

Package ‘bambu’

May 28, 2020

Type Package

Title Reference-guided isoform reconstruction and quantification for long read RNA-Seq data

Version 0.1.0

Description Multi-sample transcript discovery and quantification using long read RNA-Seq data.

License GPL-3

Encoding UTF-8

LazyData true

Depends R(>= 3.5.0),
data.table(>= 1.1.8),
dplyr,
SummarizedExperiment(>= 1.1.6),
GenomicRanges,
BiocManager,
GenomicFeatures,
ggplot2

Suggests knitr,
rmarkdown,
fs,
testthat,
ComplexHeatmap,
circlize,
ggbio,
RColorBrewer,
gridExtra

Enhances parallel

SystemRequirements

biocViews FeatureExtraction,
GeneExpression,
GeneExpressionWorkflow,
GenomeAnnotation,
ImmunoOncology,
Normalization,

RNASeq,
Regression,
Sequencing,
Software,
Transcription,
Transcriptomics

bugReports <https://github.com/GoekeLab/bambu/issues>

URL <https://github.com/GoekeLab/bambu>

RoxygenNote 7.1.0

LinkingTo Rcpp,
RcppArmadillo,
RcppProgress

Imports S4Vectors(>= 0.22.1),
IRanges,
GenomicAlignments,
glmnet,
Rsamtools,
Rcpp,
RcppArmadillo,
RcppProgress,
progress

VignetteBuilder knitr,
rmarkdown

R topics documented:

bambu	2
plot.bambu	4
prepareAnnotations	5
prepareAnnotationsFromGTF	6
SEtoGTF	7
transcriptToGeneExpression	7

Index	8
--------------	----------

bambu	<i>long read isoform reconstruction and quantification</i>
-------	--

Description

This function takes bam file of genomic alignments and performs isoform recontruction and gene and transcript expression quantification. It also allows saving of read class files of alignments, extending provided annotations, and quantification based on extended annotations. When multiple samples are provided, extended annotations will be combined across samples to allow comparison.

Usage

```

bambu(
  reads = NULL,
  readClass.file = NULL,
  readClass.outputDir = NULL,
  annotations = NULL,
  genomeSequence = NULL,
  stranded = FALSE,
  ncore = 1,
  yieldSize = NULL,
  isoreParameters = NULL,
  emParameters = NULL,
  extendAnnotations = TRUE,
  verbose = FALSE
)

```

Arguments

reads A string or a vector of strings specifying the paths of bam files for genomic alignments, or a [BamFile](#) object or a [BamFileList](#) object (see [Rsamtools](#)).

readClass.file A string or a vector of strings specifying the read class files that are saved during previous run of [bambu](#).

readClass.outputDir A string variable specifying the path to where read class files will be saved.

annotations A [TxDb](#) object or A [GRangesList](#) object obtained by [prepareAnnotations](#) or [prepareAnnotationsFromGTF](#).

genomeSequence A fasta file or a [BSGenome](#) object.

ncore specifying number of cores used when parallel processing is used, defaults to 1.

yieldSize see [Rsamtools](#).

isoreParameters

A list of controlling parameters for isoform reconstruction process:

- **prefix** specifying prefix for new gene Ids (`genePrefix.number`), defaults to empty
- **remove.subsetTx** indicating whether filter to remove read classes which are a subset of known transcripts(), defaults to TRUE
- **min.readCount** specifying minimum read count to consider a read class valid in a sample, defaults to 2
- **min.readFractionByGene** specifying minimum relative read count per gene, highly expressed genes will have many high read count low relative abundance transcripts that can be filtered, defaults to 0.05
- **min.sampleNumber** specifying minimum sample number with minimum read count, defaults to 1
- **min.exonDistance** specifying minum distance to known transcript to be considered valid as new, defaults to 35
- **min.exonOverlap** specifying minimum number of bases shared with annotation to be assigned to the same gene id, defaults 10 base pairs

emParameters A list of controlling parameters for quantification algorithm estimation process:

- **maxiter** specifying maximum number of run iterations, defaults to 10000.
- **bias** specifying whether to correct for bias, defaults to FALSE.
- **conv** specifying the convergence threshold control, defaults to 0.0001.

extendAnnotations A logical variable indicating whether annotations are to be extended for quantification.

verbose A logical variable indicating whether processing messages will be printed.

Details

Main function

Value

A list of two SummarizedExperiment object for transcript expression and gene expression.

Examples

```
## Not run:
## =====
## More stringent new gene/isoform discovery: new isoforms are identified with at least 5 read count in 1 sample
## Increase EM convergence threshold to 10-6
seOutput <- bambu(reads, annotationGrangesList,
  genomeSequence, isoreParameters = list(min.readCount=5),
  emParameters = list(conv = 10-6))

## End(Not run)
```

plot.bambu

plot.bambu

Usage

```
## S3 method for class 'bambu'
plot(
  se,
  group.variable = NULL,
  type = c("annotation", "pca", "heatmap"),
  gene_id = NULL,
  transcript_id = NULL
)
```

Arguments

`se` An summarized experiment object obtained from [bambu](#) or [transcriptToGene](#).}

`\item{group.variable}`{ Variable for grouping in plot, has be to provided if choos-
ing to plot PCA.}

`\item{type}`{plot type variable, a values of annotation for a single gene with
heatmap for isoform expressions, pca, or heatmap, see [details](#).}

`\item{gene_id}`{specifying the gene_id for plotting gene annotation, either gene_id
or transcript_id has to be provided when type = "annotation".}

`\item{transcript_id}`{specifying the transcript_id for plotting transcript annota-
tion, either gene_id or transcript_id has to be provided when type = "annota-
tion"} } { A heatmap plot for all samples } { plotSEOuptut } { [type](#) indicates
the type of plots to be plotted. There are two types of plots can be chosen, PCA
or heatmap. }

prepareAnnotations	<i>prepare annotations from txdb object</i>
--------------------	---

Description

Function to prepare tables and genomic ranges for transcript reconstruction using a txdb object

Usage

```
prepareAnnotations(txdb)
```

Arguments

`txdb` a [TxDb](#) object

Value

A [GrangesList](#) object

Examples

```
## Not run:
library(TxDb.Hsapiens.UCSC.hg38.knownGene)
txdb <- TxDb.Hsapiens.UCSC.hg38.knownGene
prepareAnnotations(txdb)

## End(Not run)
```

```
prepareAnnotationsFromGTF
```

prepare annotations from gtf file

Description

Prepare annotations from gtf

Usage

```
prepareAnnotationsFromGTF(  
  gtf.file,  
  dataSource = NA,  
  organism = "Homo sapiens",  
  taxonomyId = NA,  
  chrominfo = NULL,  
  miRBaseBuild = NA,  
  metadata = NULL,  
  dbxrefTag,  
  ...  
)
```

Arguments

<code>gtf.file</code>	A string variable indicates the path to a gtf file.
<code>dataSource</code>	as described in makeTxDbFromGFF .
<code>organism</code>	as described in makeTxDbFromGFF .
<code>taxonomyId</code>	as described in makeTxDbFromGFF .
<code>chrominfo</code>	as described in makeTxDbFromGFF .
<code>miRBaseBuild</code>	as described in makeTxDbFromGFF .
<code>metadata</code>	as described in makeTxDbFromGFF .
<code>dbxrefTag</code>	as described in makeTxDbFromGFF .
<code>...</code>	see makeTxDbFromGFF .

Value

A [GrangesList](#) object

SEtoGTF	<i>transcript to gene expression</i>
---------	--------------------------------------

Description

Outputs a GTF dataframe for the nanorna-bam nextflow pipeline

Usage

SEtoGTF(se)

Arguments

se a summarizedExperiment object from [bamby](#)

Value

gtf a GTF dataframe

transcriptToGeneExpression	<i>transcript to gene expression</i>
----------------------------	--------------------------------------

Description

Reduce transcript expression to gene expression

Usage

transcriptToGeneExpression(se)

Arguments

se a summarizedExperiment object from [bamby](#)

Index

bambu, [2](#), [3](#), [5](#), [7](#)

BamFile, [3](#)

BamFileList, [3](#)

details, [5](#)

GrangesList, [5](#), [6](#)

makeTxDbFromGFF, [6](#)

plot.bambu, [4](#)

prepareAnnotations, [3](#), [5](#)

prepareAnnotationsFromGTF, [3](#), [6](#)

Rsamtools, [3](#)

SEtoGTF, [7](#)

transcriptToGene, [5](#)

transcriptToGeneExpression, [7](#)

TxDb, [3](#), [5](#)

type, [5](#)