Package 'InPAS'

October 9, 2015

Type Package

Title Identification of Novel alternative PolyAdenylation Sites (PAS)

Version 1.0.6

Date 2014-09-12

Author Jianhong Ou, Sung Mi Park, Michael R. Green and Lihua Julie Zhu

Maintainer Jianhong Ou <jianhong.ou@umassmed.edu>

Description Alternative polyadenylation (APA) is one of the important post-transcriptional regulation mechanisms which occurs in most human genes. InPAS facilitates the discovery of novel APA sites from RNAseq data. It leverages cleanUpdTSeq to fine tune identified APA sites.

biocViews RNASeq, Sequencing, AlternativeSplicing, Coverage, DifferentialSplicing, GeneRegulation, Transcription

License GPL (>= 2)

Lazyload yes

Imports AnnotationDbi, BSgenome, cleanUpdTSeq, Gviz, seqinr, limma, IRanges, GenomeInfoDb

Depends R (>= 3.1), GenomicRanges, GenomicFeatures, BiocParallel, S4Vectors

Suggests RUnit, BiocGenerics, BiocStyle, BSgenome.Hsapiens.UCSC.hg19, BSgenome.Mmusculus.UCSC.mm10, org.Hs.eg.db, org.Mm.eg.db, TxDb.Hsapiens.UCSC.hg19.knownGene, TxDb.Mmusculus.UCSC.mm10.knownGene, rtracklayer

NeedsCompilation no

R topics documented:

`InPAS-package` *alternative polyadenylation and cleavage estimations*

Description

predict and estimate the alternative polyadenylation and cleavage site for mRNA-seq data

Details

Package:	InPAS
Type:	Package
Version:	1.0
Date:	2014-09-12
License:	GPL (>= 2)

Author(s)

Jianhong Ou, Sung Mi Park, Michael R. Green and Lihua Julie Zhu

Maintainer: Jianhong Ou <jianhong.ou@umassmed.edu>

References

Sheppard S, Lawson N and Zhu L (2013). Accurate identification of polyadenylation sites from 3' end deep sequencing using a naive Bayes classifier. Bioinformatics, 29(20), pp. 2564. ISSN 1460-2059

`coverageFromBedGraph` *read coverage from bedGraph files*

Description

read coverage from bedGraph files and save as a list.

Usage

```
coverageFromBedGraph(bedgraphs, tags, genome, hugeData=FALSE, ...)
```

Arguments

bedgraphs	The file names of bedgraphs generated by bedtools. eg: bedtools genomecov -bg -split -ibam $bam -g mm10.size.txt > $bedgraph
tags	the names for each input bedgraphs
genome	an object of BSgenome
hugeData	is this dataset consume too much memory? if it is TRUE, the coverage will be saved into tempfiles.
...	parameters can be passed into tempfile. This is useful when you submit huge dataset to cluster.

Value

return a list of coverage for each bedgraph files. For each item in the list, it is a list of coverage for each chromosome. And the chromosome must start from "chr".

Author(s)

Jianhong Ou

Examples

```
if(interactive()){
    library(BSgenome.Mmusculus.UCSC.mm10)
    path <- file.path(find.package("InPAS"), "extdata")
    bedgraphs <- file.path(path, "Baf3.extract.bedgraph")
    data(utr3.mm10)
    tags <- "Baf3"
    genome <- BSgenome.Mmusculus.UCSC.mm10
    coverage <-
        coverageFromBedGraph(bedgraphs, tags, genome, hugeData=FALSE)
}
```

CPsites *predict the cleavage and polyadenylation(CP) site*

Description

predict the alternative cleavage and polyadenylation (CP or APA) site.

Usage

```
CPsites(coverage, gp1, gp2=NULL, genome, utr3,
    window_size=100, search_point_START=50, search_point_END=NA,
    cutStart=window_size, cutEnd=0, search_distal_polyA_end=FALSE,
    coverage_threshold=5, long_coverage_threshold=2,
    gcCompensation=NA, mappabilityCompensation=NA,
    FFT=FALSE, fft.sm.power=20,
  PolyA_PWM=NA, classifier=NA, classifier_cutoff=.8, shift_range=window_size,
    BPPARAM=NULL)
```

Arguments

coverage	coverage for each sample, output of coverageFromBedGraph
gp1	tag names involved in group 1
gp2	tag names involved in group 2
genome	an object of BSgenome
utr3	output of utr3Annotation
window_size	window size for noval distal position searching and adjusted polyA searching, default: 100
search_point_START	start point for searching
search_point_END	end point for searching
cutStart	how many nucleotides should be removed from the start before search, 0.1 means 10 percent, 25 means cut first 25.
cutEnd	how many nucleotides should be removed from the end before search, 0.1 means 10 percent.
search_distal_polyA_end	If true, adjust distal polyA end by cleanUpdTSeq
coverage_threshold	cutoff coverage threshold for first 100 nucleotides. If the coverage of first 100 nucleotides is lower than coverage_threshold, that transcript will be dropped.
long_coverage_threshold	cutoff threshold for coverage in the region of long form. If the coverage in the region of long form is less than long_coverage_threshold, that transcript will be dropped.
gcCompensation	GC content compensation vector
mappabilityCompensation	mappability compensation vector
FFT	use Fast Fourier Transform Algorithm to smooth the data or not. default: FALSE
fft.sm.power	if FFT is TRUE, the frequency should be removed
PolyA_PWM	Position Weight Matrix of polyA
classifier	An object of class "PASclassifier"

`classifier_cutoff`
: This is the cutoff used to assign whether a putative pA is true or false. This can be any floating point number between 0 and 1. For example, classifier_cutoff = 0.5 will assign an putative pA site with prob.1 > 0.5 to the True class (1), and any putative pA site with prob.1 <= 0.5 as False (0).

`shift_range`
: the shift range for polyA site searching

`BPPARAM`
: An optional `BiocParallelParam` instance determining the parallel back-end to be used during evaluation, or a list of BiocParallelParam instances, to be applied in sequence for nested calls to bplapply.

Value

return an object of GRanges contain the estimated CP sites.

Author(s)

Jianhong Ou

References

ref: Cheung MS, Down TA, Latorre I, Ahringer J. Systematic bias in high-throughput sequencing data and its correction by BEADS. Nucleic Acids Res. 2011 Aug;39(15):e103. doi: 10.1093/nar/gkr425. Epub 2011 Jun 6. PubMed PMID: 21646344; PubMed Central PMCID: PMC3159482.

mappability could be calculated by [GEM](http://algorithms.cnag.cat/wiki/Man:gem-mappability)

ref: Derrien T, Estelle J, Marco Sola S, Knowles DG, Raineri E, Guigo R, Ribeca P. Fast computation and applications of genome mappability. PLoS One. 2012;7(1):e30377. doi: 10.1371/journal.pone.0030377. Epub 2012 Jan 19. PubMed PMID: 22276185; PubMed Central PMCID: PMC3261895.

Examples

```
if(interactive()){
    library(BSgenome.Mmusculus.UCSC.mm10)
    path <- file.path(find.package("InPAS"), "extdata")
    bedgraphs <- file.path(path, "Baf3.extract.bedgraph")
    data(utr3.mm10)
    tags <- "Baf3"
    genome <- BSgenome.Mmusculus.UCSC.mm10
    coverage <-
        coverageFromBedGraph(bedgraphs, tags, genome, hugeData=FALSE)
    CP <- CPsites(coverage=coverage, gp1=tags, gp2=NULL, genome=genome,
        utr3=utr3.mm10, coverage_threshold=5, long_coverage_threshold=5)
}
```

inPAS *do estimation of alternative polyadenylation and cleavage site in one step*

Description

do estimation of alternative polyadenylation and cleavage site in one step

Usage

```
inPAS(bedgraphs, tags, genome, utr3, gp1, gp2, txdb,
    window_size=100, short_coverage_threshold=NA,
    long_coverage_threshold=NA,
    adjusted.P_val.cutoff=0.05, dPDUI_cutoff=0.3,
    PDUI_logFC_cutoff=0.59,
    search_point_START=50, search_point_END=NA,
    cutStart=100, cutEnd=0, search_distal_polyA_end=FALSE,
    coverage_threshold=5,
    gcCompensation=NA, mappabilityCompensation=NA,
    FFT=FALSE, fft.sm.power=20, hugeData=FALSE,
    PolyA_PWM=NA, classifier=NA, classifier_cutoff=.8, shift_range=0,
    BPPARAM=NULL)
```

Arguments

bedgraphs	The file names of bedgraphs generated by bedtools. eg: bedtools genomecov -bg -split -ibam $bam -g mm10.size.txt > $bedgraph
tags	the names for each input bedgraphs
genome	an object of BSgenome
utr3	output of utr3Annotation
gp1	tag names involved in group 1
gp2	tag names involved in group 2
txdb	an object of TxDb
window_size	window size for noval distal position searching and adjusted polyA searching, default: 100
short_coverage_threshold	cutoff threshold for coverage in thre region of short form
long_coverage_threshold	cutoff threshold for coverage in thre region of long form
adjusted.P_val.cutoff	cutoff value for adjusted p.value
dPDUI_cutoff	cutoff value for differenctial PAS(polyadenylation signal) usage index
PDUI_logFC_cutoff	cutoff value for log2 fold change of PAS(polyadenylation signal) usage index

search_point_START
: start point for searching

search_point_END
: end point for searching

cutStart
: how many nucleotides should be removed from the start before search, 0.1 means 10 percent.

cutEnd
: how many nucleotides should be removed from the end before search, 0.1 means 10 percent.

search_distal_polyA_end
: If true, adjust distal polyA end by cleanUpdTSeq

coverage_threshold
: cutoff coverage threshold for fist 100 nucleotides of each transcript

gcCompensation
: GC content compensation vector

mappabilityCompensation
: mappability compensation vector

FFT
: use Fast Fourier Transform Algorithm to smooth the data or not. default: FALSE

fft.sm.power
: if FFT is TRUE, the frequency should be removed

hugeData
: is this dataset consume too much memory? if it is TRUE, the coverage will be saved into tempfiles.

PolyA_PWM
: Position Weight Matrix of polyA

classifier
: An object of class "PASclassifier"

classifier_cutoff
: This is the cutoff used to assign whether a putative pA is true or false. This can be any floating point number between 0 and 1. For example, classifier_cutoff = 0.5 will assign an putative pA site with prob.1 > 0.5 to the True class (1), and any putative pA site with prob.1 <= 0.5 as False (0).

shift_range
: the shift range for polyA site searching

BPPARAM
: An optional BiocParallelParam instance determining the parallel back-end to be used during evaluation, or a list of BiocParallelParam instances, to be applied in sequence for nested calls to bplapply.

Value

return an object of GRanges

Author(s)

Jianhong Ou

Examples

```
if(interactive()){
    library(BSgenome.Mmusculus.UCSC.mm10)
    library(TxDb.Mmusculus.UCSC.mm10.knownGene)

    path <- file.path(find.package("InPAS"), "extdata")
```

```
    bedgraphs <- file.path(path, "Baf3.extract.bedgraph")
    data(utr3.mm10)
    res <- inPAS(bedgraphs=bedgraphs, tags=c("Baf3"),
              genome=BSgenome.Mmusculus.UCSC.mm10,
              utr3=utr3.mm10, gp1="Baf3", gp2=NULL,
              txdb=TxDb.Mmusculus.UCSC.mm10.knownGene,
              search_point_START=200,
              short_coverage_threshold=15,
              long_coverage_threshold=3,
              cutStart=0, cutEnd=.2,
              hugeData=FALSE)
    res
}
```

`usage4plot` *prepare coverage data and fitting data for plot*

Description

prepare coverage data and fitting data for plot

Usage

```
usage4plot(gr, coverage, proximalSites, genome,
           gp1, gp2,
           gcCompensation=NA, mappabilityCompensation=NA,
           FFT=FALSE, fft.sm.power=20)
```

Arguments

`gr`	an object of GRanges
`coverage`	coverage for each sample
`proximalSites`	proximal sites
`genome`	an object of BSgenome
`gp1`	tag names involved in group 1
`gp2`	tag names involved in group 2
`gcCompensation`	GC content compensation vector
`mappabilityCompensation`	mappability compensation vector
`FFT`	use FFT to smooth the data or not. default: FALSE
`fft.sm.power`	if FFT is TRUE, the frequency should be removed

Value

Formal class 'GRanges' [package "GenomicRanges"] with metadata:

`dat`	matrix, first column is the fit data, the other columns are coverage data for each sample
`offset`	offset from the start of 3UTR

Author(s)

Jianhong Ou

Examples

```
library(BSgenome.Mmusculus.UCSC.mm10)
path <- file.path(find.package("InPAS"), "extdata")
bedgraphs <- c(file.path(path, "Baf3.extract.bedgraph"),
           file.path(path, "UM15.extract.bedgraph"))
coverage <- coverageFromBedGraph(bedgraphs, tags=c("Baf3", "UM15"),
                            genome=Mmusculus, hugeData=FALSE)
gr <- GRanges("chr6", IRanges(128846245, 128850081), strand="-")
dat <- usage4plot(gr, coverage, proximalSites=128849148, Mmusculus)
data <- dat$dat[[1]]
op <- par(mfrow=c(3, 1))
plot(data[,1], type="l", xlab="", ylab="The fitted value")
abline(v=dat$offset)
plot(data[,2], type="l", xlab="", ylab="Baf3")
plot(data[,3], type="l", xlab="", ylab="UM15")
par(op)
```

---

utr3.hg19 *3'UTR annotation for hg19 obtained from utr3Annotation*

Description

3'UTR annotation obtained from utr3Annotation by TxDb.Hsapiens.UCSC.hg19.knownGene and org.Hs.eg.db

Usage

```
data(utr3.hg19)
```

Format

GRanges with slot start holding the start position of the 3'UTR, slot end holding the end position of the 3'UTR, slot names holding transcripts and gene names of 3'UTR, slot seqnames holding the chromosome location where the 3'UTR is located and slot strand for strand of 3'UTR. In addition, the following variables are included.

`feature` should be unknown or proximalCP_XXXXXXXX

`id` should be utr3 or next.exon.gap

`exon` exon id

`transcript` trnscript id

`gene` entriz gene id

`symbol` gene symbol

Details

used in the examples Annotation data obtained by: library(TxDb.Hsapiens.UCSC.hg19.knownGene)

library(org.Hs.eg.db)

utr3Annotation(TxDb.Hsapiens.UCSC.hg19.knownGene, org.Hs.egSYMBOL)

Value

an object of GRanges.

Examples

```
data(utr3.hg19)
head(utr3.hg19)
```

---

utr3.mm10 *3'UTR annotation for mm10 obtained from utr3Annotation*

Description

3'UTR annotation obtained from utr3Annotation by TxDb.Mmusculus.UCSC.mm10.knownGene and org.Mm.eg.db

Usage

```
data(utr3.mm10)
```

Format

GRanges with slot start holding the start position of the 3'UTR, slot end holding the end position of the 3'UTR, slot names holding transcripts and gene names of 3'UTR, slot seqnames holding the chromosome location where the 3'UTR is located and slot strand for strand of 3'UTR. In addition, the following variables are included.

`feature` should be unknown or proximalCP_XXXXXXXX

`id` should be utr3 or next.exon.gap

`exon` exon id

`transcript` trnscript id

`gene` entriz gene id

`symbol` gene symbol

Details

used in the examples Annotation data obtained by: library(TxDb.Mmusculus.UCSC.mm10.knownGene)

library(org.Mm.eg.db)

utr3Annotation(TxDb.Mmusculus.UCSC.mm10.knownGene, org.Mm.egSYMBOL)

Value

an object of GRanges.

Examples

```
data(utr3.mm10)
head(utr3.mm10)
```

`utr3Annotation` *extract 3'UTR from TxDb object*

Description

extract 3'UTR from a TxDb object. The 3'UTR is defined as the last 3'UTR fragment for each transcript and it will be cut if there is any overlaps with other exons.

Usage

```
utr3Annotation(txdb, orgDbSYMBOL, MAX_EXONS_GAP = 10000)
```

Arguments

`txdb`	an object of TxDb
`orgDbSYMBOL`	a string indicates org SYMBOL to entriz id map
`MAX_EXONS_GAP`	maximul exon gap for distal CP site

Value

return an object of GRanges

Author(s)

Jianhong Ou

Examples

```
if(interactive()){
    library(TxDb.Mmusculus.UCSC.mm10.knownGene)

    library(org.Mm.eg.db)

    utr3Annotation(TxDb.Mmusculus.UCSC.mm10.knownGene, "org.Mm.egSYMBOL")
}
```

`utr3UsageEstimation` *estimation of 3'UTR usage for each region*

Description

estimation of 3'UTR usage for short form and long form

Usage

```
utr3UsageEstimation(CPsites, coverage, genome, utr3,
    gp1, gp2=NULL,
    short_coverage_threshold = 10,
    long_coverage_threshold = 2,
    adjusted.P_val.cutoff = 0.05,
    dPDUI_cutoff = 0.3,
    PDUI_logFC_cutoff=0.59, BPPARAM=NULL)
```

Arguments

`CPsites`	outputs of CPsites
`coverage`	coverage for each sample, outputs of coverageFromBedGraph
`genome`	an object of BSgenome
`utr3`	output of utr3Annotation
`gp1`	tag names involved in group 1
`gp2`	tag names involved in group 2
`short_coverage_threshold`	cutoff threshold for coverage in thre region of short form
`long_coverage_threshold`	cutoff threshold for coverage in thre region of long form
`adjusted.P_val.cutoff`	cutoff value for adjusted p.value
`dPDUI_cutoff`	cutoff value for differenctial PAS(polyadenylation signal) usage index
`PDUI_logFC_cutoff`	cutoff value for log2 fold change of PAS(polyadenylation signal) usage index
`BPPARAM`	An optional `BiocParallelParam` instance determining the parallel back-end to be used during evaluation, or a list of BiocParallelParam instances, to be applied in sequence for nested calls to bplapply.

Value

return an object of GRanges

Author(s)

Jianhong Ou

Examples

```
if(interactive()){
    library(BSgenome.Mmusculus.UCSC.mm10)
    path <- file.path(find.package("InPAS"), "extdata")
    bedgraphs <- file.path(path, "Baf3.extract.bedgraph")
    data(utr3.mm10)
    tags <- "Baf3"
    genome <- BSgenome.Mmusculus.UCSC.mm10
    coverage <-
        coverageFromBedGraph(bedgraphs, tags, genome, hugeData=FALSE)
    CP <- CPsites(coverage=coverage, gp1=tags, gp2=NULL, genome=genome,
        utr3=utr3.mm10, coverage_threshold=5, long_coverage_threshold=5)
    res <- utr3UsageEstimation(CP, coverage,
        utr3.mm10, genome, gp1=tags, gp2=NULL)
}
```

Index