

Package ‘RNAModR’

May 13, 2016

Type Package

Title Functional Analysis of RNA Modifications

Version 0.1.0

Author Maurits Evers <maurits.evers@anu.edu.au>

Maintainer Maurits Evers <maurits.evers@anu.edu.au>

Description Transcriptome-wide analysis of RNA modifications.

Imports AnnotationDbi, beanplot, Biostrings, GenomeInfoDb, gplots,
methods, RSQLite, rtracklayer

Depends GenomicRanges, GenomicFeatures, IRanges

Suggests knitr, testthat

License GPL-3 | file LICENSE

LazyData false

RoxygenNote 5.0.1.9000

VignetteBuilder knitr

R topics documented:

BuildTx	2
GenerateNull	3
GetEEJunct	4
GetMotifLoc	5
GetNumberOfLoci	5
Info	6
PlotGC	7
PlotOverlap	8
PlotRelDistDistribution	8
PlotRelDistEnrichment	9
PlotSectionDistribution	9
PlotSectionEnrichment	10
PlotSeqLogo	11
PlotSpatialDistribution	12
PlotSpatialEnrichment	13

ReadBED	14
SmartMap	14
txLoc-class	15
WriteTxLocToBED	16
Index	17

BuildTx	<i>Build a custom transcriptome.</i>
---------	--------------------------------------

Description

Build a custom transcriptome. See 'Details'.

Usage

```
BuildTx(genomeVersion = c("hg38", "hg19", "mm10", "mm9"), force = FALSE,
        sanityCheck = FALSE)
```

Arguments

- genomeVersion A character string; refers to a specific reference genome assembly version; default is "hg38".
- force A logical scalar; if TRUE force rebuild of transcriptome; this will overwrite existing data.
- sanityCheck A logical scalar; if TRUE perform sanity checks.

Details

The function builds a custom transcriptome that consists of one transcript per unique Entrez ID; the transcript is determined from all RefSeq-based isoforms as the transcript with the longest CDS, and adjoining upstream/downstream longest UTRs. Transcript segments are stored per transcript section, and written to a .RData file.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

GenerateNull	<i>Generate a null distribution of transcript sites.</i>
--------------	--

Description

Generate a null distribution of transcript sites. See 'Details'.

Usage

```
GenerateNull(locus, id = NULL, method = c("ntAbund", "perm"), nt = "C",
  showPb = TRUE)
```

Arguments

locus	A txLoc object.
id	A character string; identifier for null sites; if NULL then id = "null"; default is NULL.
method	A character string; specifies the method used to generate null distribution; if method == "ntAbund" the position of all nucleotides specified by nt will be used as null sites; if method == "perm" then null sites will be generated by uniform-randomly shuffling candidate positions from locus within the corresponding transcript region; default is "ntAbund".
nt	A single character; if method == "ntAbund", use nt to derive distribution of null sites.
showPb	A logical scalar; if TRUE show a progress bar; default is TRUE.

Details

The function generates a null distribution of single-nucleotide sites in transcript sections. Two different methods can be employed:

1. If method == "ntAbund" sites are generated based on the position of all nucleotides nt in those transcript sections that also contain a site in locus.
2. If method == "perm" sites are generated based on sites in locus by permuting the position of those sites uniformly across the associated transcript section.

It is import to emphasise that any downstream enrichment analysis may depend critically on the choice of the null distribution. For example, a position permutation-based null distribution may not be a valid null distribution, if the position distribution is highly non-uniform across a transcript section. This is the case e.g. for the spatial distribution of cytosines across the 5'UTR, CDS and/or 3'UTR. In this case, a better null distribution would be to consider all cytosines in transcript regions that contain also a site in locus. The return object is a txLoc of transcript loci in different transcript sections.

Value

A txLoc object. See 'Details'.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

Examples

```
## Not run:
bedFile <- system.file("extdata",
                       "miCLIP_m6A_Linder2015_hg38.bed",
                       package = "RNAModR");
sites <- ReadBED(bedFile);
posSites <- SmartMap(sites, id = "m6A", refGenome = "hg38");
negSites <- GenerateNull(posSites, method = "ntAbund", nt = "A");

## End(Not run)
```

GetEEJunct

Get exon-exon junctions from transcriptome.

Description

Get exon-exon junctions from transcriptome. See 'Details'.

Usage

```
GetEEJunct(refGenome = "hg38", filter = "CDS")
```

Arguments

refGenome	A character string; specifies a specific reference genome assembly version based on which the matching transcriptome is loaded; default is "hg38".
filter	A character vector; only consider transcript sections specified in filter; filter = NULL corresponds to filter = c("5'UTR", "CDS", "3'UTR").

Details

This function extracts exon-exon junction positions from a reference transcriptome, and returns a txLoc object.

Value

A txLoc object. See 'Details'.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

GetMotifLoc	<i>Get loci of motif(s) from transcriptome.</i>
-------------	---

Description

Get loci of motif(s) from transcriptome.

Usage

```
GetMotifLoc(motif, refGenome = "hg38", maxMM = 0, filter = NULL,
            showPb = TRUE)
```

Arguments

motif	A character vector; specifies the motif(s) that will be matched against the transcriptome.
refGenome	A character string; specifies the reference genome version; default is "hg38".
maxMM	An integer scalar; specifies the maximum number of mismatches that are allowed during the motif matching; default is 0.
filter	A character vector; only consider transcript sections specified in filter; filter = NULL corresponds to filter = c("5'UTR", "CDS", "3'UTR").
showPb	A logical scalar; if TRUE show a progress bar; default is TRUE.

Details

This function returns a txLoc object of loci of the specified motif(s) within transcript sections. Motif sequence matching allows for mismatches through the parameter maxMM.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

GetNumberOfLoci	<i>Method "GetNumberOfLoci" for S4 object txLoc.</i>
-----------------	--

Description

Get the number of loci of a txLoc object in every transcript section.

Usage

```
GetNumberOfLoci(x)
```

```
## S4 method for signature 'txLoc'
GetNumberOfLoci(x)
```

Arguments

x A txLoc object.

Value

A named integer vector with the number of sites per transcript section.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

Info

Method "Info" for S4 object txLoc.

Description

Print general information of a txLoc object.

Usage

Info(x)

```
## S4 method for signature 'txLoc'
Info(x)
```

Arguments

x A txLoc object.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

Examples

```
## Not run:
bedFile <- system.file("extdata",
                       "miCLIP_m6A_Linder2015_hg38.bed",
                       package = "RNAModR");
sites <- ReadBED(bedFile);
posSites <- SmartMap(sites, id = "m6A", refGenome = "hg38");
Info(posSites);

## End(Not run)
```

PlotGC	<i>Plot GC content.</i>
--------	-------------------------

Description

Plot GC content.

Usage

```
PlotGC(locPos, locNeg, flank = 10, filter = NULL, geneNorm = FALSE,  
       subsample = TRUE)
```

Arguments

locPos	A txLoc object.
locNeg	A txLoc object.
flank	An integer scalar; see 'Details'.
filter	A logical scalar; only consider loci in transcript regions specified in filter; Default is NULL.
geneNorm	A logical scalar; if TRUE normalise GC content in window to GC content of transcript section; default is FALSE.
subsample	A logical scalar; if TRUE a subsample of locNeg is used instead of the full set; the subsample size is dynamically determined based on the total number of sites in locPos; default is TRUE.

Details

The function calculates GC contents within a region around every site from two txLoc objects. The window is defined by extending the position of every transcript locus upstream and downstream by `flank` nucleotides (if possible). The means of resulting GC content distributions are compared and assessed using a two-tailed t-test.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

PlotOverlap	<i>Plot overlap of two txLoc objects.</i>
-------------	---

Description

Plot overlap of two txLoc objects.

Usage

```
PlotOverlap(loc1, loc2)
```

Arguments

loc1	A txLoc object.
loc2	A txLoc object.

Details

This function plots the overlap of entries from two txLoc objects as Venn diagrams.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

PlotRelDistDistribution	<i>Plot distribution of relative distances between sites from two txLoc objects.</i>
-------------------------	--

Description

Plot distribution of relative distances between sites from two txLoc objects.

Usage

```
PlotRelDistDistribution(loc1, loc2, flank = 1000, binWidth = 20,  
  doBootstrap = TRUE)
```

Arguments

loc1	A txLoc object.
loc2	A txLoc object.
flank	An integer scalar; specifies the absolute maximum relative distance used as a cutoff; default is 1000.
binWidth	An integer scalar; specifies the spatial width by which distances will be binned; default is 20.
doBootstrap	A logical scalar; if YES calculate 95 region; default is TRUE.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

PlotRelDistEnrichment *Perform and plot enrichment analysis of relative distances of "positive" and "negative" control sites relative to "reference" sites.*

Description

Perform and plot enrichment analysis of relative distances of "positive" and "negative" control sites relative to "reference" sites.

Usage

```
PlotRelDistEnrichment(locPos, locNeg, locRef, flank = 1000, binWidth = 20)
```

Arguments

locPos	A txLoc object.
locNeg	A txLoc object.
locRef	A txLoc object.
flank	An integer scalar; specifies the absolute maximum relative distance used as a cutoff; default is 1000.
binWidth	An integer scalar; specifies the spatial width by which distances will be binned; default is 20.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

PlotSectionDistribution
Plot piechart of the number of loci in every transcript section.

Description

Plot piechart of the number of loci in every transcript section.

Usage

```
PlotSectionDistribution(locus, filter = NULL)
```

Arguments

locus	A txLoc object.
filter	A character vector; only consider transcript sections specified in filter; if NULL consider all sections.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

Examples

```
## Not run:
bedFile <- system.file("extdata",
                       "miCLIP_m6A_Linder2015_hg38.bed",
                       package = "RNAModR");
sites <- ReadBED(bedFile);
posSites <- SmartMap(sites, id = "m6A", refGenome = "hg38");
PlotSectionDistribution(posSites);

## End(Not run)
```

PlotSectionEnrichment *Perform transcript section enrichment analysis and plot results.*

Description

Perform transcript section enrichment analysis and plot results. Enrichment/depletion is evaluated using (multiple) Fisher's exact test(s). Multiple hypothesis testing correction is applied following the method of Benjamini and Hochberg.

Usage

```
PlotSectionEnrichment(locPos, locNeg, filter = NULL,
                      withExtendedAxisLabel = 2)
```

Arguments

locPos	A txLoc object. These should be the positive control sites.
locNeg	A txLoc object. These should be the negative control sites.
filter	Only plot loci in transcript regions specified in filter. Default is NULL.
withExtendedAxisLabel	Plot extended axis labels. Default is 2. See ??? for details.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

Examples

```
## Not run:
bedFile <- system.file("extdata",
                       "miCLIP_m6A_Linder2015_hg38.bed",
                       package = "RNAModR");
sites <- ReadBED(bedFile);
posSites <- SmartMap(sites, id = "m6A", refGenome = "hg38");
negSites <- GenerateNull(posSites, method = "nuclAbundance");
PlotSectionEnrichment(posSites, negSites,
                      filter = c("5'UTR", "CDS", "3'UTR"));

## End(Not run)
```

PlotSeqLogo

Plot sequence logo.

Description

Plot sequence logo.

Usage

```
PlotSeqLogo(locus, flank = 5, filter = NULL, ylim = c(0, 2))
```

Arguments

locus	A txLoc object.
flank	An integer scalar; see 'Details'.
filter	A character vector; only plot sequence logos of sections specified in filter; if NULL plot all sections; default is NULL.
ylim	An integer vector; specifies limits for the y-axis; automatically determined if ymin = NULL; default is c(0, 2).

Details

The function determines the sequence logo within a window defined by extending sites from locus upstream and downstream by flank nucleotides. By default logos are shown for every transcript section from locus. Use filter to specify specific transcript sections.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

PlotSpatialDistribution

Plot spatial distribution of loci from txLoc object.

Description

Plot spatial distribution of loci from txLoc object within every transcript section.

Usage

```
PlotSpatialDistribution(locus, filter = NULL, nbreaks = 100,
  absolute = FALSE, binWidth = NULL, posMax = 1000, doBootstrap = TRUE,
  ...)
```

Arguments

locus	A txLoc object.
filter	Only plot loci in transcript regions specified in filter.
nbreaks	Number of spatial bins. Default is 100.
absolute	Plot spatial distribution in absolute coordinates. Default is FALSE.
binWidth	Spatial bin width. Overrides nbreaks if not NULL.
posMax	If absolute == TRUE, show spatial distribution within a window given by posMax from the 5'/3' position of the transcript feature. Default is 1000 nt.
doBootstrap	Calculate 95 sites within transcript region. Default is TRUE.
...	Additional parameters passed to plot.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

Examples

```
## Not run:
bedFile <- system.file("extdata",
  "miCLIP_m6A_Linder2015_hg38.bed",
  package = "RNAModR");
sites <- ReadBED(bedFile);
posSites <- SmartMap(sites, id = "m6A", refGenome = "hg38");
PlotSpatialDistribution(posSites);
PlotSpatialDistribution(posSites,
  absolute = TRUE,
  filter = c("5'UTR", "CDS", "3'UTR"),
  ylim = c(0, 200));

## End(Not run)
```

PlotSpatialEnrichment *Perform spatial enrichment analysis and plot results.*

Description

Perform spatial enrichment analysis and plot results. Enrichment/depletion is evaluated using (multiple) Fisher's exact test(s). Multiple hypothesis testing correction is applied following the method of Benjamini and Hochberg.

Usage

```
PlotSpatialEnrichment(locPos, locNeg, filter = NULL, binWidth = 20,  
  posMax = 1000)
```

Arguments

locPos	A txLoc object. These should be the positive control sites.
locNeg	A txLoc object. These should be the negative control sites.
filter	Only plot loci in transcript regions specified in filter.
binWidth	Spatial bin width. Default is 20 nt.
posMax	Evaluate enrichment within a window given by posMax. Default is 1000 nt.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

Examples

```
## Not run:  
bedFile <- system.file("extdata",  
  "miCLIP_m6A_Linder2015_hg38.bed",  
  package = "RNAModR");  
sites <- ReadBED(bedFile);  
posSites <- SmartMap(sites, id = "m6A", refGenome = "hg38");  
negSites <- GenerateNull(posSites, method = "permutation");  
PlotSpatialEnrichment(posSites, negSites,  
  filter = c("5'UTR", "CDS", "3'UTR"));  
  
## End(Not run)
```

ReadBED

Read BED-formatted file.

Description

Open and read a BED file, and store the annotation features in a GRanges object.

Usage

```
ReadBED(file, collapseRange = FALSE)
```

Arguments

file	A character string; specifies the input BED file.
collapseRange	A logical scalar; if TRUE loci spanning more than one nucleotide are collapsed to a single nucleotide locus corresponding to the midpoint of the range.

Value

A GRanges object.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

Examples

```
bedFile <- system.file("extdata",
                       "miCLIP_m6A_Linder2015_hg38.bed",
                       package = "RNAModR");
sites <- ReadBED(bedFile);
```

SmartMap

Map genome coordinates to transcript coordinates.

Description

Map genome coordinates to transcript coordinates.

Usage

```
SmartMap(locus, id = NULL, refGenome = "hg38", ignore.strand = FALSE)
```

Arguments

locus	A GRanges object; specifies the list of genomic features to be mapped.
id	A character string; specifies a name for loci from locus; if NULL then id = ""; default is NULL.
refGenome	A character string; specifies a specific reference genome assembly version based on which the matching transcriptome is loaded; default is "hg38".
ignore.strand	A logical scalar; if TRUE strand information is ignored when mapping genome coordinates to transcript coordinates; default is FALSE.

Details

The function maps genomic coordinates from locus to transcript section coordinates. The function automatically loads a reference transcriptome based on refGenome. An error is thrown if a reference transcriptome could not be found. This usually means that BuildTx was not yet run successfully. The function returns a txLoc object of mapped positions.

Value

A txLoc object. See 'Details'.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

Examples

```
## Not run:
bedFile <- system.file("extdata",
                       "miCLIP_m6A_Linder2015_hg38.bed",
                       package = "RNASModR");
sites <- ReadBED(bedFile);
txSites <- SmartMap(sites, id = "m6A", refGenome = "hg38");

## End(Not run)
```

txLoc-class	<i>txLoc object. The S4 object stores information about mapped loci per transcript section. Meta-data such as an identifier and the reference genome are stored in separate slots.</i>
-------------	--

Description

txLoc object. The S4 object stores information about mapped loci per transcript section. Meta-data such as an identifier and the reference genome are stored in separate slots.

Slots

loci A list of dataframe objects; specifies the list of loci per transcript section.
id A character string; identifier for loci in txLoc object.
refGenome A character string; gives the reference genome upon which transcriptome-derived positions are based.
version A character string; can be used to flag specific version, e.g. using the current system time & date.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

WriteTxLocToBED

Write txLoc object to a BED file.

Description

Write txLoc object to a BED file.

Usage

```
WriteTxLocToBED(locus, file = NULL, formatChr = "noChrName")
```

Arguments

locus	A txLoc object.
file	Filename of output BED file. If NULL then file = "sites.bed".
formatChr	Set output format of chromosome column. Default is "noChrName".

Details

This function writes entries from a txLoc object to a BED file. This process is not "splice-aware", i.e. if an entry spans an intron the BED entry gives the left and right-most genomic coordinate of the feature.

Author(s)

Maurits Evers, <maurits.evers@anu.edu.au>

Index

BuildTx, [2](#)

GenerateNull, [3](#)

GetEEJunct, [4](#)

GetMotifLoc, [5](#)

GetNumberOfLoci, [5](#)

GetNumberOfLoci, txLoc-method
(GetNumberOfLoci), [5](#)

Info, [6](#)

Info, txLoc-method (Info), [6](#)

PlotGC, [7](#)

PlotOverlap, [8](#)

PlotRelDistDistribution, [8](#)

PlotRelDistEnrichment, [9](#)

PlotSectionDistribution, [9](#)

PlotSectionEnrichment, [10](#)

PlotSeqLogo, [11](#)

PlotSpatialDistribution, [12](#)

PlotSpatialEnrichment, [13](#)

ReadBED, [14](#)

SmartMap, [14](#)

txLoc-class, [15](#)

WriteTxLocToBED, [16](#)