Package 'maser'

November 14, 2021

Type Package

Title Mapping Alternative Splicing Events to pRoteins

Version 1.12.0

Description This package provides functionalities for downstream analysis, annotation and visualizaton of alternative splicing events generated by rMATS.

Depends R (>= 3.5.0), ggplot2, GenomicRanges

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports dplyr, rtracklayer, reshape2, Gviz, DT, GenomeInfoDb, stats, utils, IRanges, methods, BiocGenerics, parallel, data.table

Suggests testthat, knitr, rmarkdown, BiocStyle, AnnotationHub

VignetteBuilder knitr

URL https://github.com/DiogoVeiga/maser

BugReports https://github.com/DiogoVeiga/maser/issues

RoxygenNote 6.0.1

biocViews AlternativeSplicing, Transcriptomics, Visualization

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

git_branch RELEASE_3_14

git_last_commit 254cfbe

git_last_commit_date 2021-10-26

Date/Publication 2021-11-14

Author Diogo F.T. Veiga [aut, cre]

Maintainer Diogo F.T. Veiga <diogof.veiga@gmail.com>

R topics documented:

`annotation,Maser-method`

Retrieve annotation of splicing events from a maser object.

Description

Retrieve annotation of splicing events from a maser object.

Usage

```
## S4 method for signature 'Maser'
annotation(object, type = c("A3SS", "A5SS", "SE", "RI",
  "MXE"))
```

Arguments

`object`	a maser object.
`type`	a character indicating the splice type. Possible values are `c("A3SS","A5SS","SE","RI","MXE")`.

Value

a data.frame.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
head(annotation(hypoxia, "SE"))
```

---

availableFeaturesUniprotKB *Query available human protein features in UniprotKB.*

Description

Query available human protein features in UniprotKB.

Usage

```
availableFeaturesUniprotKB()
```

Value

a data.frame.

Examples

```
head(availableFeaturesUniprotKB(), 10)
```

---

boxplot_PSI_levels *Boxplots of PSI distributions by splicing type.*

Description

Boxplots of PSI distributions by splicing type.

Usage

```
boxplot_PSI_levels(events, type = c("A3SS", "A5SS", "SE", "RI", "MXE"))
```

Arguments

events	a maser object.
type	character indicating splice type. Possible values are c("A3SS","A5SS","SE","RI","MXE")

Value

a ggplot object.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_filt <- filterByCoverage(hypoxia, avg_reads = 5)
boxplot_PSI_levels(hypoxia_filt, type = "RI")
```

---

counts,Maser-method *Retrieve raw read counts values from a maser object.*

Description

Retrieve raw read counts values from a maser object.

Usage

```
## S4 method for signature 'Maser'
counts(object, type = c("A3SS", "A5SS", "SE", "RI", "MXE"))
```

Arguments

object	a maser object.
type	a character indicating the splice type. Possible values are c("A3SS","A5SS","SE","RI","MXE").

Value

a matrix.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
head(counts(hypoxia, "SE"))
```

`display` *Visualization of splicing events annotation using an interactive data table.*

Description

Visualization of splicing events annotation using an interactive data table.

Usage

```
display(events, type = c("A3SS", "A5SS", "SE", "RI", "MXE"))
```

Arguments

`events`	a maser object.
`type`	character indicating splice type. Possible values are `c("A3SS","A5SS","SE","RI","MXE")`

Value

a datatables object.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_filt <- filterByCoverage(hypoxia, avg_reads = 5)
hypoxia_top <- topEvents(hypoxia_filt)
display(hypoxia_top, type = "SE")
```

`dotplot` *Dotplot representation of splicing events.*

Description

Dotplot representation of splicing events.

Usage

```
dotplot(events, type = c("A3SS", "A5SS", "SE", "RI", "MXE"), fdr = 0.05,
  deltaPSI = 0.1)
```

Arguments

`events`	a maser object.
`type`	character indicating splice type. Possible values are `c("A3SS","A5SS","SE","RI","MXE")`
`fdr`	numeric, FDR (False Discovery Rate) cutoff.
`deltaPSI`	numeric, absolute minimum PSI (Percent spliced-in) change

Value

a ggplot object.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_filt <- filterByCoverage(hypoxia, avg_reads = 5)
dotplot(hypoxia_filt, type = "SE")
```

---

filterByCoverage *Filter splicing events based on coverage.*

Description

Filter splicing events based on coverage.

Usage

```
filterByCoverage(events, avg_reads = 5)
```

Arguments

events	a maser object.
avg_reads	numeric, average number of reads covering the splice event.

Value

a maser object.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_filt <- filterByCoverage(hypoxia, avg_reads = 5)
```

filterByEventId *Filter splicing events based on event identifier and type.*

Description

Filter splicing events based on event identifier and type.

Usage

```
filterByEventId(events, event_id, type = c("A3SS", "A5SS", "SE", "RI", "MXE"))
```

Arguments

events	a maser object.
event_id	numeric vector of event identifiers.
type	character indicating splice type. Possible values are c("A3SS","A5SS","SE","RI","MXE").

Value

a maser object.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
filterByEventId(hypoxia, 33208, "SE")
```

geneEvents *Retrieve splicing events for a given gene.*

Description

Retrieve splicing events for a given gene.

Usage

```
geneEvents(events, geneS, fdr = 0.05, deltaPSI = 0.1)
```

Arguments

events	a maser object.
geneS	a character indicating the gene symbol.
fdr	numeric, FDR cutoff.
deltaPSI	numeric, minimum PSI change.

Value

a maser object.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_mib2 <- geneEvents(hypoxia, "MIB2")
```

---

granges,Maser-method *Retrieve genomic ranges of splicing events from a maser object.*

Description

Retrieve genomic ranges of splicing events from a maser object.

Usage

```
## S4 method for signature 'Maser'
granges(x, type = c("A3SS", "A5SS", "SE", "RI", "MXE"), ...)
```

Arguments

x	a maser object.
type	a character indicating the splice type. Possible values are c("A3SS","A5SS","SE","RI","MXE").
...	additional arguments.

Value

a GRangesList.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
head(granges(hypoxia, type = "SE"))
```

mapProteinFeaturesToEvents *Mapping of splice events to UniprotKB protein features.*

Description

Mapping of splice events to UniprotKB protein features.

Usage

```
mapProteinFeaturesToEvents(events, tracks, by = c("feature", "category"),
  ncores = 1)
```

Arguments

events	a maser object with transcript and protein identifiers.
tracks	a character vector indicating valid UniprotKB features or categories.
by	a character vector, possible values are `c("feature","category")`.
ncores	number of cores for multithreading (available only in OSX and Linux machines). If Windows, `ncores` will be set to 1 automatically.

Details

This function performs mapping of splicing events to protein features available in the UniprotKB database. Annotation tracks of protein features mapped to the hg38 build of the human genome are retrieved from the public UniprotKB FTP. The function will overlap exons involved in the splice event with the feature genomic coordinates retrieved from UniprotKB.

Annotation can be executed either by feature or category. If categories are provided, all features within the category group will be included for annotation.

Thus, batch annotation is enabled either by using `by = category` or by providing mutilple features in the `tracks` argument.

Visualization of protein features can be done using `plotUniprotKBFeatures`.

Value

a maser object with protein feature annotation.

See Also

`plotUniprotKBFeatures`

Examples

```
## Create the maser object
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_filt <- filterByCoverage(hypoxia, avg_reads = 5)

## Ensembl GTF annotation for SRSF6
gtf_path <- system.file("extdata", file.path("GTF", "SRSF6_Ensembl85.gtf"),
 package = "maser")
ens_gtf <- rtracklayer::import.gff(gtf_path)

## Retrieve gene specific splice events
srsf6_events <- geneEvents(hypoxia_filt, geneS = "SRSF6")

## Map splicing events to transcripts
srsf6_mapped <- mapTranscriptsToEvents(srsf6_events, ens_gtf)

## Annotate splice events with protein domains
srsf6_annot <- mapProteinFeaturesToEvents(srsf6_mapped, tracks = "domain")
head(annotation(srsf6_annot, "SE"))
```

`mapTranscriptsToEvents` *Mapping of splice events to Ensembl transcripts.*

Description

Mapping of splice events to Ensembl transcripts.

Usage

```
mapTranscriptsToEvents(events, gtf, ncores = 1)
```

Arguments

`events`	a maser object.
`gtf`	a `GRanges` object obtained from an Ensembl or Gencode GTF file using the hg38 build of the human genome.
`ncores`	number of cores for multithreading (available only in OSX and Linux machines). If Windows, `ncores` will be set to 1 automatically.

Details

This function performs mapping of splice events in the maser object to Ensembl transcripts by overlapping exons involved in the splice event to the transcript models provided in the GTF.

Each type of splice event requires a specific mapping procedure (described below).

The mapping will also add Uniprot identifiers when the ENST transcript encodes for a protein.

Visualization of affected transcripts can be done using plotTranscripts.

Exon skipping

Inclusion transcript(s) Transcript(s) overlapping the cassette exon, as well both flanking exons (i.e upstream and downstream exons).

Skipping transcript(s) Transcript(s) overlapping both flanking exons but not the cassettte exon.

Intron retention

Retention transcript(s) Transcript(s) overlapping exactly the retained intron.

Skipping transcript(s) Transcript(s) where intron is spliced out and overlapping both flanking exons.

Mutually exclusive exons

Exon1 transcript(s) Transcript(s) overlapping the first exon and both flanking exons.

Exon2 transcript(s) Transcript(s) overlapping the second exon and both flanking exons.

Alternative 3' and 5' splice sites

Short exon transcript(s) Transcript(s) overlapping both short and downstream exons.

Long exon transcript(s) Transcript(s) overlapping both long and downstream exons.

Value

a maser object with transcript and protein identifiers.

See Also

plotTranscripts

Examples

```
## Create the maser object
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_filt <- filterByCoverage(hypoxia, avg_reads = 5)

## Ensembl GTF annotation for SRSF6
gtf_path <- system.file("extdata", file.path("GTF",
 "Ensembl85_examples.gtf.gz"), package = "maser")
ens_gtf <- rtracklayer::import.gff(gtf_path)

## Retrieve gene specific splice events
srsf6_events <- geneEvents(hypoxia_filt, geneS = "SRSF6")

## Map splicing events to transcripts
srsf6_mapped <- mapTranscriptsToEvents(srsf6_events, ens_gtf)
head(annotation(srsf6_mapped, "SE"))
```

`maser` *Create a maser object by importing rMATS splicing events.*

Description

Create a maser object by importing rMATS splicing events.

Usage

```
maser(path, cond_labels, ftype = c("ReadsOnTargetAndJunctionCounts",
  "JunctionCountOnly", "JCEC", "JC"))
```

Arguments

`path`	a character specifiying the folder containing rMATS output files.
`cond_labels`	a character vector of length 2 describing labels for experimental conditions.
`ftype`	a character indicating the rMATS file type. Possible values are `c("ReadsOnTargetAndJunctionCounts"`

Details

This function creates a maser object by importing rMATS output. `ftype` indicates which rMATS files to import. `ReadsOnTargetandJunction` or `JunctionCountOnly` are used in rMATS 3.2.5 or lower. Newer versions (>4.0.1) use `"JCEC"` or `"JC"` nomenclature.

Value

A maser object.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
```

`Maser-class` *S4 class to represent splicing events imported from rMATS.*

Description

S4 class to represent splicing events imported from rMATS.

pca *Prinicipal component analysis of PSI distributions.*

Description

Prinicipal component analysis of PSI distributions.

Usage

```
pca(events, type = c("A3SS", "A5SS", "SE", "RI", "MXE"))
```

Arguments

events	a maser object.
type	character indicating splice type. Possible values are c("A3SS","A5SS","SE","RI","MXE")

Value

a ggplot object.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_filt <- filterByCoverage(hypoxia, avg_reads = 5)
pca(hypoxia_filt, type = "RI")
```

plotGenePSI *Boxplots of Percent spliced-in levels for gene events.*

Description

Boxplots of Percent spliced-in levels for gene events.

Usage

```
plotGenePSI(events, type = c("A3SS", "A5SS", "SE", "RI", "MXE"),
  show_replicates = TRUE)
```

Arguments

events	a maser object.
type	character indicating splice type. Possible values are c("A3SS","A5SS","SE","RI","MXE")
show_replicates	logical, add data points for individual replicates

Value

a ggplot object.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_filt <- filterByCoverage(hypoxia, avg_reads = 5)
hypoxia_mib2 <- geneEvents(hypoxia_filt, geneS = "MIB2")
plotGenePSI(hypoxia_mib2, type = "SE", show_replicates = TRUE)
```

---

plotTranscripts *Mapping and visualization of Ensembl transcripts affected by splicing.*

Description

Mapping and visualization of Ensembl transcripts affected by splicing.

Usage

```
plotTranscripts(events, type = c("A3SS", "A5SS", "SE", "RI", "MXE"), event_id,
  gtf, zoom = FALSE, show_PSI = TRUE)
```

Arguments

events	a maser object.
type	character indicating splice type. Possible values are c("A3SS","A5SS","SE","RI","MXE").
event_id	numeric, event identifier.
gtf	a GRanges, Ensembl or Gencode GTF using the hg38 build of the human genome.
zoom	logical, zoom to the genomic coordinates of the splice event.
show_PSI	logical, display the PSI track.

Details

This is a wrapper function for performing both mapping and visualization of Ensembl transcripts that are compatible with the splice event. This function calls mapTranscriptsToEvents for transcript mapping, which in turn uses findOverlaps for transcript overlapping. The GViz package is used for creating annotation tracks for genomic visualization of splicing events.

Each type of splice event requires a specific overlapping rule (described below), #' and a customized Gviz plot is created for each splicing type.

Exon skipping

Inclusion track Transcript(s) overlapping the cassette exon, as well both flanking exons (i.e upstream and downstream exons).

Skipping track Transcript(s) overlapping both flanking exons but not the cassettte exon.

Intron retention

Retention track Transcript(s) overlapping exactly the retained intron.

Skipping track Transcript(s) where intron is spliced out and overlapping both flanking exons.

Mutually exclusive exons

Exon1 track Transcript(s) overlapping the first exon and both flanking exons.

Exon2 track Transcript(s) overlapping the second exon and both flanking exons.

Alternative 3' and 5' splice sites

Short exon track Transcript(s) overlapping both short and downstream exons.

Long exon track Transcript(s) overlapping both long and downstream exons.

Value

a Gviz object.

See Also

mapTranscriptsToEvents

Examples

```
## Create the maser object
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_filt <- filterByCoverage(hypoxia, avg_reads = 5)

## Ensembl GTF annotation for SRSF6
 gtf_path <- system.file("extdata", file.path("GTF",
   "SRSF6_Ensembl85.gtf"), package = "maser")
 ens_gtf <- rtracklayer::import.gff(gtf_path)

## Retrieve gene specific splicing events
srsf6_events <- geneEvents(hypoxia_filt, geneS = "SRSF6")

## Plot exon skipping event
plotTranscripts(srsf6_events, type = "SE", event_id = 33209, gtf = ens_gtf)
```

---

plotUniprotKBFeatures *Mapping and visualization of UniprotKB protein features affected by splicing.*

Description

Mapping and visualization of UniprotKB protein features affected by splicing.

Usage

```
plotUniprotKBFeatures(events, type = c("A3SS", "A5SS", "SE", "RI", "MXE"),
  event_id, gtf, features, zoom = FALSE, show_transcripts = FALSE,
  show_PSI = TRUE, ncores = 1)
```

Arguments

events	a maser object.
type	character indicating splice type. Possible values are c("A3SS","A5SS","SE","RI","MXE").
event_id	numeric, event identifier.
gtf	a GRanges, Ensembl or Gencode GTF using the hg38 build of the human genome.
features	a character vector indicating valid UniprotKB features.
zoom	logical, zoom to the genomic coordinates of the splice event.
show_transcripts	logical, display transcripts track.
show_PSI	logical, display the PSI track.
ncores	number of cores for multithreading (available only in OSX and Linux machines). If Windows, ncores will be set to 1 automatically.

Details

This is a wrapper function for performing both mapping and visualization of protein features affected by the splice event. This function calls mapProteinFeaturesToEvents for mapping of protein features to splicing events.

The GViz package is used for creating annotation tracks for genomic visualization.

Multiple protein annotation tracks can be created using the features argument.

Value

a Gviz object.

See Also

mapProteinFeaturesToEvents

Examples

```
## Create the maser object
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_filt <- filterByCoverage(hypoxia, avg_reads = 5)

## Ensembl GTF annotation for SRSF6
gtf_path <- system.file("extdata", file.path("GTF",
   "SRSF6_Ensembl85.gtf"), package = "maser")
ens_gtf <- rtracklayer::import.gff(gtf_path)
```

```
## Retrieve gene specific splicing events
srsf6_events <- geneEvents(hypoxia_filt, geneS = "SRSF6")

## Map splicing events to transcripts
srsf6_mapped <- mapTranscriptsToEvents(srsf6_events, ens_gtf)

## Plot splice event, transcripts and protein features
plotUniprotKBFeatures(srsf6_mapped, "SE", event_id = 33209, gtf = ens_gtf,
  features = c("domain"), show_transcripts = TRUE)
```

PSI *Retrieve PSI (percent spliced in) values from a maser object.*

Description

Retrieve PSI (percent spliced in) values from a maser object.

Usage

```
PSI(events, type)
```

Arguments

events	a maser object.
type	a character indicating the splice type. Possible values are `c("A3SS","A5SS","SE","RI","MXE")`.

Value

a matrix.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
head(PSI(hypoxia, "SE"))
```

PSI,Maser,character-method
Retrieve PSI (percent spliced in) values from a maser object.

Description

Retrieve PSI (percent spliced in) values from a maser object.

Usage

```
## S4 method for signature 'Maser,character'
PSI(events, type = c("A3SS", "A5SS", "SE", "RI",
  "MXE"))
```

Arguments

events	a maser object.
type	a character indicating the splice type. Possible values are c("A3SS","A5SS","SE","RI","MXE").

Value

a matrix.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
head(PSI(hypoxia, "SE"))
```

splicingDistribution *Proportion of events by splicing type.*

Description

Proportion of events by splicing type.

Usage

```
splicingDistribution(events, fdr = 0.05, deltaPSI = 0.1)
```

Arguments

events	a maser object.
fdr	numeric, FDR (False Discovery Rate) cutoff.
deltaPSI	numeric, absolute minimum PSI (Percent spliced-in) change

Value

a ggplot object.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_filt <- filterByCoverage(hypoxia, avg_reads = 5)
splicingDistribution(hypoxia_filt)
```

---

summary,Maser-method *Retrieve rMATS stats of differential splicing from a maser object.*

Description

Retrieve rMATS stats of differential splicing from a maser object.

Usage

```
## S4 method for signature 'Maser'
summary(object, type = c("A3SS", "A5SS", "SE", "RI", "MXE"))
```

Arguments

`object`	a maser object.
`type`	a character indicating the splice type. Possible values are `c("A3SS","A5SS","SE","RI","MXE")`.

Value

a data.frame.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
head(summary(hypoxia, "SE"))
```

topEvents *Filter splicing events based on false discovery rate and PSI change.*

Description

Filter splicing events based on false discovery rate and PSI change.

Usage

```
topEvents(events, fdr = 0.05, deltaPSI = 0.1)
```

Arguments

events	a maser object.
fdr	numeric, FDR (False Discovery Rate) cutoff.
deltaPSI	numeric, absolute minimum PSI (Percent spliced-in) change

Value

a maser object.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))

## To select all events with minimum 10% change in PSI, and FDR < 0.01
hypoxia_top <- topEvents(hypoxia, fdr = 0.01, deltaPSI = 0.1)
```

volcano *Volcano plot of splicing events.*

Description

Volcano plot of splicing events.

Usage

```
volcano(events, type = c("A3SS", "A5SS", "SE", "RI", "MXE"), fdr = 0.05,
  deltaPSI = 0.1)
```

Arguments

events	a maser object.
type	character indicating splice type. Possible values are c("A3SS","A5SS","SE","RI","MXE")
fdr	numeric, FDR (False Discovery Rate) cutoff.
deltaPSI	numeric, absolute minimum PSI (Percent spliced-in) change

Value

a ggplot object.

Examples

```
path <- system.file("extdata", file.path("MATS_output"), package = "maser")
hypoxia <- maser(path, c("Hypoxia 0h", "Hypoxia 24h"))
hypoxia_filt <- filterByCoverage(hypoxia, avg_reads = 5)
volcano(hypoxia_filt, type = "SE")
```

Index