

FraseR

July 8, 2017

Type Package

Title Find RAre Splicing Events in RNA-Seq Data

Version 0.99.1

Date 2017-05-27

Author Christian Mertes <mertes@in.tum.de>, Prof. Julien Gagneur
<gagneur@in.tum.de>

Maintainer Christian Mertes <mertes@in.tum.de>

Description Detection of rare aberrant splicing events in transcriptome profiles. The workflow aims to assist the diagnostics in the field of rare diseases where RNA-seq is performed to identify splicing defects.

License Artistic-2.0

Depends R (>= 3.4), BiocParallel, data.table, Rsamtools,
SummarizedExperiment (>= 1.6.3)

Imports BBmisc, biomaRt, DelayedArray, DT, GenomicAlignments, GenomicRanges,
IRanges, graphics, gplots, htmlwidgets, HDF5Array, methods, parallel,
plotly (>= 4.6.0), rhdf5, Rsubread, R.utils, shiny, stats, tidy,
tools, VGAM

biocViews RNASeq, AlternativeSplicing, Sequencing,
Software, Genetics, Coverage

URL <https://il2g-gagneurweb.in.tum.de/gitlab/mertes/FraseR/>

RoxygenNote 6.0.1

Collate 'FraseR-package.R'
'FraseRDataSet-class.R'
'AllGenerics.R'
'Fraser-pipeline.R'
'accessor-methods.R'
'annotationOfRanges.R'
'beta-binomial-testing.R'
'calculatePSIValue.R'
'calculateStats.R'
'countRNAseqData.R'
'example_functions.R'
'getURLs.R'
'helper-functions.R'
'plotResults.R'
'saveHDF5Objects.R'
'shiny.R'

VignetteBuilder knitr

Suggests BiocStyle, knitr, testthat, covr, RSelenium

R topics documented:

annotateRanges	3
assayNames,FraseRDataSet-method	3
assays,FraseRDataSet-method	3
bamFile,FraseRDataSet-method	4
calculatePSIValues	4
calculatePValues	5
calculateZScores	5
cleanCache	5
condition,FraseRDataSet-method	6
countRNAData	7
counts,FraseRDataSet-method	7
createFullLinkTable	8
createTestFraseRDataSet	8
createTestFraseRSettings	8
FraseR	9
FraseRDataSet	10
FraseRDataSet-class	10
FraseRShinyApp	11
length,FraseRDataSet-method	11
loadFraseRDataSet	11
mcols,FraseRDataSet-method	12
method,FraseRDataSet-method	12
na2false	13
name,FraseRDataSet-method	13
nonSplicedReads,FraseRDataSet-method	14
parallel,FraseRDataSet-method	14
plotCountsAtSite	15
plotSampleResults	15
results	15
samples,FraseRDataSet-method	16
saveFraseRDataSet	16
scanBamParam,FraseRDataSet-method	17
strandSpecific,FraseRDataSet-method	18
subsetFraseR	18
workingDir,FraseRDataSet-method	19

annotateRanges	<i>Annotates the given FraseRDataSet with the HGNC symbol with biomaRt</i>
----------------	--

Description

Annotates the given FraseRDataSet with the HGNC symbol with biomaRt

Usage

```
annotateRanges(fds, feature = "hgnc_symbol",
               biotype = list("protein_coding"), ensembl = NULL)
```

assayNames, FraseRDataSet-method	<i>Returns the assayNames of FraseR</i>
----------------------------------	---

Description

Returns the assayNames of FraseR

Usage

```
## S4 method for signature 'FraseRDataSet'
assayNames(x)
```

assays, FraseRDataSet-method	<i>Returns the assay corresponding to the given name/index of the FraseRDataSet</i>
------------------------------	---

Description

Returns the assay corresponding to the given name/index of the FraseRDataSet

Usage

```
## S4 method for signature 'FraseRDataSet'
assays(x, ..., type = NULL, withDimnames = TRUE)
```

```
bamFile, FraseRDataSet-method
  Get/Set the bamFile
```

Description

Get/Set the bamFile

Usage

```
## S4 method for signature 'FraseRDataSet'
bamFile(object)

## S4 replacement method for signature 'FraseRDataSet'
bamFile(object) <- value
```

Arguments

object A FraseRDataSet object.

Value

A vector with the bamFile for each sample

Author(s)

Christian Mertes <mertes@in.tum.de>

Examples

```
settings <- createTestFraseRSettings()
bamFile(settings)
bamFile(settings) <- file.path("bamfiles", samples(settings), "rna-seq.bam")
```

```
calculatePSIValues This function calculates the PSI values for each junction and splice
site based on the FraseRDataSet object
```

Description

This function calculates the PSI values for each junction and splice site based on the FraseRDataSet object

Usage

```
calculatePSIValues(fds)
```

Examples

```
fds <- countRNAData(createTestFraseRSettings())
fds <- calculatePSIValues(fds)
```

calculatePValues	<i>calculates the P-Value for the given FraseR dataset object The P-Value calculation is based on the given method in the FraseRSettings object</i>
------------------	---

Description

calculates the P-Value for the given FraseR dataset object The P-Value calculation is based on the given method in the FraseRSettings object

Usage

```
calculatePValues(fds, internBPPARAM = SerialParam(), ...)
```

Examples

```
fds <- countRNAData(createTestFraseRSettings())
fds <- calculatePSIValues(fds)
fds <- calculatePValues(fds)
```

calculateZScores	<i>Calculate the zscore for each PSI value.</i>
------------------	---

Description

Calculate the zscore for each PSI value.

Usage

```
calculateZScores(fds)
```

Examples

```
fds <- countRNAData(createTestFraseRSettings())
fds <- calculatePSIValues(fds)
fds <- calculateZScores(fds)
```

cleanCache	<i>clear the files in the cache to start fresh</i>
------------	--

Description

clear the files in the cache to start fresh

Usage

```
cleanCache(fds, all = FALSE, cache = TRUE, assays = FALSE,
  results = FALSE)
```

Examples

```
fds <- createTestFraseRSettings()  
cleanCache(fds)
```

condition, FraseRDataSet-method

Get the condition for each sample

Description

Get the condition for each sample

Usage

```
## S4 method for signature 'FraseRDataSet '  
condition(object)  
  
## S4 replacement method for signature 'FraseRDataSet '  
condition(object) <- value
```

Arguments

object A FraseRDataSet object.

Value

A vector with the condition per sample. If the condition column is not set, sequence of numbers is returned.

Author(s)

Christian Mertes <mertes@in.tum.de>

Examples

```
fds <- createTestFraseRSettings()  
condition(fds)  
condition(fds) <- 1:dim(fds)[2]
```

countRNAData	<i>Count RNA-seq data</i>
--------------	---------------------------

Description

This method extracts and counts the split reads and non spliced reads from a RNA bam file

Usage

```
countRNAData(fds, NcpuPerSample = 1, junctionMap = NULL)
```

Arguments

fds	A FraseRDataSet object with all the information how and what to count
NcpuPerSample	A BiocParallel param object or a positive integer to configure the parallel back-end of the internal loop per sample
junctionMap	A object or file containing a map of all junctions of interest across all samples

Details

TODO

Value

FraseRDataSet

Examples

```
countRNAData(createTestFraseRSettings())
countRNAData(createTestFraseRSettings(), 5)
```

counts, FraseRDataSet-method
<i>getter for count data</i>

Description

getter for count data

Usage

```
## S4 method for signature 'FraseRDataSet'
counts(object, type = NULL, side = c("ofInterest",
  "otherSide"))
```

```
createFullLinkTable
      get links
```

Description

get links

Usage

```
createFullLinkTable(data, addHRef = TRUE)
```

```
createTestFraseRDataSet
```

Create a test case dataset based on the test sample annotation filled with counts to be used in the vignette and to explore the functionality of the FraseR package.

Description

Create a test case dataset based on the test sample annotation filled with counts to be used in the vignette and to explore the functionality of the FraseR package.

Usage

```
createTestFraseRDataSet(BPPARAM = NULL)
```

Value

a FraseRDataSet object which contains a test case

Examples

```
createTestFraseRDataSet()
```

```
createTestFraseRSettings
```

Create a test case dataset (sample information only) to be used in the vignette and to explore the functionality of the FraseR package.

Description

Create a test case dataset (sample information only) to be used in the vignette and to explore the functionality of the FraseR package.

Usage

```
createTestFraseRSettings()
```


Value

a FraseRSettings object which contains a test case

Examples

```
createTestFraseRSettings()
```

FraseR	<i>FraseR: A package providing a workflow to detect aberrant splicing events in RNA sequencing data in a rare disease cohort</i>
--------	--

Description

FraseR: A package providing a workflow to detect aberrant splicing events in RNA sequencing data in a rare disease cohort

This function performs a default analysis of RNA-seq data

Usage

```
FraseR(settings = createTestFraseRSettings(), NcpuPerSample = 1)
```

Arguments

settings	A FraseRDataSet object with all the information how and what to count
NcpuPerSample	A BiocParallel param object to configure the parallel backend of the internal loop for counting

Value

FraseRDataSet

Examples

```
fds <- FraseR(createTestFraseRSettings())
plotSampleResults(fds)
```

FraserDataSet

The constructor function for FraserSettings

Description

The constructor function for FraserSettings

Usage

```
FraserDataSet(colData = NULL, ...)
```

Arguments

... Any parameters corresponding to the slots and their possible values. See [FraserDataSet](#)

Value

A FraserDataSet object.

Author(s)

Christian Mertes <mertes@in.tum.de>

Examples

```
fraser <- FraserDataSet()  
fraser <- countRNADData(createTestFraserSettings())
```

FraserDataSet-class*FraserDataSet*

Description

This class is designed to store the whole Fraser data set needed for an analysis of a disease cohort

Author(s)

Christian Mertes <mertes@in.tum.de>

FraseRShinyApp	<i>Present the FraseR results as shiny app</i>
----------------	--

Description

Present the FraseR results as shiny app

Usage

```
FraseRShinyApp(fds, fdsres = NULL, server = !interactive(), ...)
```

Examples

```
fds <- createTestFraseRDataSet()

# for interactive sessions
myShinyApp <- FraseRShinyApp(fds)

# for running a shiny application as server
myShinyAppObj <- FraseRShinyApp(fds, server=TRUE)
```

length, FraseRDataSet-method	<i>retrive the length of the object (aka number of junctions)</i>
------------------------------	---

Description

retrive the length of the object (aka number of junctions)

Usage

```
## S4 method for signature 'FraseRDataSet'
length(x)
```

loadFraseRDataSet	<i>Load a saved FraseR object into memory</i>
-------------------	---

Description

Load a saved FraseR object into memory

Usage

```
loadFraseRDataSet(dir, name = NULL)
```

Arguments

dir	a path to the working directory of FraseR
-----	---

Examples

```
loadFraseRDataSet(file.path(Sys.getenv("HOME"), "FraseR"))
```

```
mcols,FraseRDataSet-method
```

getter and setter for mcols

Description

getter and setter for mcols

Usage

```
## S4 method for signature 'FraseRDataSet'
mcols(x, type = NULL, ...)
```

```
method,FraseRDataSet-method
```

Get/Set the statistical method to use for P-value calculation from the FraseRDataSet object

Description

Get/Set the statistical method to use for P-value calculation from the FraseRDataSet object

Usage

```
## S4 method for signature 'FraseRDataSet'
method(object)

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

Arguments

object A FraseRDataSet object.

Value

A character string representing the statistical method

Author(s)

Christian Mertes <mertes@in.tum.de>

Examples

```
settings <- createTestFraseRSettings()
method(settings)
method(settings) <- "betaBin"
```

na2false	<i>convert all NA's of a input vector or of a single dimension matrix/data.table to FALSE</i>
----------	---

Description

convert all NA's of a input vector or of a single dimension matrix/data.table to FALSE

Usage

```
na2false(x)
```

Examples

```
a <- c(TRUE, FALSE, NA, TRUE, NA)
na2false(a)

dt <- data.table(a)
na2false(dt)
```

name, FraseRDataSet-method	<i>Get/Set the name of the analysis</i>
----------------------------	---

Description

Get/Set the name of the analysis

Usage

```
## S4 method for signature 'FraseRDataSet'
name(object)

## S4 replacement method for signature 'FraseRDataSet'
name(object) <- value
```

Arguments

object A FraseRDataSet object.

Value

A character string representing the name of the analysis

Author(s)

Christian Mertes <mertes@in.tum.de>

Examples

```
settings <- createTestFraserSettings()
name(settings)
name(settings) <- "My Analysis"
```

```
nonSplicedReads,FraserDataSet-method
```

accessor for the non spliced reads object within the FraserDataSet object

Description

accessor for the non spliced reads object within the FraserDataSet object

setter for the non spliced reads object within the FraserDataSet object

Usage

```
## S4 method for signature 'FraserDataSet '
nonSplicedReads(object)

## S4 replacement method for signature 'FraserDataSet '
nonSplicedReads(object) <- value
```

```
parallel,FraserDataSet-method
```

Get/Set the parallel param object from the FraserDataSet object

Description

Get/Set the parallel param object from the FraserDataSet object

Usage

```
## S4 method for signature 'FraserDataSet '
parallel(object)

## S4 replacement method for signature 'FraserDataSet '
parallel(object) <- value
```

Arguments

object A FraserDataSet object.

Value

A parallel param object

Author(s)

Christian Mertes <mertes@in.tum.de>

Examples

```
settings <- createTestFraseRSettings()
parallel(settings)
parallel(settings) <- SerialParam()
```

plotCountsAtSite	<i>plot count distribution</i>
------------------	--------------------------------

Description

plot count distribution

Usage

```
plotCountsAtSite(gr, fds, type, sample = NULL, plotLog = TRUE)
```

plotSampleResults	<i>Plot the results of the FraseR analysis pipeline All three types are plotted into one HTML file based on plotly.</i>
-------------------	---

Description

Plot the results of the FraseR analysis pipeline All three types are plotted into one HTML file based on plotly.

Usage

```
plotSampleResults(fds, sampleID = NULL, file = NULL, dir = NULL,
  browseIt = FALSE)
```

Examples

```
fds <- FraseR()
plotSampleResults(fds, "sample1")
plotSampleResults(fds, "sample1", "result.html")
```

results	<i>obtain the results for the given analysis pipeline</i>
---------	---

Description

obtain the results for the given analysis pipeline

Usage

```
results(fds, sampleIDs = samples(fds), pvalueCut = 1e-05, zscoreCut = 2,
  psiType = c("psi3", "psi5", "psiSite"), redo = FALSE)
```

```
samples, FraserDataSet-method
```

Getter/Setter for the sampleIDs

Description

Getter/Setter for the sampleIDs

Usage

```
## S4 method for signature 'FraserDataSet'
samples(object)

## S4 replacement method for signature 'FraserDataSet'
samples(object) <- value
```

Arguments

object A FraserDataSet object.

Value

A vector with all sample IDs

Author(s)

Christian Mertes <mertes@in.tum.de>

Examples

```
fds <- createTestFraserSettings()
samples(fds)
samples(fds) <- 1:dim(fds)[2]
```

```
saveFraserDataSet    Saves the FraserDataSet object on disk under the given working dir.
                     It furthermore uses HDF5 to save all internal assays
```

Description

After saving this can be loaded again with the corresponding load function

Usage

```
saveFraserDataSet(fds, dir = NULL)
```

Arguments

fds A FraserDataSet object to be saved
 dir a directory name where to save the objects (replaces the working directory)

Examples

```
fds <- countRNAData(createTestFraseRSettings())
saveFraseRDataSet(fds)
```

scanBamParam, FraseRDataSet-method

Get/Set the ScanBamParam object from the FraseRDataSet object

Description

Get/Set the ScanBamParam object from the FraseRDataSet object

Usage

```
## S4 method for signature 'FraseRDataSet'
scanBamParam(object)

## S4 replacement method for signature 'FraseRDataSet'
scanBamParam(object) <- value
```

Arguments

object A FraseRDataSet object.

Value

A ScanBamParam object

Author(s)

Christian Mertes <mertes@in.tum.de>

Examples

```
settings <- createTestFraseRSettings()
scanBamParam(settings)
scanBamParam(settings) <- ScanBamParam(mapqFilter=30)
```

```
strandSpecific, FraseRDataSet-method
```

Get/Set if the analysis is strand specific or not

Description

Get/Set if the analysis is strand specific or not

Usage

```
## S4 method for signature 'FraseRDataSet '
strandSpecific(object)

## S4 replacement method for signature 'FraseRDataSet '
strandSpecific(object) <- value
```

Arguments

object A FraseRDataSet object.

Value

A logical value if the analysis is strand specific

Author(s)

Christian Mertes <mertes@in.tum.de>

Examples

```
settings <- createTestFraseRSettings()
strandSpecific(settings)
strandSpecific(settings) <- TRUE
```

```
subsetFraseR                    Subsetting by indices for junctions
```

Description

Providing subsetting by indices through the single-bracket operator

Usage

```
subsetFraseR(x, i, j)
```

Arguments

x A FraseRDataSet object
i A integer vector to subset the rows/ranges
j A integer vector to subset the columns/samples

Value

A subsetting FraseRDataSet object

Examples

```
fds <- countRNAData(createTestFraseRSettings())
fds[1:10, 1:10]
fds[, samples(fds) %in% c("sample1", "sample2")]
```

workingDir, FraseRDataSet-method

Get/Set the working directory from the FraseRDataSet object

Description

Get/Set the working directory from the FraseRDataSet object

Usage

```
## S4 method for signature 'FraseRDataSet'
workingDir(object)

## S4 replacement method for signature 'FraseRDataSet'
workingDir(object) <- value
```

Arguments

object A FraseRDataSet object.

Value

A path

Author(s)

Christian Mertes <mertes@in.tum.de>

Examples

```
settings <- createTestFraseRSettings()
workingDir(settings)
workingDir(settings) <- tempdir()
```

Index

annotateRanges, [3](#)
assayNames, FraseRDataSet-method, [3](#)
assays, FraseRDataSet-method, [3](#)

bamFile, FraseRDataSet-method, [4](#)
bamFile<-, FraseRDataSet-method
(bamFile, FraseRDataSet-method), [4](#)

calculatePSIValues, [4](#)
calculatePValues, [5](#)
calculateZScores, [5](#)
cleanCache, [5](#)
condition, FraseRDataSet-method, [6](#)
condition<-, FraseRDataSet-method
(condition, FraseRDataSet-method), [6](#)

countRNAData, [7](#)
counts, FraseRDataSet-method, [7](#)
createFullLinkTable, [8](#)
createTestFraseRDataSet, [8](#)
createTestFraseRSettings, [8](#)

FraseR, [9](#)
FraseRDataSet, [10](#), [10](#)
FraseRDataSet-class, [10](#)
FraseRShinyApp, [11](#)

length, FraseRDataSet-method, [11](#)
loadFraseRDataSet, [11](#)

mcols, FraseRDataSet-method, [12](#)
method, FraseRDataSet-method, [12](#)
method<-, FraseRDataSet-method
(method, FraseRDataSet-method), [12](#)

na2false, [13](#)
name, FraseRDataSet-method, [13](#)
name<-, FraseRDataSet-method
(name, FraseRDataSet-method), [13](#)

nonSplicedReads, FraseRDataSet-method, [14](#)
nonSplicedReads<-, FraseRDataSet-method
(nonSplicedReads, FraseRDataSet-method), [14](#)

parallel, FraseRDataSet-method, [14](#)
parallel<-, FraseRDataSet-method
(parallel, FraseRDataSet-method), [14](#)

plotCountsAtSite, [15](#)
plotSampleResults, [15](#)

results, [15](#)

samples, FraseRDataSet-method, [16](#)
samples<-, FraseRDataSet-method
(samples, FraseRDataSet-method), [16](#)

saveFraseRDataSet, [16](#)
scanBamParam, FraseRDataSet-method, [17](#)
scanBamParam<-, FraseRDataSet-method
(scanBamParam, FraseRDataSet-method), [17](#)

strandSpecific, FraseRDataSet-method, [18](#)
strandSpecific<-, FraseRDataSet-method
(strandSpecific, FraseRDataSet-method), [18](#)

subsetFraseR, [18](#)

workingDir, FraseRDataSet-method, [19](#)
workingDir<-, FraseRDataSet-method
(workingDir, FraseRDataSet-method), [19](#)