb)
txTable <- select(filtered.txdb, keys=names(trans.exon.range),
                   columns=c("TXID","TXNAME","GENEID","TXSTART","TXEND"), keytype="TXID")
Altvalue <- findAlternative("ENSG00000170889",txTable,trans.exon.range,trans.intron.range,19)
```
**findOversnp**

| findOversnp | Find SNPs which belong to alternative exons and flanking introns of them. |

**Description**

Find SNPs which belong to alternative exons and flanking introns of them.

**Usage**

```r
findOversnp(altInvalue = NULL, snprange = NULL)
```

**Arguments**

- `altInvalue` A list data set from the `findAlternative` function.
- `snprange` A matrix of SNP ranges.

**Value**

This function returns a matrix with SNPs which are located in alternative exons and flanking introns and ranges of those SNPs.

**Author(s)**

Seonggyun Han, Sangsoo Kim

**See Also**

`findOverlaps`

**Examples**

```r
combo <- system.file("extdata", "sampleDB", package="IVAS")
sample.Txdb <- loadDb(combo)
data(samplesnplocus)
data(samplesnp)
filtered.txdb <- chrseparate(sample.Txdb, 19)
trans.exon.range <- exonsBy(filtered.txdb, by="tx")
trans.intron.range <- intronsByTranscript(filtered.txdb)
txTable <- select(filtered.txdb, keys=names(trans.exon.range),
                  columns=c("TXID","TXNAME","GENEID","TXSTART","TXEND"), keytype="TXID")
ch.snp.locus <- as.matrix(samplesnplocus[samplesnplocus[,2] == 19,])
ch.snps <- matrix(ch.snp.locus[is.element(ch.snp.locus[,1],rownames(samplesnp))],nrow=3,byrow=FALSE)
ch.snps.range <- GRanges(seqnames=Rle(19), ranges=IRanges(start=as.integer(ch.snps[,3]),
                                            end=as.integer(ch.snps[,3])),metadata=ch.snps[,1])
Altvalue <- findAlternative("ENSG00000170889",txTable,trans.exon.range,trans.intron.range,19)
overlapsnp <- findOversnp(Altvalue,ch.snps.range)
```
IVAS-deprecated  Deprecated functions in package ‘IVAS’

**Description**

These functions are provided for compatibility with older versions of ‘IVAS’ only, and will be defunct at the next release.

**Details**

The following functions are deprecated and will be made defunct; use the replacement indicated below:

- MsqtlFinder: `sQTLsFinder`
- sqtlfinder: `sQTLsFinder`
- calSignificant: `Splicingfinder`

---

### MsqtlFinder  Deprecated

**Description**

This function is deprecated and will be made defunct. Instead, use `sQTLsFinder`.

---

### RatioFromFPKM  Estimate relative expression ratio.

**Description**

With the FPKM expression data set of transcripts, this function estimates relative expression ratio between transcripts with and without alternatively spliced exons based on splicing models of the ASdb object

**Usage**

```
RatioFromFPKM(GTFdb = NULL, ASdb = NULL, Total.expdata = NULL, CalIndex = NULL, Ncor = 1, out.dir = NULL)
```
Arguments

GTFdb A TxDB object in the GenomicFeatures package.
ASdb A ASdb object including "SplicingModel" slot from the Splicingfinder function.
Total.expdata A data frame of expression data.
CalIndex An index number in the ASdb object which will be tested in this function.
Ncor The number of cores for multi-threads function.
out.dir An output directory.

Value

ASdb with the slot (labeled by "Ratio") containing results from the the RatioFromFPKM 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, Sangsoo Kim

See Also

isActiveSeq, seqinfo, Splicingfinder

Examples

```r
sampleDB <- system.file("extdata", "sampleDB", package="IVAS")
sample.Txdb <- loadDb(sampleDB)
data(sampleexp)
ASdb <- Splicingfinder(sample.Txdb)
ASdb <- RatioFromFPKM(sample.Txdb,ASdb,sampleexp)
```
Description

CEU expression data including 78 individuals

Usage

data("sampleexp")

Format

A data frame with 64 transcript expressions on the 78 individuals

Value

A data frame with 64 transcript expressions on the 78 individuals

Source

The data was generated by GEUVADIS (Genetic European Variation in Health and Disease, A European Medical Sequencing Consortium) RNA sequencing project for 1000 Genomes samples (http://www.geuvadis.org/web/geuvadis/RNAseq-project).

References


Examples

data(sampleexp)

Description

CEU genotype data including 78 individuals

Usage

data("samplesnp")
samplesnplocus

**Format**

A data frame with 11 SNPs on the 78 individuals

**Value**

A data frame with 11 SNPs on the 78 individuals

**Source**

The data has 1000 genomes Phages 1 dataset and was imputed by GEUVADIS (Genetic European Variation in Health and Disease, A European Medical Sequencing Consortium) RNA sequencing project for 1000 Genomes samples (http://www.geuvadis.org/web/geuvadis/RNaseq-project).

**References**


**Examples**

```r
data(samplesnp)
```

<table>
<thead>
<tr>
<th>samplesnploycus</th>
<th>snplocus</th>
</tr>
</thead>
</table>

**Description**

snplocus

**Usage**

```r
data("samplesnploycus")
```

**Format**

A data frame with 11 SNPs and locus of them

**Value**

A data frame with 11 SNPs and locus of them

**Examples**

```r
data(samplesnploycus)
```
saveBplot

Description
Save boxplots

Usage
saveBplot(ASdb=ASdb, Total.snpdata=NULL, Total.snplocus=NULL,
CalIndex=NULL, out.dir=NULL)

Arguments
- ASdb: A ASdb object including "sQTLs" slot from the sQTLsFinder function.
- Total.snpdata: A data frame of genotype data.
- Total.snplocus: A data frame containing locus information of SNP markers in the snpdata.
- CalIndex: An index number in the ASdb object which will be tested in this function.
- out.dir: An output directory.

Value
This function draws the boxplot

Author(s)
Seonggyun Han, Sangsoo Kim

See Also
boxplot

Examples
```r
sampleDB <- system.file("extdata", "sampleDB", package="IVAS")
sample.Txdb <- loadDb(sampleDB)
data(sampleexp)
data(samplesnp)
data(samplesnplocus)
ASdb <- Splicingfinder(sample.Txdb)
ASdb <- RatioFromFPKM(sample.Txdb, ASdb, sampleexp)
ASdb <- sQTLsFinder(ASdb, samplesnp, samplesnplocus, method="lm")
saveBplot(ASdb=ASdb, Total.snpdata=samplesnp, Total.snplocus=samplesnplocus, CalIndex="ASS7", out.dir="./result")
```
Splicingfinder

Find alternatively spliced exons based on GTF reference transcript models.

Usage

```
Splicingfinder(GTFdb = NULL, txTable = NULL, calGene = NULL, Ncor = 1, out.dir = NULL)
```

Arguments

- **GTFdb**: A TxDb object in the `GenomicFeatures` package.
- **txTable**: A matrix of transcripts including transcript IDs, gene names, transcript names, transcript start sites, and transcript end sites based on a GTF reference transcript model file.
- **calGene**: An interest of a gene that will be tested. If calGene is inputted by a single gene, the splicing pattern for the only gene is tested. If not, the splicing patterns for total of genes are tested.
- **Ncor**: The number of cores for multi-threads.
- **out.dir**: An output directory.

Value

ASdb with the slot (labeled by "SplicingModel") containing results from the `Splicingfinder` function. The "Splicingfinder" 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), 1st_des (shorter spliced target exons in a representative exon), 2nd_des (second alternatively spliced target exons in a representative exon), Do_des (downstream exons in a representative exon), Up_des (upstream exons in a representative exon), and Types (splicing type).

- **ASS**: A data frame for the result of Alternative splice site, 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 downstream or upstream exons), Short_des (shorter spliced target exons in a representative exon), Long_des (longer spliced target exons in a representative exon), Neighbor_des (neighboring downstream or upstream exons in a representative exon), and Types (splicing type).
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), Retain_des (retained intron exons in a representative exon), Do_des (downstream exons in a representative exon), Up_des (upstream exons in a representative exon), and Types (splicing type).

Author(s)
Seonggyun Han, Sangsoo Kim

See Also
isActiveSeq, seqinfo

Examples
```r
sampleDB <- system.file("extdata", "sampleDB", package="IVAS")
sample.Txdb <- loadDb(sampleDB)
ASdb <- Splicingfinder(sample.Txdb)
```

---

This function is deprecated and will be made defunct. Instead, use `sQTLsFinder`.

---

Find significant SNPs using the calSignificant function.

Usage
```r
sQTLsFinder(ASdb, Total.snpdata = NULL, Total.snplocus = NULL, GroupSam = NULL, method = "lm", CalIndex = NULL, Ncor = 1, out.dir = NULL)
```
**sQTLsFinder**

**Arguments**

- **ASdb**
  A ASdb object including "SplicingModel" and "Ratio" slots from the `Splicingfinder` and `RatioFromFPKM` functions, respectively.

- **Total.snpdata**
  A data frame of genotype data.

- **Total.snplocus**
  A data frame containing locus information of SNP markers in the snpdata.

- **GroupSam**
  A list object of a conditions for each individual. If GroupSam is not NULL, the odds ratio and its confidence intervals between conditions are calculated.

- **method**
  An option for statistical models and boxplot. ("lm" : analysis using linear regression model, "glm" : analysis using generalized linear mixed model, "both" : "lm" and "glm", and "boxplot" : for writing boxplot).

- **CalIndex**
  An index number in the ASdb object which will be tested in this function.

- **Ncor**
  The number of cores for multi-threads function.

- **out.dir**
  An output directory.

**Value**

ASdb with the slot (labeled by "sQTLs") containing results from the the `sQTLsFinder` function. The "Splicingfinder" 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), pByGeno (P-values of "lm" or "glm" test for association PSI values and genotypes), FdrByGeno (pByGeno), diff ("diff" if differential median > 0.1 and "Nondiff" otherwise), pByGroups (P-values of chi-square test for association between genotypes of two groups), fdrByGroups (FDR values for the pByGroups column), OR (odds ratio), lowCI(low confidence interval), highCI(high confidence interval), and methods ("lm" or "glm").

- **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), pByGeno (P-values of "lm" or "glm" test for association PSI values and genotypes), FdrByGeno (pByGeno), diff ("diff" if differential median > 0.1 and "Nondiff" otherwise), pByGroups (P-values of chi-square test for association between genotypes of two groups), fdrByGroups (FDR values for the pByGroups column), OR (odds ratio), lowCI(low confidence interval), highCI(high confidence interval), and methods ("lm" or "glm").

- **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), pByGeno (P-values of "lm" or...
"glm" test for association PSI values and genotypes), FdrByGeno (pByGeno),
diff ("diff" if differential median > 0.1 and "Nondiff" otherwise), pByGroups (P-
values of chi-square test for association between genotypes of two groups), fdr-
ByGroups (FDR values for the pByGroups column), OR (odds ratio), lowCI(low
confidence interval), highCI(high confidence interval), and methods ("lm" or
"glm").

The boxplot method returns matrix data with relative ratio values and genotypes of samples.

Author(s)
Seonggyun Han, Sangsoo Kim

References

See Also
lm, glmer

Examples

```r
sampleDB <- system.file("extdata", "sampleDB", package="IVAS")
sample.Txdb <- loadDb(sampleDB)
data(sampleexp)
data(samplesnp)
data(samplesnplocus)
ASdb <- Splicingfinder(sample.Txdb)
ASdb <- RatioFromFPKM(sample.Txdb, ASdb, sampleexp)
ASdb <- sQTLsFinder(ASdb, samplesnp, samplesnplocus, method="lm")
```
Index

* datasets
  sampleexp, 10
  samplesnp, 10
  samplesnplocus, 11

* package
  IVAS-package, 2

ASdb-class, 2
boxplot, 12
calSignificant, 3
CalSigSNP, 3
chrseparate, 4
exonsBy, 6
findAlternative, 5, 7
findOverlaps, 7
findOversnp, 7

glmer, 4, 16
GRanges, 6
intronsByTranscript, 6
IRanges, 6
isActiveSeq. 4, 5, 9, 14
IVAS (IVAS-package), 2
IVAS-deprecated, 8
IVAS-package, 2

lm, 3, 4, 16
MsqtlFinder, 8
RatioFromFPKM, 2, 3, 8, 9, 15

sampleexp, 10
samplesnp, 10
samplesnplocus, 11
saveBplot, 12
seqinfo, 5, 9, 14

Splicingfinder, 2, 3, 8, 9, 13, 14, 15
sqtlfinder, 14
sQTLsFinder, 2, 3, 8, 12, 14, 14, 15