Package 'InPAS'

April 15, 2020

Type Package

Title InPAS: a bioconductor package for the identification of novel alternative PolyAdenylation Sites (PAS) using RNA-seq data

Version 1.18.0

Author Jianhong Ou, Sungmi M. Park, Michael R. Green and Lihua Julie Zhu

Maintainer Jianhong Ou <jianhong.ou@duke.edu>, Lihua Julie Zhu <Julie.Zhu@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 and the differential usage of APA sites from RNA-Seq data. It leverages cleanUpdTSeq to fine tune identified APA sites by removing false sites due to internal-priming.

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

License GPL (>= 2)

Lazyload yes

Imports AnnotationDbi, BSgenome, cleanUpdTSeq, Gviz, seqinr, preprocessCore, IRanges, GenomeInfoDb, depmixS4, limma, BiocParallel

Depends R (>= 3.1), methods, Biobase, GenomicRanges, GenomicFeatures, 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, knitr

VignetteBuilder knitr

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

git_branch RELEASE_3_10

git_last_commit 205b776

git_last_commit_date 2019-10-29

Date/Publication 2020-04-14

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, 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

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

---

coverageRate *coverage rate of genes and 3UTRs*

Description

calculate coverage rate of gene and 3UTRs. This function is used for quality control of the coverage. The coverage rate can show the complexity of RNA-seq library.

Usage

```
coverageRate(coverage, txdb, genome,
                      cutoff_readsNum=1,
                      cutoff_expdGene_cvgRate=0.1,
                      cutoff_expdGene_sampleRate=0.5,
                      which=NULL, ...)
```

Arguments

coverage	coverage for each sample, output of coverageFromBedGraph
txdb	an object of TxDb
genome	an object of BSgenome
cutoff_readsNum	cutoff reads number. If the coverage in the location is greater than cutoff_readsNum, the location will be treated as coveraged by signal.
cutoff_expdGene_cvgRate,cutoff_expdGene_sampleRate	cutoff_expdGene_cvgRate and cutoff_expdGene_sampleRate are the parameters used to calculate which gene is expressed in all input dataset. cutoff_expdGene_cvgRate set the cutoff value for the coverage rate of each gene; cutoff_expdGene_sampleRate set the cutoff value for ratio of numbers of expressed and all samples for each gene. for example, by default, cutoff_expdGene_cvgRate=0.1 and cutoff_expdGene_sampleRate=0.5 surppose there are 4 samples, for one gene, if the coverage rates by base are: 0.05, 0.12, 0.2, 0.17, this gene will be count as expressed gene because mean(c(0.05, 0.12, 0.2, 0.17) > cutoff_expdGene_cvgRate) > cutoff_expdGene_sampleRate if the coverage rates by base are: 0.05, 0.12, 0.07, 0.17, this gene will be count as un-expressed gene because mean(c(0.05, 0.12, 0.07, 0.17) > cutoff_expdGene_cvgRate) <= cutoff_expdGene_sampleRate
which	an object of GRanges or NULL. If it is not NULL, only the exons overlapping the given ranges are used.
...	not used.

Value

return a datafrom with colnames : gene.coverage.rate: coverage per base for all genes, expressed.gene.coverage.rate: coverage per base for expressed genes, UTR3.coverage.rate: coverage per base for all 3' UTRs, UTR3.expressed.gene.subset.coverage.rate: coverage per base for 3' UTRs of expressed genes. and rownames: the names of coverage.

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 <- c(file.path(path, "Baf3.extract.bedgraph"),
                   file.path(path, "UM15.extract.bedgraph"))
    hugeData <- FALSE

    coverage <- coverageFromBedGraph(bedgraphs,
                            tags=c("Baf3", "UM15"),
                            genome=BSgenome.Mmusculus.UCSC.mm10,
                            hugeData=hugeData)
    coverageRate(coverage,
                txdb=TxDb.Mmusculus.UCSC.mm10.knownGene,
                genome=BSgenome.Mmusculus.UCSC.mm10,
                which = GRanges("chr6", ranges=IRanges(98013000, 140678000)))
}
```

`covThreshold` *calculate the cutoff threshold of coverage*

Description

calculate the cutoff threshold of coverage for long form and short form

Usage

```
covThreshold(coverage, genome, txdb, utr3,
                          chr="chr1", hugeData, groupList)
```

Arguments

`coverage`	coverage for each sample, output of coverageFromBedGraph
`genome`	an object of BSgenome
`txdb`	an object of TxDb
`utr3`	output of utr3Annotation
`chr`	chromosome to be used for calculation, default is "chr1"
`hugeData`	is this dataset consume too much memory? if it is TRUE, the coverage will be saved into tempfiles.
`groupList`	group list of tag names

Value

a numeric vector

Author(s)

Jianhong Ou

See Also

`CPsite_estimation`

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

Description

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

Usage

```
CPsites(coverage, groupList=NULL, genome, utr3,
    window_size=100, search_point_START=50, search_point_END=NA,
    cutStart=window_size, cutEnd=0, adjust_distal_polyA_end=TRUE,
    coverage_threshold=5, long_coverage_threshold=2,
    background=c("same_as_long_coverage_threshold",
                 "1K", "5K", "10K", "50K"),
    txdb=NA,
    PolyA_PWM=NA, classifier=NA, classifier_cutoff=.8, step=1,
    two_way=FALSE,
    shift_range=window_size,
    BPPARAM=NULL, tmpfolder=NULL, silence=TRUE)
```

Arguments

coverage	coverage for each sample, output of coverageFromBedGraph
groupList	group list of tag names
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.
adjust_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.
background	the range for calculating cutoff threshold of local background
txdb	an object of TxDb
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).
step	adjust step, default 1, means adjust by each base by cleanUpdTSeq.

two_way	Search the proximal site from both direction or not.
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.
tmpfolder	temp folder could save and reload the analysis data for resume analysis.
silence	report progress or not. default not report.

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)
}
```

CPsite_estimation *estimate the cpsites*

Description

estimate the cpsites for a giving chromosome

Usage

```
CPsite_estimation(chr.cov, utr3, MINSIZE, window_size, search_point_START,
search_point_END, cutStart, cutEnd, adjust_distal_polyA_end,
background, z2s, coverage_threshold, long_coverage_threshold,
PolyA_PWM, classifier, classifier_cutoff, shift_range,
depth.weight, genome, step=1, two_way=FALSE,
tmpfolder=NULL, silence=TRUE)
```

Arguments

chr.cov	coverage list for one chromosome
utr3	output of utr3Annotaion
MINSIZE	min size of short form
window_size	window size
search_point_START	search start point
search_point_END	search end point
cutStart	cut from start
cutEnd	cut from end
adjust_distal_polyA_end	adjust distal site or not
background	how to get the local background
z2s	output of zScoreThrethold
coverage_threshold	cutoff value for coverage
long_coverage_threshold	cutoff value for long form
PolyA_PWM	polyA PWM
classifier	classifier
classifier_cutoff	classifier cutoff
shift_range	shift range
depth.weight	output of depthWeight
genome	a BSgenome object
step	adjust step, default 1, means adjust by each base by cleanUpdTSeq.
two_way	Search the proximal site from both direction or not.
tmpfolder	temp folder could save and reload the analysis data for resume analysis.
silence	report progress or not. default not report.

Value

a data.frame

Author(s)

Jianhong Ou

See Also

CPsites, searchProximalCPs, proximalAdj, proximalAdjByPWM, proximalAdjByCleanUpdTSeq, PAscore,PAscore2

`depthWeight` *calculate the depth weight for each example*

Description

calculate the depth weight for each example

Usage

```
depthWeight(coverage, hugeData, groupList=NULL)
```

Arguments

`coverage`	a list. output of coverageFromBedGraph
`hugeData`	is it a huge dataset?
`groupList`	group list for huge dataset

Value

a numeric vector with depth weight

Author(s)

Jianhong Ou

`distalAdj` *adjust distal CP sites by cleanUpdTSeq*

Description

adjust distal CP sites by cleanUpdTSeq

Usage

```
distalAdj(distalCPs, classifier, classifier_cutoff, shift_range, genome, step=1)
```

Arguments

`distalCPs`	the output of searchDistalCPs
`classifier`	cleanUpdTSeq classifier
`classifier_cutoff`	cutoff value of the classiffier
`shift_range`	the searching range for the better CP sites
`genome`	a BSgenome object
`step`	adjust step, default 1, means adjust by each base by cleanUpdTSeq.

Value

a list could be input of searchProximalCPs

Author(s)

Jianhong Ou

See Also

searchDistalCPs, PAscore2

---

filterRes *filter results*

Description

filter results of testUsage

Usage

```
filterRes(res,
          gp1, gp2,
          background_coverage_threshold=2,
          P.Value_cutoff=0.05,
          adj.P.Val_cutoff=0.05,
          dPDUI_cutoff=0.3,
          PDUI_logFC_cutoff)
```

Arguments

res	output of testUsage
gp1	tag names involved in group 1
gp2	tag names involved in group 2
background_coverage_threshold	background coverage cut off value. for each group, more than half of the long form should greater than background_coverage_threshold. for both group, at leat in one group, more than half of the short form should greater than background_coverage_threshold.
P.Value_cutoff	cutoff of P value
adj.P.Val_cutoff	cutoff of adjust P value
dPDUI_cutoff	cutoff of dPDUI
PDUI_logFC_cutoff	cutoff of PDUI log2 transformed fold change

Value

a data.frame

Author(s)

Jianhong Ou

See Also

testUsage

Examples

```
path <- file.path(find.package("InPAS"), "extdata")
load(file.path(path, "CPs.MAQC.rda"))
load(file.path(path, "coverage.MAQC.rda"))
library(BSgenome.Hsapiens.UCSC.hg19)
data(utr3.hg19)
res <- testUsage(CPsites=CPs,
                 coverage=coverage,
                 genome=BSgenome.Hsapiens.UCSC.hg19,
                 utr3=utr3.hg19,
                 method="fisher.exact",
                 gp1=c("Brain.auto", "Brain.phiX"),
                 gp2=c("UHR.auto", "UHR.phiX"))
filterRes(res,
          gp1=c("Brain.auto", "Brain.phiX"),
          gp2=c("UHR.auto", "UHR.phiX"),
          background_coverage_threshold=2,
          P.Value_cutoff=0.05,
          adj.P.Val_cutoff=0.05,
          dPDUI_cutoff=0.3,
          PDUI_logFC_cutoff=.59)
```

---

fisher.exact.test *do fisher exact test for two group datasets*

Description

do fisher exact test for two group datasets

Usage

```
fisher.exact.test(UTR3eset, gp1, gp2)
```

Arguments

UTR3eset	output of getUTR3eSet
gp1	tag names of group 1
gp2	tag names of group 2

Value

a matrix of test results

Author(s)

Jianhong Ou

See Also

singleSampleAnalyze,singleGroupAnalyze, limmaAnalyze

Examples

```
path <- file.path(find.package("InPAS"), "extdata")
load(file.path(path, "eset.MAQC.rda"))
tags <- colnames(eset$PDUI.log2)
res <- fisher.exact.test(eset, gp1=tags[1:2], gp2=tags[3:4])
```

---

get.regions.coverage *claculate coverage for giving region*

Description

claculate coverage for giving region

Usage

```
get.regions.coverage(chr, utr3.regions.chr,
                                  hugeData, coverage, phmm=FALSE)
```

Arguments

chr	chromosome
utr3.regions.chr	the GRanges of region to be extracted
hugeData	is it a huge dataset?
coverage	output of coverageFromBedGraph
phmm	prepare data for singleSample analysis?

Value

GRanges with coverage data

Author(s)

Jianhong Ou

getCov *extract coverage from bedgraph file*

Description

extract coverage from bedgraph file

Usage

```
getCov(bedgraph, genome, seqLen)
```

Arguments

bedgraph	bedGraph file names
genome	an object BSgenome
seqLen	lengthes of each chromosome

Value

a Rle object for a sample coverage

Author(s)

Jianhong Ou

See Also

coverageFromBedGraph

getUTR3eSet *prepare dataset for test*

Description

Generate a UTR3eSet object with PDUI infomation for statistic test

Usage

```
getUTR3eSet(CPsites, coverage, genome, utr3,
            normalize=c("none", "quantiles", "quantiles.robust",
                        "mean", "median"),
            ...,
            BPPARAM=NULL, singleSample=FALSE)
```

Arguments

CPsites	outputs of CPsites
coverage	coverage for each sample, outputs of coverageFromBedGraph
genome	an object of BSgenome
utr3	output of utr3Annotation
normalize	normalization method
...	parameter can be passed into normalize.quantiles.robust
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.
singleSample	prepare data for singleSample analysis? default is FALSE

Value

An object of UTR3eSet which contains following elements:

usage: an GRanges object with CP sites info.

PDUI: a matrix of PDUI

PDUI.log2: log2 transformed PDUI matrix

short: a matrix of usage of short form

long: a matrix of usage of long form

if singleSample is TRUE, one more element, signals, will be included.

Author(s)

Jianhong Ou

Examples

```
path <- file.path(find.package("InPAS"), "extdata")
load(file.path(path, "CPs.MAQC.rda"))
load(file.path(path, "coverage.MAQC.rda"))
library(BSgenome.Hsapiens.UCSC.hg19)
data(utr3.hg19)
getUTR3eSet(CPsites=CPs,
            coverage=coverage,
            genome=BSgenome.Hsapiens.UCSC.hg19,
            utr3=utr3.hg19)
```

---

getUTR3region *extract long and short 3UTR region*

Description

extract long and short 3UTR region

Usage

```
getUTR3region(.grs)
```

Arguments

.grs	output of CPsites

Value

GRanges with short form and long form

Author(s)

Jianhong Ou

---

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, genome, utr3, txdb=NA,
      tags, hugeData=FALSE, ...,

      gp1, gp2,

      window_size=100,
      search_point_START=50, search_point_END=NA,
      cutStart=window_size, cutEnd=0,
      coverage_threshold=5, long_coverage_threshold=2,
      background=c("same_as_long_coverage_threshold",
                   "1K", "5K", "10K", "50K"),
      adjust_distal_polyA_end=TRUE,
      PolyA_PWM=NA, classifier=NA, classifier_cutoff=.8,
      shift_range=window_size,

      method=c("limma", "fisher.exact",
               "singleSample", "singleGroup"),
      normalize=c("none", "quantiles", "quantiles.robust",
                  "mean", "median"),
      design, contrast.matrix, coef=1,

      P.Value_cutoff=0.05,
      adj.P.Val_cutoff=0.05,
      dPDUI_cutoff=0.3,
      PDUI_logFC_cutoff=0.59,

      BPPARAM=NULL)
```

Arguments

bedgraphs	The file names of bedgraphs generated by bedtools. eg: bedtools genomecov -bg -split -ibam $bam -g mm10.size.txt > $bedgraph
genome	an object of BSgenome
utr3	output of utr3Annotation
txdb	an object of TxDb
tags	the names for each input bedgraphs
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.
gp1	tag names involved in group 1
gp2	tag names involved in group 2
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.
cutEnd	how many nucleotides should be removed from the end before search, 0.1 means 10 percent.
coverage_threshold	cutoff threshold for coverage in the region of short form
long_coverage_threshold	cutoff threshold for coverage in thre region of long form
background	the range for calculating cutoff threshold of local background
adjust_distal_polyA_end	If true, adjust distal polyA end by cleanUpdTSeq
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
method	test method. see singleSampleAnalyze,singleGroupAnalyze, fisher.exact.test,limmaAnalyz
normalize	normalization method
design	the design matrix of the experiment, with rows corresponding to arrays and columns to coefficients to be estimated. Defaults to the unit vector meaning that the arrays are treated as replicates. see model.matrix

`contrast.matrix`
: numeric matrix with rows corresponding to coefficients in fit and columns containing contrasts. May be a vector if there is only one contrast. see makeContrasts

`coef`
: column number or column name specifying which coefficient or contrast of the linear model is of interest. see more topTable. default value: 1

`P.Value_cutoff`
: cutoff of P value

`adj.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)
    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
}
```

---

lastCDSusage *extract coverage of last CDS exon region*

Description

extract coverage of last CDS exon region

Usage

```
lastCDSusage(CDS, coverage, hugeData, BPPARAM=NULL, phmm=FALSE)
```

Arguments

CDS	GRanges object of CDS
coverage	output of coverageFromBedGraph
hugeData	is it a huge dataset?
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.
phmm	prepare data for singleSample analysis?

Value

the average coverage of last CDS for each transcript

Author(s)

Jianhong Ou

---

limmaAnalyze *use limma to analyze the PDUI*

Description

use limma to analyze the PDUI

Usage

```
limmaAnalyze(UTR3eset, design, contrast.matrix, coef=1, robust=FALSE, ...)
```

Arguments

UTR3eset	an UTR3eSet object
design	the design matrix of the experiment, with rows corresponding to arrays and columns to coefficients to be estimated. Defaults to the unit vector meaning that the arrays are treated as replicates. see model.matrix
contrast.matrix	numeric matrix with rows corresponding to coefficients in fit and columns containing contrasts. May be a vector if there is only one contrast. see makeContrasts
coef	column number or column name specifying which coefficient or contrast of the linear model is of interest. see more topTable. default value: 1
robust	logical, should the estimation of the empirical Bayes prior parameters be robustified against outlier sample variances?
...	other arguments are passed to lmFit.

Value

fit results of eBayes by limma. It is an object of class MArrayLM containing everything found in fit. see eBayes

Author(s)

Jianhong Ou

See Also

singleSampleAnalyze,singleGroupAnalyze, fisher.exact.test

Examples

```
library(limma)
path <- file.path(find.package("InPAS"), "extdata")
load(file.path(path, "eset.MAQC.rda"))
tags <- colnames(eset$PDUI.log2)
g <- factor(gsub("\\..*$", "", tags))
design <- model.matrix(~-1+g)
colnames(design) <- c("Brain", "UHR")
contrast.matrix <- makeContrasts(contrasts="Brain-UHR",levels=design)
res <- limmaAnalyze(eset, design, contrast.matrix)
head(res)
```

---

optimalSegmentation *calculate SSE*

Description

calculate SSE values

Usage

```
optimalSegmentation(.ele, search_point_START, search_point_END, n = 1, savedID = NA)
```

Arguments

.ele	3UTR coverage
search_point_START	start position to calculate
search_point_END	end position to calculate
n	the length of output
savedID	the proximal CPsites for noval distal events

Value

a list of SSE and idx

Author(s)

Jianhong Ou

PAscore *calculate the CP score*

Description

calculate the CP score by PWM

Usage

```
PAscore(seqname, pos, str, idx, PWM, genome, ups = 50, dws = 50)
```

Arguments

seqname	sequence names
pos	genomic positions
str	strands
idx	offset postion
PWM	polyA position weight matrix
genome	an object of BSgenome
ups	upstream base
dws	downstream base

Value

idx list after filter

Author(s)

Jianhong Ou

See Also

PAscore2

PAscore2 *calculate the CP score*

Description

calculate CP score by cleanUpdTSeq

Usage

```
PAscore2(seqname, pos, str, idx, idx.gp, genome, classifier, classifier_cutoff)
```

Arguments

seqname	sequence names
pos	genomic positions
str	strands
idx	offset postion
idx.gp	group number of the offset postion
genome	an object of BSgenome
classifier	a cleanUpdTSeq classifier
classifier_cutoff	classifier cutoff value

Value

a data.frame

Author(s)

Jianhong Ou

See Also

PAscore

---

polishCPs *polish the searching results of CP sites*

Description

remove the multiple positions of CP sites for same 3UTRs and only keep the best CP sites for proximal and distal.

Usage

```
polishCPs(CPs)
```

Arguments

CPs	output of searchProximalCPs or proximalAdj

Value

a matrix with columes: "fit_value", "Predicted_Proximal_APA", "Predicted_Distal_APA", "utr3start", "utr3end", "type"

Author(s)

Jianhong Ou

See Also

CPsite_estimation, searchProximalCPs, proximalAdj, proximalAdjByPWM, proximalAdjByCleanUpdTSeq, PAscore,PAscore2

prepare4GSEA *prepare the files for GSEA analysis*

Description

output the log2 transformed delta PDUI txt file and chip file for GSEA analysis

Usage

```
prepare4GSEA(eset, groupList, Preranked=TRUE,
             folder=".",
             rnkFilename="InPAS.rnk",
             chipFilename="InPAS.chip",
             dataFilename="dPDUI.txt",
             PhenFilename="group.cls")
```

Arguments

eset	a UTR3eSet object
groupList	group list of tag names
Preranked	logical value, out preranked or not
folder	output folder
rnkFilename	filename of preranked file
chipFilename	filename of chip
dataFilename	filename of dataset
PhenFilename	filename of Phenotype labels

Value

None

Author(s)

Jianhong Ou

Examples

```
file <- system.file("extdata", "eset.MAQC.rda", package="InPAS")
load(file)
gp1=c("Brain.auto", "Brain.phiX")
gp2=c("UHR.auto", "UHR.phiX")
groupList <- list(Brain=gp1, UHR=gp2)
prepare4GSEA(eset, groupList=groupList, Preranked=FALSE)
```

proximalAdj *adjust the proximal CP sites*

Description

adjust the proximal CP sites by PolyA PWM and cleanUpdTSeq

Usage

```
proximalAdj(CPs, MINSIZE, PolyA_PWM, genome, classifier, classifier_cutoff,
            shift_range, search_point_START, step=1)
```

Arguments

CPs	the outputs of searchProximalCPs
MINSIZE	min size for short from
PolyA_PWM	PolyA position weight metrix
genome	a BSgenome object
classifier	cleanUpdTSeq classifier
classifier_cutoff	cutoff value of the classiffier
shift_range	the searching range for the better CP sites
search_point_START	just in case there is no better CP sites
step	adjust step, default 1, means adjust by each base by cleanUpdTSeq.

Value

keep same as searchProximalCPs, which can be handled by polishCPs.

Author(s)

Jianhong Ou

See Also

searchProximalCPs, polishCPs, proximalAdjByPWM, proximalAdjByCleanUpdTSeq, PAscore,PAscore2

`proximalAdjByCleanUpdTSeq` *adjust the proximal CP sites by cleanUpdTseq*

Description

adjust the proximal CP sites by cleanUpdTseq

Usage

```
proximalAdjByCleanUpdTSeq(idx.list, cov_diff.list, seqnames, starts, strands,
                          genome, classifier, classifier_cutoff,
                          shift_range, search_point_START, step=1)
```

Arguments

`idx.list`	the offset of positions of CP sites
`cov_diff.list`	the SSE values
`seqnames`	sequence names
`starts`	starts
`strands`	strands
`genome`	a BSgenome object
`classifier`	cleanUpdTSeq classifier
`classifier_cutoff`	cutoff value of the classiffier
`shift_range`	the searching range for the better CP sites
`search_point_START`	just in case there is no better CP sites
`step`	adjust step, default 1, means adjust by each base by cleanUpdTSeq.

Details

the step for calculating is 10, can not do every base base it is really very slow.

Value

the offset of positions of CP sites after filter

Author(s)

Jianhong Ou

See Also

`proximalAdjByPWM`, `proximalAdj`, `PAscore2`

`proximalAdjByPWM` *adjust the proximal CP sites by PWM*

Description

adjust the proximal CP sites by polyA Position Weight Metrix. It only need the PWM get match in upstream or downstream shift_range nr.

Usage

```
proximalAdjByPWM(idx, PolyA_PWM, seqnames, starts, strands, genome,
                 shift_range, search_point_START)
```

Arguments

`idx`	the offset of positions of CP sites
`PolyA_PWM`	polyA PWM
`seqnames`	sequence names
`starts`	start position in the genome
`strands`	strands
`genome`	an BSgenome object
`shift_range`	the shift range of PWM hits
`search_point_START`	Not use

Details

the hits is searched by matchPWM and the cutoff is 70%

Value

the offset of positions of CP sites after filter

Author(s)

Jianhong Ou

See Also

`proximalAdjByCleanUpdTSeq`, `proximalAdj`, `PAscore`

removeUTR3__UTR3 *remove the candicates LIKE UTR3___UTR3*

Description

some of the results is from connected two UTR3. We want to remove them. However, the algorithm need to be improved.

Usage

```
removeUTR3__UTR3(x)
```

Arguments

x	the distal 3UTR coverage

Value

the 3UTR coverage after removing the next 3UTR

Author(s)

Jianhong Ou

searchDistalCPs *search distal CP sites*

Description

search distal CP sites

Usage

```
searchDistalCPs(chr.cov.merge, conn_next_utr3,
            curr_UTR, window_size,
            depth.weight,
            long_coverage_threshold,
            background, z2s)
```

Arguments

chr.cov.merge	coverage of current chromosome
conn_next_utr3	joint to next 3UTR or not (used for removeUTR3__UTR3)
curr_UTR	GRanges of current 3UTR
window_size	window size
depth.weight	output of depthWeight
long_coverage_threshold	cutoff value for coverage of long form 3UTR
background	local background range
z2s	cut off background scores. see zScoreThrethold

Value

a list

Author(s)

Jianhong Ou

See Also

distalAdj, PAscore2

---

searchProximalCPs *search proximal CPsites*

Description

search proximal CPsites

Usage

```
searchProximalCPs(CPs, curr_UTR, window_size,
                  MINSIZE, cutEnd,
                  search_point_START,
                  search_point_END,
                  two_way=FALSE)
```

Arguments

CPs	output of searchDistalCPs or distalAdj
curr_UTR	GRanges of current 3UTR
window_size	window size
MINSIZE	MINSIZE for short form
cutEnd	how many nucleotides should be removed from the end before search, 0.1 means 10 percent.
search_point_START	start point for searching
search_point_END	end point for searching
two_way	Search the proximal site from both direction or not.

Value

a list

Author(s)

Jianhong Ou

See Also

proximalAdj, polishCPs, proximalAdjByPWM, proximalAdjByCleanUpdTSeq, PAscore,PAscore2

seqLen *get sequence lengths*

Description

get sequence lengths from a BSgenome object

Usage

```
seqLen(genome)
```

Arguments

genome	an object of BSgenome

Value

a numeric vector

Author(s)

Jianhong Ou

See Also

seqlengths

singleGroupAnalyze *do analysis for single group samples*

Description

do analysis for single group samples by anova test

Usage

```
singleGroupAnalyze(UTR3eset)
```

Arguments

UTR3eset	must be the output of getUTR3eSet

Value

a metrix of test results

Author(s)

Jianhong Ou

See Also

UTR3eSet, getUTR3eSet

Examples

```
path <- file.path(find.package("InPAS"), "extdata")
load(file.path(path, "eset.MAQC.rda"))
res <- singleGroupAnalyze(eset)
```

`singleSampleAnalyze` *do analysis for single sample*

Description

do analysis for single sample by a hidden Markov model

Usage

```
singleSampleAnalyze(UTR3eset)
```

Arguments

UTR3eset	must be the output of getUTR3eSet

Details

the test will be performed by a two states hidden Markov model.

Value

a metrix of test results

Author(s)

Jianhong Ou

See Also

UTR3eSet, getUTR3eSet, depmix

Examples

```
path <- file.path(find.package("InPAS"), "extdata")
load(file.path(path, "eset.MAQC.rda"))
res <- singleSampleAnalyze(eset)
```

sortGR *sort GRanges*

Description

sort a GRanges by chromosome and start position

Usage

```
sortGR(.ele)
```

Arguments

.ele	an object of GRanges

Value

an sorted object of GRanges

Author(s)

Jianhong Ou

testUsage *do test for dPDUI*

Description

do test for dPDUI

Usage

```
testUsage(CPsites, coverage, genome, utr3, BPPARAM=NULL,
          method=c("limma", "fisher.exact",
                   "singleSample", "singleGroup"),
          normalize=c("none", "quantiles", "quantiles.robust",
                      "mean", "median"),
          design, contrast.matrix, coef=1, robust=FALSE, ...,
          gp1, gp2)
```

Arguments

CPsites	outputs of CPsites
coverage	coverage for each sample, outputs of coverageFromBedGraph
genome	an object of BSgenome
utr3	output of utr3Annotation
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.

method	test method. see singleSampleAnalyze,singleGroupAnalyze, fisher.exact.test,limmaAnalyz
normalize	normalization method
design	the design matrix of the experiment, with rows corresponding to arrays and columns to coefficients to be estimated. Defaults to the unit vector meaning that the arrays are treated as replicates. see model.matrix
contrast.matrix	numeric matrix with rows corresponding to coefficients in fit and columns containing contrasts. May be a vector if there is only one contrast. see makeContrasts
coef	column number or column name specifying which coefficient or contrast of the linear model is of interest. see more topTable. default value: 1
robust	logical, should the estimation of the empirical Bayes prior parameters be robustified against outlier sample variances?
...	other arguments are passed to lmFit.
gp1	tag names involved in group 1
gp2	tag names involved in group 2

Details

if method is "limma", design matrix and contrast is required. if method is "fisher.exact", gp1 and gp2 is required.

Value

a list with test results. the output of test results is a matrix.

Author(s)

Jianhong Ou

See Also

singleSampleAnalyze,singleGroupAnalyze, fisher.exact.test,limmaAnalyze

Examples

```
library(limma)
path <- file.path(find.package("InPAS"), "extdata")
load(file.path(path, "CPs.MAQC.rda"))
load(file.path(path, "coverage.MAQC.rda"))
library(BSgenome.Hsapiens.UCSC.hg19)
data(utr3.hg19)
tags <- names(coverage)
g <- factor(gsub("\\..*$", "", tags))
design <- model.matrix(~-1+g)
colnames(design) <- c("Brain", "UHR")
contrast.matrix<-makeContrasts(contrasts="Brain-UHR",levels=design)
res <- testUsage(CPsites=CPs,
             coverage=coverage,
             genome=BSgenome.Hsapiens.UCSC.hg19,
             utr3=utr3.hg19,
             method="limma",
             design=design,
             contrast.matrix=contrast.matrix)
```

totalCoverage *total coverage*

Description

for huge dataset, it will read in the coverage from tmp files and merge them by groups

Usage

```
totalCoverage(coverage, genome, hugeData, groupList=NULL)
```

Arguments

coverage	coverage for each sample, outputs of coverageFromBedGraph
genome	an object of BSgenome
hugeData	hugeData or not
groupList	tag names involved in each groups

Value

a coverage list

Author(s)

Jianhong Ou

trimSeqnames *trim the sequence names*

Description

only ^chr[0-9XY]+$ is OK.

Usage

```
trimSeqnames(genome)
```

Arguments

genome	an BSgenome object

Value

an character vector with trimmed seqnames

Author(s)

Jianhong Ou

usage4plot *prepare coverage data and fitting data for plot*

Description

prepare coverage data and fitting data for plot

Usage

```
usage4plot(gr, coverage, proximalSites, genome, groupList)
```

Arguments

gr	an object of GRanges
coverage	coverage for each sample
proximalSites	proximal sites
genome	an object of BSgenome
groupList	the list of sample names

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.danRer10` *3'UTR annotation for danRer10 obtained from utr3Annotation*

Description

3'UTR annotation obtained from utr3Annotation by TxDb.Drerio.UCSC.danRer10.refGene and org.Dr.eg.db

Usage

```
data(utr3.danRer10)
```

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.Drerio.UCSC.danRer10.refGene)

library(org.Dr.eg.db)

utr3Annotation(TxDb.Drerio.UCSC.danRer10.refGene, "org.Dr.egSYMBOL")

Value

an object of GRanges.

Examples

```
data(utr3.danRer10)
head(utr3.danRer10)
```

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 with 7 metadata columns: feature (utr3, next.exon.gap, CDS), annotatedProximalCP (unknown, proximalCP_<coordinate>), exon (<transcript id>_<index>), transcript, gene (entrez_id), symbol, truncated (logical).

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")
}
```

UTR3eSet-class *Class* UTR3eSet

Description

An object of class UTR3eSet represents the results of 3UTR usage

Objects from the Class

Objects can be created by calls of the form new("UTR3eSet",usage,PDUI,PDUI.log2,short,long,signals,testRes

Slots

`usage` an GRanges object with CP sites info.

`PDUI` a matrix of PDUI

`PDUI.log2` log2 transformed PDUI matrix

`short` a matrix of usage of short form

`long` a matrix of usage of long form

`signals` signals used for single sample

`testRes` a matrix of test results of testUsage

Methods

$, $<- Get or set the slot of UTR3eSet

as("UTR3eSet", "ExpressionSet") Convert a UTR3eSet to an ExpressionSet.

as("UTR3eSet", "GRanges") Convert a UTR3eSet to an GRanges.

Author(s)

Jianhong Ou

`UTR3TotalCoverage` *extract coverage of 3UTR for CP sites prediction*

Description

extract 3UTR coverage from totalCov according and GRanges object utr3.

Usage

```
UTR3TotalCoverage(utr3, totalCov, gcCompensation = NA,
                  mappabilityCompensation = NA,
                  FFT = FALSE, fft.sm.power = 20)
```

Arguments

`utr3`	an GRanges object. must be the output of utr3Annotation
`totalCov`	total coverage of each sample. must be the output of totalCoverage
`gcCompensation`	GC compensation vector. Not support yet.
`mappabilityCompensation`	mappability compensation vector. Not support yet.
`FFT`	Use FFT smooth or not.
`fft.sm.power`	the cut-off frequence of FFT smooth.

Value

a list. level 1: chromosome; level 2: each transcripts; level3: data matrix

Author(s)

Jianhong Ou

UTR3usage *calculate the usage of long and short form of UTR3*

Description

calculate the usage of long and short form of UTR3 for the results of CPsites

Usage

```
UTR3usage(CPsites, coverage, hugeData, BPPARAM = NULL, phmm = FALSE)
```

Arguments

CPsites	outputs of CPsites
coverage	coverage for each sample, outputs of coverageFromBedGraph
hugeData	is this dataset consume too much memory? if it is TRUE, the coverage will be saved into tempfiles.
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.
phmm	prepare data for singleSample analysis? default is FALSE

Value

GRanges object

Author(s)

Jianhong Ou

See Also

CPsites

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)
}
```

`valley` *get the local minimal square standard error (SSE)*

Description

For a giving numeric vectors, calculate the top N local minimal square standard error. It will also include the saved ID if it is in the range of (ss, se)

Usage

```
valley(x, ss, se, n = 1, savedID = NA, filterByPval = TRUE)
```

Arguments

`x`	numeric vector
`ss`	start searching position
`se`	end searching position
`n`	the length of output. If n=-1, output all the local minimal SSE positions.
`savedID`	saved positions
`filterByPval`	logical. Filter the positions by p value or not.

Value

a numeric vector, position list.

Author(s)

Jianhong Ou

`zScoreThrethold` *calculate local background cutoff value*

Description

calculate local background cutoff value based on z-score

Usage

```
zScoreThrethold(background, introns, totalCov, utr3, z = 2)
```

Arguments

`background`	background range
`introns`	GRanges of introns
`totalCov`	total coverage of output of totalCoverage
`utr3`	output of utr3Annotation
`z`	z score cut off value

Value

a numeric vector

Author(s)

Jianhong Ou

Index