Package 'APAlyzer'

February 5, 2025

Type Package

Title A toolkit for APA analysis using RNA-seq data

Version 1.21.0

Description Perform 3'UTR APA, Intronic APA and gene expression analysis using RNA-seq data.

biocViews Sequencing, RNASeq, DifferentialExpression, GeneExpression, GeneRegulation, Annotation, DataImport, Software

Imports GenomicRanges, GenomicFeatures, GenomicAlignments, DESeq2, ggrepel, SummarizedExperiment, Rsubread, stats, ggplot2, methods, rtracklayer, VariantAnnotation, dplyr, tidyr, repmis, Rsamtools, HybridMTest

Suggests knitr, rmarkdown, BiocStyle, org.Mm.eg.db, AnnotationDbi, TBX20BamSubset, testthat, pasillaBamSubset

URL https://github.com/RJWANGbioinfo/APAlyzer/

BugReports https://github.com/RJWANGbioinfo/APAlyzer/issues

VignetteBuilder knitr

License LGPL-3

Encoding UTF-8

Depends R (>= 3.5.0)

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

git_branch devel

git_last_commit 7a59a45

git_last_commit_date 2024-10-29

Repository Bioconductor 3.21

Date/Publication 2025-02-05

Author Ruijia Wang [cre, aut] (ORCID: <https://orcid.org/0000-0002-4211-5207>), Bin Tian [aut], Wei-Chun Chen [aut]

Maintainer Ruijia Wang <rjwang.bioinfo@gmail.com>

Contents

`APABox` *APABox, APA RED Box plotting*

Description

APA RED Box plotting

Usage

```
APABox (df, xlab = "APAreg", ylab = "RED",
     plot_title = NULL)
```

Arguments

`df`	a dataframe of APAdiff output
`xlab`	lable of x-axis, default is 'APAreg'
`ylab`	lable of y-axis, default is 'RED'
`plot_title`	Main title of plot

Value

The function `APABox` return a Box plot.

Author(s)

Ruijia Wang

Examples

```
    library("TBX20BamSubset")
    library("Rsamtools")
    flsall = getBamFileList()
    extpath = system.file("extdata",
    "mm9_TBX20.APAout.RData", package="APAlyzer")
    load(extpath)
    sampleTable1 = data.frame(samplename = c(names(flsall)),
        condition = c(rep("NT",3),rep("KD",3)))
    sampleTable2 = data.frame(samplename = c("SRR316184","SRR316187"),
        condition = c("NT","KD"))
    ## 3'UTR APA plot
    test_3UTRmuti=APAdiff(sampleTable1,DFUTRraw,
    conKET='NT',trtKEY='KD',PAS='3UTR',CUTreads=0)
UTR_APA_PLOTBOX=APABox(test_3UTRmuti, plot_title='3UTR APA')

    ## IPA plot
    test_IPAmuti=APAdiff(sampleTable1,IPA_OUTraw,
    conKET='NT',trtKEY='KD',PAS='IPA',CUTreads=0)
IPA_PLOTBOX=APABox(test_IPAmuti, plot_title='IPA')
```

`APAdiff` *APAdiff, calculate delta relative expression (RED) and statistics significance between two sample groups*

Description

Calculate delta relative expression (RED) and statistics significance between two sample groups.

Usage

```
 APAdiff(sampleTable,mutiraw, conKET='NT',
trtKEY='KD', PAS='3UTR', CUTreads=0, p_adjust_methods="fdr", MultiTest='unpaired t-test')
```

Arguments

`sampleTable`	a dataframe of sample table containing 8 colmuns for Intronic PASs: 'samplename','condition'
`mutiraw`	a dataframe output obtained using either PASEXP_3UTR or PASEXP_IPA
`conKET`	the name of control in the sampletable, default is 'NT'
`trtKEY`	the name of control in the sampletable, default is 'KD'
`PAS`	type of PAS analyzed, either '3UTR' or 'IPA', default is '3UTR'
`CUTreads`	reads cutoff used for the analysis, default is 0

p_adjust_methods
: p value correction method, the method can be "holm", "hochberg", "hommel", "bonferroni", "BH", "BY","fdr", "none", default is "fdr"

MultiTest
: statistics testing method for muti-replicates designs, the method can be "unpaired t-test", "paired t-test", "ANOVA", default is "unpaired t-test"

Value

The function APAdiff return a dataframe containning RED, pvalue and regulation pattern (UP, DN or NC) for either each gene (3'UTR APA) or each PAS (IPA).

Author(s)

Ruijia Wang

Examples

```
 library("TBX20BamSubset")
 library("Rsamtools")
 flsall = getBamFileList()
 extpath = system.file("extdata",
 "mm9_TBX20.APAout.RData", package="APAlyzer")
 load(extpath)
 sampleTable1 = data.frame(samplename = c(names(flsall)),
     condition = c(rep("NT",3),rep("KD",3)))
 sampleTable2 = data.frame(samplename = c("SRR316184","SRR316187"),
     condition = c("NT","KD"))
 ## Analysis 3'UTR APA between KD and NT group using muti-replicates
 test_3UTRmuti=APAdiff(sampleTable1,DFUTRraw,
conKET='NT',trtKEY='KD',PAS='3UTR',CUTreads=0,p_adjust_methods="fdr",MultiTest='unpaired t-test')

 ## Analysis 3'UTR APA between KD and NT group without replicates
 test_3UTRsing=APAdiff(sampleTable2,DFUTRraw,
 conKET='NT',trtKEY='KD',PAS='3UTR',CUTreads=0,p_adjust_methods="fdr")

 ## Analysis IPA between KD and NT group
 test_IPAmuti=APAdiff(sampleTable1,IPA_OUTraw,
conKET='NT',trtKEY='KD',PAS='IPA',CUTreads=0,p_adjust_methods="fdr",MultiTest='unpaired t-test')

 ## Analysis IPA between KD and NT group without replicates
 test_IPAsing=APAdiff(sampleTable2,IPA_OUTraw,
 conKET='NT',trtKEY='KD',PAS='IPA',CUTreads=0,p_adjust_methods="fdr")
```

APAVolcano *APAVolcano, APA Volcano plotting*

Description

APA Volcano plotting

Usage

```
APAVolcano (df, Pcol = "pvalue",PAS='3UTR',
     top = -1, markergenes = NULL,
     y_cutoff = 0.05,xlab = "RED", ylab = "-Log10(P-value)",
     PAScolor = c("gray80", "red", "blue"),
     alpha = 0.75, plot_title = NULL,
     width = 4, height = 2.5)
```

Arguments

df	a dataframe of APAdiff output
Pcol	p-value column used to for y-axis of volcano plot, default is 'pvalue'
top	number of genes/IPA to label in the plot, default is -1, which don't lable top genes, user can set it >0, e.g., top = 5
markergenes	a set of genes to label in the plot
PAS	type of PAS analyzed, either '3UTR' or 'IPA', default is '3UTR'
y_cutoff	y cutoff line, default is 0.05
xlab	lable of x-axis, default is 'RED'
ylab	lable of y-axis, default is '-Log10(P-value)'
PAScolor	dot color for 'NC','UP' and 'DN' gene/IPAs, default is "gray80", "red", and "blue"
alpha	alpha of the dot, default is 0.75
plot_title	Main title of plot
width	width of the dot, default is 4
height	height of the dot, default is 2.5

Value

The function APAVolcano return a Volcano plot.

Author(s)

Ruijia Wang

Examples

```
library("TBX20BamSubset")
library("Rsamtools")
flsall = getBamFileList()
extpath = system.file("extdata",
"mm9_TBX20.APAout.RData", package="APAlyzer")
load(extpath)
sampleTable1 = data.frame(samplename = c(names(flsall)),
    condition = c(rep("NT",3),rep("KD",3)))
sampleTable2 = data.frame(samplename = c("SRR316184","SRR316187"),
    condition = c("NT","KD"))
```

```
   ## 3'UTR APA plot
   test_3UTRmuti=APAdiff(sampleTable1,DFUTRraw,
   conKET='NT',trtKEY='KD',PAS='3UTR',CUTreads=0)
UTR_APA_PLOT=APAVolcano(test_3UTRmuti, PAS='3UTR', Pcol = "pvalue", top=5, plot_title='3UTR APA')

   ## IPA plot
   test_IPAmuti=APAdiff(sampleTable1,IPA_OUTraw,
   conKET='NT',trtKEY='KD',PAS='IPA',CUTreads=0)
IPA_PLOT=APAVolcano(test_IPAmuti, PAS='IPA', Pcol = "pvalue", top=5, plot_title='IPA')
```

`download_testbam` *download_testbam, download bam files of mouse testis and heart*

Description

download bam files of mouse testis and heart

Usage

```
download_testbam()
```

Value

The function `download_testbam` download test data bam files.

Author(s)

Ruijia Wang

Examples

```
download_testbam()
```

`GENEXP_CDS` *GENEXP_CDS, count reads mapped to CDS regions and calculate TPM for coding gene*

Description

Map reads to CDS regions and calculate TPM for each gene.

Usage

```
GENEXP_CDS(CDSbygene, flS, Strandtype="NONE")
```

Arguments

CDSbygene	a genomic ranges of CDS regions for each coding gene
flS	bamfile lists containing the file and path of bam files
Strandtype	strand type of the bam file; "forward" is forwand sequencing, "invert" is reverse sequencing and "NONE" is non-strand specific, Default is "NONE".

Value

The function GENEXP_CDS() return a dataframe containing reads count, TPM for each gene

Author(s)

Ruijia Wang

Examples

```
## count reads mapped to CDS regions and calculate TPM for each gene
## using forward sequencing
    library("TBX20BamSubset")
    library("Rsamtools")
    library("GenomicAlignments")
    library("GenomicFeatures")
    library("org.Mm.eg.db")
    flsall = getBamFileList()
    extpath = system.file("extdata", "mm9.chr19.refGene.R.DB", package="APAlyzer")
    txdb = loadDb(extpath, packageName='GenomicFeatures')
    IDDB = org.Mm.eg.db
    CDSdbraw = REFCDS(txdb,IDDB)
    DFGENEraw = GENEXP_CDS(CDSdbraw, flsall, Strandtype="forward")
```

PAS2GEF *PAS2GEF, build reference regions for 3'UTR PASs*

Description

Build 3'UTR PAS and IPA (IPA and LE) Reference using GTF file.

Usage

```
PAS2GEF(GTFfile,AnnoMethod="V2")
```

Arguments

GTFfile	GTF file of gene annotation
AnnoMethod	annotation method used to build PAS reference, either 'legacy' or 'V2', default is 'V2'

Value

The function PAS2GEF() returns 3 input tables of PAS references: PASREF$refUTRraw is for 3'UTR PAS, PASREF$dfIPA and PASREF$dfLE are for IPA references.

Author(s)

Ruijia Wang

Examples

```
## build Reference ranges for 3'UTR PASs in mouse
 download.file(url='ftp://ftp.ensembl.org/pub/release-99/gtf/mus_musculus/Mus_musculus.GRCm38.99.gtf.gz',
              destfile='Mus_musculus.GRCm38.99.gtf.gz')
 GTFfile="Mus_musculus.GRCm38.99.gtf.gz"

 PASREF=PAS2GEF(GTFfile, AnnoMethod="V2")
 refUTRraw=PASREF$refUTRraw
    dfIPA=PASREF$dfIPA
 dfLE=PASREF$dfLE
```

PASEXP_3UTR *PASEXP_3UTR, calculate relative expression of aUTR and cUTR regions*

Description

Map reads to 3'UTR APA regions and calculate relative expression of aUTR and cUTR regions.

Usage

```
PASEXP_3UTR(UTRdb, flS, Strandtype="NONE")
```

Arguments

UTRdb	a genomic ranges of aUTR(pPAS to dPAS) and cUTR(cdsend to pPAS) regions for each gene
flS	bamfile lists containing the file and path of bam files
Strandtype	strand type of the bam file; "forward" is forwand sequencing, "invert" is reverse sequencing and "NONE" is non-strand specific, Default is "NONE".

Value

The function PASEXP_3UTR() return a dataframe containning reads count, RPKM and relative expression of aUTR and cUTR for each gene

Author(s)

Ruijia Wang

Examples

```
## count reads mapped to 3'UTR APA regions and
## calculate relative expression of aUTR and cUTR regions
## using forward sequencing
    library("TBX20BamSubset")
    library("Rsamtools")
    library("GenomicAlignments")
 library("repmis")
    flsall = getBamFileList()
 URL="https://github.com/RJWANGbioinfo/PAS_reference_RData/blob/master/"
 file="mm9_REF.RData"
 source_data(paste0(URL,file,"?raw=True"))
    refUTRraw = refUTRraw[which(refUTRraw$Chrom=='chr19'),]
    UTRdbraw = REF3UTR(refUTRraw)
    DFUTRraw = PASEXP_3UTR(UTRdbraw, flsall, Strandtype="forward")
```

PASEXP_IPA *PASEXP_IPA, calculate relative expression of IPA regions*

Description

Map reads to IPA regions and calculte relative expression of aUTR and cUTR regions.

Usage

```
PASEXP_IPA(dfIPAraw, dfLEraw, flS, Strandtype="NONE", nts=1, minMQS=0, SeqType = "SingleEnd")
```

Arguments

dfIPAraw	a dataframe containing 8 colmuns for Intronic PASs: 'PASid', 'gene_symbol', 'Chrom', 'Strand', 'Pos', 'upstreamSS', 'downstreamSS'. 'upstreamSS' means closest 5'/3' splice site to IPA, 'downstreamSS' means closest 3' splice site
dfLEraw	a dataframe containing 5 colmuns for 3' least exon: 'gene_symbol', 'Chrom', 'Strand', 'LEstart', 'TES'. 'LEstart' means the start position of last 3' exon.
flS	bamfile lists containing the file and path of bam files
Strandtype	strand type of the bam file; "forward" is forwand sequencing, "invert" is reverse sequencing and "NONE" is non-strand specific, Default is "NONE".
nts	number of threads used for computing, parameter used by featureCounts, nthread option, Default is 1
minMQS	minimum mapping quality score of counted reads, parameter used by featureCounts, minMQS option, Default is 0
SeqType	set the sequencing type of reads in bam files can be either 'SingleEnd' (default) or 'ThreeMostPairEnd'.

Value

The function PASEXP_IPA() return a dataframe containning reads count, RPKM and relative expression of aUTR and cUTR for each gene

Author(s)

Ruijia Wang

Examples

```
## count reads mapped to IPA regions and
## calculte relative expression of aUTR and cUTR regions
## using forward sequencing
    library("TBX20BamSubset")
    library("Rsamtools")
    library("GenomicAlignments")
 library("repmis")
    flsall = getBamFileList()
 URL="https://github.com/RJWANGbioinfo/PAS_reference_RData/blob/master/"
 file="mm9_REF.RData"
 source_data(paste0(URL,file,"?raw=True"))
    IPA_OUTraw=PASEXP_IPA(dfIPA, dfLE, flsall, Strandtype="forward", nts=1)
```

REF3UTR *REF3UTR, build reference regions for 3'UTR PASs*

Description

Build 3'UTR PAS Reference for distal and proximal PAS.

Usage

```
REF3UTR(refUTR)
```

Arguments

refUTR	a dataframe containing 6 colmuns for 3'UTR PASs: 'gene_symbol', 'Chrom', 'Strand', 'Proximal', 'Distal', 'cdsend'

Value

The function REF3UTR() returns a genomic ranges of aUTR(pPAS to dPAS) and cUTR(cdsend to pPAS) regions for each gene

Author(s)

Ruijia Wang

Examples

```
## build Reference ranges for 3'UTR PASs in mouse
 library(repmis)
 URL="https://github.com/RJWANGbioinfo/PAS_reference_RData/blob/master/"
 file="mm9_REF.RData"
 source_data(paste0(URL,file,"?raw=True"))
    refUTRraw=refUTRraw[which(refUTRraw$Chrom=='chr19'),]
    UTRdbraw=REF3UTR(refUTRraw)
```

REF4PAS *REF4PAS, build reference regions for 3'UTR and Intronic PAS using dataframe formated input*

Description

build reference regions for 3'UTR and Intronic PAS using dataframe formated input

Usage

```
REF4PAS(refUTRraw, dfIPAraw, dfLEraw)
```

Arguments

refUTRraw	a dataframe containing 6 colmuns for 3'UTR PASs: 'gene_symbol', 'Chrom', 'Strand', 'Proximal', 'Distal', 'cdsend'
dfIPAraw	a dataframe containing 8 colmuns for Intronic PASs: 'PASid', 'gene_symbol', 'Chrom', 'Strand', 'Pos', 'upstreamSS', 'downstreamSS'. 'upstreamSS' means closest 5'/3' splice site to IPA, 'downstreamSS' means closest 3' splice site
dfLEraw	a dataframe containing 5 colmuns for 3' least exon: 'gene_symbol', 'Chrom', 'Strand', 'LEstart', 'TES'. 'LEstart' means the start position of last 3' exon.

Value

The function `REF4PAS()` returns list a genomic ranges of 3'UTR, Intronic PAS and last 3'exon regions for each gene

Author(s)

Ruijia Wang

Examples

```
## build Reference ranges for 3'UTR and Intronic PAS in mouse (mm9)
 library(repmis)
 URL="https://github.com/RJWANGbioinfo/PAS_reference_RData/blob/master/"
 file="mm9_REF.RData"
 source_data(paste0(URL,file,"?raw=True"))
    refUTRraw=refUTRraw[which(refUTRraw$Chrom=='chr19'),]
```

```
dfIPAraw=dfIPA[which(dfIPA$Chrom=='chr19'),]
dfLEraw=dfLE[which(dfLE$Chrom=='chr19'),]
   PASREF=REF4PAS(refUTRraw,dfIPAraw,dfLEraw)
UTRdbraw=PASREF$UTRdbraw
   dfIPA=PASREF$dfIPA
dfLE=PASREF$dfLE
```

REFCDS *REFCDS, build reference regions for CDS of protein coding genes*

Description

Build CDS reference for protein coding genes.

Usage

```
REFCDS(txdb,IDDB)
```

Arguments

txdb	a TranscriptDb generate using GenomicFeatures
IDDB	Genome annotation of the corresponding species, e.g., "org.Hs.eg.db"

Value

The function `REFCDS()` returns a genomic ranges of CDS regions for each coding gene

Author(s)

Ruijia Wang

Examples

```
## build Reference ranges for CDS in mouse coding genes
    library("GenomicFeatures")
    library("org.Mm.eg.db")
    extpath = system.file("extdata", "mm9.chr19.refGene.R.DB", package="APAlyzer")
    txdb = loadDb(extpath, packageName='GenomicFeatures')
    IDDB = org.Mm.eg.db
    CDSdbraw = REFCDS(txdb,IDDB)
```

ThreeMostPairBam *ThreeMostPairBam, extract 3 prime most alignment of a paired-end bam file*

Description

extract 3 prime most alignment of a paired-end bam file and saved into a new bam file.

Usage

```
ThreeMostPairBam(BamfilePath, OutDirPath, StrandType="NONE")
```

Arguments

BamfilePath	file path of a bam file
OutDirPath	output folder path
StrandType	strand type of the bam file; "forward-reverse": read 1 forward but read 2 is reverse sequencing, "reverse-forward": read 2 forward but read 1 is reverse sequencing, and "NONE" is non-strand specific, Default is "NONE".

Value

The function ThreeMostPairBam() return a single-end bam file containning 3 prime most alignment of the input paired-end file

Author(s)

Ruijia Wang

Examples

```
## Extract 3 prime most alignment of a paired-end
## bam file and saved into a new bam file
    library("pasillaBamSubset")

    ThreeMostPairBam (BamfilePath=untreated3_chr4(),
      OutDirPath=getwd(),
      StrandType='forward-reverse')
```

Index