Package 'Guitar'

November 5, 2026

Type Package

Title Guitar

Version 2.28.0

Date 2019-04-26

Author Xiao Du, Hui Liu, Lin Zhang, Jia Meng

Maintainer Jia Meng <jia.meng@xjtlu.edu.cn>

Description The package is designed for visualization
of RNA-related genomic features with respect to the landmarks of RNA transcripts, i.e.,
transcription starting site, start codon, stop codon and transcription ending site.

License GPL-2

Depends GenomicFeatures, rtracklayer,AnnotationDbi, GenomicRanges,
magrittr, ggplot2, methods, stats,utils ,knitr,dplyr

import GenomicFeatures, rtracklayer,AnnotationDbi, GenomicRanges,
magrittr, ggplot2, methods, stats,utils,dplyr

biocViews Sequencing, SplicedAlignment, Alignment, DataImport, RNASeq,
MethylSeq, QualityControl, Transcription

NeedsCompilation no

Encoding UTF-8

RoxygenNote 6.1.1

VignetteBuilder knitr

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

git_branch RELEASE_3_23

git_last_commit 28a7533

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-04

Contents

GRangesListmapToTranscripts 2
GuitarPlot 3
makeGuitarTxdb 6
normalize 8
samplePoints 9

Index **11**

`GRangesListmapToTranscripts` *GRangesListmapToTranscripts*

Description

Map the input data to the transcript and select the data with a length equal to the length of the site when entering the data.

Usage

```
GRangesListmapToTranscripts(site,
                            mapFilterTranscript = FALSE,
                            transcripts)
```

Arguments

`site`	A GRangeslist object, the content of the sites information.
`mapFilterTranscript`	Whether to filter the length of transcripts equal the original site. Default: FALSE.
`transcripts`	A type of transcripts from the Guitartxdb.

Value

A GRangeslist object, the content is the site data filtered by the mapping.

Examples

```
# read transcript information and generate guitartxdb.
txdb_file <- system.file("extdata", "mm10_toy.sqlite", package="Guitar")
txdb <- loadDb(txdb_file)
guitarTxdb <- makeGuitarTxdb(txdb)

#read the gene feature file.
stBedFile <- system.file("extdata", "m6A_mm10_exomePeak_1000peaks_bed12.bed", package="Guitar")
site <-  blocks(import(stBedFile))
sitesGRanges <- GRangesListmapToTranscripts(site, mapFilterTranscript = FALSE,transcripts = guitarTxdb$tx$tx)
```

GuitarPlot *GuitarPlot*

Description

Plot the transcriptomic distribution of genomic features

Usage

```
GuitarPlot(txGTF = NULL,
            txGFF = NULL,
            txGenomeVer = NULL,
            txTxdb = NULL,
            txGuitarTxdb = NULL,
            txGuitarTxdbSaveFile = NA,
            stBedFiles = NULL,
            stGRangeLists = NULL,
            stGroupName = NULL,
            stAmblguity = 5,
            stSampleNum = 10,
            stSampleModle = "Equidistance",
            #stSampleModle = "random",
            txfiveutrMinLength = 100,
            txcdsMinLength = 100,
            txthreeutrMinLength = 100,
            txlongNcrnaMinLength = 100,
            txlncrnaOverlapmrna = FALSE,
            txpromoterLength = 1000,
            txtailLength = 1000,
            txAmblguity = 5,
            txPrimaryOnly = FALSE,
            txTxComponentProp = NULL,
            txMrnaComponentProp = NULL,
            txLncrnaComponentProp = NULL,
            mapFilterTranscript = TRUE,
            headOrtail = TRUE,
            enableCI = TRUE,
            pltTxType =  c("tx","mrna","ncrna"),
            overlapIndex = 1,
            siteLengthIndex = 1,
            adjust = 1,
            CI_ResamplingTime = 1000,
            CI_interval = c(0.025,0.975),
            miscOutFilePrefix = NA)
```

Arguments

txGTF	GTF file as a source of transcripts.
txGFF	txGFF file as a source of transcripts.
txGenomeVer	Allows direct provision of genomic assembly numbers, such as "hg19", which is automatically downloaded from makeTxDbFromUCSC.
txTxdb	txTxdb file as a source of transcripts.
txGuitarTxdb	The processed GuitarCoordinate is used as the source of the transcriptome.
txGuitarTxdbSaveFile	Return the generated GuitarTxdb to a file.
stBedFiles	BED file as a source of Site.
stGRangeLists	GRange data structure as a source of Site.
stGroupName	Group names of Sites.
stAmblguity	Maximum overlap between sites. Default: 5.
stSampleNum	The number of bases sampled at each Site. Default: 3.
stSampleModle	sampling "Equidistance",sampling "random". Default:"Equidistance".
txfiveutrMinLength	5'UTR length. Default: 100.
txcdsMinLength	CDS length. Default: 100.
txthreeutrMinLength	3'UTR length. Default: 100.
txlongNcrnaMinLength	lncrna length. Default: 100.
txlncrnaOverlapmrna	Whether to allow lncRNA to overlap with mRNA. Default: FALSE.
txpromoterLength	promoter length. Default: 1000.
txtailLength	tail length. Default: 1000.
txAmblguity	Maximum overlap between Tx. Default: 5.
txPrimaryOnly	Whether to use only the main Tx. Default: FALSE.
txTxComponentProp	If it is "NULL", the proportion of the promoter/tx/tail of TX is automatically calculated according to the transcriptome, otherwise, the user specifies the proportion of each part.
txMrnaComponentProp	If it is "NULL", the proportion of promoter/5'UTR/CDS/3'UTR/tail of mrna is automatically calculated according to the transcriptome. Otherwise, the user specifies the proportion of each part.
txLncrnaComponentProp	If it is "NULL", the proportion of promoter/tx/tail of lncRNA is automatically calculated according to the transcriptome, otherwise the user specifies the proportion of each part.

`mapFilterTranscript`
: Whether to filter the length of transcripts equal the original site. Default: TRUE.

`headOrtail`
: Whether to retain promoter and tail. Default: TRUE.

`enableCI`
: Whether to add a CI curve. Default: TRUE.

`pltTxType`
: Which transcript is to be drawn on. If there is no such transcript in the genome, it cannot be drawn even if specified. Default: c("tx","mrna","ncrna").

`overlapIndex`
: Index of site overlapping times. Default: 1.

`siteLengthIndex`
: Index of site length. Default: 1.

`CI_ResamplingTime`
: Resampling times in density drawing mode. Default: 1000.

`CI_interval`
: Upper and lower limits of the confidence interval. Default : c(0.025,0.975).

`adjust`
: Curve's smooth level. Default: 1.

`miscOutFilePrefix`
: Save as a PDF name prefix, If the prefix is not specified, we will default the PDF name to "Guitar-type-test.PDF".

Details

This function plots the transcriptomic distribution of genomic features. It is designed for a fast usage of the Guitar package, When you need to specify a parameter, you need to define the function when the package is called.

Value

A figure showing the transcriptomic distribution of the genomic features will be generated. Post-editing with Adobe Illustrator or other graphic software is recommended.

Author(s)

Xiao Du <xiao.du@cumt.edu.cn>

Examples

```
# read transcript information
txdb_file <- system.file("extdata", "mm10_toy.sqlite", package="Guitar")
txdb <- loadDb(txdb_file)
# read genomic features
stGRangelist<-list()
stBedFiles <- list(system.file("extdata", "m6A_mm10_exomePeak_1000peaks_bed12.bed", package="Guitar"))
for (i in 1:length(stBedFiles)) {
  stGRangelist[[i]] <-  blocks(import(stBedFiles[[i]]))
}
#plot
GuitarPlot(txTxdb = txdb,
               stGRangeLists = stGRangelist,
               stGroupName = c("Group1"))
```

makeGuitarTxdb *makeGuitarTxdb*

Description

Make a Guitar Coordinates from TranscriptDb object, i.e., making Guitar coordinates for 3 different type, including, tx, mRNA, lncRNA, tx include three component, Pomoter,RNA, Tail, mRNA include five component, Pomoter, 5'UTR, CDS, 3'UTR, Tail, lncRNA include three component, Pomoter, ncRNA, Tail. Additional filters will discard transcripts that are too short or has too much ambigous on Genome to increase the sensitivity of the analysis.

Usage

```
makeGuitarTxdb(txdb,
               txfiveutrMinLength = 100,
               txcdsMinLength = 100,
               txthreeutrMinLength = 100,
               txlongNcrnaMinLength = 100,
               txlncrnaOverlapmrna = FALSE,
               txpromoterLength = 1000,
               txtailLength = 1000,
               txAmblguity = 5,
               txTxComponentProp = NULL,
               txMrnaComponentProp = NULL,
               txLncrnaComponentProp = NULL,
               txPrimaryOnly = FALSE,
               pltTxType = c("tx","mrna","ncrna"),
               withTxContext = TRUE
               )
```

Arguments

txdb	A transcriptDb object, which can be generated from makeTxDbFromUCSC or other functions.
txfiveutrMinLength	5'UTR simulation length. Unfortunately, the 5'UTR length of some mRNAs does not provide an effective resolution for analysis. These mRNAs will be filtered out of the analysis. Default: 100.
txcdsMinLength	CDS simulation length. Unfortunately, the CDS length of some mRNAs does not provide an effective resolution for analysis. These mRNAs will be filtered out of the analysis. Default: 100.
txthreeutrMinLength	3'UTR simulation length. Unfortunately, the 3'UTR length of some mRNAs does not provide an effective resolution for analysis. These mRNAs will be filtered out of the analysis. Default: 100.

`txlongNcrnaMinLength`
: non-coding RNAs with length smaller than this value will not be used in the analysis.

`txlncrnaOverlapmrna`
: Whether to allow lncRNA to overlap with mRNA. Default: FALSE.

`txpromoterLength`
: promoter simulator length. Default: 1000.

`txtailLength`
: tail simulator length. Default: 1000.

`txAmblguity`
: If a transcript overlap with more number of transcripts than this number, this transcript will be used in the analysis. By filtering out a number of transcripts, this filter also decrease memory usage and computation time. Default: 5.

`txTxComponentProp`
: If it is "NULL", the proportion of the promoter/tx/tail of TX is automatically calculated according to the transcriptome, otherwise, the user specifies the proportion of each part.

`txMrnaComponentProp`
: If it is "NULL", the proportion of promoter/5'UTR/CDS/3'UTR/tail of mrna is automatically calculated according to the transcriptome. Otherwise, the user specifies the proportion of each part.

`txLncrnaComponentProp`
: If it is "NULL", the proportion of promoter/tx/tail of lncRNA is automatically calculated according to the transcriptome, otherwise the user specifies the proportion of each part.

`txPrimaryOnly`
: Whether to use only the main Tx. Default: TRUE.

`pltTxType`
: Which transcript is to be drawn on. If there is no such transcript in the genome, it cannot be drawn even if specified. Default: c("tx","mrna","ncrna").

`withTxContext`
: Whether to add an extended area to the transcript.

Value

A Guitar coordiantes (GRanges object) will be returned, with 3 different type, with Transcript ID, the relative position of each GRanges on the RNA transcript, the interval (bp) between different coordiantes on a transcript component.

Author(s)

Xiao Du <xiao.du@cumt.edu.cn>

Examples

```
# read transcript information
txdb_file <- system.file("extdata", "mm10_toy.sqlite", package="Guitar")
txdb <- loadDb(txdb_file)
guitarTxdb <- makeGuitarTxdb(txdb)
```

normalize *normalize*

Description

This function Comparing the sampled site with the normalized guitar coordinate system to find the relative position of each site in the zero to one coordinate system, and calculating the weight of each site at this position.

Usage

```
normalize(sitesGRanges,
          guitarTxdb,
          txType,
          overlapIndex,
          siteLengthIndex)
```

Arguments

sitesGRanges	A kind of GRange object, generated by sampling the site.
guitarTxdb	Sites mapping on guitartxdb and finding the corresponding location on the transcript.
txType	Transcript type comparing with guitar coordinate's transcript type.
overlapIndex	Index of site overlapping times. Default: 1.
siteLengthIndex	Index of site length. Default: 1.

Value

A data list object contains two types of data information, one is the location information of the site, and the other is the weight information of the site.

Author(s)

Xiao Du<xiao.du@cumt.edu.cn>

Examples

```
# read genomic features
stBedFile <- system.file("extdata", "m6A_mm10_exomePeak_1000peaks_bed12.bed", package="Guitar")
site <-  blocks(import(stBedFile))
# read transcript information
txdb_file <- system.file("extdata", "mm10_toy.sqlite", package="Guitar")
txdb <- loadDb(txdb_file)
guitarTxdb <- makeGuitarTxdb(txdb)
sitesGRanges <- samplePoints(list(site),
                             stSampleNum = 5,
```

```
                              stAmblguity = 5,
                              pltTxType = c("mrna"),
                              stSampleModle = "Equidistance",
                              mapFilterTranscript = FALSE,
                              guitarTxdb)
sitesNormlize <- normalize(sitesGRanges,
                           guitarTxdb,
                           txType = "mrna",
                           overlapIndex = 1,
                           siteLengthIndex = 1)
```

`samplePoints` *samplePoints*

Description

This function calculates the length of each sites and then samples the points from sites by using two sampling methods based on the number of sampling points. One sampling method is equidistant sampling, the N sampling points are sampled at equal intervals according to the length of the site. Another method is random sampling, and N sampling points are randomly distributed on the site.

Usage

```
samplePoints(sitesGrangelists,
             stSampleNum = 5,
             stAmblguity = 5,
             pltTxType = c("tx","mrna","ncrna"),
             stSampleModle = "Equidistance",
             mapFilterTranscript = FALSE,
             guitarTxdb)
```

Arguments

`sitesGrangelists`	A GRangeslist object, the content of the sites information.
`stSampleNum`	The number of bases sampled at each Site. Default: 3.
`stAmblguity`	Maximum overlap between sites. Default: 5.
`pltTxType`	Transcript type mapped to guitartxdb.
`stSampleModle`	sampling "Equidistance",sampling "random". Default:"Equidistance".
`mapFilterTranscript`	Whether to filter the length of transcripts equal the original site. Default: FALSE.
`guitarTxdb`	Sites mapping on guitartxdb and finding the corresponding location on the transcript.

Value

A GRange data list, the content of the list is the sampling point, the width is 1, the total number of data is N times the input data.

Author(s)

Xiao Du<xiao.du@cumt.edu.cn>

Examples

```
# read genomic features
stBedFile <- system.file("extdata", "m6A_mm10_exomePeak_1000peaks_bed12.bed", package="Guitar")
site <-  blocks(import(stBedFile))
#generate GuitarTxdb
txdb_file <- system.file("extdata", "mm10_toy.sqlite", package="Guitar")
txdb <- loadDb(txdb_file)
guitarTxdb <- makeGuitarTxdb(txdb)
# sample site points
sitesPoints <- samplePoints(list(site),
                            stSampleNum = 5,
                            stAmblguity = 5,
                            pltTxType = c("tx","mrna","ncrna"),
                            stSampleModle = "Equidistance",
                            mapFilterTranscript = FALSE,
                            guitarTxdb)
```

Index

GRangesListmapToTranscripts, 2
GuitarPlot, 3

makeGuitarTxdb, 6

normalize, 8

samplePoints, 9