Package 'kissDE'

November 12, 2026

Version 1.32.0

Title Retrieves Condition-Specific Variants in RNA-Seq Data

Description Retrieves condition-specific variants in RNA-seq data (SNVs, alternative-splicings, indels). It has been developed as a post-treatment of 'KisSplice' but can also be used with user's own data.

Imports aods3, Biobase, DESeq2, DSS, ggplot2, gplots, graphics, grDevices, matrixStats, stats, utils, foreach, doParallel, parallel, shiny, shinycssloaders, ade4, factoextra, DT, rlang

Suggests BiocStyle, quarto, testthat

License GPL (>= 2)

Contact Vincent Lacroix <vincent.lacroix@univ-lyon1.fr>

Encoding UTF-8

biocViews AlternativeSplicing, DifferentialSplicing, ExperimentalDesign, GenomicVariation, RNASeq, Transcriptomics

URL https://github.com/lbbe-software/kissDE

BugReports https://github.com/lbbe-software/kissDE/issues

VignetteBuilder quarto

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

git_branch RELEASE_3_23

git_last_commit b642202

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Author Clara Benoit-Pilven [aut],
Camille Marchet [aut],
Janice Kielbassa [aut],
Lilia Brinza [aut],
Audric Cologne [aut],
Aurelie Siberchicot [aut, cre] (ORCID:
<https://orcid.org/0000-0002-7638-8318>),

Vincent Lacroix [aut],
Frank Picard [ctb],
Laurent Jacob [ctb],
Vincent Miele [ctb]

Maintainer Aurelie Siberchicot <aurelie.siberchicot@univ-lyon1.fr>

Contents

kissDE-package 2
diffExpressedVariants 3
exploreResults 5
kissDE 6
kissplice2counts 8
qualityControl 10
writeOutputKissDE 11

Index **13**

`kissDE-package` *Retrieves condition-specific variants in RNA-seq data*

Description

The **kissDE** package retrieves condition-specific variants in RNA-seq data. Each variation (SNVs, alternative splicing events) is represented as a pair of variants. The quantification of each variant is summarized as a count, in each condition and each replicate where it was measured. The package tests for enrichment of a variant in a condition. Data counts are modelled using either a poisson or a negative binomial. Likelihood ratio tests are then performed using the GLM (Generalized Linear Model) framework.

Details

Main functions:

diffExpressedVariants(countsData, conditions, pvalue = 1, filterLowCountsVariants = 10, flagLowCountsConditions = 10, technicalReplicates = FALSE)

qualityControl(countsData, conditions, storeFigs = FALSE)

kissplice2counts(fileName, counts = 0, pairedEnd = FALSE, order = NULL, exonicReads = TRUE, k2rg = FALSE, keep = c("All"), remove = NULL)

writeOutputKissDE(resDiffExprVariant, output, adjPvalMax = 1, dPSImin = 0, writePSI = FALSE)

kissDE(fileName, conditions, output, counts = 2, pairedEnd = FALSE, order = NULL, exonicReads = TRUE, k2rg = FALSE, keep = c("All"), remove = NULL, pvalue = 1, filterLowCountsVariants = 10, flagLowCountsConditions = 10, technicalReplicates = FALSE, nbCore = 1, adjPvalMax = 1, dPSImin = 0, writePSI = FALSE)

Note

Authors of the package: Clara Benoit-Pilven, Camille Marchet, Janice Kielbassa, Lilia Brinza, Audric Cologne and Vincent Lacroix all contributed code and ideas.

Contributors of the package: Franck Picard and Laurent Jacob provided statistical expertise for the models underlying kissDE. Vincent Miele provided expertise for the development of the R package.

Maintainer of the package: Aurélie Siberchicot

`diffExpressedVariants` *Retrieve condition-specific variants in RNA-seq data*

Description

Function that retrieves condition-specific variants in RNA-seq data.

Usage

```
diffExpressedVariants(countsData, conditions, pvalue = 1,
    filterLowCountsVariants = 10, flagLowCountsConditions = 10,
    technicalReplicates = FALSE,
    nbCore = 1)
```

Arguments

`countsData`	a data frame containing the counts in the appropriate format (see Details below).
`conditions`	a character vector containing the experimental conditions.
`pvalue`	a numerical value indicating the p-value threshold below which the events will be kept in the final data frame.
`filterLowCountsVariants`	a numerical value indicating the global variant count value (see Details below) below which events are filtered out in order to increase statistical power of the analysis. Both variant must have a read coverage below this value in order to remove the event. This filter is done after the normalization and the overdispersion estimation.
`flagLowCountsConditions`	a numerical value indicating the global condition count value (see Details below) below which we flag events as 'lowCounts' in the final data frame. At least n-1 conditions (over n conditions) must have low counts to flag the event as 'lowCounts'.
`technicalReplicates`	a boolean value indicating if the counts in `countsData` come from technical replicates only or not.
`nbCore`	an integer indicating the number of cores to use for the model fitting step.

Details

The countsData data frame must be formatted as follows:

- Column 1: names of the events
- Column 2: lengths (in bp) of the variants
- Column 3 to n: counts corresponding to each replicate of each experimental condition of one variant

Each row corresponds to one variant, thus an event correspond to two rows with the longest variant (or inclusion variant) in the first row (thus denotated as upper path: UP) and the smallest variant (or exclusion variant) in the second row (thus denotated as lower path: LP). This data frame can be obtained using kissplice2counts function.\ The global variant count is the minimal number of reads that cover one or the other variant across all the replicates (sum by variant).\ The global condition count is the minimal number of reads that cover one or the other condition (sum by replicates for each conditions).

Value

diffExpressedVariants returns a list of 6 objects:

finalTable — a data frame containing the columns

- ID: the variation identifier
- Length_diff: the size of the variable region
- UP_Condi_Rj_Norm (resp LP_Condi_Rj_Norm): returns the normalized counts of the first variant (UP, resp. second variant: LP), for the condition i (Condi) and the replicate j (Rj)
- Adjusted_pvalue: p-value adjusted for multiple testing with Benjamini & Hochberg method
- Deltaf/DeltaPSI: difference of relative abundance of variants across conditions. For instance if there are 2 conditions, deltaPSI returns relative abudance in condition 2 - relative abundance in condition 1. Inclusion variant's counts are corrected for the length of the variant so that we do not overestimate the PSI value.
- lowcounts: a column that flag low counts in data. If TRUE, at least n-1 conditions over n conditions have less than 10 reads.

correctedPval — a numeric vector containing p-values after correction for multiple testing

uncorrectedPVal — a numeric vector containing p-values before correction for multiple testing

resultFitNBglmModel — a data frame containing the results of the fitting of the model to the data

f/psiTable — a data frame containing the allele frequency (f)/Percent Spliced In (PSI) of each replicate

k2rgFile — a string containing either the KisSplice2RefGenome file path and name or NULL if no KisSplice2RefGenome input file was given

References

Lopez-Maestre et al., 2016. Snp calling from rna-seq data without a reference genome: identification, quantification, differential analysis and impact on the protein sequence. *Nucleic Acids Research*, **44**(19):e148. doi:10.1093/nar/gkw655

Examples

```
fpath1 <- system.file("extdata", "output_kissplice_SNV.fa", package = "kissDE")
mySNVcounts <- kissplice2counts(fpath1, counts = 0, pairedEnd = TRUE)
mySNVconditions <- c("EUR", "EUR", "TSC", "TSC")
# diffSNV <- diffExpressedVariants(mySNVcounts, mySNVconditions)

fpath2 <- system.file("extdata", "table_counts_alt_splicing.txt",
package = "kissDE")
mySplicingconditions <- c("C1", "C1", "C2", "C2")
mySplicingcounts <- read.table(fpath2, header = TRUE)
# diffSplicing <- diffExpressedVariants(mySplicingcounts, mySplicingconditions)
```

---

exploreResults *Print kissDE results in an interactive Shiny application*

Description

This function will read the rds file created by writeOutputKissDE and creat an interactive Shiny application allowing users to explore and plot the results of kissDE.

Usage

```
exploreResults(rdsFile)
```

Arguments

rdsFile	a string indicating the path to the rds file outputed by writeOutputKissDE.

Value

None.

Examples

```
kissplice2refgenome_file <- system.file("extdata",
    "output_k2rg_alt_splicing.txt", package="kissDE")
mySplicingconditions <- c("C1", "C1", "C2", "C2")
counts <- kissplice2counts(fileName=kissplice2refgenome_file, counts=2,
    pairedEnd=TRUE, k2rg=TRUE)
# res <- diffExpressedVariants(countsData=counts,
#   conditions=mySplicingconditions)
# writeOutputKissDE(res, output="results.tsv")
# exploreResults("results.tsv.rds")
```

kissDE *Run the whole kissDE analysis*

Description

This function will sequentially call the kissplice2counts, diffExpressedVariants and writeOutputKissDE functions in order to run a complete kissDE analysis. It may also call the qualityControl and the exploreResults functions.

Usage

```
kissDE(fileName, conditions, output, counts = 2, pairedEnd = FALSE,
     order = NULL, exonicReads = TRUE, k2rg = FALSE, keep = c("All"), remove = NULL,
       pvalue = 1, filterLowCountsVariants = 10, flagLowCountsConditions = 10,
       technicalReplicates = FALSE, nbCore = 1, adjPvalMax = 1, dPSImin = 0,
       writePSI = TRUE, doQualityControl = TRUE, resultsInShiny=TRUE)
```

Arguments

fileName	a string indicating the path to the KisSplice (.fa) or the KisSplice2RefGenome (tab-delimited) file.
conditions	a character vector containing the experimental conditions.
output	a character indicating the path and file name to save writeOutputKissDE output.
counts	an interger (0, 1 or 2) corresponding to the KisSplice counts option used (see Details below).
pairedEnd	a logical indicating if the data is paired-end (FALSE, default). If set to TRUE, the sum of the counts from the pair of reads will be computed. It can be used along with counts option. By default, it is assumed that, in the KisSplice command line, two reads of the same pair has been inputed as following each other. If it is not the case, see option order.
order	a numeric vector indicating the actual order of the corresponding paired reads in the columns of the KisSplice output such that they can be summed. This option goes along with pairedEnd = TRUE, if the read pairs are not in the expected order (see pairedEnd option). It has as many elements as there are samples in total. For more information on this parameter, see Details in kissplice2counts.
exonicReads	a logical indicating if exonic/intronic read counts will be kept (TRUE, default) or discareded (FALSE). This option only works if counts = 2.
k2rg	a logical indicating if the input file is a KisSplice2RefGenome (TRUE) output or a KisSplice (FALSE, default) output file.
keep	a character vector listing the names of the events to be kept for the statistical test (for k2rg = TRUE, analyses all of the events by default). The test will be more sensitive the selected events. Event(s) name(s) must be part of this list: deletion, insertion, indel, IR, ES, altA, altD, altAD, alt, - (for unclassified events). For more information on this parameter, see Details in kissplice2counts.

`remove` a character vector listing the names of the events to remove for the statistical test (for `k2rg = TRUE`, does not remove any event by default). The test will be more sensitive for the non-selected events. Event(s) name(s) must be part of this list: deletion, insertion, indel, IR, ES, altA, altD, altAD, alt, - (for unclassified events), MULTI. This option can not be used along with the `keep` option, unless ES is one of the events to be kept. In this case, the `remove` option will work on specific ES events. For more information on this parameter, see Details in `kissplice2counts`.

`pvalue` a numerical value indicating the p-value threshold below which the events will be kept in the final data frame.

`filterLowCountsVariants` a numerical value indicating the global variant count value (see Details below) below which events are filtered out in order to increase statistical power of the analysis. Both variant must have a read coverage below this value in order to remove the event. This filter is done after the normalization and the overdispersion estimation.

`flagLowCountsConditions` a numerical value indicating the global condition count value (see Details below) below which we flag events as 'lowCounts' in the final data frame. At least n-1 conditions (over n conditions) must have low counts to flag the event as 'lowCounts'.

`technicalReplicates` a boolean value indicating if the counts in `countsData` come from technical replicates only or not.

`nbCore` an integer indicating the number of cores to use for the model fitting step.

`adjPvalMax` a double indicating the threshold for adjusted p-value. Only SNVs/splicing events with an adjusted p-value lower than this threshold will be kept in the output file.

`dPSImin` a double indicating the threshold for the deltaPSI. Only SNVs/splicing events having an absolute value of deltaf/deltaPSI higher than this threshold will be kept in the output file.

`writePSI` a boolean indicating if the user wants the f/PSI table to be printed (`TRUE`, default) along with the final table (`FALSE`).

`doQualityControl` a boolean indicating if the user wants quality control plots to be written in the output folder (`TRUE`, default) or not (`FALSE`). See details in `qualityControl`.

`resultsInShiny` a boolean indicating if the user wants the results to be printed in a Shiny application (`TRUE`, default) or not (`FALSE`). See details in `exploreResults`.

Value

None.

Examples

```
kissplice2refgenome_file <- system.file("extdata",
    "output_k2rg_alt_splicing.txt", package="kissDE")
```

```
mySplicingconditions <- c("C1", "C1", "C2", "C2")
#kissDE(fileName=kissplice2refgenome_file, conditions=mySplicingconditions,
#       output="results.tsv", counts=2, pairedEnd=TRUE, k2rg=TRUE)
```

---

kissplice2counts *Conversion of* KisSplice *or* KisSplice2RefGenome *outputs*

Description

Function that converts KisSplice (.fa) output or KisSplice2RefGenome (tab-delimited) output to a counts data frame that can be used by other functions of the KissDE package.

Usage

```
kissplice2counts(fileName, counts = 2, pairedEnd = FALSE, order = NULL,
    exonicReads = TRUE, k2rg = FALSE, keep = c("All"), remove = NULL)
```

Arguments

fileName	a string indicating the path to the KisSplice (.fa) or the KisSplice2RefGenome (tab-delimited) file.
counts	an interger (0, 1 or 2) corresponding to the KisSplice counts option used (see Details below).
pairedEnd	a logical indicating if the data is paired-end (FALSE, default). If set to TRUE, the sum of the counts from the pair of reads will be computed. It can be used along with counts option. By default, it is assumed that, in the KisSplice command line, two reads of the same pair has been inputed as following each other. If it is not the case, see option order.
order	a numeric vector indicating the actual order of the corresponding paired reads in the columns of the KisSplice output such that they can be summed. This option goes along with pairedEnd = TRUE, if the read pairs are not in the expected order (see pairedEnd option). It has as many elements as there are samples in total. For more information on this parameter, see Details below.
exonicReads	a logical indicating if exonic/intronic read counts will be kept (TRUE, default) or discareded (FALSE). This option only works if counts = 2.
k2rg	a logical indicating if the input file is a KisSplice2RefGenome (TRUE) output or a KisSplice (FALSE, default) output file.
keep	a character vector listing the names of the events to be kept for the statistical test (for k2rg = TRUE, analyses all of the events by default). The test will be more sensitive the selected events. Event(s) name(s) must be part of this list: deletion, insertion, indel, IR, ES, altA, altD, altAD, alt, - (for unclassified events). For more information on this parameter, see Details below.

`remove` a character vector listing the names of the events to remove for the statistical test (for `k2rg = TRUE`, does not remove any event by default). The test will be more sensitive for the non-selected events. Event(s) name(s) must be part of this list: deletion, insertion, indel, IR, ES, altA, altD, altAD, alt, - (for unclassified events), MULTI. This option can not be used along with the `keep` option, unless ES is one of the events to be kept. In this case, the `remove` option will work on specific ES events. For more information on this parameter, see Details below.

Details

The `counts` parameter:

By default, as in `KisSplice`, the `counts` option is set to 0, assuming there is no special counting option. Below, an example of the upper path counts format output by `KisSplice` when `counts` is set to 2:

|AS1_0|SB1_0|S1_0|ASSB1_0|AS2_27|SB2_41|S2_0|ASSB2_21|

AS3_0|SB3_0|S3_0|ASSB3_0|AS4_7|SB4_8|S4_0|ASSB4_2.

In a regular `KisSplice` output (`counts = 0`), it would be:

|C1_0|C2_47|C3_1|C4_13 (with 47 = 27+41+0-21 and 13 = 7+8+0-2)

The `order` parameter:

If the reads corresponding to a paired-end fragments have not been passed to `Kissplice` next to each other, the order needs to be explicitly given to the `kissplice2counts` function. For instance, if there are two paired-end samples and if the input in `Kissplice` has been: `-r sample1_readPair1.fa -r sample2_readPair1.fa -r sample1_readPair2.fa -r sample2_readPair2.fa`, the input is not organised with the reads of one pair next to each other. The vector `order` to give would be `order = c(1, 2, 1, 2)`.

The `keep` and `remove` parameters:

The options `keep` and `remove` allow the user to select the type of alternative splicing events from `KisSplice2RefGenome` that have to be analysed. To work only with intron retention events, the vector should be: `keep = c("IR")`. To work on all events except insertions and deletions, the vector should be `remove = c("insertion","deletion")`. To work specifically on single exon skipping (ES) events (only one exon can be included or excluded), both `keep` and `remove` options must be used. The `keep` option should be set to `c("ES")` and the `remove` option should be set to `c("alt","altA","altD","altAD","MULTI")`.

Value

`kissplice2counts` returns a list of 4 objects:

`countsEvents` a data frame containing several columns: a first column (`events.names`) with the name of the event based on `KisSplice` notation, a second one (`events.length`) containing the length of the event, and the remaining others columns (`counts1` to `countsN`) with the counts corresponding to the replicates of the conditions.

`psiInfo` a data frame containing information to compute the PSI values. This data frame is used only when `counts` is different from 0.

`exonicReadsInfo` a logical indicating if `exonicReads` are used.

k2rgFile — a string containing the KisSplice2RefGenome path and file name. It is equal to NULL if the input file comes from KisSplice.

Only countsEvents is shown when kissplice2counts output is printed.

Examples

```
fpath <- system.file("extdata", "output_kissplice_SNV.fa", package="kissDE")
mySNVcounts <- kissplice2counts(fpath, counts = 0, pairedEnd=TRUE)
names(mySNVcounts)
str(mySNVcounts)
head(mySNVcounts$countsEvents)
```

---

qualityControl *Quality control plots*

Description

Function that provide some helpful plots about the data to ensure that the experimental plan passed to diffExpressedVariants is correct and to control the quality of the data. This function should be run before launching diffExpressedVariants, to validate the data.

Usage

```
qualityControl(countsData, conditions, storeFigs = FALSE, returnPCAdata = FALSE)
```

Arguments

countsData — the output of the kissplice2counts function or a data frame containing the counts in the appropriate format (see Details below).

conditions — a character vector that gives the conditions' order. It has as many elements as there are samples in total.

storeFigs — a logical or a string indicating if the plots should be stored and in which directory. If the qualityControl function is a part of an automatised worflow, we recommand to set this option to TRUE or to a user defined value. If storeFigs is TRUE, the figures will be stored in a kissDEFigures directory which is created in a temporary directory. This directory will be removed when the R session is closed. If storeFigs is a path (a string, e.g. '/path/to/figs'), this directory is created to store the figures. Note that if the directory already exist, it will be overwritten. Plots are stored in .png format. By default (FALSE), the interactive mode is enabled and plots are returned to the graphics device.

returnPCAdata — a logical indicating if the data frame used in the PCA analysis should be returned. By default (FALSE), the data frame is not returned.

Details

countsData input must be formatted as follows: in its first column the names of the events, in its second column the lengths of the events, and in the following columns, the counts corresponding to each replicate of each experimental condition of one variant. Each row corresponds to one variant.

Value

The figures are saved or displayed in the R session.

See Also

diffExpressedVariants

Examples

```
fpath <- system.file("extdata", "output_kissplice_SNV.fa", package="kissDE")
mySNVcounts <- kissplice2counts(fpath, counts = 0, pairedEnd=TRUE)
mySNVconditions <- c("EUR", "EUR", "TSC", "TSC")
qualityControl(mySNVcounts, mySNVconditions)
```

---

writeOutputKissDE *Create and store the output of the* diffExpressedVariants *function in a file and in a rds object.*

Description

If a KisSplice fasta file was used as input for the analysis, writeOutputKissDE will output a tab-delimited file containing one alternative splicing event/SNV per line. The columns are: the ID of the variation, the variable part length, the counts of each variant for each condition, the adjusted p-value (FDR), the deltaPSI and a boolean indicating if the splicing event/SNV was sufficiently expressed (controled by the flagLowCountsConditions option from the diffExpressedVariants function).

If a KisSplice2RefGenome file was used as input for the analysis, this function will add five columns to the KisSplice2RefGenome file, with the following KissDE results: normalized counts, PSI computed from normalized counts, adjusted p-value, deltaPSI and a boolean indicating if the splicing event/SNV was sufficiently expressed in at least half of the conditions (controled by the flagLowCountsConditions option from the diffExpressedVariants function).

In both cases, an rds object is saved in the output folder, that can be used to explore the results of kissDE through a Shiny application with the exploreResults function.

Usage

```
writeOutputKissDE(resDiffExprVariant, output, adjPvalMax = 1, dPSImin = 0,
    writePSI = FALSE)
```

Arguments

resDiffExprVariant	a list, returned by diffExpressedVariants.
output	a character indicating the path and file name to save writeOutputKissDE output.
adjPvalMax	a double indicating the threshold for adjusted p-value. Only SNVs/splicing events with an adjusted p-value lower than this threshold will be kept in the output file.

dPSImin	a double indicating the threshold for the deltaPSI. Only SNVs/splicing events having an absolute value of deltaf/deltaPSI higher than this threshold will be kept in the output file.
writePSI	a boolean indicating if the user wants the f/PSI table to be printed (TRUE) instead of the final table (FALSE, default).

Value

None.

Examples

```
kissplice2refgenome_file <- system.file("extdata",
    "output_k2rg_alt_splicing.txt", package="kissDE")
mySplicingconditions <- c("C1", "C1", "C2", "C2")
counts <- kissplice2counts(fileName=kissplice2refgenome_file, counts=2,
    pairedEnd=TRUE, k2rg=TRUE)
# res <- diffExpressedVariants(countsData=counts,
#   conditions=mySplicingconditions)
# writeOutputKissDE(res, output="results.tsv")
# writeOutputKissDE(res, output="significants_results.tsv",
#   adjPvalMax=0.05, dPSImin=0.1)
# writeOutputKissDE(res, output="psi_results.tsv", adjPvalMax=0.05,
#   dPSImin=0.1, writePSI=TRUE)
```

Index

∗ **package**
kissDE-package, 2

diffExpressedVariants, 3, *4*, *6*, *10*, *11*

exploreResults, 5, *6*, *7*, *11*

kissDE, 6
kissDE-package, 2
kissplice2counts, *4*, *6*, *7*, 8, *9*, *10*

qualityControl, *6*, *7*, *10*, 10

writeOutputKissDE, *5*, *6*, *11*, 11